MetaPose: Fast 3D Pose from Multiple Views without 3D Supervision

Ben Usman[1,2] Andrea Tagliasacchi[2,3] Kate Saenko[1,4] Avneesh Sud[2]

[1]Boston University [2]Google Research [3]Simon Fraser University [4]MIT-IBM Watson AI Lab

Abstract

In the era of deep learning, human pose estimation from multiple cameras with unknown calibration has received little attention to date. We show how to train a neural model to perform this task with high precision and minimal latency overhead. The proposed model takes into account joint location uncertainty due to occlusion from multiple views, and requires only 2D keypoint data for training. Our method outperforms both classical bundle adjustment and weakly-supervised monocular 3D baselines on the well-established Human3.6M dataset, as well as the more challenging in-the-wild Ski-Pose PTZ dataset.

1. Introduction

We tackle the problem of estimating 3D coordinates of human joints from RGB images captured using synchronized (potentially moving) cameras with unknown positions, orientations, and intrinsic parameters. We additionally assume having access to a training set with *only* 2D positions of joints labeled on captured images.

Historically, real-time capture of the human 3D pose has been undertaken only by large enterprises that could afford expensive specialized motion capture equipment [18]. In principle, if camera calibrations are available [3], human body joints can be triangulated directly from camera-space observations [26, 33]. One scenario in which camera calibration cannot easily be estimated is sports capture, in which close-ups of players are captured in front of *low-texture backgrounds*, with *wide-baseline, moving cameras*. Plain backgrounds preclude calibration via classical multi-camera SfM [21], as not sufficiently many feature correspondences can be detected across views; see Figure 1.

In this work, we propose a neural network to simultaneously predict 3D human and relative camera poses from multiple views; see Figure 1. Our approach uses *human body* joints as a source of information for camera calibration. As joints often become occluded, *uncertainty* must be carefully accounted for, to avoid bad calibration and consequent erroneous 3D pose predictions. As we assume a synchronized multi-camera setup at test-time, our algorithm should also be able to effectively *aggregate* information from different viewpoints. Finally, our approach supervised by 2D annotations *alone*, as ground-truth annotation of 3D data is unwieldy. As summarized in Figure 2, and detailed in what follows, none of the existing approaches fully satisfies these fundamental requirements.

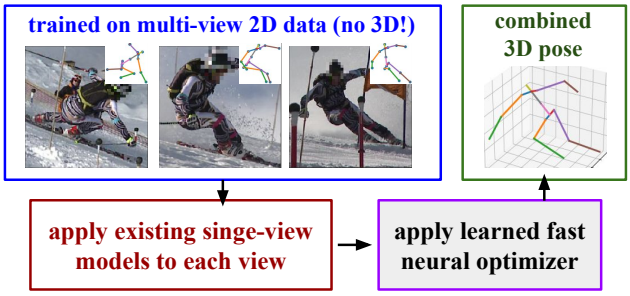

Figure 1. We show how to train a neural network that can aggregate outputs of multiple single-view methods, takes prediction uncertainty into consideration, has minimal latency overhead, and requires only 2D supervision for training. Our method mimics the structure of bundle-adjustment solvers, but using the joints of the human body to drive camera calibration, and by implementing a bundle-like solver with a simple feed-forward neural network.

Fully-supervised 3D pose estimation approaches yield the lowest estimation error, but make use of known 3D camera specification during either training [65] or both training and inference [26]. However, the prohibitively high cost of 3D joint annotation and full camera calibration in-the-wild makes it difficult to acquire large enough labeled datasets representative of specific environments [30, 53], therefore rendering supervised methods not applicable in this setup.

Monocular 3D methods [25, 37, 62] and 2D-to-3D lifting networks [10, 61], relax data constraints to enable 3D pose inference using just multi-view 2D data without calibration at train time. Unfortunately, at inference time, these methods can only be applied to a single view at a time, therefore unable to leverage cross-view information and uncertainty.

Classical SfM (structure from motion) approaches to 3D pose estimation [33] iteratively refine both the camera and

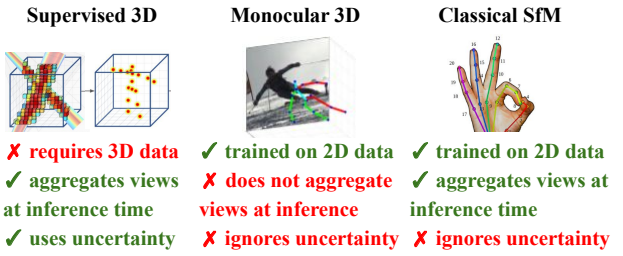

Figure 2. *Prior work* — Existing solutions either require 3D annotations [26], perform inference on a single view at a time [62], or ignore uncertainty in joint coordinates due to occlusions [33].

the 3D pose from noisy 2D observations. However, these methods are often much slower than their neural counterparts, since they have to perform several optimization steps during inference. Further, most of them do not consider uncertainty estimates, resulting in sub-par performance.

To overcome these limitation we propose *MetaPose*; see Figure 1. Our method for 3D pose estimation aggregates pose predictions and uncertainty estimates across multiple views, requires no 3D joint annotations or camera parameters at both train and inference time, and adds very little latency to the resulting pipeline.

Overall, we propose the feed-forward neural architecture that can accurately estimate the 3D human pose *and* the relative cameras configuration from multiple views, taking into account joint occlusions and prediction uncertainties, and uses only 2D joint annotations for training. We employ an off-the-shelf weakly-supervised 3D network to form an *initial* guess about the pose and the camera setup, and a neural *meta*-optimizer that iteratively *refines* this guess using 2D joint location probability heatmaps generated by an off-the-shelf 2D pose estimation network. This modular approach not only yields low estimation error, leading to state-of-the-art results on Human3.6M [24] and Ski-Pose PTZ [53], but also has low latency, as inference within our framework executes as a feed-forward neural network.

2. Related Work

In this section, we review only multi-view 3D human pose estimation methods, and refer our readers to the supplementary Sec. 7.4 for an extended review of learned neural optimizers and human body priors, and to Joo et al. [30] for a survey of 3D human pose estimation in the wild.

Full supervision. Supervised methods [11, 26, 60] yield the lowest 3D pose estimation errors on multi-view single person [24] and multi-person [6, 11, 29] datasets, but require precise camera calibration during both training and inference. Other approaches [65] use datasets with full 3D annotations and a large number of annotated cameras to train models that can adapt to novel camera setups in visually similar environments, relaxing camera calibration requirements. Martinez et al. [46] use pre-trained 2D pose networks [49] to take advantage of existing datasets with 2D pose annotations. Epipolar transformers [22] use only 2D keypoint supervision, but require camera calibration to incorporate 3D information in the 2D feature extractors.

Weak and self-supervision. Some approaches do not use full 3D GT poses for training. Many augment limited 3D annotations with 2D labels [32, 48, 66, 69]. Fitting-based methods [32, 38, 40, 66] jointly fit a statistical 3D human body model and 3D human pose to monocular images. Analysis-by-synthesis methods [27, 41, 52] learn to predict 3D human pose by estimating appearance in a novel view. Most related to our work are approaches that exploit the structure of multi-view image capture. EpipolarPose [37] uses epipolar geometry to obtain 3D pose estimates from multi-view 2D predictions, and subsequently uses them to directly supervise 3D pose regression. Iqbal et al. [25] proposes a weakly-supervised baseline to predict pixel coordinates of joints and their depth in each view and penalized the discrepancy between rigidly aligned predictions for different views during training. The self-supervised CanonPose [62] further advances state-of-the-art by decoupling 3D pose estimation in "canonical" frame. Drover et al. [15] learn a "dictionary" mapping 2D pose projections into corresponding realistic 3D poses, using a large collection of simulated 3D-to-2D projections. RepNet [61] and Chen et al. [10] train similar "2D-to-3D lifting networks" with more realistic data constraints. While all the aforementioned methods use multi-view consistency for *training*, they do not allow pose *inference* from multiple images.

