Temporally Consistent 3D Human Pose Estimation Using Dual 360° Cameras

Matthew Shere[1] Hansung Kim[1,2] Adrian Hilton[1]

1. Centre for Vision, Speech and Signal Processing
University of Surrey, UK
{m.shere, a.hilton}@surrey.ac.uk

2. Electronics and Computer Science
University of Southampton, UK
h.kim@soton.ac.uk

Abstract

This paper presents a 3D human pose estimation system that uses a stereo pair of 360° sensors to capture the complete scene from a single location. The approach combines the advantages of omnidirectional capture, the accuracy of multiple view 3D pose estimation and the portability of monocular acquisition. Joint monocular belief maps for joint locations are estimated from 360° images and are used to fit a 3D skeleton to each frame. Temporal data association and smoothing is performed to produce accurate 3D pose estimates throughout the sequence. We evaluate our system on the Panoptic Studio dataset, as well as real 360° video for tracking multiple people, demonstrating an average Mean Per Joint Position Error of 12.47cm with 30cm baseline cameras. We also demonstrate improved capabilities over perspective and 360° multi-view systems when presented with limited camera views of the subject.

1. Introduction

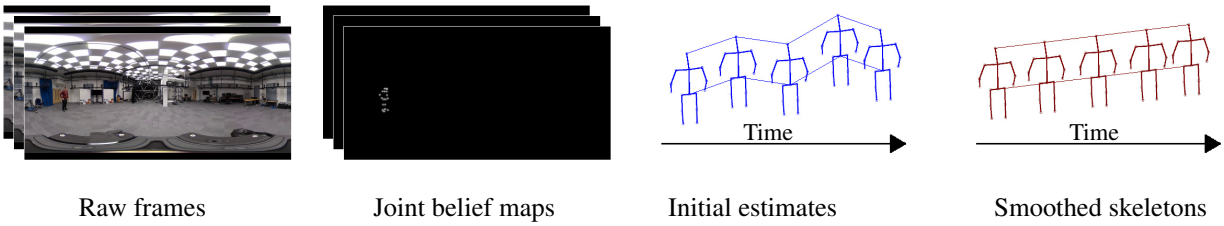
3D human pose estimation is an often studied area of Computer Vision, wherein we attempt to find the 3D coordinates of a skeleton or model of a person, given only 2D images or videos. Whilst its main application is in the entertainment sector, 3D pose estimates can also be used in medical and assisted living fields, as well as in bio-mechanical applications. While entertainment fields traditionally used special clothing or other actor-mounted markers, this is time consuming and often impractical in unconstrained environments. To this end, markerless techniques were devised, and are a major focus of modern 3D pose estimation systems.

Modern 3D pose estimation systems can be broadly described in two categories, dependent on the camera arrangement. Monocular systems use a single input camera, combined with either deep learning approaches and/or a large corpus of example poses to produce a 3D pose estimate. These systems are lightweight and portable, due to the use of a single camera, but suffer from depth ambiguity unless the system is tuned to a specific performer[29][9]. Conversely, stereo or multi-view systems combine input from 2 or more cameras to provide accurate pose estimates, removing depth ambiguity and placing the pose in a constant world coordinate system. However, such systems are expensive, since achieving total scene coverage currently requires a network of perspective cameras together with calibration, synchronisation and data storage, and thus are generally located in purpose built studios. More pertinently, both of these arrangements have a major drawback of capture space, restricted to either a single perspective camera in the case of monocular systems, or to a tightly defined capture volume in the case of multi-view systems. In this paper, we propose a single pair of 360° cameras that provide complete scene coverage for 3D human pose estimation, one that is free of these restrictions.

The motivation of this work, therefore, is to leverage 360° sensors to gain the benefits of both systems, while also overcoming the capture space problem. We arrange a pair of 360° sensors vertically (Figure 2a) on a single tripod, which can be placed at the centre of any "capture space". Due to the omni-directional nature of the sensors, our capture volume has no limit beyond the resolution of the camera, while the 360° camera pair provides a compact system with complete scene coverage, providing a practical, accurate and low-cost alternative to existing 3D pose estimation systems.

1.1. Contribution

Our main contribution is a temporally consistent 3D human pose estimation system for a complete scene captured using a pair of vertically aligned 360° cameras with a narrow baseline. State of the art approaches require a network of more than 2 cameras for complete scene coverage, while stereo estimation systems generally require a horizontal baseline in order to reliably estimate depth information. The proposed approach overcomes these limitations with a single pair of vertically displaced 360° cameras. This configuration combines the lightweight portability of a monocular system with the depth accuracy of a multi-view system,



Raw frames Joint belief maps Initial estimates Smoothed skeletons

Figure 1: An overview of our system. Joint belief maps are obtained from input images using Openpose[4], then approximated as 2D Gaussians. Skeletons are then fitted to these Gaussians in each frame, before being smoothed across the entire sequence.

while capturing the entire scene without the need of a large camera count.

2. Related Works

At its core, 3D pose estimation needs to accurately reconstruct the position and motion of a given subject at each frame. More specifically, we need to identify the individual body components (as joints, body segments etc.), calculate their 3D position and use the information to reason about the location of any unidentified points, thus estimating the full 3D pose in each frame.

As noted in Section 1, 3D human pose estimation systems can be split into two categories. Single camera, or monocular pose estimation systems, use a single camera and "lift" estimated 2D pose information to a 3D pose. Stereo camera, or multi-view pose estimation systems, triangulate between multiple cameras from different angles to capture images that allow a 3D pose to be calculated from multiple 2D pose estimates.

