This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the version available on IEEE Xplore.

Detailed Full-Body Reconstructions of Moving People from Monocular RGB-D Sequences

Federica Bogo[1] Michael J. Black[1] Matthew Loper[1,2] Javier Romero[1]
[1]Max Planck Institute for Intelligent Systems, Tübingen, Germany
[2]Industrial Light & Magic, San Francisco, CA

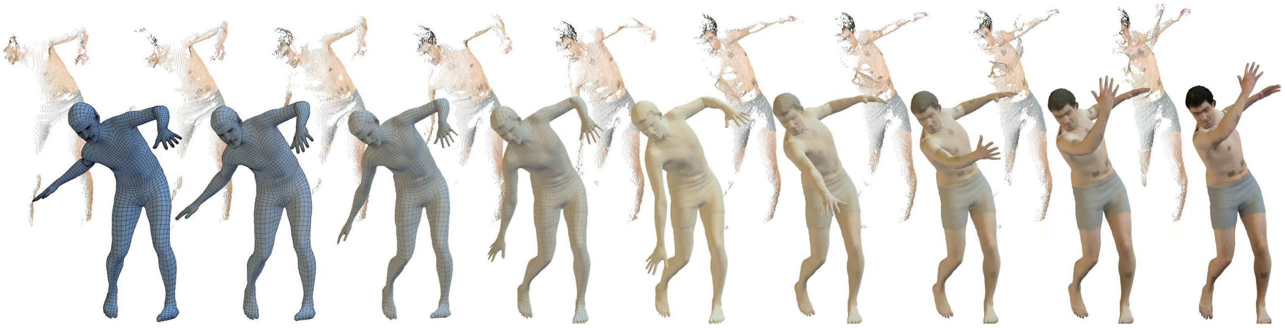

Figure 1: From a monocular RGB-D sequence (background), we estimate a low-dimensional parametric model of body shape (left), detailed 3D shape (middle), and a high-resolution texture map (right).

Abstract

We accurately estimate the 3D geometry and appearance of the human body from a monocular RGB-D sequence of a user moving freely in front of the sensor. Range data in each frame is first brought into alignment with a multi-resolution 3D body model in a coarse-to-fine process. The method then uses geometry and image texture over time to obtain accurate shape, pose, and appearance information despite unconstrained motion, partial views, varying resolution, occlusion, and soft tissue deformation. Our novel body model has variable shape detail, allowing it to capture faces with a high-resolution deformable head model and body shape with lower-resolution. Finally we combine range data from an entire sequence to estimate a high-resolution displacement map that captures fine shape details. We compare our recovered models with high-resolution scans from a professional system and with avatars created by a commercial product. We extract accurate 3D avatars from challenging motion sequences and even capture soft tissue dynamics.

1. Introduction

Accurate 3D body shape and appearance capture is useful for applications ranging from special effects, to fashion, to medicine. High-resolution scanners can capture human body shape and texture in great detail but these are bulky and expensive. In contrast, inexpensive RGB-D sensors are proliferating but are of much lower resolution. Scanning a full body from multiple partial views requires that the subject stands still or that the system precisely registers deforming point clouds captured from a non-rigid and articulated body. We propose a novel method that estimates body shape with the realism of a high-resolution body scanner by allowing a user to move freely in front of a single commodity RGB-D sensor.

Several previous methods have been proposed for 3D full-body scanning using range data [9, 10, 21, 23, 28, 30, 32, 34], but our method provides a significant increase in detail, realism, and ease of use as illustrated in Fig. 1. We work with RGB-D sequences from a single camera (Fig. 1, background). We exploit both depth and color data to combine information across an entire sequence to accurately estimate pose and shape from noisy sensor measurements. By allowing people to move relative to the sensor, we obtain data of varying spatial resolution. This lets us estimate a high-resolution detail for regions such as the face. By tracking the person we are able to cope with large portions of the body being outside the sensor's field of view.

To achieve this, we develop a new parametric 3D body model, called **Delta**, that is based on SCAPE [6] but contains several important innovations. First, we define a parametric shape model at multiple resolutions that enables the estimation of body shape and pose in a coarse-to-fine process. Second, we define a variable-detail shape model that models faces with higher detail; this is important for realistic avatars. Figure 1 (left) shows the high resolution body shape estimated from the sequence. Third, Delta combines a relatively-low polygon count mesh with a high-resolution displacement map to capture realistic shape details (Fig. 1 middle). Finally, Delta also includes a high-resolution texture map that is estimated from the sequence (Fig. 1 right).

Optimization is performed in three stages. Stage 1 estimates the body shape and pose in each frame by first fitting a low-resolution body and using this to initialize a higher-resolution model. Stage 2 uses the variable-detail shape model at the highest resolution and simultaneously estimates the texture map, a single body shape, and the pose at every frame to minimize an objective function containing both shape and appearance terms. We improve accuracy by solving for the shape and color of a textured avatar that, when projected into all the RGB images, minimizes an appearance error term. Stage 3 uses the estimated body shape and pose at every frame to register the sequence of point clouds to a common reference pose, creating a virtual high-resolution scan. From this we estimate the displacement map used in Fig. 1 (middle).

The method extracts more information from monocular RGB-D sequences than previous approaches with fewer constraints on the user's motion. The resulting model is realistic, detailed and textured, making it appropriate for many applications. We estimate models from a wide variety of challenging sequences and obtain reliable body pose estimates in situations where the Kinect pose estimation fails, *e.g*. when the person turns around or large parts of the body are out of the frame. We visually and quantitatively compare our models with scans acquired using a high-resolution scanning system and with avatars created using a commercial product. Moreover, we show how our approach captures the dynamics of full-body soft tissue motion.

2. Related Work

Shape reconstruction can be roughly divided into model-free and model-based approaches. Here we focus on methods that capture 3D body shape. Model-free methods register multiple depth frames, from different viewpoints, to obtain a complete scan. Model-based approaches fit the shape and pose parameters of a body model to multiple partial views. Many systems use multiple high-quality cameras and controlled lighting environments to capture the complex, dynamic, and detailed geometry of non-rigid human motion (*e.g*. [11, 13, 20, 31, 33]). The availability of consumer depth cameras, however, motivates more "lightweight" capture systems with fewer constraints. While some approaches employ multiple devices [12, 32, 35], we focus on methods that use a single RGB-D sensor.