Iterative refinement. Estimating camera and pose simultaneously is a long-standing problem in vision [54]. One of the more recent successful attempts is the work of Bridgeman et al. [8] that proposed an end-to-end network that refines the initial calibration guess using center points of multiple players in the field. In the absence of such external calibration signals, Takahashi et al. [57] performs bundle adjustment with bone length constraints, but do not report results on a public benchmark. AniPose [33] performs joint 3D pose and camera refinement using a modified version of the robust 3D registration algorithm of Zhou et al. [68]. Such methods ignore predicted uncertainty for faster inference, but robustly iteratively estimate outlier 2D observations and ignores them during refinement. In Section 5, we show that these classical approaches struggle in ill-defined settings, such as when we have a small number of cameras. More recently, SPIN [40], HUND [67] and Holopose [19] incorporate iterative pose refinement for *monocular* inputs, however, the refinement is tightly integrated into the pose estimation network. MetaPose effectively regularizes the *multi-view* pose estimation problem with a finite-capacity neural network resulting in both faster inference and higher precision than the classical refinement.

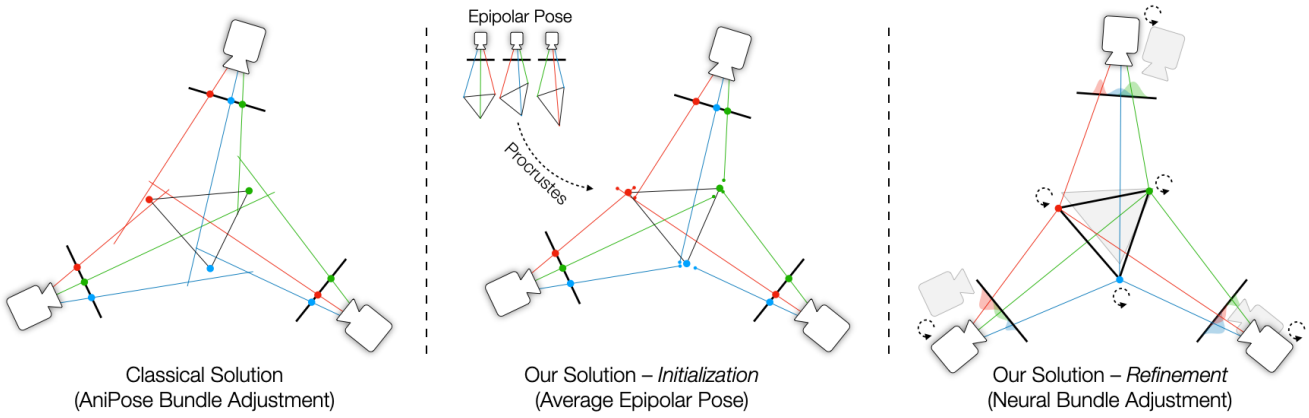

Figure 3. **Method** – We illustrate our method with a simple 2D example of regressing the 3D vertices of an equilateral triangle given multi-view observations. (left) AniPose [33] performs classical bundle adjustment to identify camera positions and 3D vertices that minimize reprojection error to 2D landmarks on the input images. Conversely, our technique *emulates* classical bundle adjustment in a "neural" fashion by a meta-optimizer: first (middle), the EpipolarPose [37] neural network obtains a per-frame 3D estimate of the joints, which we co-align via procrustes to obtain an initial guess for both camera parameters and joint locations; then (right), a neural network meta-optimizer performs bundle adjustment and refines both joints and cameras, using per-view keypoint localization heatmaps as input. Additional prior information, such as the fact that the triangle is equilateral, can be elegantly integrated in the meta-optimizer training.

3. Method

As illustrated in Figure 3, given a collection of $\{\mathcal{I}_c\}$ images, we seek to optimize, up to a global rotation, scale, and shift:

- $\mathbf{J} = \{\mathbf{j}_j \in \mathbb{R}^3\}_{j=1}^{J}$: the 3D coordinates of 3D body joints,
- $\mathbf{C} = \{\mathbf{c}_c \in \mathbb{R}^P\}_{c=1}^{C}$: parameters of each camera.

Having also observed:

- $\mathbf{H} = \{\mathbf{h}_c \in \mathbb{R}^{J\times H\times W}\}_{c=1}^{C}$: a set of 2D heatmaps of locations on images $\{\mathcal{I}_c\}$ captured using these cameras,

And assuming that, at training time, we are provided with:

- $\mathbf{K} = \{\mathbf{k}_{j,c}\}$: the ground truth 2D locations of the projection of joint $\mathbf{j}_j$ in camera $\mathbf{c}_c$.

Bayesian model. Formally, assuming that heatmaps depend on camera parameters and joint positions $\mathbf{J}$ only through 2D keypoint locations (i.e. $p(\mathbf{H}|\mathbf{K},\mathbf{J},\mathbf{C}) = p(\mathbf{H}|\mathbf{K})$), the joint distribution can be factorized as:

$$p(\mathbf{J},\mathbf{C},\mathbf{K},\mathbf{H}) = p(\mathbf{H}|\mathbf{K})\,p(\mathbf{K}|\mathbf{J},\mathbf{C})\,p(\mathbf{J})\,p(\mathbf{C}) \quad (1)$$

Joints and keypoints are assumed to be related by:

$$p(\mathbf{K}|\mathbf{J},\mathbf{C}) = \prod_{j,c}\delta(\mathbf{k}_{j,c} - \pi(\mathbf{j}_j,\mathbf{c}_c)) \quad (2)$$

where δ is the Dirac distribution, and $\pi(\mathbf{j},\mathbf{c})$ projects a joint $\mathbf{j}$ to the 2D coordinates in camera $\mathbf{c}$. We use a weak-projection camera model , hence, each camera is defined by a tuple of rotation matrix $\mathbf{R}$, pixel shift vector $\mathbf{t}$, and single scale parameter s, *i.e.* $\mathbf{c} = [\mathbf{R},\mathbf{t},s]$, and the projection operator is defined as $\pi(\mathbf{j},(\mathbf{R},\mathbf{t},s)) = s\cdot\mathbf{I}_{[0:1]}\cdot R\cdot\mathbf{j} + t$ where $\mathbf{I}_{[0:1]}$ is a truncated identity matrix that discards the third dimension of the multiplied vector. This choice of the camera model simplifies initialization of camera parameters from single-view 3D pose estimates (Section 3.2) and eliminates re-projection singularities (supplementary Sec. 7.6). In Section 5 we show experimentally what fraction of the final error comes from this choice of camera model.