2.1. Single Camera 3D Human Pose Estimation

Monocular images have an inherent depth ambiguity, in that without prior knowledge of the subject (for example, their height), we cannot accurately determine their position relative to the camera. A simple approach to solving this problem is the use of a body model with the correct dimensions. Bogo *et al*. [3] use a CNN to estimate 2D joint locations, then optimise the parameters of a Skinned Multi-Person Linear Model (SMPL) body model, using a capsule approximation of SMPL to stop implausible body poses from occurring. However, they require the camera focal length in advance, as well as assuming the subject is standing parallel to the the image plane.

Tome *et al*. [23] created a probabilistic skeletal model from by applying Principal Component Analysis (PCA) on the Human 3.6M dataset, then predict 2D pose positions from an input image, iteratively fitting 3D model parameters to the 2D joint positions and refining. This in turn informs the 2D joint positions, which can be updated if the 3D pose demonstrates their inaccuracy (for example, transposing the left and right hand).

Using a mesh fitting approach rather than keypoint fitting, Kanazawa *et al*. [14] attempt to fit the projection of a 3D body mesh to a 2D image. SMPL model parameters are optimised, then the resulting body model projected as a mesh onto the input image, as well as being fed into a discriminator, which determines if the resulting mesh constitutes a person. This prevents results that minimise reprojection error but that are still improbable or anatomically impossible.

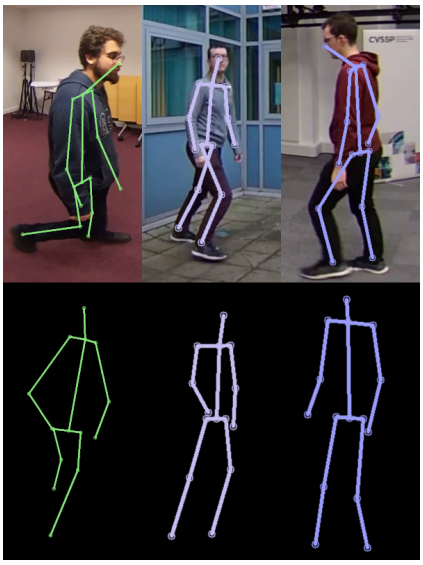
Incorporating temporal information, Pavllo *et al*. [19] use a multi-dimensional CNN with dilated convolutions to optimise a skeletal path that best explains 2D pose estimates across a block of hundreds of frames. An auto-encoder is also trained to project unlabelled 2D joint information to 3D joint positions, along with an auto-decoder that performs the reverse. Skeletal path and 3D positions are then jointly optimised during training.

Taking a hat mounted wide angle lens, Xu *et al*. [30] make use of a CNN, trained on a large and bespoke synthetic dataset. This trained CNN is capable of estimating 2D joint locations from a fisheye projection, using two separate branches to account for the distortion. These joint belief maps are then fed into a distance module which estimates the joint distance, before a full 3D pose is estimated. This produces results relative to the camera, however given camera position it becomes very easy to self occlude limbs.

Kolotouros *et al*. [15] provides another mesh deformation technique, again using the SMPL model. In contrast to previous techniques, model parameters are not directly optimised, but rather a proposed Graph CNN is utilized to find a mesh deformation matching the input image. Once found, this deformed mesh can be used to fit the parameters of a SMPL model for use in other applications. This also manages to achieve near real time inference, however challenging poses and multiple people in frame cause issues.

Xu *et al*. [28] makes use of a series of neural networks to perform pose estimation. An initial network is trained to learn the intrinsic parameters of a video sequence by comparing the projection of 3D ground truth joint positions to the 2D joint estimate. This network is then used to 'correct'

(a) Example 360° camera configuration



(b) Example 3D estimates from 360° images captures in indoor and outdoor environments, and novel view from behind

Figure 2: Camera configuration, capable of complete scene coverage, and produced pose estimates

potentially unreliable 2D joint estimates. Next, a trained kinematic skeleton is used to refine 3D joint positions according to typical human motion, and a third network then temporally interpolates the refined 3D joint information, replacing any 3D point found to be too inaccurate.

Finally, Habermann *et al*. [9] jointly optimise skeletal pose and model distortion, fitting a pre scanned subject to both Openpose[4] and a clean foreground segmentation of the subject. Their system is trained on multi-view data, but produces an inference from a single view. This combination produces good depth results, however requires a large set of motion captured in a studio environment by the subject, as well as a 3D scan of the subject, limiting its portability.

2.2. Multi Camera 3D Human Pose Estimation

Multi camera setups have the advantage on levels of input data, solving depth problems as well as giving little to no occlusion of body parts. Early applications of this include Carranza *et al*. [5], taking 7 calibrated cameras, obtaining subject silhouettes and then optimising a pre-constructed human body model to them. Temporal consistency is obtained by minimising the energy required to move to the next frame. Conversely, Starck and Hilton[22] use 16 cameras with a chroma-key background to construct a visual hull, before refining the visual hull using keypoints detected on the subject.

De Aguiar *et al*. [6] and Vlasic *et al*. [27] take similar approaches, fitting a pre scanned mesh and/or a template mesh to image keypoints obtained from calibrated cameras. This approach allows for loose clothing to be accounted for, albeit at the cost of either laser scanning the subject or manual editing of the final mesh to tidy it up.