Model-free systems like KinectFusion [18, 26] create detailed 3D reconstructions of rigid scenes, including high-quality appearance models [38], in real time from a moving RGB-D sensor. Several body scanning methods draw inspiration from KinectFusion [10, 21, 30, 36]. Such methods are not ideal for human body scanning because the user either must hold still while an operator moves the sensor, rotate in front of the device while trying to maintain a roughly rigid pose, or be rotated on a turntable. Partial data captured from different viewpoints is merged to produce a single mesh, using non-rigid registration to correct for small changes in shape between views.

Full-body scanning presents special challenges. If the object is small, like a hand or face, then it is easy for the sensor to see all of it (from one side) at once. For example, Li et al. [19] reconstruct non-rigid surface deformations from high-resolution monocular depth scans, using a smooth template as a geometric prior. Zollhöfer et al. [39] capture an initial template of small objects or body parts, acquired with a custom RGB-D camera, and then continuously reconstruct non-rigid motions by fitting the template to each frame in real time. Recently, [25] extends KinectFusion to capture dynamic 3D shapes including partial views of moving people. They only show slow and careful motions, do not use or capture appearance, and do not perform a quantitative analysis of the recovered shapes.

Less effort has been devoted to reconstruct the motion of full human bodies, including their soft tissue deformations. Several methods recover 3D deformable objects (including humans) from dynamic monocular sequences but test only on synthetic bodies [8, 22], or with high-quality scan systems for small volumes [8]. Helten et al. [16] estimate a personalized body shape model from two Kinect depth images and then use it to track the subject's pose in real time from a stream of depth images. The system fails when the subject does not face the camera or when parts of the body are outside the recording volume of the Kinect.

Model-based techniques [9, 34] fit pose and shape parameters to multiple frames in order to recover complete models from partial data. Perbet et al. [28] learn a mapping from depth images to initial body shape and pose parameters. They then refine a parametric model by fitting it to a single depth scan. Zhang et al. [37] register several Kinect scans of a subject in multiple poses and use these registrations to train a personalized body model, that is then fit to dynamic data. While model-based methods can handle a wider range of poses than model-free methods, their use of a low-dimensional shape space smooths out high-frequency geometry (*e.g*. subject-specific face details).

To capture full-body appearance from the Kinect, current methods average RGB information from different views [10] and blend texture between views [21, 30, 32, 37]. Existing methods capture only low-resolution texture. In contrast, we estimate a high-resolution texture map that combines images from multiple views, different poses, and

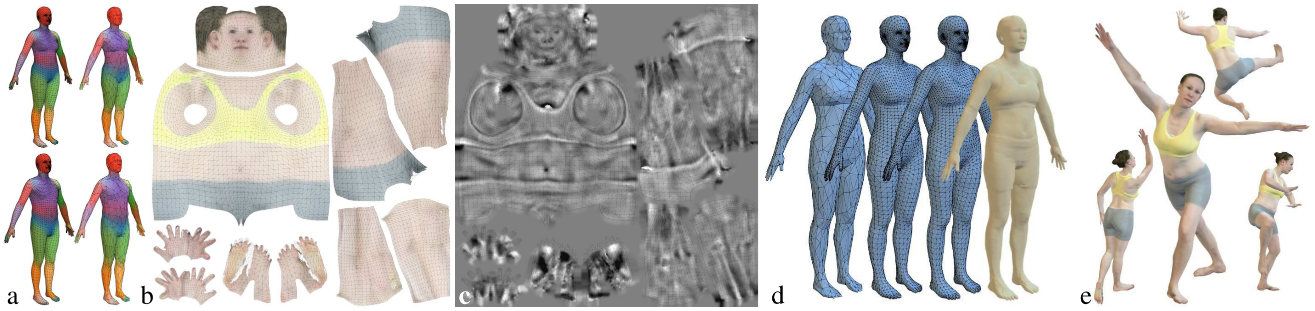

Figure 2: **Delta body model.** (a) Average male and female shapes at resolutions 1 and 2 (6890 and 863 vertices respectively). Color coding illustrates the segmentation into parts and the blend weights. (b) High-resolution texture map, U. (c) High-resolution displacement map, D. (d) Estimated body shape represented with 10 low-res shape basis vectors, 20 full-body high-res and 20 head basis vectors, personalized shape S, and S with the displacement map. (e) Textured model reposed.

varying distances from the sensor. We also use this texture to improve pose and shape estimation.

3. Body Model

We extend the BlendSCAPE body model introduced in [17], which is a version of the original SCAPE model [6]. We go beyond previous work to introduce a multi-resolution body model, variable detail in the shape space of the body parts, and a displacement map to capture fine shape detail. These changes allow us to capture realistic body shape while keeping optimization tractable by progressively adding detail. These improvements, together with a texture map as in [7], comprise our **Delta** body model (Fig. 2).

Multi-resolution mesh. We take an artist-designed triangulated template mesh and decimate it using Qslim [14] to construct a low-resolution version with a known mapping between low and high resolution. Let T_1^* and T_2^* be the high- and low-resolution templates with 6890 and 863 vertices respectively. The meshes have artist-designed segmentations and blend weights as illustrated in Fig. 2(a).

Like SCAPE, Delta factorizes the deformations that transform a template mesh, $T^*_{\{1,2\}}$, into a new body shape and pose. These pose- and shape-dependent deformations are represented by 3×3 deformation matrices. Each body part can undergo a rotation represented as a 3-element axis-angle. The rotations for the whole body are stacked into a 72-element pose vector $\boldsymbol{\theta}$, which is independent of mesh resolution. Pose-dependent deformations are modeled as in BlendSCAPE as a weighted linear function of the pose parameters. We train these linear functions from a database of approximately 1800 high-quality scans of 60 people that are all aligned (registered) to the template at the high resolution. The low-resolution pose-dependent deformations are trained with decimated meshes generated from the high-resolution model to ensure model compatibility.

SCAPE represents the body shape of different people in a low-dimensional deformation space. We register T_1^* to 3803 scans of subjects from the US and EU CAESAR datasets [29] and normalize the pose. We vectorize all the deformation matrices representing the shape of a subject. We compute the mean deformation, $\boldsymbol{\mu}_1$, across all subjects and use principal component analysis (PCA) to compute a low-dimensional linear subspace of deformations. Then a body shape at resolution 1 is a function of a vector of linear coefficients, $\boldsymbol{\beta}$:

$$S_1(\boldsymbol{\beta}) = \sum_{i=1}^{N} \beta_i B_{1,i} + \boldsymbol{\mu}_1, \tag{1}$$

where $B_{1,i}$ is the i^{th} principal component at resolution 1, β_i is a scalar coefficient, and $N << 3803$ is the dimensionality of the subspace. In Delta, we additionally learn a low-resolution shape subspace with directions $B_{2,i}$ and mean $\boldsymbol{\mu}_2$ trained to follow the direction of the components $B_{1,i}$, such that the shape coefficients are shared across resolutions. We learn separate shape spaces for men and women. Figure 2(a) shows the male and female mean shapes at both resolutions.