Inference task. Our inference task is then to estimate the $\mathbf{J}$ and $\mathbf{C}$ from observed heatmaps $\mathbf{H}$. We first introduce a probabilistic bundle adjustment formulation to handle joint position uncertainty, then propose a regression model that models complex interactions between joint positions and observed heatmaps. The overall inference task can be framed as finding the maximum of the posterior probability of the pose and camera parameters given observed heatmaps, marginalized over possible keypoint locations:

$$\max_{\mathbf{J},\mathbf{C}} p(\mathbf{J},\mathbf{C}|\mathbf{H}) = \int \frac{p(\mathbf{k}|\mathbf{H})\,p(\mathbf{k}|\mathbf{J},\mathbf{C})\,p(\mathbf{J})\,p(\mathbf{C})}{p(\mathbf{k})}\,\mathrm{d}\mathbf{k} \quad (3)$$

where, assuming that no prior information over camera parameters, keypoint locations, and poses is given (i.e. constant $p(\mathbf{C})$, $p(\mathbf{K})$ and $p(\mathbf{J})$) and using (2) we get:

$$p(\mathbf{J},\mathbf{C}|\mathbf{H}) \propto \prod_{c,j} p(\mathbf{k}_{j,c} = \pi(\mathbf{j}_j,\mathbf{c}_c)|\mathbf{H}) \quad (4)$$

Further, assuming that each keypoint $\mathbf{k}_{c,j}$ is affected only by a corresponding heatmap $\mathbf{h}_{c,j}$, and more specifically that the conditional probability density is proportional to the corresponding value of the heatmap:

$$p(\mathbf{k}_{j,c}|\mathbf{H}) = p(\mathbf{k}_{j,c}|\mathbf{h}_{j,c}) \propto \mathbf{h}_{j,c}[\mathbf{k}_{j,c}] \quad (5)$$

we get a probabilistic bundle adjustment problem:

$$\max_{\mathbf{J},\mathbf{C}} \prod_{c,j} \mathbf{h}_{j,c}[\pi(\mathbf{j}_j, \mathbf{c}_c)] \tag{6}$$

As we will show in Section 5, better estimation *accuracy* with *faster* inference time can be archived if assume that each keypoint can be affected by any heatmap via the following functional relation up to a normally distributed residual:

$$p(\mathbf{K}|\mathbf{H}, \theta) = \mathcal{N}(\mathbf{K} \mid \pi(\mathbf{J}_\theta(\mathbf{H}), \mathbf{C}_\theta(\mathbf{H})), \mathbf{I}) \tag{7}$$

where $\mathbf{J}_\theta, \mathbf{C}_\theta$ are joint and camera regression models (e.g. neural networks) parameterized by an unknown parameter θ, and $\mathcal{N}$ is a multivariate normal density. Parameters of this model can be found via maximum likelihood estimation using observations from $p(\mathbf{K}, \mathbf{H})$ available during training

$$\theta_{\text{MLE}} = \arg\max_\theta \; p(\mathbf{H}, \mathbf{K}|\theta) = \arg\max_\theta p(\mathbf{K}|\mathbf{H}, \theta) \tag{8}$$

$$= \arg\min_\theta \; \mathbb{E}_{\mathbf{K},\mathbf{H}} \|\mathbf{K} - \pi(\mathbf{J}_\theta(\mathbf{H}), \mathbf{C}_\theta(\mathbf{H}))\|_2^2 \tag{9}$$

Then the test-time inference reduces to evaluation of the regression model at given heatmaps:

$$\arg\max_{\mathbf{J},\mathbf{C}} \; p(\mathbf{J}, \mathbf{C}|\mathbf{H}, \theta) = \mathbf{J}_\theta(\mathbf{H}), \mathbf{C}_\theta(\mathbf{H}) \tag{10}$$

Intuitively, the parametric objective enables complex interactions between all observed heatmaps and all predicted joint locations. The resulting model outperforms the probabilistic bundle adjustment both in terms of speed and accuracy, as we show in Section 5.

Solver. To solve the highly non-convex problem in (9), and to do so *efficiently*, we employ a modular *two stages* approach; see Figure 3:

Stage 1 (S1): Initialization – Section 3.2: We first acquire an *initial* guess $(\mathbf{J}_{\text{init}}, \mathbf{C}_{\text{init}})$ using single-view 3D pose estimates for the camera configuration and the 3D pose by applying rigid alignment to per-view 3D pose estimates obtained using a pre-trained weakly-supervised single-view 3D network, e.g. [37, 62]

Stage 2 (S2): Refinement – Section 3.3: We then train a neural network f_θ to predict a series of *refinement* steps for camera and pose, staring from the initial guess so to optimize (9).

Advantages. This approach has several key advantages:

1) it *primes* the refinement stage with a "good enough" guess to start from the correct basin of the highly non-convex pose likelihood objective given multi-view heatmaps;
2) it provides us with a *modular* framework, letting us swap pre-trained modules for single-view 2D and 3D *without* re-training the entire pipeline whenever a better approach becomes available;
3) the neural optimizer provides orders of magnitude *faster inference* than classical iterative refinement, and allows the entire framework to be written within the same coherent computation framework (i.e. neural networks vs. neural networks *plus* classical optimization).

3.1. Pre-processing

We assume that we have access to a 2D pose estimation model (e.g. PoseNet [50]) that produces 2D localization heatmaps $\mathbf{h}_{j,c}$ for each joint j from RGB image $\mathcal{I}_c$. We approximate each heatmap $\mathbf{h}_{j,c}$ with an M-component mixture of spherical Gaussians $\mathbf{g}_{j,c}$. This *compressed* format reduces the dimensionality of the input to the neural optimizer (Section 3.3). To fit parameters $\mathbf{g}_{j,c}$ of a mixture of spherical Gaussians to a localization 2D histogram $\mathbf{h}_{j,c}$, we treat the heatmap as a regular grid of 2D pixel coordinates weighted by corresponding probabilities, and apply weighted EM algorithm [17] to these weighted coordinates, as described in the supplementary Section 7.5.

Single-view pose estimation. To initialize camera parameters via rigid alignment (Section 3.2), we need a single-image 3D pose estimation model trained without 3D supervision (e.g. EpipolarPose [37]) that produces per-camera rough 3D pose estimates $\mathbf{Q} = \{\mathbf{q}_{c,j}\}$ given an image $\mathcal{I}_c$ from that camera. These single-image estimates $\mathbf{q}_{c,j}$ are assumed to be in the camera frame, meaning that first two spatial coordinates of $\mathbf{q}_{c,j}$ correspond to *pixel coordinates* of joint j on image $\mathcal{I}_c$, and the third coordinate corresponds to its single-image relative zero-mean *depth* estimate.