Liu *et al*. [16] use pre-scanned individuals, creating an articulated skeleton which can then be fitted onto 2D image segmentations. This approach has the advantage of being able to handle multiple people in close proximity, assuming they are all scanned in advance. Trumble *et al*. [24] utilizes a deep learning approach, creating a probabilistic visual hull from multiview image data, to which joint positions are fitted.

Making use of additional information in the form of Inertial Measurement Units (IMUs), Marcard *et al*. [26] expand from traditional multi-view stereo and combined the IMU data, allowing all joints to be tracked across all input frames, even in the event of total occlusion. Trumble *et al*. [25] expands upon their previous work, adding IMU data and a Long Short Term Memory (LSTM) to reduce temporal noise across both image and IMU data. Malleson *et al*. [17] takes input video and IMU data to create a real time robust and temporally consistent skeletal fitting system.

Huang *et al*. [10] expand upon the Bogo *et al*. monocular approach, fitting a SMPL model to the subject's silhouette from each view per frame, effectively fitting the SMPL model to a visual hull, then applying temporal reasoning across frame batches. This produces accurate results, as well as maintaining the underlying capability to operate on monocular sequences. Their system requires accurate silhouettes of the input images, however, as well as requiring camera focal length information.

Performing a form of triangulation, Iskakov *et al*. [12] create a network that learns how to triangulate. 2D joint belief maps are "unprojected" into a 3D space, creating a 3D belief map for each joint identified. A bounding cube is then placed around the central joint in order to limit search space, and a network used to predict a 3D pose based upon these 3D belief maps.

Expanding upon IMU work, Zhang *et al*. [31] estimates joint belief maps for each view independently, then uses IMU orientation data to fuse these belief maps in 3D space relative to each other, rather than attempting to use IMU information at a later stage to 'fix' poor performance. A skeletal model with fixed joint lengths is then fit to these belief maps.

2.3. 360° Camera 3D Human Pose Estimation

To our knowledge, relatively few works have attempted to specifically utilize 360° cameras with pose, rather than applying a perspective pose estimation system to 360° images. Fowler *et al*. [7] use a single 360° camera to create an affordance map of a room. An individual is identified and their activity is broadly categorised (walking, sitting, standing *etc.*). This information, along with position and depth,

is used to produce an affordance map, as well as a 3D scene reconstruction of the respective surfaces. It is limited, however, to 2D per frame pose estimation of a single individual and does not reconstruct 3D pose nor handle multiple individuals.

Shere *et al*. [21] fit a pre-measured skeleton to keypoints identified by two 360° cameras. Axis-angle rotations are optimised to produce a skeleton, for which joint reprojection error and joint movement are minimised. Additionally, joint angle acceleration is also minimised, in order to prevent gross scale changes caused by joint mis-identification. Notably, their system uses cameras placed 1-2 metres apart, as well as needing the cameras to be close to the same horizontal plane.

Both of these systems still maintain the weaknesses of their perspective counterparts. Fowler *et al*. take some prior knowledge of the subject, as well as being limited to indoor scenes. Shere *et al*. still use two separate camera locations, resulting in a large footprint for the capture system, and placing limitations on the capture volume in terms of minimum size.

Our proposed system expands upon Shere *et al*. in both representation and realism, by using non-rigid joint lengths and allowing for minor motion in 3D joint positions without penalty. The addition of the Principal Component Analysis (PCA) model restricts the system to plausible poses in the first stage, while our reprojection error uses joint belief maps rather than absolute (2D) joint positions, allowing us to exploit uncertainty when combined with flexible joint lengths. Finally, our temporal consistency is vastly improved with our smoothing step, since we use information from both the previous and next frames. All of this is achieved on a significantly lower baseline, 30cm for our system against ≈3m for Shere *et al*. while demonstrating increased accuracy.

3. Method

Our method takes a two step approach, first producing a per frame 3D skeletal estimate, then performing temporal smoothing across the entire video sequence, optimising each joint against its temporal neighbours. In both steps, we perform a least squares optimisation using Ceres[2]

3.1. Initial Skeletal Estimate

The initial stage of our pipeline is to obtain our skeletal model to be used in our optimisation. We take a series of capture sequences from the Panoptic Studio dataset[13], each containing a single individual performing a series of poses. This gives us a collection of 3D skeletal poses $s \in D$, where s is a 3D skeleton consisting of J joints and D is the collection of 3D skeletons. Each joint $j_i = [x, y, z] \in s$ is the 3D position of the joint, with i being the index of the joint. We translate each skeleton such that the neck joint is located at the origin, then rotate each skeleton so that the vector from the left hip to the right hip is consistent in each skeleton. Using this database D, we perform principal component analysis (PCA)[20]. Combined with a translation vector and rotation scalar, this allows us to express a skeleton as $P(\boldsymbol{\psi}, r, \boldsymbol{t})$, where $\boldsymbol{\psi}$ is a vector of parameters with length p, r is the rotation about the z (vertical) axis, and $\boldsymbol{t}$ is a 3D vector that translates the 3D skeleton to any point in 3D space.