Given a set of shape deformations, $S(\boldsymbol{\beta})$, and a pose, $\boldsymbol{\theta}$, the Delta model produces a mesh, $M(S(\boldsymbol{\beta}), \boldsymbol{\theta})$, by applying the deformations to the triangles of the template, rotating the triangles of each part, applying pose-dependent deformations, and solving for a consistent mesh (see [6, 17]).

Variable detail model. We want to capture body shape as well as fine head detail since accurate reconstruction of the face is important for a realistic avatar. However, capturing fine face detail with a full-body model would require many principal components, $B_{1,i}$. Because estimating body and face shape from low-resolution RGB-D data is challenging, we want to keep the dimensionality low.

To address this, Delta uses a second, head-specific and overcomplete shape space. We simply build a second PCA model for head identity deformations (*i.e.* across subjects, not facial expressions). We do this by setting to zero, for each shape vector, all the elements corresponding to non-head triangles and then performing PCA. We then represent the body and head with different levels of shape fidelity in one linear equation:

$$S_1(\boldsymbol{\beta}) = \sum_{i=1}^{N} \beta_i B_{1,i} + \boldsymbol{\mu}_1 + \sum_{j=1}^{K} \beta_{N+j} H_{1,j} \qquad (2)$$

where $H_{1,j}$ are the principal components of head shape at resolution 1, $\beta_{N+1} \ldots \beta_{N+K}$ are the head shape coefficients. $H_{1,j}$ are vectors of the same size as $B_{1,i}$ but with zeros in all areas but the head. Note that the same idea could be applied just to face triangles or to other body parts.

In practice we only use the head shape model at resolution 1 with $N = K = 20$ components. Achieving comparable face fidelity with full-body components would require many more components (*i.e.* more than 40) and would make optimization more difficult. Furthermore, to capture the face detail using a full-body model, PCA would also capture body shape detail unnecessary for many applications.

Note that head/face shape is correlated with body shape and this is represented in the full-body shape basis, $B_{1,i}$. This is useful because we capture people moving around in front of the sensor and their face may be out of view or they may have their back to the sensor. In these scenarios, the full-body space helps the optimization keep track of the head. Then, when the face is in view, the head space allows us to capture more detail.

Resolution 2 only captures rough body shape and pose. Consequently we do not use a detailed head shape model and use only 10 principal components, $B_{2,i}, i = 1 \ldots 10$. This allows a *coarse-to-fine* fitting approach.

A low-dimensional shape space smooths out personalized shape details. To capture more detail, at the finest level, we allow the shape to deform away from the low-dimensional space to better fit scan data. We denote this personalized shape by S, dropping the dependency on the coefficients $\boldsymbol{\beta}$. Figure 2(d) summarizes the levels of detail.

Fine detail. For efficient rendering and inference, a template mesh should have a low polygon count. To capture realistic detail we use a high-resolution 2D texture map, U, and a displacement map, D (Fig. 2(b,c)). U is 2048×2048 texels while D is 512×512. Note that we define these only for the high-resolution model.

The final Delta model, $M(S, \boldsymbol{\theta}, U, D)$, deforms the body mesh, rotates the parts, applies pose-dependent deformations, and finally applies the displacement and texture maps.

4. Method

Input data. We use a Kinect One, which provides 512×424 depth images and 1920×1080 RGB images, at 30fps. We compute depth and RGB camera calibration parameters using a customized version of [3]. For each frame t, the sensor produces a depth image Z^t and a RGB image I^t. Given the camera calibration, we process Z^t to obtain a point cloud, P^t, with one 3D point per depth pixel. For each sequence, we acquire a background shot. We denote the background point cloud and color image by P_{bg} and I_{bg}, respectively.

Stage 1 – Pose and shape estimation in low-dimensional space. Stage 1 subdivides the initial sequence, of length n, into short intervals of $n' = 3$ consecutive frames and estimates the body shape and pose in each interval in a coarse-to-fine manner. Given an interval extending from frame t to frame $t' = t + n' - 1$, we solve for the pose parameters for each frame $\{\boldsymbol{\theta}^i\}_{i=t}^{t'}$ and the shape vector $\boldsymbol{\beta}^t$ minimizing:

$$\begin{aligned} \underset{\{\boldsymbol{\theta}^i\}_{i=t}^{t'}, \boldsymbol{\beta}^t}{\arg\min} \; & \lambda_S \sum_i E_S(M(S_j(\boldsymbol{\beta}^t), \boldsymbol{\theta}^i); P^i, P_{bg}) + \qquad (3) \\ & \lambda_{vel} E_{vel}(\{\boldsymbol{\theta}^i\}) + \lambda_\theta \sum_i E_\theta(\boldsymbol{\theta}^i) + \lambda_\beta E_\beta(\boldsymbol{\beta}^t) \end{aligned}$$

where we first set $j = 2$ and solve for the shape $S_2(\boldsymbol{\beta}^t)$, which is approximated with 10 principal components.

The geometric term E_S penalizes the distance in 3D between P^i and the surface of $M(S_j(\boldsymbol{\beta}^t), \boldsymbol{\theta}^i)$. We compute E_S over model surface points visible from the camera, considering also the background:

$$E_S(M(S_j(\boldsymbol{\beta}^t), \boldsymbol{\theta}^i); P^i, P_{bg}) = \sum_{\boldsymbol{v} \in P^i} \rho \left(\min_{\boldsymbol{x} \in V} ||\boldsymbol{v} - \boldsymbol{x}|| \right) \qquad (4)$$

where V is the set of visible points on the union of meshes $M(S_j(\boldsymbol{\beta}^t), \boldsymbol{\theta}^i)$ and P_{bg}, and ρ is a robust penalty function [15], useful when dealing with noisy Kinect data (*e.g.*, to ignore outliers at object boundaries). E_{vel} encourages smooth pose changes within the interval:

$$E_{vel}(\{\boldsymbol{\theta}^i\}) = \sum_{t<i<t'} ||2\boldsymbol{\theta}^i - \boldsymbol{\theta}^{i-1} - \boldsymbol{\theta}^{i+1}||^2. \qquad (5)$$

$E_\theta(\boldsymbol{\theta}^i)$ is a prior on pose. We compute the mean $\boldsymbol{\mu}_\theta$ and covariance Σ_θ of the poses from 39 subjects across more than 700 mocap sequences from the CMU dataset [4] and penalize the squared Mahalanobis distance between $\boldsymbol{\theta}^i$ and this distribution. The shape prior E_β penalizes the squared Mahalanobis distance between $\boldsymbol{\beta}^t$ and the distribution of CAESAR shapes with mean $\boldsymbol{\mu}_1$ and covariance Σ_β.

