BLSM: A Bone-Level Skinned Model of the Human Mesh

Haoyang Wang[1,2], Riza Alp Güler[1,2], Iasonas Kokkinos[1],
George Papandreou[1], and Stefanos Zafeiriou[1,2]

[1] Ariel AI, London, UK
{hwang, alpguler, iasonas, gpapan, szafeiriou}@arielai.com
[2] Imperial College London
{haoyang.wang15, r.guler, s.zafeiriou}@imperial.ac.uk

Abstract. We introduce BLSM, a bone-level skinned model of the human body mesh where bone scales are set prior to template synthesis, rather than the common, inverse practice. BLSM first sets bone lengths and joint angles to specify the skeleton, then specifies identity-specific surface variation, and finally bundles them together through linear blend skinning. We design these steps by constraining the joint angles to respect the kinematic constraints of the human body and by using accurate mesh convolution-based networks to capture identity-specific surface variation. We provide quantitative results on the problem of reconstructing a collection of 3D human scans, and show that we obtain improvements in reconstruction accuracy when comparing to a SMPL-type baseline. Our decoupled bone and shape representation also allows for out-of-box integration with standard graphics packages like Unity, facilitating full-body AR effects and image-driven character animation. Additional results and demos are available from the project webpage: http://arielai.com/blsm

Keywords: 3D human body modelling, Graph convolutional networks

1 Introduction

Mesh-level representations of the human body form a bridge between computer graphics and computer vision, facilitating a broad array of applications in motion capture, monocular 3D reconstruction, human synthesis, character animation, and augmented reality. The articulated human body deformations can be captured by rigged modelling where a skeleton animates a template shape; this is used in all graphics packages for human modelling and animation, and also in state-of-the-art statistical models such as SMPL or SCAPE [6, 23].

Our work aims at increasing the accuracy of data-driven rigged mesh representations. Our major contribution consists in revisiting the template synthesis process prior to rigging. Current models, such as SMPL, first synthesize the template mesh in a canonical pose through an expansion on a linear basis. The skeleton joints are then estimated post-hoc by regressing from the synthesized mesh to the joints. Our approach instead disentangles bone length variability from acquired body traits dependent e.g. on exercise or dietary habits.

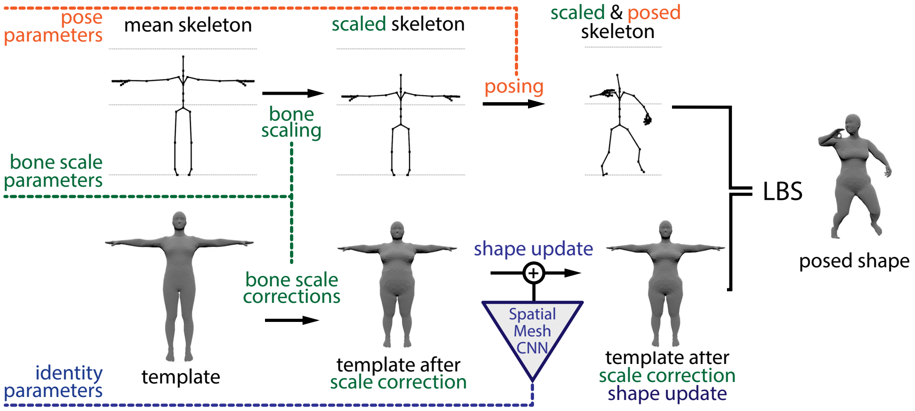

Fig. 1: Overview of our Bone-Level Skinned Model (BLSM): The top row shows skeleton synthesis: starting from a canonical, bind pose, we first scale the bone lengths and then apply an articulated transformation. The bottom row shows shape control: the canonical mesh template is affected by the bone scaling transform through Bone-Scaling Blend Shapes, and then further updated to capture identity-specific shape variation. The skeleton drives the deformation of the resulting template through Linear Blend Skinning, yielding the posed shape.

Based on this, we first model bone length-driven mesh variability in isolation, and then combine it with identity-specific updates to represent the full distribution of bodies. As we show experimentally, this disentangled representation results in more compact models, allowing us to obtain highly-accurate reconstructions with a low parameter count.

Beyond this intuitive motivation, decoupling bone lengths from identity-specific variation is important when either is fixed; e.g. when re-targeting an outfit to a person we can scale the rigged outfit's lengths to match those of the person, while preserving the bone length-independent part of the outfit shape. In particular, we model the mesh synthesis as the sequential specification of identity-specific bone length, pose-specific joint angles, and identity-specific surface variation, bundled together through linear blend skinning.

We further control and strengthen the individual components of this process: Firstly, we constrain joint angles to respect the kinematic constraints of human body, reducing body motion to 43 pose atoms, amounting to joint rotations around a single axis. Alternative techniques require either restricting the form of the regressor [14] or penalizing wrong estimates through adversarial training [18].

Secondly, we introduce accurate mesh convolution-based networks to capture identity-specific surface variation. We show that these largely outperform their linear basis counterparts, demonstrating for the first time the merit of mesh convolutions in rigged full-body modelling (earlier works [21] were applied to the setup of the face mesh).

We provide quantitative results on the problem of reconstructing a collection of 3D human scans, and show that we obtain systematic gains in average vertex reconstruction accuracy when comparing to a SMPL-type baseline. We note that this is true even though we do not use the pose-corrective blendhapes of [23]; these can be easily integrated, but we leave this for future work.

Beyond quantitative evaluation, we also show that our decoupled bone and shape representation facilitates accurate character animation in-the-wild. Our model formulation allows for out-of-box integration with standard graphics packages like Unity, leading to full-body augmented reality experiences.