Using this representation, we can then perform our optimisation on a video sequence V, captured on two 360° cameras α, β, and consisting of N frames, each of height h and width w. Given a frame $f_n \in V$, where n is the frame number, we obtain joint belief maps using Openpose[4], then fit a 2D Gaussian $\delta_i(a, x_{max}, y_{max}, x_\sigma, y_\sigma)$ to each belief map, with a being the maximum amplitude, x_{max}, y_{max} being the x, y coordinates of the maximum amplitude, and x_σ, y_σ being the standard deviation in the x and y axis respectively. This produces a set of J Gaussians, denoted as Δ, and with each $\delta_i \in \Delta$ corresponding to a joint j_i. Given a suitable skeleton size $l_{(j_1,j_2)} \in L$, where $l_{(j_1,j_2)}$ is the length between joints j_1 and j_2, we attempt to minimise the error function $\epsilon_{solve}(\boldsymbol{\psi}_n, r_n, \boldsymbol{t}_n)$ for each frame f_n, as denoted in Equation 1

$$\begin{aligned}\epsilon_{solve}(\boldsymbol{\psi}_n, r_n, \boldsymbol{t}_n) = {} & \epsilon_{proj}(s_n, \Delta_n, \alpha)^{u_{proj}} \\ & + \epsilon_{proj}(s_n, \Delta_n, \beta)^{u_{proj}} \\ & + \epsilon_{move}(\boldsymbol{t}_n, \boldsymbol{t}_{n-1})^{u_{move}} \\ & + \epsilon_{rot}(r_n, r_{n-1})^{u_{rot}} \\ & + \epsilon_{size}(s_n)^{u_{size}} + \epsilon_{length}(s_n, L)^{u_{length}}\end{aligned} \tag{1}$$

where $s_n = P(\boldsymbol{\psi}_n, r_n, \boldsymbol{t}_n)$. The first two components of this error function are our reprojection errors for cameras $c = \alpha, \beta$, outlined in Equation 2

$$\epsilon_{proj}(s, \Delta, c) = \sum_{i=0}^{J} \sqrt{\Gamma_{dist}(\rho(j_i, c), \delta_i^{max}) \cdot \gamma_i(\delta_i, \rho(j_i, c))} \tag{2}$$

where $\rho(j, c)$ is the projection of joint $j \in s$ onto camera c, $\gamma(\delta, C)$ is the Gaussian distribution value of 2D coordinate $C = [x, y]$ with using the Gaussian $\delta_i \in \Delta$, and

$$\begin{aligned}\Gamma_1 &= ||\mathbf{a} - \mathbf{b}|| \\ \Gamma_2 &= \begin{cases} ||(\mathbf{a} - [\mathbf{w}, \mathbf{0}]) - \mathbf{b}|| : \text{for } a_x > b_x \\ ||(\mathbf{a} + [\mathbf{w}, \mathbf{0}]) - \mathbf{b}|| : \text{for } a_x \leq b_x \end{cases} \\ \Gamma_{dist}(\mathbf{a}, \mathbf{b}) &= min(\Gamma_1, \Gamma_2)\end{aligned} \tag{3}$$

is the shortest circular l_2 norm between coordinates $\mathbf{a}, \mathbf{b}$, with $||\cdot||$ being the l_2 norm.

The third and fourth components of Equation 1 minimise the amount of joint movement from the previous frame, both in terms of translation and rotation, and are defined as

$$\epsilon_{move}\left(\boldsymbol{t}_n - \boldsymbol{t}_{n-1}\right) = \max\left(\left|\left|\boldsymbol{t}_n, \boldsymbol{t}_{n-1}\right|\right|, \omega_{move}\right) \quad (4)$$

$$\epsilon_{rot}\left(r_n, r_{n-1}\right) = \max\left(\Gamma_{rot}\left(r_n, r_{n-1}\right), \omega_{rotation}\right) \quad (5)$$

where $\omega_{move}, \omega_{rot}$ are the amount of translation and rotation permitted between frames before accumulating error respectively, and

$$\begin{aligned} \Gamma_1 &= \theta_1 - \theta_2 \\ \Gamma_2 &= \begin{cases} \Gamma_1 + 360 : \text{for } \Gamma_1 < 0 \\ \Gamma_1 - 360 : \text{for } \Gamma_1 \geq 0 \end{cases} \\ \Gamma_{rot}\left(\theta_1, \theta_2\right) &= |min(\Gamma_1, \Gamma_2)| \end{aligned} \quad (6)$$

is the shortest circular angle between two angles.

The fifth component of the error function constrains the x, y size of the skeleton against skeletal centroid, and is defined as

$$\epsilon_{size}\left(s\right) = \sum_{i=0}^{J} \max\left(\left|\left|j_i - \frac{\sum_{k=0}^{J} j_k}{J}\right|\right|, \omega_{size}\right) \quad (7)$$

where ω_{size} is the distance from the centroid the joint can have before accumulating error.

Finally, we have the joint length error, which restricts the skeletal joint lengths against those expected in L

$$\epsilon_{length}\left(s, L\right) = \sum \max\left(\left|L_{i,k} - \left|\left|j_i - j_k\right|\right|\right|, L_{i,k} \cdot \omega_{length}\right) \quad \text{for joint pairs } i, k \in L \quad (8)$$

with ω_{length} as the maximum joint length variation before error is accumulated. Note that a joint may have multiple lengths associated for more rigid sections of the skeleton (such as the left shoulder linking to the neck, right shoulder, left hip and right hip)

Each error component is then weighted in order to emphasize that portion of the optimisation. From our experiments, we found $u_{proj} = 3, u_{move} = 1, u_{rot} = 1, u_{size} = 3, u_{length} = 3, \omega_{move} = 10, \omega_{rot} = 1, \omega_{size} = 40, \omega_{length} = 0.1$ produced good results across all experiments.