After solving for $\boldsymbol{\beta}^t$ and the poses for the low-resolution model, we use them as initialization and minimize (3) at resolution 1. See Fig. 3 (b) and (c).

We minimize (3) for each frame in the sequence, starting from the first frame and proceeding sequentially with overlapping intervals, initializing each interval with the values optimized for the previous one. This gives a body shape $\boldsymbol{\beta}^t$ and three estimates of the pose at nearly every frame. To output a single body shape from stage 1, we average the shape coefficients of the high-resolution models

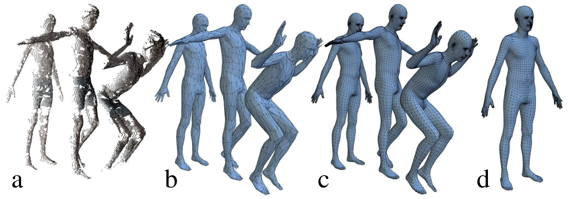

Figure 3: **Stage 1.** Three input point clouds (a) and the corresponding low- (b) and high-resolution (c) models obtained after optimizing objective (3). Also shown is the final output of stage 1 – a consistent high-resolution shape (d).

(Fig. 3). We similarly average the three estimated poses at each frame; this works well since the estimates tend to be very similar.

Stage 2 – Appearance-based refinement. Given the initial guess from above we now solve for a more detailed body shape that is no longer constrained to the PCA subspace. From here on we only work at resolution 1. Let S be the vector of body shape deformations we seek (no longer a function of $\boldsymbol{\beta}$). To compute S, we directly optimize vertex positions of a freely deforming mesh, which we call an "alignment", T^t. Alignments have the same topology as T_1^*. As in [17], they are regularized towards the model, but their vertices can deviate from it to better fit the data. We optimize T^t's vertices together with model parameters:

$$\begin{aligned}\underset{\{T^t\}_{t=1}^n,\Theta,S,U}{\arg\min} & \sum_t \lambda_S E_S(T^t; P^t, P_{bg}) + \\ & \sum_t (\lambda_U E_U(T^t, U; I^t, I_{bg}) + \lambda_\theta E_\theta(\boldsymbol{\theta}^t)) \\ & \sum_t \lambda_{cpl} E_{cpl}(T^t, S, \boldsymbol{\theta}^t) + \lambda_{sh} E_{sh}(S)\end{aligned} \quad (6)$$

where $\Theta = \{\boldsymbol{\theta}^t\}_{t=1}^n$, the geometric term E_S is as in Eq. (4) and we add a photometric term, E_U, plus a set of regularization terms.

E_U penalizes the discrepancy between the real image I^t and the rendered image $\widetilde{I}^t = \widetilde{I}(T^t, U; I_{bg})$, obtained by projecting T^t, textured with U, over the background image I_{bg} [7]. To mitigate problems due to shadowing we contrast-normalize I^t and $\widetilde{I}^t$ with a Ratio-of-Gaussians filter Γ:

$$E_U(T^t, U; I^t, I_{bg}) = ||\Gamma(I^t) - \Gamma(\widetilde{I}(T^t, U; I_{bg}))||_F^2 \quad (7)$$

where $||\cdot||_F$ is the Frobenius norm (cf. [7]).

E_{cpl} is a "coupling" term that encourages consistency between T^t and the posed mesh, $M(S, \boldsymbol{\theta}^t)$, with shape S:

$$E_{cpl}(T^t, S, \boldsymbol{\theta}^t) = \sum_{e \in V'} ||(AT^t)_e - (AM(S, \boldsymbol{\theta}^t))_e||_F^2 \quad (8)$$

where AT^t and $AM(S, \boldsymbol{\theta}^t)$ are the edge vectors of the triangles of T^t and $M(S, \boldsymbol{\theta}^t)$, respectively, e indexes edges and $V' = \text{vis}(AT^t)$ restricts the summation to visible edges.

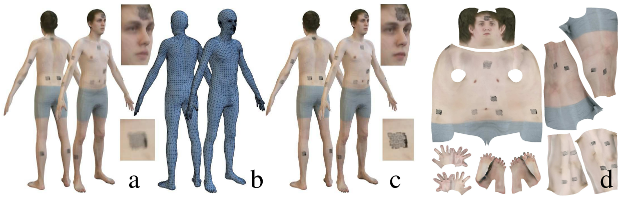

Figure 4: **Stage 2.** Output shape S (b) and texture map U (d). For comparison, S is rendered with U before optimization (a) and after optimization (c).

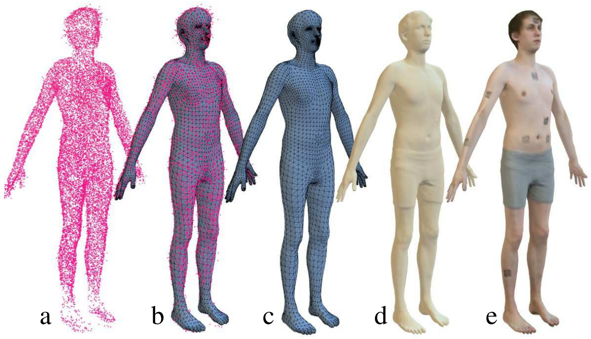

Figure 5: **Stage 3.** (a) Reposed point cloud $P^\cup$ (subsampled to 20000 points for visualization); (b) overlay $P^\cup$ / model $M(S, \boldsymbol{\theta}_{ref})$; (c) model after minimizing (10); (d) after applying the displacement D; (e) after applying D and U.

$E_{sh}(S) = \sum_{k,k'} ||S_k - S_{k'}||_F^2$ encourages smoothness of the shape deformations, where S_k and $S_{k'}$ are the deformation matrices for adjacent triangles k and k', and $||\cdot||_F$ is defined as in Eq. (7). $E_\theta(\boldsymbol{\theta}^t)$ is defined as above.