3.2. Initialization – Figure 4

The goal of this stage is to acquire an initial guess for the 3D pose and cameras $(\mathbf{J}_{\text{init}}, \mathbf{C}_{\text{init}})$ using single-view rough camera-frame 3D pose estimates $\mathbf{Q}$ made by a model trained without 3D supervision [37, 62]. We assume fixed initial parameters of the first camera

$$\mathbf{c}_0^{\text{init}} = (\mathbf{R}_0^{\text{init}}, \mathbf{t}_0^{\text{init}}, s_0^{\text{init}}) = (\mathbf{I}, \bar{0}, 1) \tag{11}$$

and define initial estimates of rotations, scales and translations of remaining cameras as solutions the following orthogonal rigid alignment problem:

$$\arg\min_{\mathbf{R}_c, \mathbf{t}_c, s_c} \sum_j \|\mathbf{q}_{c,j} - (s_c \cdot \mathbf{R}_c \cdot \mathbf{q}_{0,j} + \mathbf{I}_{[0:1]}^T \cdot \mathbf{t}_c)\|^2 \tag{12}$$

that can be solved using SVD of the outer product of mean-centered 3D poses [55]. The initial guess for the 3D pose $\mathbf{J}_{\text{init}}$ then is the average of single-view 3D pose predictions $\mathbf{Q}$ rigidly aligned back into the first camera frame by corresponding estimated optimal rotations, scales and shifts:

$$\mathbf{J}^{\text{init}} = \tfrac{1}{C} \sum_c \mathbf{R}_c^T \cdot (\mathbf{q}_c - \mathbf{I}_{[0:1]}^T \cdot \mathbf{t}_c))/s_c \tag{13}$$

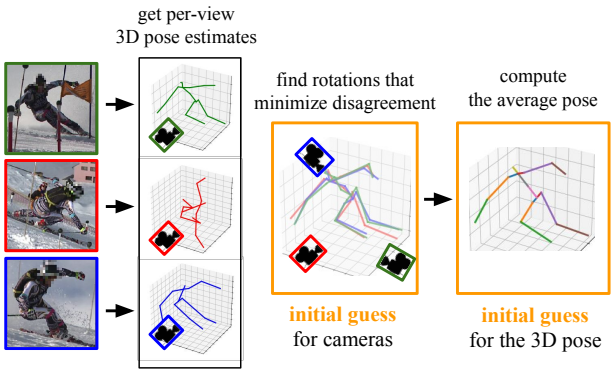

Figure 4. **Initialization** – We form an initial guess for the 3D pose *and* the cameras by taking the mean of rigid aligned 3D poses estimated from each RGB image using an external single-view weakly-supervised 3D pose estimation network [37, 62].

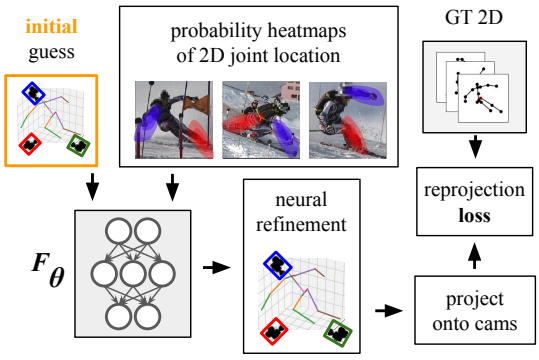

Figure 5. **Refinement** – We train a *neural optimizer* f_θ to predict iterative refinement that minimizes the reprojection error with the ground truth re-projection, using the current guess and joint heatmaps as an input. During inference, we *do not* need ground truth 2D projections.

3.3. Refinement – Figure 5

We train a neural network f_θ to predict a series of updates to 3D pose and camera estimates that leads to a refined estimate starting from the initialization from Section 3.2:

$$\mathbf{J}^{(i+1)} = \mathbf{J}^{(i)} + \mathrm{d}\mathbf{J}^{(i)}, \qquad \mathbf{J}^{(0)} = \mathbf{J}_{\text{init}} \tag{14}$$

$$\mathbf{C}^{(i+1)} = \mathbf{C}^{(i)} + \mathrm{d}\mathbf{C}^{(i)}, \qquad \mathbf{C}^{(0)} = \mathbf{C}_{\text{init}}. \tag{15}$$

To ensure that inferred camera parameters $\mathbf{C}$ stay valid under any update $\mathrm{d}\mathbf{C}$ predicted by a network, camera scale (always positive) is represented in log-scale, and camera rotation uses a continuous 6D representation [70], see Sec. 7.9.

At each refinement step $\mathrm{d}\mathbf{J}^{(i)}, \mathrm{d}\mathbf{C}^{(i)} = \mathcal{F}_\theta^{(i)}(\dots)$ the sub-network $\mathcal{F}_\theta^{(i)}$ of the overall network f_θ is provided with as much information as possible to perform a meaningful update towards the optimal solution:

- $(\mathbf{J}^{(i)}, \mathbf{C}^{(i)})$ – the current estimate to be refined;
- $\mathbf{G}{=}\{\mathbf{g}_{j,c}\}$ – a collection of Gaussian mixtures compactly representing the heatmaps density distributions;
- $\mathbf{K}^{(i)}{=}\{\mathbf{k}_{j,c}^{(i)} = \pi(\mathbf{j}_j^{(i)}, \mathbf{c}_c^{(i)})\}$ – the set of projections of each joint $\mathbf{j}^{(i)}$ into each camera frame $\mathbf{c}^{(i)}$;
- $\mathcal{L}(\mathbf{J}^{(i)}, \mathbf{C}^{(i)}|\mathbf{G})$ – the likelihood of the current estimate of joints given the heatmap mixture parameters.

These learnt updates seek to minimize the L2 distance between predicted and ground truth 2D coordinates of keypoints in each frame, mirroring the maximum likelihood objective (9) we defined earlier:

$$\arg\min_\theta \mathcal{L}_{\mathbf{k}}(\theta) = \sum_{(i)} \sum_{j,c} \|\mathbf{k}_{j,c}^{(i+1)} - \mathbf{k}_{j,c}^{\text{gt}}\|_2^2 \tag{16}$$

where, in practice, we train refinement steps $\mathcal{F}_\theta{}^{(i)}$ progressively, one after the other, as discussed in suppl. Sec. 7.7.

Architecture design. The architecture of $\mathcal{F}_\theta$ needs to be very carefully designed to respect the symmetries of the problem at hand. The inferred updates to $\mathbf{J}^{(i+1)}$ ought to be *invariant* to the order of cameras, while updates to $\mathbf{C}^{(i+1)}$ ought to be *permutation-equivariant* w.r.t. the current estimates of $\mathbf{C}^{(i)}$, rows of $\mathbf{K}^{(i)}$, and Gaussian mixtures $\mathbf{G}$. Formally, for any inputs and permutation of cameras σ:

$$\mathrm{d}\mathbf{J}, \mathrm{d}\mathbf{C} = \mathcal{F}_\theta(\mathbf{J}^{(i)}, \mathbf{C}^{(i)}, \mathbf{G}, \mathbf{K}^{(i)}, \mathcal{L}) \tag{17}$$

$$\mathrm{d}\mathbf{J}', \mathrm{d}\mathbf{C}' = \mathcal{F}_\theta(\mathbf{J}^{(i)}, \mathbf{C}_\sigma^{(i)}, \mathbf{G}_\sigma, \mathbf{K}_\sigma^{(i)}, \mathcal{L}) \tag{18}$$

we need to guarantee that $\mathrm{d}\mathbf{J} = \mathrm{d}\mathbf{J}'$ and $\mathrm{d}\mathbf{C} = \mathrm{d}\mathbf{C}'_\sigma$. To archive this, we concatenate view-invariant inputs $\mathbf{J}^{(i)}$ and $\mathcal{L}$ to each row of view-dependant inputs $\mathbf{C}^{(i)}, \mathbf{G}, \mathbf{K}^{(i)}$, pass them though a permutation-equivariant MLP [13, 31] with aggregation layers concatenating first and second moments of feature vectors back to these feature vectors, and apply mean aggregation and a non-permutation-equivariant MLP to get the final pose update, as illustrated in Figure 6.

Limitations. We assume a weak camera model, making our method less accurate on captures shot using wide-angle (short-focus) lenses. To achieve best performance, our method requires accurate 2D keypoint ground truth for training, but we also report performance without using GT keypoints during training (Table 2). We implicitly assume that the subject is completely in the frame and of comparable size (in pixels) across all views, and expect that manual re-weighting of different components of the reprojection loss (16) might be necessary otherwise.