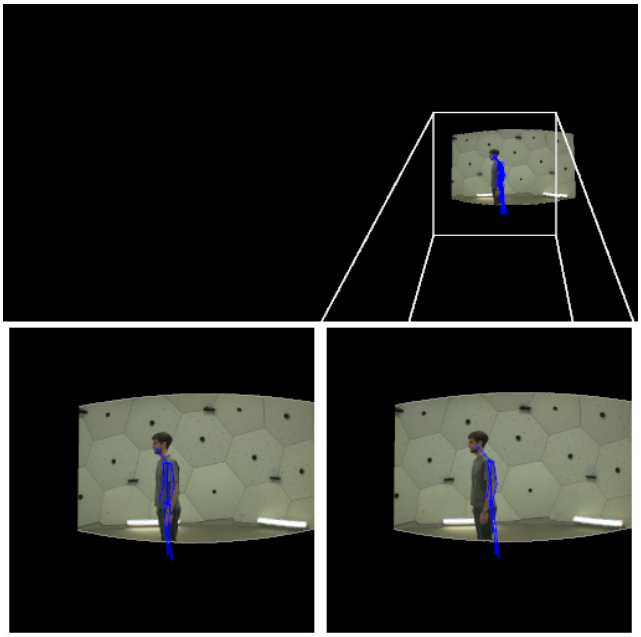


Figure 3: Example of a protocol 1 pose estimate from an (unused) reference camera. Top: Equirectangular frame. Left: Ground Truth. Right: Ours. Note the slightly rearward skeletal position, caused by errors introduced during projection process

3.2. Temporal Smoothing

Once we have initial skeletal estimates for each frame (section 3.1), we perform temporal association and smoothing. Unlike the previous section, our optimisation function uses information from both previous and future frames to smooth out errors introduced from either poor input or optimisation. We also run our optimisation in several passes, in order to allow our smoothing to replace the aforementioned poor data. This stage therefore acts as a refinement algorithm to our initial estimates produced in Section 3.1.

In order to give our system maximum flexibility, we optimise each joint position directly in terms of its x, y, z coordinates. We therefore optimise the vector $\boldsymbol{\Psi}$ of length $3J$, instead of the PCA parameters $P\left(\boldsymbol{\psi}, r, \boldsymbol{t}\right)$ as in Equation 1. On each pass, we attempt to minimise the function $\epsilon_{smooth}(\boldsymbol{\Psi}_n)$ as defined in Equation 9

$$\begin{aligned} \epsilon_{smooth}\left(\boldsymbol{\Psi}_n\right) = \epsilon_{proj}\left(s_n, \Delta_n, \alpha\right) + \epsilon_{proj}\left(s_n, \Delta_n, \beta\right) \\ + \epsilon_{shift}\left(s_n, s_{n-1}\right) + \epsilon_{align}\left(s_{n+1}, s_n, s_{n-1}\right) \\ + \epsilon_{size}\left(s_n\right) + \epsilon_{length}\left(s_n, L\right) \end{aligned} \quad (9)$$

where s is the joints optimised on the current pass, and s_{n-1}, s_{n+1} are the skeletons produced on the previous pass for the previous and next frame (if they exist).

Dataset	MPJPE (mm)					
	Triangulated	Iskakov[12]	Pavllo[19]	Shere[21]	Tome[23]	Ours
171204_pose1	184.39	331.54	188.96	332.63	209.76	**141.53**
171026_pose3	**184.48**	430.70	286.38	455.52	266.68	218.38
161029_tools1	203.06	382.27	419.07	388.18	333.88	**165.05**
161029_sports1	238.74	520.84	575.02	362.33	336.72	**151.62**
Synthetic_1	717.10	N/A	590.31	733.99	667.44	**492.58**
Synthetic_2	901.59	N/A	685.82	778.75	795.59	**239.60**

Table 1: Protocol 1 results of each approach. MPJPE is averaged across all frames containing one person

$\epsilon_{proj}, \epsilon_{size}$ and ϵ_{length} are defined in Equations 2, 7 and 8 respectively, while $\epsilon_{shift}\left(s_n, s_{n-1}\right)$ is defined as

$$\epsilon_{shift}\left(s_n, s_{n-1}\right) = \sum_{i=0}^{J} \max\left(\left|\left|j_i - k_i\right|\right|, \omega_{shift}\right) \quad j_i \in s_n, k_i \in s_{n-1} \tag{10}$$

and is the total movement of each joint with respect to the previous frame, where ω_{shift} is the maximum allowed movement before error is accumulated. Finally

$$\epsilon_{align}\left(s_{n+1}, s_n, s_{n-1}\right) = \sum_{i=0}^{J} \left|\left| g_i - \left(g_i + \left(\left(\frac{k_i - g_i}{||k_i, g_i||} \right) \cdot \left(j_i - g_i\right)^T \right) \frac{k_i - g_i}{||k_i, g_i||} \right) \right|\right| \quad g \in s_{n-1}, j \in s_n, k \in s_{n+1} \tag{11}$$

is the distance of the current frame joint to the vector defined by the previous frame joint and the future frame joint. Note we only move the current frame joint to be co-linear with the past/future joints, rather than the midpoint of the past and future joints.