We use the shape and pose vectors obtained in stage 1 as initialization when minimizing (6). To initialize the appearance, U, we leverage shape and poses estimated in stage 1. As in [7], we blend (average) color from all frames on a per-texel basis, weighting each contribution according to the angle between surface normal and viewing direction.

This works well except for the face, which has a lot of high-frequency detail. Stage 1 may not produce precise head poses because the model resolution is low, leading to blurred face detail. To address this we use an average face per gender computed from a training set in the face region of U and minimize (6) over the head pose parameters only.

We then alternate between optimizing (6) with respect to Θ and $\{T^t\}_{t=1}^n$, S and U. For U we compute an average texture map given $\{T^t\}_{t=1}^n$ and Θ as described above. Note the alignments are allowed to deviate from S and thus can capture more pose-specific shape detail and produce a sharper texture map. Figure 4 shows the shape, S, and texture map, U, estimated in stage 2.

Stage 3 – High-resolution displacement mapping. Stage 3 uses the alignments from the previous stage to "repose" all the point clouds in the sequence, $\{P^t\}_{t=1}^n$, and to "fuse"

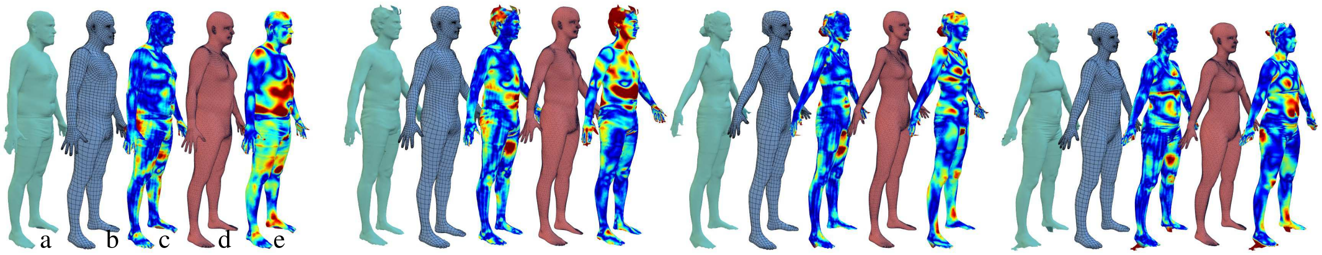

Figure 6: **Shape evaluation for Seq. 1.** Comparison between ground-truth scans (a) in green, our estimated models (b) in blue, and BodySnap models (d) in red for 4 subjects. Heat maps (c) and (e) beside each model show the scan-to-model registration error for our method and BodySnap, respectively (blue means 0mm, red means $\geq$ 1cm).

them in a common reference frame, thus obtaining a single high-resolution (but noisy) point cloud $P^{\cup}$ (Fig. 5). To do this, we define a mapping between mesh local surface geometry and the 3D world. Consider a point cloud P^t and the corresponding alignment T^t. We express each point $\boldsymbol{v}$ of P^t according to an orthonormal basis, defined at its closest point $\boldsymbol{x}$ on T^t. The basis vectors are the surface normal at $\boldsymbol{x}$ and two orthogonal vectors tangential to the surface at $\boldsymbol{x}$, chosen according to [24]. We denote by $\Delta(\boldsymbol{v}, T^t)$ the projection of $\boldsymbol{v}$ according to the basis defined by T^t, and by Δ^{-1} its inverse – from local surface geometry to 3D world.

As a common reference frame, we use the mesh, $M(S, \boldsymbol{\theta}_{ref})$, obtained using shape S from stage 2, posed according to a reference pose $\boldsymbol{\theta}_{ref}$ (note that the choice of $\boldsymbol{\theta}_{ref}$ is arbitrary). We compute $P^{\cup}$ (Fig. 5(a)) by reposing all point clouds in the sequence according to $\boldsymbol{\theta}_{ref}$:

$$P^{\cup} = \cup_t(\cup_{\boldsymbol{v} \in P^t} \Delta^{-1}(\Delta(\boldsymbol{v}, T^t), M(S, \boldsymbol{\theta}_{ref}))). \qquad (9)$$

The resolution of $P^{\cup}$ is far beyond the resolution of our body model or any of the individual point clouds. We now use $P^{\cup}$ to estimate a highly detailed body shape in two steps. First, we use it to refine shape S by minimizing:

$$\underset{T^{\cup}, S}{\arg\min}\, \lambda_S E_S(T^{\cup}; P^{\cup}) + \lambda_{cpl} E_{cpl}(T^{\cup}, S; \boldsymbol{\theta}_{ref}) \qquad (10)$$

where $T^{\cup}$ is an alignment for the point cloud $P^{\cup}$, and E_S, E_{cpl} are defined as above. With respect to (6), now we exploit all frames simultaneously during shape optimization.

The level of detail we recover from $P^{\cup}$ is bounded by our mesh resolution. In a final step, we transfer the high-resolution details of $P^{\cup}$ to our model computing a displacement map D. Let texel $\boldsymbol{y}$ in D be associated to the surface point $\boldsymbol{x}_y$ on the model. We compute the set of all points $\boldsymbol{p}$ in $P^{\cup}$ such that $\boldsymbol{x}_y = \arg\min_{\boldsymbol{x} \in M(S, \boldsymbol{\theta}_{ref})} ||\boldsymbol{x} - \boldsymbol{p}||^2$, and $\boldsymbol{p}$ is closer than 1cm to $\boldsymbol{x}_y$. After computing for each $\boldsymbol{p}$ its projection $\Delta(\boldsymbol{p}, M(S, \boldsymbol{\theta}_{ref}))$, we take the median along the normal at $\boldsymbol{x}_y$ and assign this to $\boldsymbol{y}$. Displacement maps substantially enhance high-frequency shape details (Fig. 5).

Optimization. We minimize objective (3) using Powell's dogleg method [27] with Gauss-Newton Hessian approximation. We compute function gradients using the Chumpy auto-differentiation package [2]. In stage 2, minimizing (6) with respect to $\{T^t\}_{t=1}^n$ and $\{\boldsymbol{\theta}^t\}_{t=1}^n$ corresponds to solving n independent registration subproblems. We use dogleg within the OpenDR framework [23], proceeding coarse to fine in image space (we increase the RGB resolution from a quarter to half and then to full resolution). We solve for the shape S via linear least squares. An analogous approach is used to minimize (10) iteratively with respect to $T^{\cup}$ and S. Note that we minimize (10) using 10^7 points sampled uniformly at random from $P^{\cup}$.

