

# Total Capture: A 3D Deformation Model for Tracking Faces, Hands, and Bodies (Supplementary Material) \*

Hanbyul Joo<sup>1</sup> Tomas Simon<sup>1,2</sup> Yaser Sheikh<sup>1,2</sup>

<sup>1</sup>Carnegie Mellon University <sup>2</sup>Facebook Reality Labs, Pittsburgh

[hanbyulj@cs.cmu.edu](mailto:hanbyulj@cs.cmu.edu), {tomas.simon, yaser.sheikh}@fb.com

## 1 Building the Mesh Topology of the Frank Model

To define the shape of Frank model at the rest pose, we manually align the body model and other part models as shown Fig. 1 (a). The alignment is done by selecting a set of corresponding points and computing a similarity transform matrix between parts. Then, we manually stitched the aligned meshes in a single mesh topology by removing redundant parts and modifying the vertices around the borders to make the final mesh as seamless as possible. We also modified the shape of torso and feet of the final model to better match clothed people, as shown in Fig. 1 (b). Finally, a blending  $\mathbf{C}$  used in Eq. (4) of the paper is computed to transform the aligned part models to the final Frank mesh shape.

## 2 Motion Capture with Frank Model

### 2.1 Feet Keypoint Detection

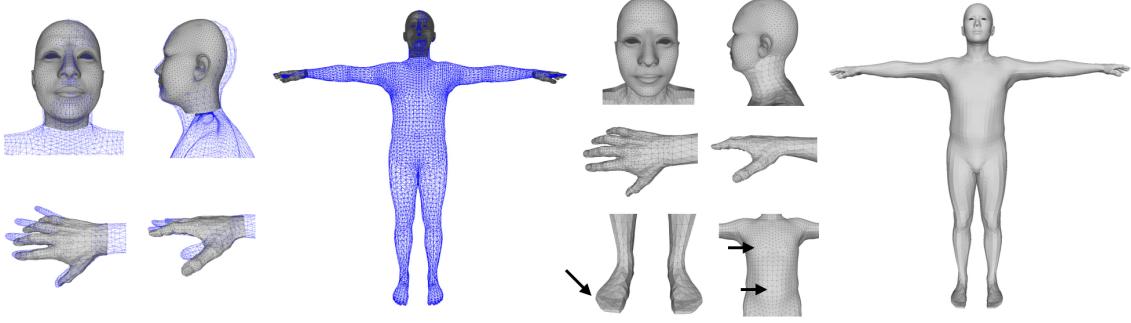
Example annotations for the feet landmarks can be seen in Fig. 2. At test time, for a given person detection, we use a bounding box centered at the mean of the knee and ankle points (defining the shins) of both legs. As the size of the bounding box, we use a square of scale 3 times the maximum difference between the point locations across all axes.

### 2.2 Objective Function

**Seam Constraints:** We encourage the vertices around the seam parts to be close by penalizing differences between the last rings of vertices around the seam of each part, and the corresponding closest point in the body model in the rest pose expressed as barycentric coordinates,

---

\*Website: <http://www.cs.cmu.edu/~hanbyulj/totalcapture>



(a) Aligned face and hand models to the body model

(b) A seamless mesh topology of Frank model

Figure 1: (a) We manually align face and hand models on the mean shape of the body model at rest pose. (b) The mean shape of the Frank model is defined to minimize the seams around the borders. In building the shape of Frank, we also modify the original shape of the body model to make the model less naked (e.g., toes and shape of torsos as pointed by arrows).



Figure 2: (Left) Definition of foot keypoints: tip of big and little toes and center of the heel bone. (Right) Example images for foot keypoint annotations.

$$E_{\text{seam}} = \sum_{(i,j) \in \mathcal{C}^{LH}} \|\mathbf{B}_i \mathbf{V}^B - (\mathbf{v}_j^{LH})^T\|^2 + \quad (1)$$

$$\sum_{(i,j) \in \mathcal{C}^{RH}} \|\mathbf{B}_i \mathbf{V}^B - (\mathbf{v}_j^{RH})^T\|^2 + \quad (2)$$

$$\sum_{(i,j) \in \mathcal{C}^F} \|\mathbf{B}_i \mathbf{V}^B - (\mathbf{v}_j^F)^T\|^2, \quad (3)$$

where the set  $\mathcal{C}^{LH}$  contains correspondences  $(i, j)$  where  $j$  indexes the last two rings of vertices around the seam of the left hand, and  $i$  denotes the corresponding closest point in the body model in the rest pose expressed as barycentric coordinates with  $\mathbf{B}_i \in \mathbb{R}^{1 \times N^B}$  (where  $\mathbf{B}_i \mathbf{1}_{NB} = 1$ ), and similarly for the right hand. For the face, we use the last three rings of vertices, as shown in Fig. 3.