This optimisation process is repeated O times, and from our experiments, we found $O = 20, \omega_{shift} = 10, \omega_{size} = 40, \omega_{length} = 0.1$ produced good results across all experiments.

4. Evaluation

We evaluate our system on three benchmark datasets. We produce a quantitative evaluation, using 4 scenes from the Panoptic Studio dataset[13], selecting two cameras (hd_00_01 and hd_00_30) that are close to vertically aligned, and projecting the frames into equirectangular space for processing. We use a fifth scene (171204_pose2) for our PCA, and compare against the pose estimates provided by the dataset.

The scenes selected (detailed in Tables 1 and 2) are broken down into sequences containing a single person, and are between 130 and 6950 frames in length. 171204_pose1 and 171026_pose3 cover a range of motion for an individual in a static position. 161029_tools1 deals with fine hand control, and 161029_sports1 contains sporting motions for baseball, tennis, juggling and break dancing. We use this dataset as, to our knowledge, no 360° datasets exist that contain both vertical pairs of 360° sensors and ground truth 3D joint locations. As such, we also evaluate on a pair of synthetic datasets (generated using Blender) using a closer camera baseline (30cm, compared to ≈ 120cm for panoptic).

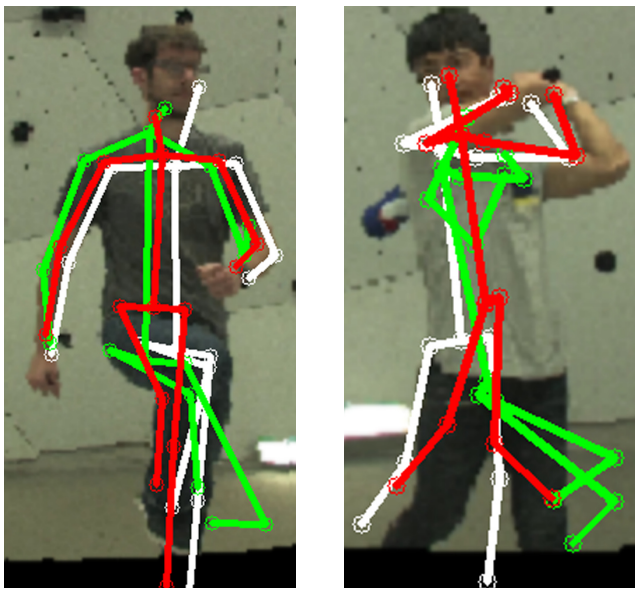


Figure 4: Example reference camera frames. Red is ours. Green is Tome *et al*. White is ground truth

We compare against Shere *et al*. [21] as a stereo 360° pose estimation system, Iskakov *et al*. [12] as a stereo perspective pose estimation system, Tome *et al*. [23] as a monocular perspective pose estimation system without temporal information, and Pavllo *et al*. [19] as a monocular perspective pose estimation system with temporal information. We also include a naive triangulation result for baseline performance. We perform an additional qualitative evaluation

Figure 5: Example pose estimates on a panoramic image (top and bottom removed for brevity). Each pose estimate was performed separately.

on scenes captured using a pair of Ricoh Theta V[1] spaced 30cm apart. These scenes cover multi person scenes in both outdoor and non-studio indoor environments, examples can be seen in Figure 2b and Figure 5. Please refer to our supplementary materials for further results.

4.1. Quantitative Evaluation

Each system is evaluated using Mean Per Joint Position Error (MPJPE), which is defined as

$$\text{MPJPE}\left(s_{est}, s_{gt}\right) = \frac{\sum_{i=0}^{J} ||g_i - k_i||}{J} \qquad (12)$$
$$g \in s_{est}, k \in s_{gt}$$

and measures the average distance between the joints of the estimated skeleton and the ground truth skeleton. Since the projection and rotation of the Panoptic Studio dataset into 360° is inexact, and error may be introduced during this process (see Figure 3), we compare our system according to 2 different protocols. In protocol 1, we measure MPJPE of the estimated skeleton against the ground truth skeleton, while in protocol 2, we first perform Procrustes Analysis[8] before comparison. This is done to eliminate projection errors for our system and Shere *et al*., as well as eliminating the depth ambiguity for monocular systems, providing a fairer comparison. No other processing is performed on the estimated skeleton.

Under protocol 1 (Table 1), our system shows good performance on the panoptic data sets, despite the inaccuracies from projection (Figure 4, 6). When the baseline is lowered in the synthetic datasets, we maintain reasonable performance while other systems struggle. As noted in Section 2.1, monocular systems struggle with depth ambiguity due to a single viewpoint, accounting for some of the error. However, when that ambiguity is removed in protocol 2 (Table 2), our system still outperforms Tome *et al*. and Pavllo *et al*., even when comparing some of our protocol 1 results to monocular protocol 2 results.

Simultaneously, both Shere *et al*. and Iskakov *et al*. show very poor performance, with protocol 2 only showing slight improvements. For Shere *et al*., this can explained by both projection issues described above, and with camera configuration, since their system was demonstrated with horizontally placed cameras. In the case of Iskakov *et al*., we tried using both the reference implementation (trained on Human 3.6M[11]) and a variant trained using only the intended vertical camera pair from panoptic, however we were unable to get close to their reported error of 13.7mm[12] on panoptic when using 4 cameras. We were also unable to obtain results for Iskakov *et al*. on the synthetic data, due to the lack of calibration information produced for this data.