Pose and shape parameters in objective (3) are initialized to the mean pose in CMU and the mean shape in CAESAR, respectively. Since we use two different models for males and females, we manually select the subject gender. Afterwards, the entire pipeline runs automatically. Optimizing (3) over three frames takes 4-5 minutes on a desktop CPU; this is the only stage requiring sequential optimization. Optimizing an alignment in (6) takes 3 minutes; optimizing (10) and computing D requires approximately 10 minutes. See also [5] for more details.

5. Experimental Evaluation

Data Acquisition. We captured 13 subjects (6 female and 7 male) who gave informed written consent. Three subjects did not give permission to show their face; these are blurred. All subjects wore tight clothing; subjects with long hair wore it tied back.

From each subject we captured at least four different sequences. In Seq. 1, subjects followed a scanning protocol that involved rotating at different distances from the sensor, walking towards it, and bending down for a face closeup. Seq. 2 and 3 are dancing and an "arbitrary" motions (*e.g.* simulating interactive videogame play), respectively. Note that *we do not use any prior information about the motion sequence* during optimization. Sequence length ranged from approximately 150 to 1100 frames. Many sequences included fast motions; subjects significantly changed orientation and distance with respect to the camera. To compare with commercial software we captured an additional "static" sequence (Seq. 4) of 8 frames, with the subject

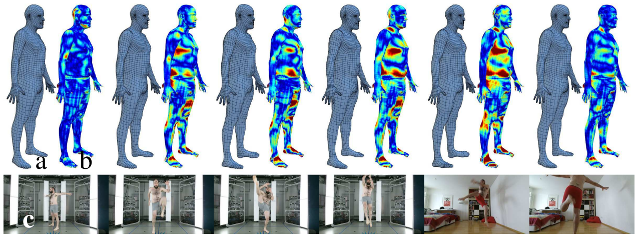

Figure 7: **Shape consistency.** Estimated shape (a) and corresponding registration error (b) (blue means 0mm, red means $\geq$ 1cm) for 6 sequences of the same subject. Images (c) show the corresponding motion.

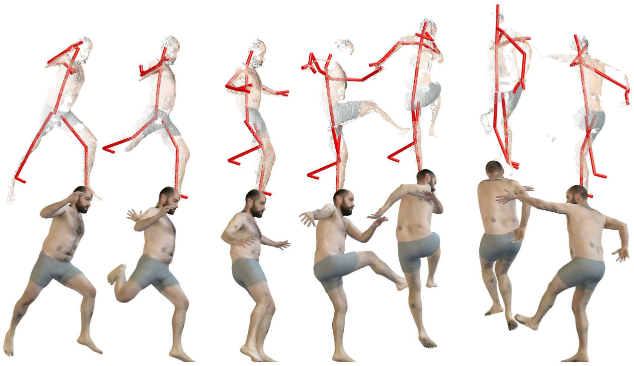

Figure 8: **Motion capture.** Poses estimated by Kinect (red skeleton, top) and by our approach (bottom).

rotating by roughly 45 degrees between frames. For one subject we captured an additional 9 challenging motion sequences. Most captures took place in a room with fairly even lighting (Fig. 11). For one subject we captured 5 additional sequences in a living room with uneven lighting (Fig. 7 and 11). For all sequences we captured a background RGB-D shot. See [5] for an overview of all sequences.

To enable the visual evaluation of our results, we applied a high-frequency pattern, using black body makeup and a woodcut stamp, on a dozen locations across the body (visible in Fig. 11). We used stamps on 11 subjects, and captured 2 subjects without the stamps to verify that the added texture was not necessary for the accuracy of our method.

Shape Estimation. To evaluate the accuracy of our estimated body shapes, we captured all subjects in a static A-pose (Fig. 6) with a full-body, 66-camera, active stereo system (**3dMD**, Atlanta, GA). The system outputs high-resolution scans (150000 vertices on average) that we take as "ground truth". We define the "registration error" of a shape S in terms of the scan-to-model distance; *i.e.* we compute the Euclidean distance between each scan vertex and its closest point on the surface of model $M(S, \boldsymbol{\theta}_{opt})$, where pose $\boldsymbol{\theta}_{opt}$ is adjusted to minimize this distance. Note that we evaluate S after optimizing objective (10) but before applying displacement maps D. We found visual improvement but no significant numerical improvement after applying D.

For 7 subjects, we compared our results against the models produced by **BodySnap** (Body Labs Inc., New York, NY) [1]. We ran it in "expert" mode, because it gave the best results. BodySnap reconstructs a complete 3D body model (with 43102 vertices) from 10 frames – the Seq. 4 protocol with 2 additional face closeups where the subject is 90cm from the device. Again we repose the result to match the ground-truth scan. BodySnap average error over the 7 subjects is 3.40mm, while our algorithm achieved an average of 2.40mm on the same 7 subjects performing Seq. 4.

These results are shown for 4 subjects in Fig. 6, which shows ground-truth scans, shape estimation and registration error both for our algorithm and BodySnap. Despite good overall accuracy, the latter captures fewer subject-specific shape details (*e.g.* see large red patches in the heat maps across the torso and on the head).

The average registration error of our algorithm for Seq. 1 computed over all 13 subjects is 2.54mm. We found little difference in accuracy between Seq. 1 results and those from more free-form motions (Seq. 2 was 2.82mm, Seq. 3 was 3.23). This suggests that a practical system could be designed around fun and engaging motions rather than a strict protocol. Errors from more restricted sequences like Seq. 4 are also comparable, 2.45mm, while they miss facial detail and cannot capture some occluded spots like the feet soles.

Figure 7 shows registration errors for one subject in 6 different sequences (2 captured in a living room). In all cases the average registration error is below 4.21mm – *i.e.* no more than 2mm worse than the error given by Seq. 4 (the left most in Fig. 7). Note that [21] and [37] report an average alignment error of about 3mm and 2.45mm, respectively, *on a mannequin.*

Motion Capture. Our approach is able to track motions where the standard Kinect pose estimation fails (Fig. 8). Tracking succeeds even in the presence of challenging poses, with large portions of the body either outside of the field of view or occluded.

Additionally, we capture the dynamics of soft tissue. Recall that we estimate alignments, $\{T^t\}_{t=1}^n$, in (6). These are constrained to be close to the model, $M(S, \boldsymbol{\theta}^t)$, but can deviate to match depth and color data in each frame. Figure 9 shows 6 such alignments; soft tissue deformation is visible on the chest and stomach. We believe that dynamic soft tissue capture with Kinect is new. Note that this particular sequence is special in the sense that we are using the model extracted from Seq. 1 instead of estimating it from scratch, as we do in the rest of the examples in this paper.