2 Related Work

3D Human Body Modelling Linear Blend Skinning (LBS) is widely used to model 3D human bodies due to its ability to represent articulated motions. Some early works have focused on synthesizing realistic 3D humans by modifying the LBS formulation. PSD [20] defines deformations as a function of articulated pose. [3] use the PSD approach learned from 3D scans of real human bodies. Other authors have focused on learning parametric model of human body shapes independently from the pose [2, 25, 30]. Following these works, [6, 11, 13, 15, 16] model both body shape and pose changes with triangle deformations. These work has been extended to also model dynamic soft-tissue motion [26].

Closely related to [4], SMPL [23] propose an LBS-based statistical model of the human mesh, working directly on a vertex coordinate space: T-posed shapes are first generated from a PCA-based basis, and then posed after updating joint locations. More recent works have focused on improving the representational power of the model by combining part models, e.g. for face and hands [17, 29], without however modifying the body model. One further contribution of [23] consists in handling artefacts caused by LBS around the joints when posing the template through the use of pose-corrective blend shapes [23]. Our formulation can be easily extended to incorporate these, but in this work we focus on our main contribution which is modelling of the shape at the bone level.

Graph Convolutions for 3D Human Bodies Different approaches have been proposed to extend convolutional neural networks to non-euclidean data such as graphs and manifolds [9, 10, 12, 22, 27, 32]. Among these, [9, 10, 22, 32] have attempted to model and reconstruct 3D human bodies using convolutional operators defined on meshes. While these methods achieve good performance on shape reconstruction and learning correspondences, their generalisation is not comparable to LBS based methods. Furthermore, the process of synthesising new articulated bodies using mesh convolutional networks is not easy to control since the latent vector typically encodes both shape and pose information.

3 Bone-Level Skinned Model

We start with a high-level overview, before presenting in detail the components of our approach. As shown in Fig. 1, when seen as a system, our model takes as input bone scales, joint angles, and shape coefficients and returns an array of 3-D vertex locations. In particular, BLSM operates along two streams, whose results are combined in the last stage. The upper stream, detailed in Sec. 3.1, determines the internal skeleton by first setting the bone scales through bone scaling coefficients $\mathbf{c}_b$, delivering a bind pose. This is in turn converted to a new pose by specifying joint angles $\boldsymbol{\theta}$, yielding the final skeleton $T(\mathbf{c}_b, \boldsymbol{\theta})$.

The bottom stream, detailed in Sec. 3.2, models the person-specific template synthesis process: Starting from a mesh corresponding to an average body type, $\bar{\mathbf{V}}$, we first absorb the impact of bone scaling by adding a shape correction term, $\mathbf{V}_b$. This is in turn augmented by an identity-specific shape update $\mathbf{V}_s$, modelled by a mesh-convolutional network. The person template is obtained as

$$\mathbf{V} = \bar{\mathbf{V}} + \mathbf{V}_b + \mathbf{V}_s \,. \tag{1}$$

Finally, we bundle the results of these two streams using Linear Blend Skinning, as described in Sec. 3.3, delivering the posed template $\hat{\mathbf{V}}$:

$$\hat{\mathbf{V}} = LBS(\mathbf{V}, T(\mathbf{c}_b, \boldsymbol{\theta})) \,. \tag{2}$$

3.1 Skeleton Modeling

Kinematic Model Our starting point for human mesh modelling is the skeleton. As is common in graphics, the skeleton is determined by a tree-structured graph that ties together human bones through joint connections.

Starting with a single bone, its ‘bind pose’ is expressed by a template rotation matrix $\mathbf{R}^t$ and translation vector $\mathbf{O}^t$ that indicate the displacement and rotation between the coordinate systems at the two bone joints. We model the transformation with respect to the bind pose through a rotation matrix $\mathbf{R}$ and a scaling factor s, bundled together in a 4×4 matrix $\mathbf{T}$:

$$\mathbf{T} = \underbrace{\begin{bmatrix} sI & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R} & 0 \\ 0 & 1 \end{bmatrix}}_{\text{deformation}} \underbrace{\begin{bmatrix} \mathbf{R}^t & \mathbf{O}^t \\ 0 & 1 \end{bmatrix}}_{\text{resting bone}} . \tag{3}$$

We note that common models for character modelling use $s = 1$ and only allow for limb rotation. Any change in object scale, or bone length is modelled by modifying the displacement at the bind pose, $\mathbf{O}$. This is done only implicitly, by regressing the bind pose joints from a 3D synthesized shape. By contrast, our approach gives us a handle on the scale of a limb through the parameter s, making the synthesis of the human skeleton explicitly controllable.

The full skeleton is constructed recursively, propagating from the root node to the leaf nodes along a kinematic chain. Every bone transformation encodes a displacement, rotation, and scaling between two adjacent bones, i and j, where i is the parent and j is the child node. To simplify notation, we will describe the modelling along a single kinematic chain, meaning $j = i+1$, and denote the local transformation of a bone by $\mathbf{T}^i$. The global transformation $\mathbf{T}_j$ from the local coordinates of bone j to world coordinates is given by: $\mathbf{T}_j = \prod_{i \leq j} \mathbf{T}^i$, where we compose the transformations for every bone on the path from the root to the j-th node. This product accumulates the effects of consecutive transformations:

for instance a change in the scale of a bone will incur the same scaling for all of its descendants. These descendants can in turn have their own scale parameters, which are combined with those of their ancestors. The 3D position of each bone j can be read from the last column of $\mathbf{T}_j$, while the upper-left 3×3 part of $\mathbf{T}_j$ provides the scaling and orientation of its coordinate system.

Parametric Bone Scaling We model human proportions by explicitly scaling each bone. For this we perform PCA on bone lengths, as detailed in Sec. 4.2 and use the resulting principal components to express individual bone scales as:

$$\mathbf{b} = \bar{\mathbf{b}} + \mathbf{c}_b \mathbf{P}_b \tag{4}$$

where $\mathbf{c}_b$ are the bone scaling coefficients, $\mathbf{P}_b$ is the bone-scaling matrix, and $\bar{\mathbf{b}}$ is the mean bone scale.