4.2. Ablation Study

We performed several experiments to test the effect of each component of our error functions in Equations 1 and 9. We eliminate error terms from each function and test their accuracy on a small subsection of the 171204_pose1, and results are presented according to Protocol 1.

As can be seen in Table 3, our presented combination is most effective in minimising MPJPE. Each row shows a specific error component removed from the initial solve, while each column shows the removal from the temporal smoother. Certain error function components appear to have little effect ($\epsilon_{length}, \epsilon_{move}$ for the solver, ϵ_{align} for the smoother), however their contribution is made more in the smoothness of qualitative the output video sequence, rather than in absolute error minimisation terms.

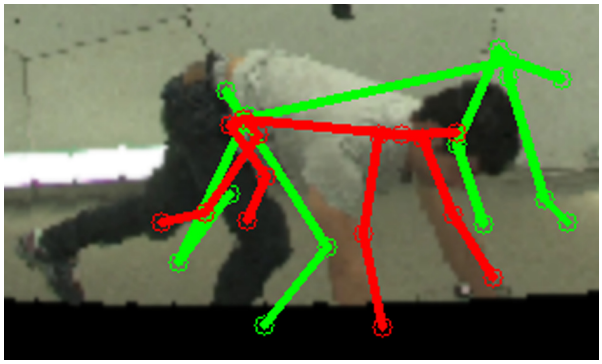


Figure 6: Reference camera frame during break dancing. Red is our skeleton. Green is Tome *et al*. Note lower legs and feet are out of frame in input images.

Dataset	PA-MPJPE (mm)					
	Triangulated	Iskakov	Pavllo	Shere	Tome	Ours
171204_pose1	66.59	219.51	116.66	273.14	76.62	**57.88**
171026_pose3	**43.29**	98.21	113.65	311.99	68.67	55.08
161029_tools1	**73.21**	136.82	117.14	256.09	79.29	77.99
161029_sports1	116.33	279.19	174.00	269.06	101.85	**73.17**
Synthetic_1	328.10	N/A	224.18	627.38	241.02	**119.14**
Synthetic_2	497.12	N/A	227.98	762.38	258.20	**130.32**

Table 2: Protocol 2 results of each approach. PA-MPJPE is averaged across all frames containing one person

Removed components	MPJPE (mm)					
	None	ϵ_{align}	ϵ_{length}	ϵ_{shift}	ϵ_{size}	All
None	**67.2**	67.5	69.4	70.2	69.4	69.4
ϵ_{length}	67.3	69.9	71.4	72.2	71.4	71.4
ϵ_{move}	67.3	67.4	69.4	70.7	69.4	69.9
ϵ_{rot}	68.9	68.9	69.4	70.4	69.4	69.8
ϵ_{size}	84.8	86.3	90.8	90.8	90.8	90.8

Table 3: Ablation study results. Rows show components removed from the solver (Equation 1), columns are components removed from the smoother (Equation 9)

4.3. Smoothing Effect

To better demonstrate this smoothing effect, we plotted the mid hip joint distance from a camera throughout a sequence where a subject walks a straight line in front of the camera and back again. From this movement, we would expect to see a double u shaped curve, as the distance starts high, decreases, then increases and finishes high, twice in succession. We compare the initial solved joints against the smoothed joints, as well as adding pure triangulation for reference.

As we can see from Figure 7, the triangulated skeleton is highly jittery, even when the person is stationary at the start of the sequence. Quantitatively, the triangulated skeleton gives a maximum depth change of 264.9mm between frames 214 and 215, and an average depth change of 59.5mm. The solved skeleton mitigates some of this, but nonetheless still shows some considerable jumps, peaking at 194.7mm between frames 186 and 187, and averaging a depth change of 29.6mm. Our smoothed skeleton produces a much smoother distance change, and while its maximum depth change is 151.0mm (between frames 109 and 110), this is the only frame with a change above 50mm, giving an average depth change of 12.0mm per frame.

5. Conclusion

We have presented a temporally consistent 3D human pose estimation system operating from a pair of 360° sensors. By aligning these sensors vertically, we have created a system that can be easily deployed yet provide good results, allowing its use in a variety of environments where time, equipment, power or space limitations may be in place. We demonstrate excellent accuracy, even when projecting perspective frames into an equirectangular projection with approximated rotations. As such, our system can also be used with traditional perspective videos, albeit with an expensive reprojection operation. For future work, we consider refining this to produce a system capable of operating from a vertical pair of arbitrary camera lens angles.

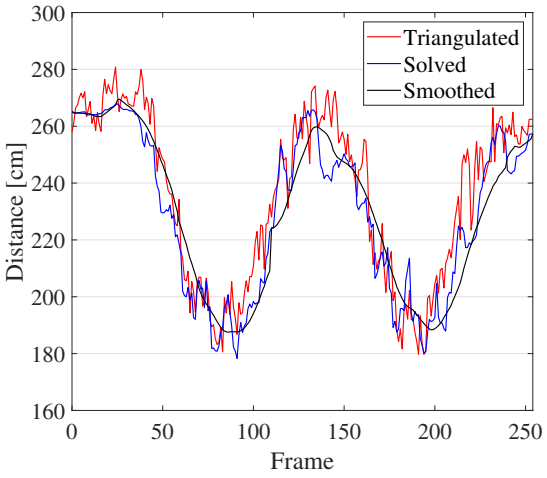


Figure 7: Plot of mid hip joint distance from camera as a subject walks a straight line in front the camera and back

6. Acknowledgments

This work is supported by both the EPSRC (grant number EP/N 509383/1) and BBC Research and Development. The authors would like to thank the reviewers for their constructive comments.