3.3.1 Pose prior (i.e. "bone-length" experiment)

We illustrate the modularity of our solution by effortlessly injecting a *subject-specific* bone-legth prior into our meta-optimizer. Given two joints $\mathbf{j}_n$ and $\mathbf{j}_m$ connected in the

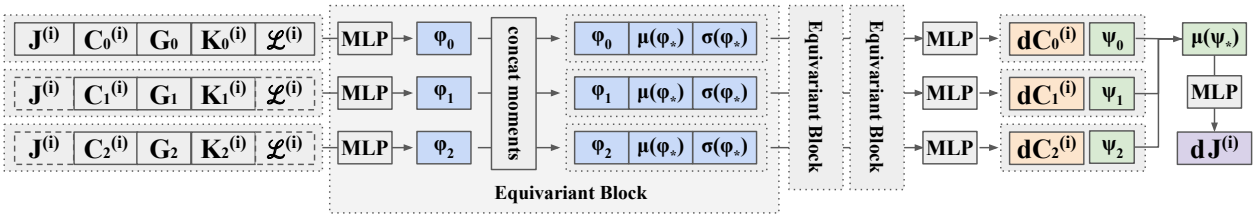

Figure 6. **Architecture** – In order for predicted updates to respect symmetries of the problem at hand, we copy and concatenate view-invariant inputs (current pose estimate, average heatmap likelihood - dashed line) to each row of view-specific inputs (current cameras and joint projections, heatmaps), pass them though a Permutation-Equivariant MLP Block shown above. To get permutation-invariant final pose update we additionally apply MLP to averaged output pose embeddings.

human skeleton $\mathcal{E}$ by an edge $e = (n, m)$, we define the bone length $b_e(\mathbf{J}) = \|\mathbf{j}_n - \mathbf{j}_m\|_2$. However, as our bundle adjustment is performed *up to scale* we ought to define *scale-invariant* bone lengths $b^N(\mathbf{J}) = b(\mathbf{J})/\hat{\mu}(b(\mathbf{J}))$ by expressing length of each bone relative to the average length of other bones $\hat{\mu}(b) = (\sum_e b_e)/|\mathcal{E}|$. If we assume that during training and inference we observe noisy normalized bone-lenghs vectors $\mathbf{B} = b^N(\mathbf{J}) + \varepsilon$, where $\varepsilon \sim \mathcal{N}(0, \sigma_b^2\mathbf{I})$. Then, the joint probability (1) becomes:

$$p(\mathbf{J}, \mathbf{C}, \mathbf{K}, \mathbf{H}, \mathbf{B}) = p(\mathbf{B}|\mathbf{J})\, p(\mathbf{H}|\mathbf{K})\, p(\mathbf{K}|\mathbf{J}, \mathbf{C})\, p(\mathbf{J})\, p(\mathbf{C})$$

and our parametric likelihood (7) becomes:

$$p(\mathbf{K}|\mathbf{H}, \mathbf{B}, \theta) \propto p(\mathbf{K}|\mathbf{H}, \theta) \cdot \mathcal{N}(b^N(\mathbf{J}_\theta(\mathbf{H}, \mathbf{B}))|\mathbf{B}, \sigma_b^2\mathbf{I})$$

and its parameters θ can be estimated equivalently to (9) via maximum over $p(\mathbf{K}, \mathbf{H}, \mathbf{B}|\,\theta)$ using observations from $p(\mathbf{K}, \mathbf{H}, \mathbf{B})$ available during training, effectively resulting in an additional loss term penalizing derivations of bone lengths of predicted poses from provided bone lengths:

$$\mathcal{L}_b(\theta) = \sum_{(i)} \left\| b^N(\mathbf{J}^{(i+1)}) - \mathbf{B} \right\|_2^2. \tag{19}$$

4. Experiments

In this section, we specify datasets and metrics we used to validate the performance of the proposed method and a set of baselines and ablation experiments we conducted to evaluate the improvement in error provided by each stage and each supervision signal.

Data. We evaluated our method on Human3.6M [24] dataset with four fixed cameras and a more challenging SkiPose-PTZ [53] dataset with six *moving* pan-tilt-zoom cameras. We used standard train-test evaluated protocol for H36M [26, 37] with subjects 1, 5, 6, 7, and 8 used for training, and 9 and 11 used for testing. We additionally pruned the H36M dataset by taking each 16-th frame from it, resulting in 24443 train and 8516 test examples, each example containing information from four cameras. We evaluated our method on the subset (1035 train / 230 test) of SkiPose [53] that was used in CanonPose [62] that excludes 280 examples with visibility obstructed by snow. In each dataset, we used the first 64 examples from the train split as a validation set. In supplementary Section 7.13, we show that among existing multi-view datasets, SkiPose is the only publicly available annotated multi-view dataset with moving cameras actively used in recent prior work.

Metrics. We report Procrustes aligned Mean Per Joint Position Error (PMPJPE) and Normalized Mean Per Joint Position Error (NMPJPE) that measure the L2-error of 3D joint estimates after applying the optimal rigid alignment (including scale) to the predicted 3D pose and the ground truth 3D pose (for NMPJPE), or only optimal shift and scale (for PMPJPE). We also report the total amount of time (Δt) it takes to perform 3D pose inference from multi-view RGB.

Baselines. On H36M we lower-bound the error with the state-of-the-art fully-supervised baseline of Iskakov et al. [26] that uses *ground truth* camera parameters to aggregate multi-view predictions during inference. We also compare the performance of our method to methods that use multi-view 2D supervision during training but only perform inference on a single view at a time: self-supervised EpipolarPose (EP) [37] and CanonPose (CP) [62], as well as the weakly supervised baselines of Iqbal et al. [25] and Rhodin et al. [53]. On SkiPose we compared our model with the only two baselines available in the literature: CanonPose [62] and Rhodin et al. [53]. We did not evaluate EpipolarPose on SkiPose because it requires fixed cameras to perform the initial self-supervised pseudo-labeling. We did not evaluate Iqbal et al. [25] on SkiPose because no code has been released to date and authors did not respond to a request to share code.

We also compared our method against the "classical" bundle adjustment initialized with ground truth extrinsic camera parameters of all cameras, and set fixed GT intrinsics, therefore putting it into *unrealistically favorable* conditions. We used the well-tested implementation of bundle adjustment in AniPose [33] that uses an adapted version of the 3D registration algorithm of Zhou et al. [68]. This approach takes point estimates of keypoint locations as an in-