From Eq. 4 we obtain individual bone scales. However, the bone scales s that appear in Eq. 3 are meant to be used through the kinematic chain recursion, meaning that the product of parent scales delivers the actual bone scale, $\mathbf{b}_j = \prod_{i \leq j} s_i$; this can be used to transform the predictions of Eq. 4 into a form that can be used in Eq. 3:

$$s_i = \begin{cases} \mathbf{b}_i / \mathbf{b}_{i-1}, & i > 0 \\ \quad 1 & i = 0 \end{cases} \tag{5}$$

Kinematically Feasible Posing We refine our modeling of joint angles to account for the kinematic constraints of the human body. For instance the knee has one degree of freedom, the wrist has two, and the neck has three. For each joint we set the invalid degrees of freedom to be identically equal to zero, and constrain the remaining angles to be in a plausble range (e.g. ±45 degrees for an elbow). In Fig. 2 we show sample meshes synthesized by posing a template along one valid degree of freedom.

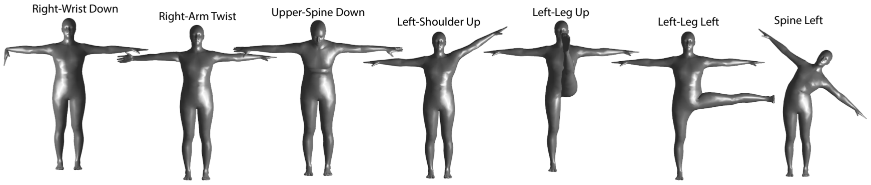

Fig. 2: 7 out of the 47 degrees of freedom corresponding to kinematically feasible joint rotations for our skeleton.

For this, for each such degree of freedom we use an unconstrained variable $x \in R$ and map it to a valid Euler angle $\theta \in [\theta_{min}, \theta_{max}]$ by using a hyperbolic tangent unit:

$$\theta = \frac{\theta_{max} - \theta_{min}}{2} \tanh(x) + \frac{\theta_{min} + \theta_{max}}{2} \tag{6}$$

This allows us to perform unconstrained optimization when fitting our model to data, while delivering kinematically feasible poses. The resulting per-joint Euler angles are converted into a rotation matrix, delivering the matrix R in Eq. 3.

Using Eq. 6 alleviates the need for restricting the regressor form [14] or adversarial training [18], while at the same time providing us with a compact, interpretable dictionary of 47 body motions. We provide samples of all such motions in the supplemental material.

3.2 Template Synthesis

Having detailed skeleton posing, we now turn to template synthesis. We start by modeling the effect of bone length on body shape, and then turn to modelling identity-specific variability.

Bone-Dependent Shape Variations Bone length can be used to account for a substantial part of body shape variability. For example, longer bones correlate with a male body-shape, while limb proportions can correlate with ectomorph, endomorph and mesomorph body-type variability. We represent the bone-length dependent deformation of the template surface through a linear update:

$$\mathbf{V}_b = \mathbf{c}_b \mathbf{P}_{bc} \tag{7}$$

where $\mathbf{P}_{bc}$ is the matrix of bone-corrective blendshapes.

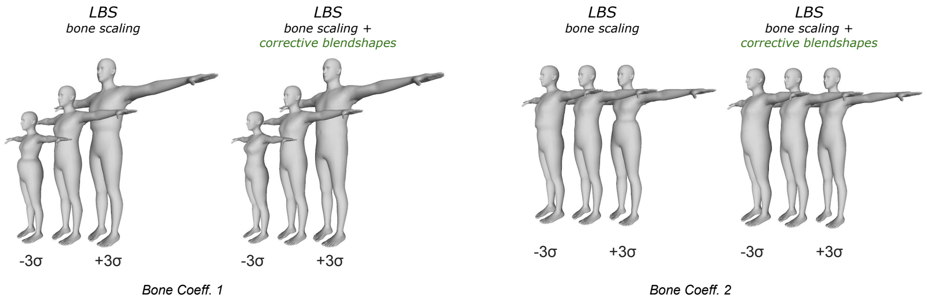

Fig. 3: Impact of bone length variation on the template. Plain linear blend skinning results in artifacts. The linear, bone-corrective blendshapes eliminate these artifacts, and capture correlations of bone lengths with gender and body type.

Graph Convolutional Shape Modelling Having accounted for the bone length-dependent part of shape variability, we turn to the remainder of the person-specific variability. The simplest approach is to use a linear update:

$$\mathbf{V}_s = \mathbf{c}_s \mathbf{P}_s \tag{8}$$

where $\mathbf{c}_s$ are the shape coefficients, and $\mathbf{P}_s$ is the matrix of shape components; we refer to this baseline as the linear model. By contrast, we propose a more powerful, mesh-convolutional update. For this we use multi-layer mesh convolution decoder that precisely models the nonlinear manifold of plausible shapes in its output space.

We represent the triangular mesh as a graph $(\mathbf{V}, \mathcal{E})$ with vertices $\mathbf{V}$ and edges $\mathcal{E}$ and denote the convolution operator on a mesh as:

$$(f \star g)_x = \sum_{l=1}^{L} g_l f(x_l) \tag{9}$$

where g_l denotes the filter weights, and $f(x_l)$ denotes the feature of the l-th neighbour of vertex x. The neighbours are ordered consistently across all meshes, allowing us to construct a one-to-one mapping between the neighbouring features and the filter weights. Here we adapt the setting of [10], where the ordering is defined by a spiral starting from the vertex x, followed by the d-ring of the vertex, i.e. for a vertex x, x_l is defined by the ordered sequence:

$$S(x) = \{x, R_1^1(x), R_2^1(x), ..., R^h_{\|R^h\|}\}, \tag{10}$$

where h is the patch radius and $R_j^d(x)$ is the j-th element in the d-ring.