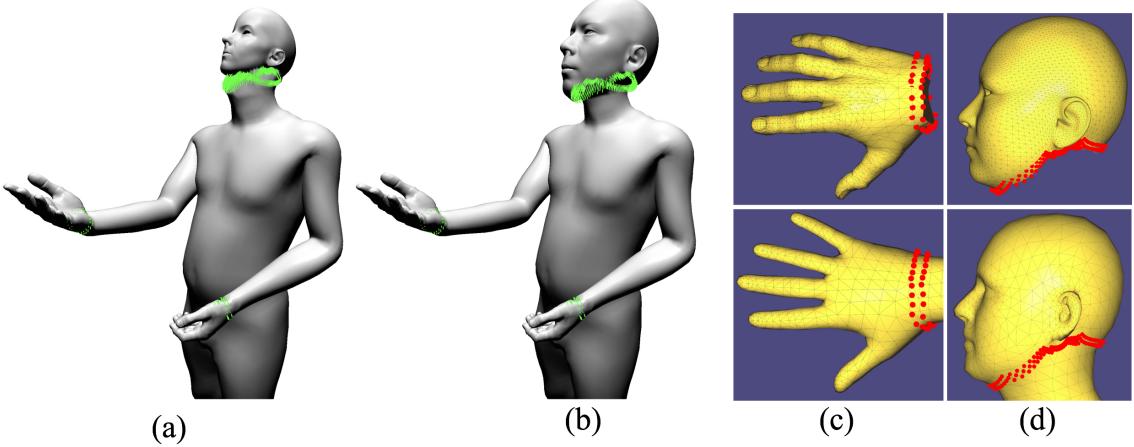


Figure 3: (a and b) Frank model keeps the original parameterization of part models and it may have artifacts around the borders (in this figures we intentionally manipulate parameters to visualize the artifact). We define a seam constraint to avoid this inconsistency among parts, which are shown in the green lines in (a) and (b). (c) and (d) show the vertices selected to define the seam constraint. In (c) and (d), the top figures show the hand and face part models and the bottom figures show the vertices in the body model.

**Prior Cost:** All PCA-derived parameters are penalized with the corresponding variance, which by construction of our basis is normalized to 1, so that:

$$E_{\text{prior}}^F = \lambda_{SF} \|\phi^F\|^2 + \lambda_{PF} \|\theta^F\|^2 \quad (4)$$

$$E_{\text{prior}}^B = \lambda_{SB} \|\phi^B\|^2 + \lambda_{PB} \|\theta^B\|^2 \quad (5)$$

$$E_{\text{prior}}^H = \lambda_{SH} \|\phi^H - 1\|^2 + \lambda_{PH} \|\theta^H\|^2 \quad (6)$$

where, for simplicity, we penalize pose parameters that deviate from the rest pose with a small weight, and we encourage hand scaling factors to be close to 1, and similarly for other part model parameters. We use different weights for the priors on different parts ( $\lambda_{SF} = \lambda_{SH} = 10^{-2}$ ,  $\lambda_{SB} = 10^{-2}$ ,  $\lambda_{PF} = 10^{-4}$ ,  $\lambda_{PB} = 10^{-6}$ , and  $\lambda_{PH} = 10^{-7}$ ).

### 2.3 Optimization Procedure

The complete model is highly nonlinear, and due to the limited degrees of freedom of the skeletal joints, the optimization can get stuck in bad local minima. Therefore, instead of optimizing the complete model initially, we fit the model in phases, starting with a subset of measurements and strong priors that are relaxed as optimization progresses.