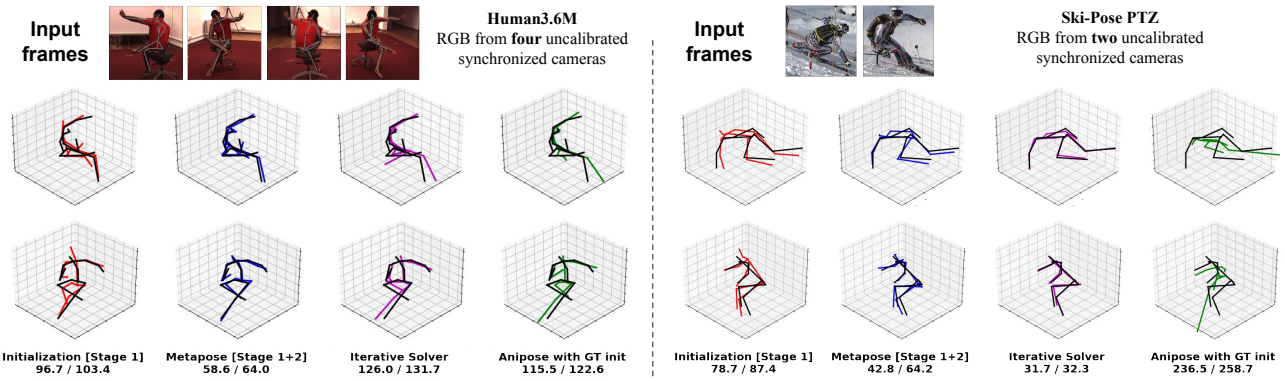

Figure 7. **Qualitative Results** – The top row shows input frames we used for pose estimation, overlayed with the GT pose (**black**). Two bottom rows show predictions made by evaluated methods on **H36M with four cameras** (left) and **SkiPose with two cameras** (right). We include predictions for Initialization (Stage 1), MetaPose (Stage 1+2), MetaPose with an Iterative Refinement (S1+IR), and AniPose initialized with GT. We also provide errors in the format: PMPJPE/NMPJPE. A video demonstration of qualitative results across both datasets can be found in the supplementary material or on the project website https://metapose.github.io/.

put (i.e. no uncertainty) and iteratively detects outliers and refines camera parameters and joint 3D positions using the second-order Trust Region Reflective algorithm [7, 9].

Architecture. For monocular 2D pose estimation, we used the stacked hourglass network [49] pre-trained on COCO pose dataset [20]. For monocular 3D estimation in Stage 1, we applied EpipolarPose [37] on Human3.6M and CanonPose [62] on SkiPosePTZ. We note that differences in the joint labeling schemes used by these monocular 3D methods and our evaluation set do not affect the quality of *camera* initialization we acquire via rigid alignment, as long as monocular 3D estimates for all views follow a consistent labeling scheme. Each neural optimizer step is trained separately, and stop gradient is applied to all inputs. We refer our readers to Section 7.7 in supplementary for a more detailed description of all components we used to train our neural optimizer and their reference performance.

5. Results – Table 1

The proposed method (MetaPose S1+S2) outperforms the classical bundle-adjustment baseline initialized with ground truth cameras (AniPose [33] w/ GT) by +40mm on H36M with four cameras, and +8mm on SkiPose with six cameras. With fewer cameras the performance gap increases further. MetaPose also outperforms semi-, weakly-, and self-supervised baselines reported in prior work [25, 37, 53, 62] by more then 10mm. We would like to re-iterate core advantages of the proposed method beyond its high performance, namely: ① that Stage 1 *primes* the neural optimizer with a good enough initialization that leads it to a good solution; ② that our solution is *modular* enabling swapping existing priming and pose estimation networks, as well as additional losses, and re-training only the neural optimizer; ③ that our method achieves **lower latency** then both classical and (GPU-accelerated) probabilistic bundle adjustment. We expand upon these and other related finding in the next subsection.

Method	PMPJPE↓		NMPJPE↓		Δt [s]
	4	2	4	2	
Isakov et al. [26]	20	-	-	-	-
AniPose [33] w/ GT	75	167	103	230	7.0
Rhodin et al. [53]	65	-	80	-	-
CanonPose [62]	53	-	82	-	-
EpipolarPose (EP) [37]	71	-	78	-	-
Iqbal et al. [25]	55	-	66	-	-
MetaPose (S1)	74	87	83	95	**0.2**
MetaPose (S1+S2)	**32**	**44**	**49**	**55**	0.3

Method	PMPJPE↓		NMPJPE↓		Δt [s]
	6	2	6	2	
AniPose [33] w/ GT	50	62	221	273	7.0
Rhodin et al. [53]	-	-	85	-	-
CanonPose (CP) [62]	90	-	128	-	-
MetaPose (S1)	81	86	140	144	**0.3**
MetaPose (S1+S2)	**42**	**50**	**53**	**59**	0.4

Table 1. **Quantitative comparison to prior work** – Performance of different methods with four and two cameras on **Human3.6M** (top) and six and two cameras **SkiPose-PTZ** (bottom), Procrustes and Normalized MPJPE in millimeters, inference time in seconds. See supplementary Table 4 for the breakdown of runtime performance and Table 6 for an extended comparison across all baselines, their supervision type, and with more decimal places.

Method	PMPJPE↓		NMPJPE↓		Δt [s]
	4	2	4	2	
MetaPose (S1+S2)	**32**	**44**	**49**	**55**	0.3
MetaPose (S1+IR)	43	53	66	75	2.0
MetaPose (S1+S2/SS)	39	50	56	63	0.3
MetaPose (S1+S2)	**32**	**44**	**49**	**55**	0.3
MetaPose (RND+S2)	36	51	52	64	0.3
MetaPose (S1+IR)	43	53	66	75	2.0
MetaPose (RND+IR)	200	385	265	444	2.0
MetaPose (GT+IR)	40	48	63	68	2.0
MetaPose (S1+S2)	32	**44**	49	**55**	0.3
MetaPose (S1+S2/MLP)	**30**	**44**	**47**	58	0.3
MetaPose (S1+S2)	32	44	**49**	55	0.3
MetaPose (S1+S2/BL)	**30**	**37**	50	**54**	0.3

Table 2. **Ablations on H36M**. Notation consistent with Table 1.

5.1. Ablations

Iterative refiner. We measured the speed gain we get from using the neural optimizer f_θ by replacing Stage 2 with a test-time GPU-accelerated gradient descent (Adam [35]) over the probabilistic bundle adjustment objective (6) with GMM-parameterized heatmaps. Section 1 in Table 2 shows that the proposed method (S1+S2) is up to *seven times* faster than the iterative refinement (S1+IR), and is at least 10mm more accurate. We also measured the contribution of keypoint supervision towards prediction accuracy of S2 compared to iterative refinement. To do that, we trained Stage 2 to minimize the same GMM-parameterized probabilistic bundle adjustment objective (6) instead of the re-projection loss (16). The resulting self-supervised model (S1+S2/SS) outperforms the iterative refinement, suggesting that the proposed architecture regularizes the pose estimation problem. Note that our self-supervised results also outperform prior work that uses weak- and self-supervision [25, 37, 62].

Random initialization. We measured the effect of replacing single-view pose estimates $\mathbf{q}_{c,j}$ used to initialize the pose and cameras in Stage 1 with random Gaussian noise. Section 2 in Table 2 shows that while the neural optimizer (RND+S2) is more resilient to poor initialization then the classical one (RND+IR), a good initialization is necessary to achieve the state-of-art performance (S1+S2). Moreover, marginally better results with GT initialization (GT+IR) show that the proposed initialization already brings the optimizer in the neighbourhood of the correct solution, and that further improvement in the quality of the initial guess will not provide significant gains in accuracy.

Non-equivariant network. We measured the effect of letting the model "memorize" the camera order by replacing equivariant blocks with MLPs that receive multi-view information as a single concatenated vector. The resulting model (S2/MLP) achieved marginally better performance on H36M and marginally worse performance on SkiPose (Table 5), likely due to fixed cameras positions in H36M and moving cameras in SkiPose.

Bone lengths. Training a model with an additional bone length prior (S1+S2/BL; see Sec. 3.3.1) improved PMPJPE with two cameras by 7mm. The two-camera setup is ill-conditioned, hence can better exploit this additional prior.