We use a convolutional mesh decoder to model the normalised deformations from the bone-updated shape. The network consists of blocks of convolution-upsampling layers similar to [27]. We pre-compute the decimated version of the template shape with quadratic edge collapse decimation to obtain the upsampling matrix. Given the latent vector $\mathbf{z}$, shape variation is represented as

$$\mathbf{V}_s = \mathcal{D}(\mathbf{z}) \tag{11}$$

where $\mathcal{D}$ is the learned mesh convolutional decoder.

3.3 Linear Blend Skinning

Having detailed the skeleton and template synthesis processes, we now turn to posing the synthesized template based on the skeleton. We use Linear Blend Skinning (LBS), where the deformation of a template mesh $\mathbf{V}$ is determined by the transformations of the skeleton. We consider that the bind pose of the skeleton is described by the matrices $\hat{\mathbf{T}}_j$, where the 3D mesh vertices take their canonical values $\mathbf{v}_i \in \mathbf{V}$, while the target pose is described by $\mathbf{T}_j$.

According to LBS, each vertex is influenced by every bone j according to a weight w_{ij}; the positions of the vertices $\hat{\mathbf{v}}_i$ at the target pose are given by:

$$\hat{\mathbf{v}}_k = \sum_j w_{ij} \mathbf{T}_j \hat{\mathbf{T}}_j^{-1} \mathbf{v}_k. \tag{12}$$

In the special case where $\mathbf{T} = \hat{\mathbf{T}}$, we recover the template shape, while in the general case, Eq. 12 can be understood as first charting every point $\mathbf{v}_k$ with respect to the bind bone (by multiplying it with $\hat{\mathbf{T}}_j^{-1}$), and then transporting to the target bone (by multiplying with $\mathbf{T}_j$).

4 Model training

Having specified BLSM, we now turn to learning its parameters from data. For this we use the CAESAR dataset [28] to train the shape model, which contains high resolution 3D scans of 4400 subjects wearing tight clothing. This minimal complexity due to extraneous factors has made CAESAR appropriate for the estimation of statistical body models, such as SMPL. For training skinning weights we use D-FAUST [8] dataset. Our training process consists in minimising the reconstruction error of CAESAR and D-FAUST through BLSM.

Since BLSM is implemented as a multi-layer network in pytorch, one could try to directly minimize the reconstruction loss with respect to the model parameters using any standard solver. Unfortunately however, this is a nonlinear optimisation problem with multiple local minima; we therefore use a carefully engineered pipeline that solves successively demanding optimization problems, as detailed below, and use automatic differentiation to efficiently compute any derivatives required during optimization.

4.1 Unconstrained Landmark-based Alignment

Each CAESAR scan $\mathbf{S}^n$ is associated with 73 anatomical landmarks, $\mathbf{L}^n$ that have been localised in 3D. We start by fitting our template to these landmarks by gradient descent on the joint angles $\boldsymbol{\theta}^n$ and bone scales $\mathbf{s}^n$, so as to minimize the 3D distances between the landmark positions and the respective template vertices. More specifically, the following optimization problem is solved:

$$\boldsymbol{\theta}^n, \mathbf{s}^n = \mathrm{argmin}_{\boldsymbol{\theta},\mathbf{s}} \|\mathbf{A}\, LBS(\mathbf{V}_T, T(\mathbf{s}, \boldsymbol{\theta})) - \mathbf{L}^n\|^2 \tag{13}$$

where $\mathbf{A}$ selects the subset of landmarks from the template.

This delivers an initial fitting which we further refine by registering our BLSM-based prediction $\hat{\mathbf{S}}^n = LBS(\mathbf{V}_T, T(\mathbf{s}^n, \boldsymbol{\theta}^n))$ to each scan $\mathbf{S}^n$ with Non-Rigid ICP (NICP) [5]. This alignment stage does not use yet a statistical model to constrain the parameter estimates, and as such can be error-prone; the following steps recover shapes that are more regularized, but the present result acts like a proxy to the scan that is in correspondence with the template vertices.

4.2 Bone Basis and Bone-corrective Blendshapes

We start learning our model by estimating a linear basis for bone scales. For each shape $\hat{\mathbf{S}}^{\mathbf{n}}$ we estimate the lengths of the bones obtained during the optimization process described in the previous section.

We perform PCA on the full set of CAESAR subjects and observe that linear bases capture 97% of bone length variability on the first three eigenvectors. We convert the PCA-based mean vector and basis results from bone lengths into the mean bone scaling factor $\bar{\mathbf{b}}$ and bone scaling basis $\mathbf{P}_b$ used in Eq. 4 by dividing them by the mean length of the respective bone along each dimension.

Having set the bone scaling basis, we use it as a regularizer to re-estimate the pose $\boldsymbol{\theta}^n$ and bone scale coefficients $\mathbf{c}_b^n$ used to match our template $\mathbf{V}_T$ to each registration $\hat{\mathbf{S}}^n$ by solving the following optimisation problem:

$$\boldsymbol{\theta}^n, \mathbf{c}_b^n = \mathrm{argmin}_{\boldsymbol{\theta},\mathbf{c}_b} \| LBS(\mathbf{V}_T, T(\mathbf{c}_b, \boldsymbol{\theta})) - \hat{\mathbf{S}}^n \|^2 \tag{14}$$

Finally we optimize over the bone-corrective basis $\mathbf{P}_b$ and mean shape $\bar{\mathbf{V}}$:

$$\mathbf{P}_{bc}^*, \bar{\mathbf{V}}^* = \mathrm{argmin}_{\mathbf{P}_b,\bar{\mathbf{V}}} \sum_{n=1}^{N} \| LBS(\bar{\mathbf{V}} + \mathbf{c}_b^n \mathbf{P}_{bc}, T(\mathbf{c}_b^n, \boldsymbol{\theta}^n)) - \hat{\mathbf{S}}^n \|^2 \tag{15}$$

Given that $\mathbf{V}_T$ and $\hat{\mathbf{S}}^n$ are in one-to-one correspondence, we no longer need ICP to optimize Eq. 14 and Eq. 15, allowing us instead to exploit automatic differentiation and GPU computation for gradient descent-based optimization.