Appearance and Fine Geometric Detail. Figure 10 shows textured models recovered for all subjects using Seq. 1, compared with ground-truth scans. The 3dMD scanner captures texture with 22 synchronized color cameras

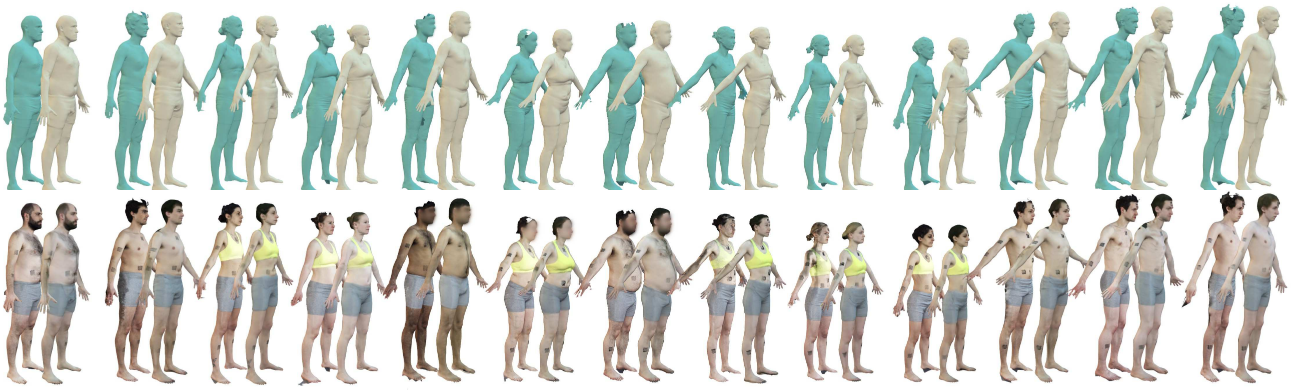

Figure 10: **High-resolution models.** Comparison between 3dMD scans (green, on the left) and our models after displacement mapping (beige, on the right) in terms of shape (top row) and texture (bottom row).

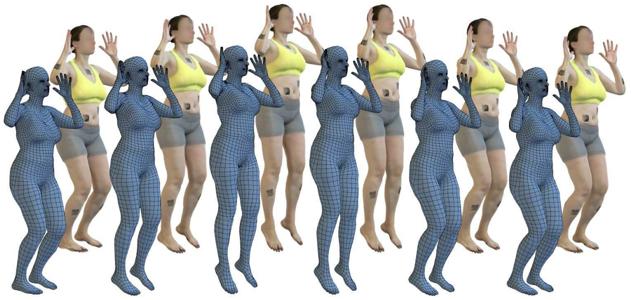

Figure 9: **Soft tissue deformations.** Shown with and without texture (better seen in the video [5]). Note the shape deformations in areas like the chest and stomach.

and LED light panels that produce smooth illumination. Despite the variety in subject appearance (skin tone, facial hair, etc.), our method recovers realistic texture maps.

Figure 11 compares real Kinect images with synthetic images rendered from our textured models over the background RGB shot. Note that, for each image, we use appearance models estimated from the sequence itself. The synthesized results are difficult to distinguish from the real data even in challenging sequences. In many cases, fine details (like the stamp pattern, with texture elements of the order of 2mm) are reconstructed. Note that sharp texture maps are reconstructed even when stamps are not used (Fig. 10).

6. Conclusion

We have presented a novel approach to estimate high-resolution 3D shape and appearance of the human body from monocular RGB-D sequences acquired with a single sensor. Our approach leverages a new parametric, multi-resolution body model, Delta, that combines a low-dimensional shape space for the full body with a second, head-specific, shape space. The model enables the estimation of body shape and pose in a coarse-to-fine manner.

In future work, we plan to extend Delta to also capture more detailed hands and feet. Additionally, we could incorporate a non-rigid face model to capture varying facial expressions. It would also be interesting to reconstruct transient per-frame high-frequency details (as in [19, 39]). Currently, our texture estimate simply blends contributions from different RGB frames. By formulating camera blur and pixel discretization in the appearance objective function, we might be able to extend super-resolution methods to non-rigid bodies. Finally, our method is fully generative. We could likely improve inference speed by using a fast discriminative method (*e.g.* the Kinect's own pose estimate) for initialization.

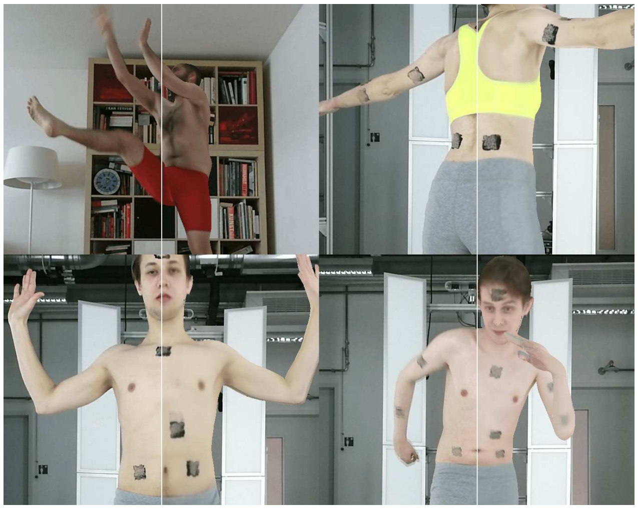

Figure 11: **Appearance estimation.** In each frame we show a real Kinect image (left half) and the corresponding synthetic image (right half) rendered from our model.

Acknowledgments. We thank R. Enficiaud, E. Holderness and A. Keller for their help with data acquisition. Conflict-of-interest disclosure: MJB is a co-founder, shareholder, and board member of Body Labs Inc., which commercializes body shape technology.