References

[1] Ricoh Theta V. https://theta360.com/uk/about/theta/v.html,

2019. Accessed: 2019-02-22.

[2] Sameer Agarwal, Keir Mierle, and Others. Ceres solver. http://ceres-solver.org.

[3] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *European Conference on Computer Vision*, pages 561–578. Springer, 2016.

[4] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[5] Joel Carranza, Christian Theobalt, Marcus A. Magnor, and Hans-Peter Seidel. Free-viewpoint video of human actors. In *ACM SIGGRAPH 2003 Papers*, SIGGRAPH '03, pages 569–577, New York, NY, USA, 2003. ACM.

[6] Edilson de Aguiar, Carsten Stoll, Christian Theobalt, Naveed Ahmed, Hans-Peter Seidel, and Sebastian Thrun. Performance capture from sparse multi-view video. *ACM Trans. Graph.*, 27(3):98:1–98:10, Aug. 2008.

[7] Sam Fowler, Hansung Kim, and Adrian Hilton. Human-centric scene understanding from single view 360 video. In *International Conference on 3DVision (3DV)*, 2018.

[8] John C Gower. Generalized procrustes analysis. *Psychometrika*, 40(1):33–51, 1975.

[9] Marc Habermann, Weipeng Xu, Michael Zollhoefer, Gerard Pons-Moll, and Christian Theobalt. Deepcap: Monocular human performance capture using weak supervision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2020.

[10] Y. Huang, F. Bogo, C. Lassner, A. Kanazawa, P. V. Gehler, J. Romero, I. Akhter, and M. J. Black. Towards accurate marker-less human shape and pose estimation over time. In *2017 International Conference on 3D Vision (3DV)*, pages 421–430, Oct 2017.

[11] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1325–1339, jul 2014.

[12] Karim Iskakov, Egor Burkov, Victor Lempitsky, and Yury Malkov. Learnable triangulation of human pose. In *International Conference on Computer Vision (ICCV)*, 2019.

[13] H. Joo, H. Liu, L. Tan, L. Gui, B. Nabbe, I. Matthews, T. Kanade, S. Nobuhara, and Y. Sheikh. Panoptic studio: A massively multiview system for social motion capture. pages 3334–3342, 2015.

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

[15] Nikos Kolotouros, Georgios Pavlakos, and Kostas Daniilidis. Convolutional mesh regression for single-image human shape reconstruction. In *CVPR*, 2019.

[16] Y. Liu, C. Stoll, J. Gall, H. Seidel, and C. Theobalt. Markerless motion capture of interacting characters using multi-view image segmentation. In *CVPR 2011*, pages 1249–1256, June 2011.

[17] C. Malleson, A. Gilbert, M. Trumble, J. Collomosse, A. Hilton, and M. Volino. Real-time full-body motion capture from video and imus. In *2017 International Conference on 3D Vision (3DV)*, pages 449–457, Oct 2017.

[18] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. Vnect: Real-time 3d human pose estimation with a single rgb camera. volume 36, 2017.

[19] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[20] Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.

[21] Matthew Shere, Hansung Kim, and Adrian Hilton. 3d human pose estimation from multi person stereo 360 scenes. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019.

[22] J. Starck and A. Hilton. Surface capture for performance-based animation. *IEEE Computer Graphics and Applications*, 27(3):21–31, May 2007.

[23] Denis Tome, Chris Russell, and Lourdes Agapito. Lifting from the deep: Convolutional 3d pose estimation from a single image. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[24] Matthew Trumble, Andrew Gilbert, Adrian Hilton, and John Collomosse. Deep convolutional networks for marker-less human pose estimation from multiple views. In *Proceedings of the 13th European Conference on Visual Media Production (CVMP 2016)*, CVMP 2016, pages 6:1–6:9, New York, NY, USA, 2016. ACM.

[25] Matthew Trumble, Andrew Gilbert, Charles Malleson, Adrian Hilton, and John Collomosse. Total capture: 3d human pose estimation fusing video and inertial sensors. In *Proceedings of 28th British Machine Vision Conference*, pages 1–13, 2017.

[26] T. v. Marcard, G. Pons-Moll, and B. Rosenhahn. Human pose estimation from video and imus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(8):1533–1547, Aug 2016.

[27] Daniel Vlasic, Ilya Baran, Wojciech Matusik, and Jovan Popović. Articulated mesh animation from multi-view silhouettes. In *ACM SIGGRAPH 2008 Papers*, SIGGRAPH '08, pages 97:1–97:9, New York, NY, USA, 2008. ACM.

[28] Jingwei Xu, Zhenbo Yu, Bingbing Ni, Jiancheng Yang, Xiaokang Yang, and Wenjun Zhang. Deep kinematics analysis for monocular 3d human pose estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

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

[30] W. Xu, A. Chatterjee, M. Zollhöfer, H. Rhodin, P. Fua, H. Seidel, and C. Theobalt. Mo2cap2: Real-time mobile 3d motion capture with a cap-mounted fisheye camera. *IEEE Transactions on Visualization and Computer Graphics*, 25(5):2093–2101, May 2019.

[31] Zhe Zhang, Chunyu Wang, Wenhu Qin, and Wenjun Zeng. Fusing wearable imus with multi-view images for human pose estimation: A geometric approach. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.