Model fitting is performed on each frame independently. To initialize the overall translation and rotation, we use four keypoints on the torso (left and right shoulders and hips)

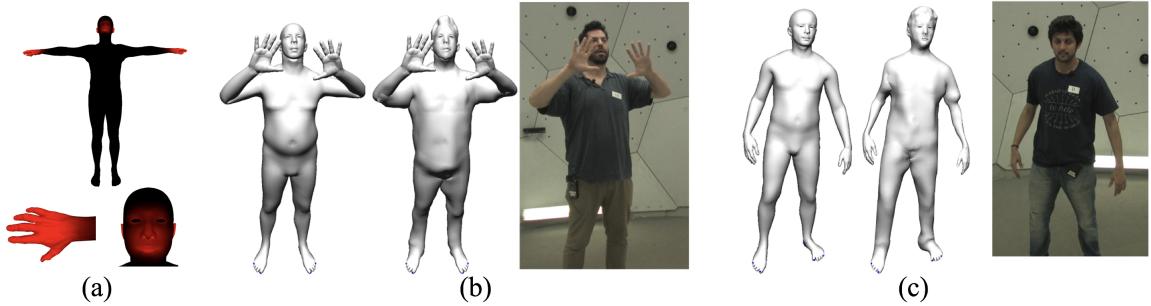


Figure 4: (a) Visualizing the weight  $\mathbf{W}$  in Eq. (15). The black region means low weight allowing more deformation and the red region means higher weight allowing fitting deformation. (b and c) Examples of out-of-shape space fitting results. For each example, the Frank model fitting result (left), after deformation (center), and input image are shown (right).

without using the ICP term, and with strong weight on the priors. Once the torso parts are approximately aligned, we use all available keypoints of all body parts, with small weight for the priors. The results at this stage already provide reasonable motion capture but do not accurately capture the shape (i.e., silhouette) of the subject. Finally, the entire optimization is performed including the ICP term to find correspondences with the 3D point cloud. We run the final optimization two times, finding new correspondences each time. For the optimization we use Levenberg-Marquardt with the Ceres Solver library [1].

### 3 Creating Adam

#### 3.1 Fitting Clothes and Hair

In Eq (15), we use a diagonal weight matrix  $\mathbf{W} \in \mathbb{R}^{N^U \times N^U}$  to avoid large deformations where the 3D point cloud has lower resolution than the original mesh, such as details in the face and hands. The visualization of this weight matrix is shown in Fig. 4 (a). The examples of deformed meshes are shown in Fig. 4 (b) and (c). We run this process for 70 subjects with 5 different poses, resulting in 350 meshes, as shown in Fig. 5.

#### 3.2 Building Joint Regression and Shape Space

The joint regression matrix  $\mathbf{J}^A$  of Adam, with the 61 joints including hand joints and their corresponding vertices are visualized in the left and center of Fig. 6. The mean shape of Adam is shown on the right of Fig. 6.