References

[1] http://bodysnapapp.com. 7

[2] http://chumpy.org. 6

[3] https://github.com/bodylabs/monocle. 4

[4] http://mocap.cs.cmu.edu. 4

[5] https://ps.is.tuebingen.mpg.de/research_projects/bodies-from-kinect. 6, 7, 8

[6] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis. SCAPE: Shape Completion and Animation of PEople. *ACM Trans. on Graph.*, 24(3):408–416, 2005. 1, 3

[7] F. Bogo, J. Romero, M. Loper, and M. J. Black. FAUST: Dataset and evaluation for 3D mesh registration. In *CVPR*, pages 3794–3801, 2014. 3, 5

[8] W. Chang and M. Zwicker. Global registration of dynamic range scans for articulated model reconstruction. *ACM Trans. on Graph.*, 30(3):187:1–187:9, 2011. 2

[9] Y. Chen, Z. Liu, and Z. Zhang. Tensor-based human body modeling. In *CVPR*, pages 105–112, 2013. 1, 2

[10] Y. Cui, W. Chang, T. Nöll, and D. Stricker. KinectAvatar: Fully automatic body capture using a single Kinect. In *ACCV Workshops*, pages 133–147, 2012. 1, 2

[11] E. De Aguiar, C. Stoll, C. Theobalt, N. Ahmed, H.-P. Seidel, and S. Thrun. Performance capture from sparse multi-view video. *ACM Trans. on Graph.*, 27(3):98:1–98:10, 2008. 2

[12] M. Dou, H. Fuchs, and J. Frahm. Scanning and tracking dynamic objects with commodity depth cameras. In *ISMAR*, pages 99–106, 2013. 2

[13] J. Gall, C. Stoll, E. De Aguiar, C. Theobalt, B. Rosenhahn, and H.-P. Seidel. Motion capture using joint skeleton tracking and surface estimation. In *CVPR*, pages 1746–1753, 2009. 2

[14] M. Garland and P. Heckbert. Surface simplification using quadric error metrics. In *SIGGRAPH*, pages 209–216, 1997. 3

[15] S. Geman and D. McClure. Statistical methods for tomographic image reconstruction. *Bulletin of the International Statistical Institute*, 52(4):5–21, 1987. 4

[16] T. Helten, A. Baak, G. Bharaj, M. Müller, H.-P. Seidel, and C. Theobalt. Personalization and evaluation of a real-time depth-based full body tracker. In *3DV*, pages 279–286, 2013. 2

[17] D. Hirshberg, M. Loper, E. Rachlin, and M. J. Black. Coregistration: Simultaneous alignment and modeling of articulated 3D shape. In *ECCV*, pages 242–255, 2012. 3, 5

[18] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, and A. Fitzgibbon. KinectFusion: Real-time 3D reconstruction and interaction using a moving depth camera. In *UIST*, pages 559–568, 2011. 2

[19] H. Li, B. Adams, L. Guibas, and M. Pauly. Robust single-view geometry and motion reconstruction. *ACM Trans. on Graph.*, 28(5):175:1–175:10, 2009. 2, 8

[20] H. Li, L. Luo, D. Vlasic, P. Peers, J. Popovic, M. Pauly, and S. Rusinkiewicz. Temporally coherent completion of dynamic shapes. *ACM Trans. on Graph.*, 31(1):2:1–2:11, 2012. 2

[21] H. Li, E. Vouga, A. Gudym, L. Luo, J. Barron, and G. Gusev. 3D self-portraits. *ACM Trans. on Graph.*, 32(6):187:1–187:9, 2013. 1, 2, 7

[22] M. Liao, Q. Zhang, H. Wang, R. Yang, and M. Gong. Modeling deformable objects from a single depth camera. In *ICCV*, pages 167–174, 2009. 2

[23] M. Loper and M. J. Black. OpenDR: An approximate differentiable renderer. In *ECCV*, pages 154–169, 2014. 1, 6

[24] M. Mikkelsen. Simulation of wrinkled surfaces revisited. Master's thesis, University of Copenhagen, 2008. 6

[25] R. Newcombe, D. Fox, and S. Seitz. DynamicFusion: Reconstruction and tracking of non-rigid scenes in real-time. In *CVPR*, pages 343–352, 2015. 2

[26] R. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon. KinectFusion: Real-time dense surface mapping and tracking. In *ISMAR*, pages 127–136, 2011. 2

[27] J. Nocedal and S. Wright. *Numerical optimization*. Springer, 2006. 6

[28] F. Perbet, S. Johnson, M.-T. Pham, and B. Stenger. Human body shape estimation using a multi-resolution manifold forest. In *CVPR*, pages 668–675, 2014. 1, 2

[29] K. Robinette, H. Daanen, and E. Paquet. The CAESAR project: A 3-D surface anthropometry survey. In *Int. Conf. on 3D Digital Imag. and Model.*, pages 380–386, 1999. 3

[30] A. Shapiro, A. Feng, R. Wang, H. Li, M. Bolas, G. Medioni, and E. Suma. Rapid avatar capture and simulation using commodity depth sensors. *Computer Animation and Virtual Worlds*, 25(3–4):201–211, 2014. 1, 2

[31] J. Starck and A. Hilton. Model-based multiple view reconstruction of people. In *ICCV*, pages 915–922, 2003. 2

[32] J. Tong, J. Zhou, L. Liu, Z. Pan, and H. Yan. Scanning 3D full human bodies using Kinects. *IEEE Trans. on Visualization and Computer Graphics*, 18(4):643–650, 2012. 1, 2

[33] D. Vlasic, P. Peers, I. Baran, P. Debevec, J. Popovic, S. Rusinkiewicz, and W. Matusik. Dynamic shape capture using multi-view photometric stereo. *ACM Trans. on Graph.*, 28(5):174:1–174:11, 2009. 2

[34] A. Weiss, D. Hirshberg, and M. J. Black. Home 3D body scans from noisy image and range data. In *ICCV*, pages 1951–1958, 2011. 1, 2

[35] G. Ye, Y. Liu, N. Hasler, X. Ji, Q. Dai, and C. Theobalt. Performance capture of interacting characters with handheld Kinects. In *ECCV*, pages 828–841, 2012. 2

[36] M. Zeng, J. Zheng, and X. Liu. Templateless quasi-rigid shape modeling with implicit loop-closure. In *CVPR*, pages 145–152, 2013. 2

[37] Q. Zhang, B. Fu, M. Ye, and R. Yang. Quality dynamic human body modeling using a single low-cost depth camera. In *CVPR*, pages 676–683, 2014. 2, 7

[38] Q. Zhou and V. Koltun. Color map optimization for 3D reconstruction with consumer depth cameras. *ACM Trans. on Graph.*, 33(4):155:1–155:10, 2014. 2

[39] M. Zollhöfer, M. Nießner, S. Izadi, C. Rehmann, C. Zach, M. Fisher, C. Wu, A. Fitzgibbon, C. Loop, C. Theobalt, and M. Stamminger. Real-time non-rigid reconstruction using an RGB-D camera. *ACM Trans. on Graph.*, 33(4):156:1–156:12, 2014. 2, 8