Inputs of neural optimizer. Unsurprisingly, among all inputs to the neural optimizer, heatmaps $\mathbf{H}$ contributed most to the final performance, but all inputs were necessary to achieve the best performance; see Table 3 in supplementary.

Further ablations (supplementary). The teacher-student loss proposed by Ma et al. [45] to draw predicted solutions into the basin of the right solution *hurts* the performance in all experiments (Table 8), suggesting that Stage 1 already provides good-enough initialization to start in the correct basin of the objective. We also ran the iterative refiner from ground truth initialization with re-projection losses with different camera models: results suggests that the weak camera model contributed to 10-15mm of error on H36M and no error on SkiPose; see Table 10. The performance of MetaPose on H36M starts to severely deteriorate at around 5% of the training data; see Table 11. Replacing GMM with a single Gaussian decreased the performance only in two-camera H36M setup by 4mm, and did not significantly influence the performance in other cases; see Table 12. We discuss sources of generalization error in supplementary Sec. 7.14.

6. Conclusions

In this paper, we propose a new modular approach to 3D pose estimation that requires only 2D supervision for training and significantly improves upon the state-of-the-art by fusing per-view outputs of singe-view modules with a simple view-equivariant neural network. Our modular approach not only enables practitioners to analyze and improve the performance of each component in *isolation*, and channel future improvements in respective sub-tasks into improved 3D pose estimation "for free", but also provides a common "bridge" that enables easy inter-operation of different schools of thought in 3D pose estimation – enriching both the "end-to-end neural world" with better model-based priors and improved interpretability, and the "iterative refinement world" with better-conditioned optimization problems, transfer-learning, and faster inference times. We provide a detailed ablation study dissecting different sources of the remaining error, suggesting that future progress in this task might come from the adoption of a full camera model, further improvements in 2D pose localization, better pose priors and incorporating temporal signals from video data.

References

[1] Jonas Adler and Ozan Öktem. Solving ill-posed inverse problems using iterative deep neural networks. *Inverse Problems*, 33(12):124007, 2017. 13

[2] Jonas Adler and Ozan Öktem. Learned primal-dual reconstruction. *IEEE transactions on medical imaging*, 37(6): 1322–1332, 2018. 13

[3] Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M. Seitz, and Richard Szeliski. Building rome in a day. *Commun. ACM*, 54(10):105–112, 2011. 1

[4] Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando De Freitas. Learning to learn by gradient descent by gradient descent. *Advances in neural information processing systems*, 29, 2016. 13

[5] Anurag Arnab, Carl Doersch, and Andrew Zisserman. Exploiting temporal context for 3d human pose estimation in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3395–3404, 2019. 13

[6] Vasileios Belagiannis, Sikandar Amin, Mykhaylo Andriluka, Bernt Schiele, Nassir Navab, and Slobodan Ilic. 3d pictorial structures for multiple human pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1669–1676, 2014. 2, 16

[7] Mary Branch, Thomas Coleman, and Yuying li. A subspace, interior, and conjugate gradient method for large-scale bound-constrained minimization problems. *SIAM Journal on Scientific Computing*, 21, 12 1999. doi: 10.1137/S1064827595289108. 7, 14

[8] Lewis Bridgeman, Marco Volino, Jean-Yves Guillemaut, and Adrian Hilton. Multi-person 3d pose estimation and tracking in sports. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 2, 16

[9] Richard H Byrd, Robert B Schnabel, and Gerald A Shultz. Approximate solution of the trust region problem by minimization over two-dimensional subspaces. *Mathematical programming*, 40(1):247–263, 1988. 7, 14

[10] Ching-Hang Chen, Ambrish Tyagi, Amit Agrawal, Dylan Drover, Stefan Stojanov, and James M Rehg. Unsupervised 3d pose estimation with geometric self-supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5714–5724, 2019. 1, 2

[11] Long Chen, Haizhou Ai, Rui Chen, Zijie Zhuang, and Shuang Liu. Cross-view tracking for multi-human 3d pose estimation at over 100 fps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[12] Ronald Clark, Michael Bloesch, Jan Czarnowski, Stefan Leutenegger, and Andrew J Davison. Ls-net: Learning to solve nonlinear least squares for monocular stereo. *arXiv preprint arXiv:1809.02966*, 2018. 13

[13] Congyue Deng, Or Litany, Yueqi Duan, Adrien Poulenard, Andrea Tagliasacchi, and Leonidas Guibas. Vector neurons: A general framework for so (3)-equivariant networks. *arXiv preprint arXiv:2104.12229*, 2021. 5

[14] Junting Dong, Qing Shuai, Yuanqing Zhang, Xian Liu, Xiaowei Zhou, and Hujun Bao. Motion capture from internet videos. In *European Conference on Computer Vision*, pages 210–227. Springer, 2020. 13

[15] Dylan Drover, Ching-Hang Chen, Amit Agrawal, Ambrish Tyagi, and Cong Phuoc Huynh. Can 3d pose be learned from 2d projections alone? In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018. 2

[16] John Flynn, Michael Broxton, Paul Debevec, Matthew DuVall, Graham Fyffe, Ryan Overbeck, Noah Snavely, and Richard Tucker. Deepview: View synthesis with learned gradient descent. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2367–2376, 2019. 14

[17] Daniel Frisch and Uwe D. Hanebeck. Gaussian mixture estimation from weighted samples, 2021. 4, 14

[18] Michael Gleicher. Animation from observation: Motion capture and motion editing. *SIGGRAPH*, 33(4):51–54, November 1999. 1

[19] Riza Alp Guler and Iasonas Kokkinos. Holopose: Holistic 3d human reconstruction in-the-wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[20] Rıza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7297–7306, 2018. 7, 14

[21] Nils Hasler, Bodo Rosenhahn, Thorsten Thormahlen, Michael Wand, Jürgen Gall, and Hans-Peter Seidel. Markerless motion capture with unsynchronized moving cameras. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 224–231. IEEE, 2009. 1

[22] Yihui He, Rui Yan, Katerina Fragkiadaki, and Shoou-I Yu. Epipolar transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7779–7788, 2020. 2

[23] Lior Fritz Imry Kissos, Matan Goldman, Omer Meir, Eduard Oks, and Mark Kliger. Beyond weak perspective for monocular 3d human pose estimation. 2020. 14

[24] 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, 2013. 2, 6, 16

[25] Umar Iqbal, Pavlo Molchanov, and Jan Kautz. Weakly-supervised 3d human pose learning via multi-view images in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5243–5252, 2020. 1, 2, 6, 7, 8, 17

[26] Karim Iskakov, Egor Burkov, Victor Lempitsky, and Yury Malkov. Learnable triangulation of human pose. In *International Conference on Computer Vision (ICCV)*, 2019. 1, 2, 6, 7, 16, 17

[27] Tomas Jakab, Ankush Gupta, Hakan Bilen, and Andrea Vedaldi. Self-supervised learning of interpretable keypoints from unlabelled videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[28] Hanbyul Joo, Tomas Simon, Xulong Li, Hao Liu, Lei Tan, Lin Gui, Sean Banerjee, Timothy Scott Godisart, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser Sheikh. Panoptic studio: A massively multiview system for social interaction capture. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. 16

[29] Hanbyul Joo, Tomas Simon, Xulong Li, Hao Liu, Lei Tan, Lin Gui, Sean Banerjee, Timothy Godisart, Bart Nabbe, Iain Matthews, and et al. Panoptic studio: A massively multiview system for social interaction capture. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(1):190–204, Jan 2019. 2