4.3 Shape Blendshapes

Once bone-corrected blendshapes have been used to improve the fit of our model to the registered shape $\hat{\mathbf{S}}^n$, the residual in the reconstruction is attributed only to identity-specific shape variability. We model these residuals as vertex displacements ${\mathbf{V}_\mathbf{D}}^n$ and estimate them for each registration $\hat{\mathbf{S}}^n$ by setting:

$$LBS(\bar{\mathbf{V}} + \mathbf{V}_b^n + {\mathbf{V}_\mathbf{D}}^n, T(\mathbf{c}_b^n, \boldsymbol{\theta}^n)) = \hat{\mathbf{S}}^n \tag{16}$$

to ensure that the residual is defined in the T-pose coordinate system.

For the linear alternative described in Sec. 3.2 the shape basis $\mathbf{P}_s$ is computed by performing PCA analysis of $\{{\mathbf{V}_\mathbf{D}}^n\}$. To train the graph convolutional system described in Sec. 3.2, we learn the parameters of the spiral mesh convolutional decoder $\mathcal{D}$ and the latent vectors $\mathbf{z}^n$ that minimize the following loss:

$$\mathrm{argmin}_{\mathcal{D},\mathbf{z}} \sum_{n=1}^{N} \| {\mathbf{V}_\mathbf{D}}^n - \mathcal{D}(\mathbf{z}^n) \| \tag{17}$$

4.4 Blending Weights

So far the blending weights of our LBS formulation are manually initialised, which can be further improved from the data. For this purpose we use the D-FAUST dataset [8], which contains registrations of a variety of identity and poses. For each registration $\mathbf{S}^n$ in the dataset, we first estimate the parameters of our model, namely $\mathbf{c}_b^n$, $\mathbf{c}_s^n$, $\boldsymbol{\theta}^n$, as well as the residual $\hat{\mathbf{V}}_D^n$ which is the error on the T-pose coordinate system after taking into account the shape blendshapes. Then we optimize instead the blending weights to minimize the following error:

$$\mathrm{argmin}_{\mathbf{W}} \sum_{n=1}^{N} \| \mathbf{S}_n - LBS_{\mathbf{W}}(\bar{\mathbf{V}} + \mathbf{V}_b^n + \mathbf{V}_s^n + \hat{\mathbf{V}}_{\mathbf{D}}^n, T(\mathbf{c}_b^n, \boldsymbol{\theta}^n)) \|^2 \tag{18}$$

where we use the mapping:

$$\mathbf{W} = \frac{f(\mathbf{W}')}{\sum_j f(\mathbf{W}')_{ij}} \quad \text{with} \quad f(\mathbf{X}) = \sqrt{\mathbf{X}^2 + \varepsilon} \tag{19}$$

to optimize freely $\mathbf{W}'$ while ensuring the output weights $\mathbf{W}$ satisfy the LBS blending weights constraints: $\sum_j \mathbf{W}_{ij} = \mathbf{1}$, and $mathbfW_{ij} \geq 0$.

5 Evaluation

5.1 Implementation Details

Baseline Implementation The publicly available SMPL model [23] has 10 shape bases, a mesh topology that is different to that of our model, and pose-corrective blendshapes, making any direct comparison to our model inconclusive.

In order to have directly comparable results across multiple shape coefficient dimensionalities we train a SMPL-like model (referred to as SMPL-reimpl) using the mesh topology, kinematic structure, and blending weight implementation of our model, and SMPL's PCA-based modeling of shape variability in the T-pose. We further remove any pose-corrective blendshape functionality, allowing us to directly assess the impact of our disentangled, bone-driven modeling of mesh variability against a baseline that does not use it.

In order to train SMPL-reimpl, we first define manually the joint regressor required by [23] by taking the mean of the ring of vertex that lies around a certain joint; we then train the blending weights and joint regressor on the D-FAUST dataset, as described in [23]. The shape blendshapes are then trained with the CAESAR dataset using the same method described in [23].

We further note that our evaluations focus on the gender-neutral versions of both SMPL-reimpl and BLSM - and may therefore be skewed in favour of BLSM's ability to easily capture large-scale, gender-dependent bone variations. This is however relevant to the performance of most downstream, CNN-driven human mesh reconstruction systems that do not know in advance the subject's gender [18] [14] [19], and have therefore adopted the neutral model. More recent work [24] has shown improvements based on exploiting gender attributes in tandem with the gender-specific SMPL models. We leave a more thorough ablation of the interplay between gender and reconstruction accuracy in future work.

Mesh Convolutional Networks For graph convolutional shape modelling, we train networks with 4 convolutional layers, with (48, 32, 32, 16) filters for each layer, respectively. Convolutional layers are followed by batch normalisation and upsampling layers with factors (2, 2, 2, 4) respectively. For the convolutional layers, we use ELU as the activation function. Finally, the output layer is a convolutional layer with filter size 3 and linear activation, which outputs the normalised vertex displacements. We train our network with an Adam optimiser, with a learning rate of 1e-3 and weight decay of 5e-5 for the network parameters, and learning rate of 0.1 and weight decay 1e-7 for the latent vectors. The learning rates are multiplied by a factor of 0.99 after each epoch.