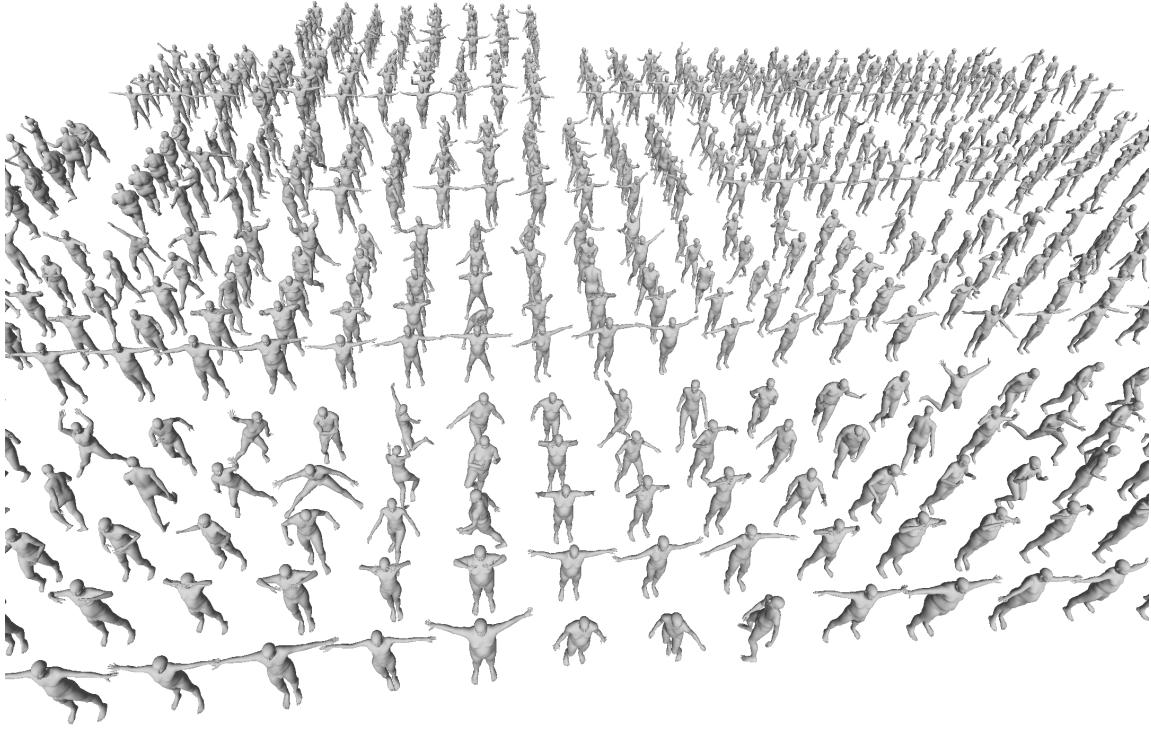


Figure 5: An overview of the reconstructed 350 shapes by fitting the Frank model.

### 3.3 Optical Flow Propagation in Tracking With Adam

While fitting each frame independently has benefits—it does not suffer from error accumulation and frames can be fit in parallel—it typically produces jittery motion. To reduce this jitter, we use optical flow to propagate the initial, per-frame fit to neighboring frames to find a smoother solution. More concretely, given the fitting results at the frame  $t$ , we propagate this mesh to frames  $t-1$  and  $t+1$  using optical flow at each vertex, which is triangulated into 3D using the method of [2]. Therefore, each vertex has at most three candidate positions: the original mesh, and the forward and backward propagated vertices (subject to a forward-backward consistency check). Given these propagated meshes, we reoptimize the model parameters by using all propagated mesh vertices as additional keypoints to find a compromise mesh.

## References

- [1] S. Agarwal, K. Mierle, and Others. Ceres solver. <http://ceres-solver.org>.

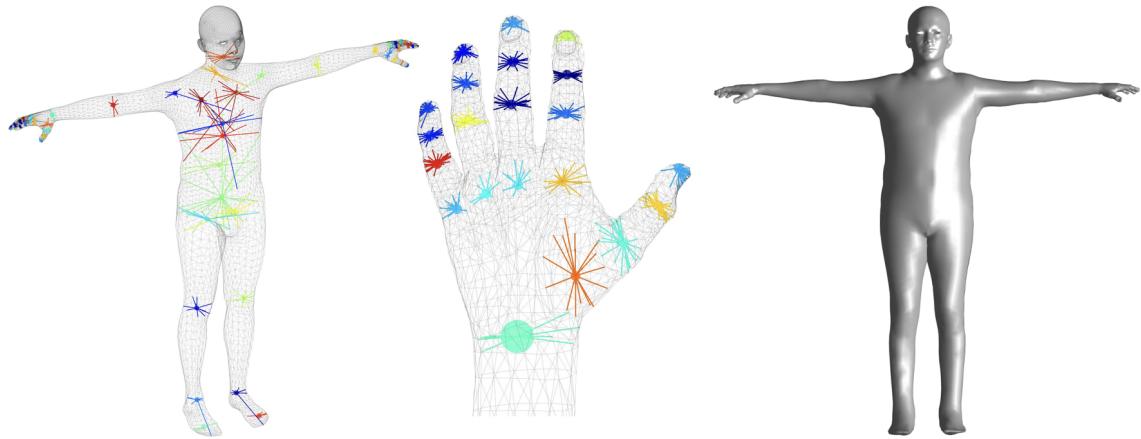


Figure 6: (Left and center) Joint locations and corresponding vertices (indicated by colors) after optimization. (Right) Mean shape of Adam model.

- [2] H. Joo, H. S. Park, and Y. Sheikh. Map visibility estimation for large-scale dynamic 3d reconstruction. In *CVPR*, 2014.