[30] Hanbyul Joo, Natalia Neverova, and Andrea Vedaldi. Exemplar fine-tuning for 3d human model fitting towards in-the-wild 3d human pose estimation, 2020. 1, 2

[31] Mor Joseph-Rivlin, Alon Zvirin, and Ron Kimmel. Momen (e) t: Flavor the moments in learning to classify shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019. 5

[32] 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. 2

[33] Pierre Karashchuk, Katie L Rupp, Evyn S Dickinson, Elischa Sanders, Eiman Azim, Bingni W Brunton, and John C Tuthill. Anipose: a toolkit for robust markerless 3d pose estimation. *BioRxiv*, 2020. 1, 2, 3, 6, 7, 17

[34] Vahid Kazemi and Josephine Sullivan. Using richer models for articulated pose estimation of footballers. In *BMVC*, 2012. 13, 16

[35] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 8, 14, 15

[36] Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. Self-normalizing neural networks. *Advances in neural information processing systems*, 30, 2017. 15

[37] Muhammed Kocabas, Salih Karagoz, and Emre Akbas. Self-supervised learning of 3d human pose using multi-view geometry. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1, 2, 3, 4, 5, 6, 7, 8, 14, 17

[38] Muhammed Kocabas, Nikos Athanasiou, and Michael J Black. Vibe: Video inference for human body pose and shape estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5253–5263, 2020. 2

[39] Muhammed Kocabas, Chun-Hao P Huang, Joachim Tesch, Lea Müller, Otmar Hilliges, and Michael J Black. Spec: Seeing people in the wild with an estimated camera. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11035–11045, 2021. 14

[40] Nikos Kolotouros, Georgios Pavlakos, Michael J Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *ICCV*, 2019. 2

[41] Jogendra Nath Kundu, Siddharth Seth, Varun Jampani, Mugalodi Rakesh, R. Venkatesh Babu, and Anirban Chakraborty. Self-supervised 3d human pose estimation via part guided novel image synthesis. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2020. 2

[42] Vincent Leroy, Philippe Weinzaepfel, Romain Brégier, Hadrien Combaluzier, and Grégory Rogez. Smply benchmarking 3d human pose estimation in the wild. In *2020 International Conference on 3D Vision (3DV)*, pages 301–310. IEEE, 2020. 13

[43] David Liebowitz and Stefan Carlsson. Uncalibrated motion capture exploiting articulated structure constraints. *International Journal of Computer Vision*, 51(3):171–187, 2003. 13

[44] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, October 2015. 13, 14

[45] Wei-Chiu Ma, Shenlong Wang, Jiayuan Gu, Sivabalan Manivasagam, Antonio Torralba, and Raquel Urtasun. Deep feedback inverse problem solver. In *ECCV*, 2020. 8, 13, 15, 16, 18

[46] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3d human pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2640–2649, 2017. 2

[47] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *3D Vision (3DV), 2017 Fifth International Conference on*. IEEE, 2017. doi: 10.1109/3dv.2017.00064. URL http://gvv.mpi-inf.mpg.de/3dhp_dataset. 16

[48] Rahul Mitra, Nitesh B. Gundavarapu, Abhishek Sharma, and Arjun Jain. Multiview-consistent semi-supervised learning for 3d human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[49] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European conference on computer vision*, pages 483–499. Springer, 2016. 2, 7, 14

[50] George Papandreou, Tyler Zhu, Liang-Chieh Chen, Spyros Gidaris, Jonathan Tompson, and Kevin Murphy. Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 269–286, 2018. 4

[51] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 13

[52] Helge Rhodin, Mathieu Salzmann, and Pascal Fua. Unsupervised geometry-aware representation for 3d human pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 2

[53] Helge Rhodin, Jorg Sporri, Isinsu Katircioglu, Victor Constantin, Frederic Meyer, Erich Mueller, Mathieu Salzmann, and Pascal Fua. Learning monocular 3d human pose estimation from multi-view images. *Conference On Computer Vision And Pattern Recognition (CVPR)*, 2018. 1, 2, 6, 7, 17

[54] Romer Rosales, Matheen Siddiqui, Jonathan Alon, and Stan Sclaroff. Estimating 3d body pose using uncalibrated cameras. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 1, pages I–I. IEEE, 2001. 2

[55] Peter H Schönemann. A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31(1):1–10, 1966. 4

[56] Jie Song, Xu Chen, and Otmar Hilliges. Human body model fitting by learned gradient descent. In *European Conference on Computer Vision*, pages 744–760. Springer, 2020. 14

[57] Kosuke Takahashi, Dan Mikami, Mariko Isogawa, and Hideaki Kimata. Human pose as calibration pattern; 3d human pose estimation with multiple unsynchronized and uncalibrated cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1775–1782, 2018. 2, 16

[58] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pages 402–419. Springer, 2020. 13

[59] Matt Trumble, Andrew Gilbert, Charles Malleson, Adrian Hilton, and John Collomosse. Total capture: 3d human pose estimation fusing video and inertial sensors. In *2017 British Machine Vision Conference (BMVC)*, 2017. 16

[60] Hanyue Tu, Chunyu Wang, and Wenjun Zeng. Voxelpose: Towards multi-camera 3d human pose estimation in wild environment, 2020. 2

[61] Bastian Wandt and Bodo Rosenhahn. Repnet: Weakly supervised training of an adversarial reprojection network for 3d human pose estimation. In *Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1, 2

[62] Bastian Wandt, Marco Rudolph, Petrissa Zell, Helge Rhodin, and Bodo Rosenhahn. CanonPose: Self-supervised monocular 3D human pose estimation in the wild. In *Computer Vision and Pattern Recognition (CVPR)*, June 2021. 1, 2, 4, 5, 6, 7, 8, 14, 17

[63] Zhe Wang, Daeyun Shin, and Charless C Fowlkes. Predicting camera viewpoint improves cross-dataset generalization for 3d human pose estimation. In *European Conference on Computer Vision*, pages 523–540. Springer, 2020. 16

[64] Donglai Xiang, Fabian Prada, Chenglei Wu, and Jessica Hodgins. Monoclothcap: Towards temporally coherent clothing capture from monocular rgb video. In *2020 International Conference on 3D Vision (3DV)*, pages 322–332. IEEE, 2020. 13

[65] Rongchang Xie, Chunyu Wang, and Yizhou Wang. Metafuse: A pre-trained fusion model for human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1, 2

[66] Andrei Zanfir, Eduard Gabriel Bazavan, Hongyi Xu, William T. Freeman, Rahul Sukthankar, and Cristian Sminchisescu. Weakly supervised 3d human pose and shape reconstruction with normalizing flows. In *Computer Vision – ECCV 2020*, pages 465–481, 2020. 2

[67] Andrei Zanfir, Eduard Gabriel Bazavan, Mihai Zanfir, William T Freeman, Rahul Sukthankar, and Cristian Sminchisescu. Neural descent for visual 3d human pose and shape. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14484–14493, 2021. 2

[68] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Fast global registration. In *European Conference on Computer Vision*, pages 766–782. Springer, 2016. 2, 6

[69] Xingyi Zhou, Qixing Huang, Xiao Sun, Xiangyang Xue, and Yichen Wei. Towards 3d human pose estimation in the wild: A weakly-supervised approach. *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 2

[70] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5745–5753, 2019. 5, 15