5.2 Quantitative Evaluation

We evaluate the representation power of our proposed BLSM model on the CAESAR dataset and compare its generalisation ability against the SMPL-type baseline on D-FAUST dataset and our in-house testset. D-FAUST contains 10 subjects, each doing 14 different motions. We further expand the testset with our in-house dataset. Captured with a custom-built multi-camera active stereo system (3dMD LLC, Atlanta, GA), our in-house testset consists of 4D sequences at 30 FPS of 20 individuals spanning different body types and poses. Each instance contains around 50K vertices. These scans are registered to our template as described in Sec. 4.1, while using NICP with temporal consistency constraints.

The models that we compare are aligned to the registered meshes by minimising the L2 distance between each vertex. We use an Adam optimiser with learning rate 0.1 to optimise parameters for all models, and reduce the learning rate by a factor of 0.9 on plateau. To avoid local minima, we use a multi-stage optimization approach as in [7]. We first fit the vertices on the torso (defined by the blending weights of the torso bones on our template) by optimising over the shape coefficients and the joint angles of the torso bones. Then for second and third stage, upper-limbs and lower-limbs are added respectively. In the last stage, all the vertices are used to fine-tune the fitted parameters. In the following, we report the mean absolute vertex errors (MABS) of gender neutral models.

CAESAR In Fig. 4 we plot the fitting errors on the CAESAR dataset as a function of the number of shape coefficients, namely shape blendshapes for SMPL-reimpl, bone blendshapes and shape blendshapes for BLSM-linear and latent space dimension for BLSM-spiral.

We observe that our BLSM-linear model attains lower reconstruction error compared to the SMPL-reimpl baseline. The sharpest decrease happens for the first three coefficients, corresponding to bone-level variability modelling. Starting from the fourth coefficient the error decreases more slowly for the linear model,

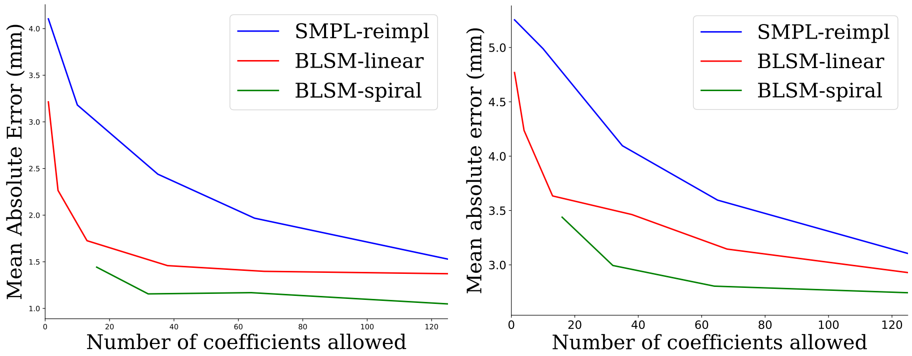

Fig. 4: Mean absolute vertex error on the CAESAR dataset (left) and our in-house testset (right) against number of shape coefficients.

but the BLSM-spiral variant further reduces errors. These results suggest that our BLSM method captures more of the shape variation with fewer coefficients compared to the SMPL-reimpl baseline.

D-FAUST For D-FAUST, we select one male and one female subject (50009 and 50021) for evaluation, and the rest for training blending weights for both models and joint regressors for SMPL-reimpl. We evaluate first shape generalisation error by fitting the models to all sequences of the test subjects (Fig. 5 left). We observe that our BLSM-linear model obtain lower generalisation error compared to SMPL-reimpl baseline, and the result is improved further with BLSM-spiral.

We also evaluate the pose generalisation error of our models (Fig. 5 right). The errors are obtained by first fitting the models to one random frame of each subject, then fit the pose parameter to rest of the frames while keeping the shape coefficients fixed. This metric suggests how well a fitted shape generalise to new poses. We observe that both of our linear and spiral models generalises better than our SMPL-reimpl baseline. We argue that by introducing bone scales to the model, the fitted poses are well regularized, thus during training it is more straight forward to decouple the shape and pose variations in the dataset, while avoiding the need to learn subject specific shapes and joints as in SMPL.

In-house Testset In Fig. 4, we also report the average MABS across all sequences in our in-house testset as a function of the number of shape coefficients used. We observe that our proposed models are able to generalise better than the SMPL-reimpl model on our testset. Our proposed models are compact and able to represent variations in our testset with a smaller number of shape coefficients than SMPL-reimpl.

In Fig. 6, we show the mean per-vertex error heatmaps on all sequences and on some example registrations in our testset. Compared to SMPL-reimpl, our

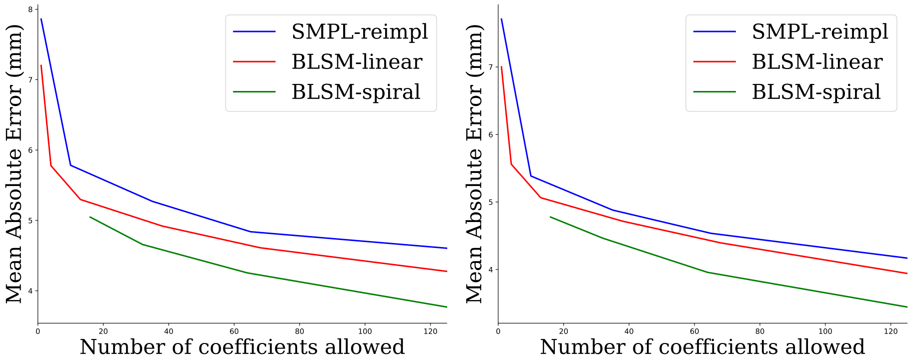

Fig. 5: Shape generalisation error (left) and pose generalisation error (right) on D-FAUST dataset against number of shape coefficients.

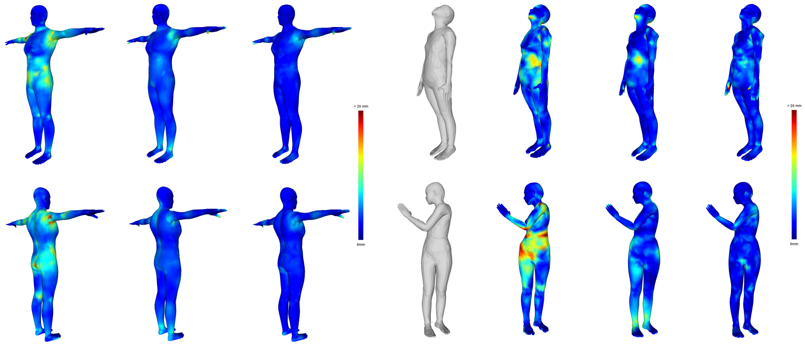

Fig. 6: Mean absolute vertex error and example of reconstructions on the testset. Left to right: SMPL-reimpl, BLSM-Linear, BLSM-Spiral. For linear models we show result with 125 coefficients allowed. For BLSM-spiral the latent size is 128.

proposed models are able to fit closely across the full body, while the SMPL-reimpl model produces larger error on some of the vertices. The result suggests that our proposed model generalise better on surface details than the SMPL-reimpl baseline model.

5.3 Qualitative Evaluation

In Fig. 8 we show samples from our linear model by varying the bone bases, as well as identity-specific shape coefficients from -3σ to +3σ. We observe that our model captures a variety of body shapes and the method successfully decouples bone length-dependent variations and identities specific shape variations.

This decoupling allows us to perform simple and accurate character animation driven by persons in unconstrained environments as shown in Fig. 7. In an offline stage, we rig several characters from [1] to our model's skeleton. Given an image of a person, we first fit our model to it using a method similar to [18]. We then apply the estimated bone transformations (scales and rotations) to the rigged characters. This allows accurate image-driven character animation within any standard graphics package like Unity.

Alternative methods require either solving a deformation transfer problem [31] [33], fixed shape assumptions, or approximations to a constant skeleton, while our approach can exactly recover the estimated skeleton position as it is part of the mesh construction.

Please note that many recent works that predict model parameters for image alignment are applicable to our model [18] [14] [19]; in this work we focus on showing the merit of our model once the alignment is obtained, and provide multiple CNN-driven results on the project's webpage.

We also assess the representational power of our mesh convolutional networks by examining the samples from each dimension of the latent space (Fig. 8). We

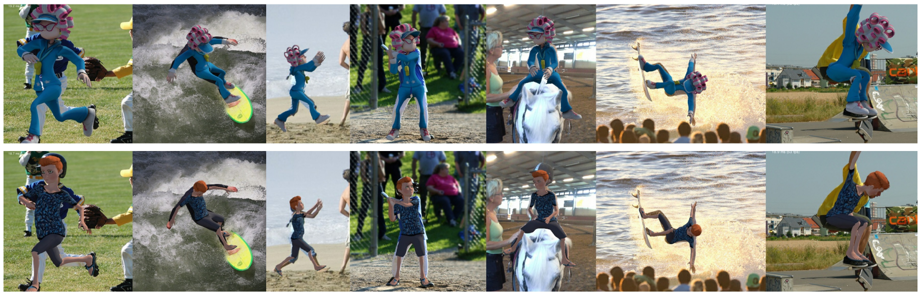

Fig. 7: Image-driven character animation: we rig two characters from [1] using our model's bone structure. This allows us to transform any person into these characters, while preserving the pose and body type of the person in the image.

observe that while capturing large deformations such as gender and body type, the network also captures details such as different body fat distributions.

6 Conclusion

In this paper we propose BLSM, a bone-level skinned model of the 3D human body mesh where bone modelling and identity-specific variations are decoupled. We introduce a data-driven approach for learning skeleton, skeleton-conditioned shape variations and identity-specific variations. Our formulation facilitates the use of mesh convolutional networks to capture identity specific variations, while explicitly modeling the range of articulated motion through built-in constraints.

We provide quantitative results showing that our model outperforms existing SMPL-like baseline on the 3D reconstruction problem. Qualitatively, we also show that by virtue of being bone-level our formulation allows us to perform accurate character retargeting in-the-wild.

Acknowledgements The work of S. Zafeiriou and R.A.Guler was funded in part by EPSRC Project EP/S010203/1 DEFORM.

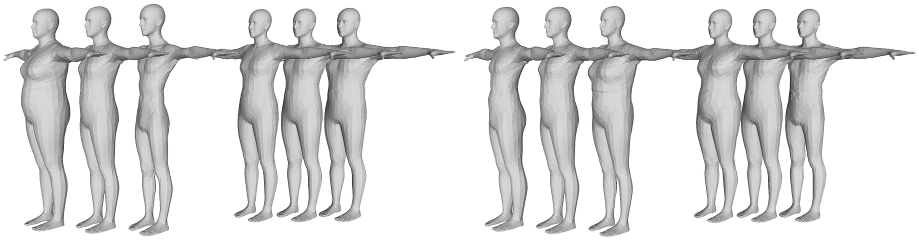

Fig. 8: Linear (left) vs. graph convolutional (right) modeling of shape variation.

References

1. Mixamo. https://www.mixamo.com (2019)
2. Allen, B., Curless, B., Curless, B., Popović, Z.: The space of human body shapes: reconstruction and parameterization from range scans. In: ACM transactions on graphics (TOG). vol. 22, pp. 587–594. ACM (2003)
3. Allen, B., Curless, B., Popović, Z.: Articulated body deformation from range scan data. In: ACM Transactions on Graphics (TOG). vol. 21, pp. 612–619. ACM (2002)
4. Allen, B., Curless, B., Popović, Z., Hertzmann, A.: Learning a correlated model of identity and pose-dependent body shape variation for real-time synthesis. In: Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation. pp. 147–156. Eurographics Association (2006)
5. Amberg, B., Romdhani, S., Vetter, T.: Optimal step nonrigid icp algorithms for surface registration. In: 2007 IEEE Conference on Computer Vision and Pattern Recognition. pp. 1–8. IEEE (2007)
6. Anguelov, D., Srinivasan, P., Koller, D., Thrun, S., Rodgers, J., Davis, J.: Scape: shape completion and animation of people. In: ACM transactions on graphics (TOG). vol. 24, pp. 408–416. ACM (2005)
7. Bogo, F., Kanazawa, A., Lassner, C., Gehler, P., Romero, J., Black, M.J.: Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In: European Conference on Computer Vision. pp. 561–578. Springer (2016)
8. Bogo, F., Romero, J., Pons-Moll, G., Black, M.J.: Dynamic faust: Registering human bodies in motion. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 6233–6242 (2017)
9. Boscaini, D., Masci, J., Rodolà, E., Bronstein, M.: Learning shape correspondence with anisotropic convolutional neural networks. In: Advances in Neural Information Processing Systems. pp. 3189–3197 (2016)
10. Bouritsas, G., Bokhnyak, S., Ploumpis, S., Bronstein, M., Zafeiriou, S.: Neural 3d morphable models: Spiral convolutional networks for 3d shape representation learning and generation. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 7213–7222 (2019)
11. Chen, Y., Liu, Z., Zhang, Z.: Tensor-based human body modeling. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 105–112 (2013)
12. Defferrard, M., Bresson, X., Vandergheynst, P.: Convolutional neural networks on graphs with fast localized spectral filtering. In: Advances in neural information processing systems. pp. 3844–3852 (2016)
13. Guan, P., Reiss, L., Hirshberg, D.A., Weiss, A., Black, M.J.: Drape: Dressing any person.
14. Guler, R.A., Kokkinos, I.: Holopose: Holistic 3d human reconstruction in-the-wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 10884–10894 (2019)
15. Hasler, N., Stoll, C., Sunkel, M., Rosenhahn, B., Seidel, H.P.: A statistical model of human pose and body shape. In: Computer graphics forum. vol. 28, pp. 337–346. Wiley Online Library (2009)
16. Hirshberg, D.A., Loper, M., Rachlin, E., Black, M.J.: Coregistration: Simultaneous alignment and modeling of articulated 3d shape. In: European Conference on Computer Vision. pp. 242–255. Springer (2012)
17. Joo, H., Simon, T., Sheikh, Y.: Total capture: A 3d deformation model for tracking faces, hands, and bodies. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8320–8329 (2018)

18. Kanazawa, A., Black, M.J., Jacobs, D.W., Malik, J.: End-to-end recovery of human shape and pose. In: Computer Vision and Pattern Regognition (CVPR) (2018)
19. Kolotouros, N., Pavlakos, G., Black, M.J., Daniilidis, K.: Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2252–2261 (2019)
20. Lewis, J.P., Cordner, M., Fong, N.: Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. In: Proceedings of the 27th annual conference on Computer graphics and interactive techniques. pp. 165–172. ACM Press/Addison-Wesley Publishing Co. (2000)
21. Li, T., Bolkart, T., Black, M.J., Li, H., Romero, J.: Learning a model of facial shape and expression from 4d scans. ACM Transactions on Graphics (TOG) **36**(6), 194 (2017)
22. Lim, I., Dielen, A., Campen, M., Kobbelt, L.: A simple approach to intrinsic correspondence learning on unstructured 3d meshes. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 0–0 (2018)
23. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: Smpl: A skinned multi-person linear model. ACM transactions on graphics (TOG) **34**(6), 248 (2015)
24. Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A.A.A., Tzionas, D., Black, M.J.: Expressive body capture: 3d hands, face, and body from a single image. In: Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (2019)
25. Pishchulin, L., Wuhrer, S., Helten, T., Theobalt, C., Schiele, B.: Building statistical shape spaces for 3d human modeling. Pattern Recognition **67**, 276–286 (2017)
26. Pons-Moll, G., Romero, J., Mahmood, N., Black, M.J.: Dyna: A model of dynamic human shape in motion. ACM Transactions on Graphics (TOG) **34**(4), 120 (2015)
27. Ranjan, A., Bolkart, T., Sanyal, S., Black, M.J.: Generating 3d faces using convolutional mesh autoencoders. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 704–720 (2018)
28. Robinette, K.M., Blackwell, S., Daanen, H., Boehmer, M., Fleming, S.: Civilian american and european surface anthropometry resource (caesar), final report. volume 1. summary. Tech. rep., SYTRONICS INC DAYTON OH (2002)
29. Romero, J., Tzionas, D., Black, M.J.: Embodied hands: Modeling and capturing hands and bodies together. ACM Transactions on Graphics (TOG) **36**(6), 245 (2017)
30. Seo, H., Cordier, F., Magnenat-Thalmann, N.: Synthesizing animatable body models with parameterized shape modifications. In: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation. pp. 120–125. Eurographics Association (2003)
31. Sumner, R.W., Popović, J.: Deformation transfer for triangle meshes. ACM Transactions on graphics (TOG) **23**(3), 399–405 (2004)
32. Verma, N., Boyer, E., Verbeek, J.: Feastnet: Feature-steered graph convolutions for 3d shape analysis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2598–2606 (2018)
33. Wang, J., Wen, C., Fu, Y., Lin, H., Zou, T., Xue, X., Zhang, Y.: Neural pose transfer by spatially adaptive instance normalization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5831–5839 (2020)