HiFECap: Monocular High-Fidelity and Expressive Capture of Human Performances – Supplemental Document –

Yue Jiang
yue.jiang@aalto.fi

Marc Habermann
Vladislav Golyanik
Christian Theobalt
{mhaberma,golyanik,theobalt}@mpi-inf.mpg.de

Max Planck Institute for Informatics
Saarland Informatics Campus

In the following, we show more qualitative results and visualize our undeformed template meshes (Sec. 1), provide details about the network architectures (Sec. 2), and give more explanations regarding the individual loss terms (Sec. 3). Last, we provide more details about the training process (Sec. 4) and limitations as well as future work (Sec. 5).

1 Qualitative Results and Template Meshes

We visualize some additional qualitative results in Fig. 1, which include different actors, clothing styles, body motions, backgrounds, facial expressions, and hand gestures. The results show that our reconstruction can jointly capture facial expressions, hand poses, and also high-frequency details, such as deforming wrinkles on the clothes. We show our final templates after further adding hands and face in Fig. 2.

2 Ablation Study

We propose a combined image- and graph-convolutional architecture, called *DisplaceNet*, to regress the per-vertex displacement field, which captures the high-frequency details on the nonrigid deforming surface. In Fig. 3, we compare our method to a purely image-based convolutional architecture [3] and a baseline, which does not use any dense per-vertex displacements. The comparison results demonstrate that our design achieves the best result as indicated by the per-vertex error and recovers richer surface details such as the wrinkles of the clothing. Note that especially in those regions, the baselines fall short in recovering the geometric details. Thus, the highest error for the baselines can be observed in these regions while our method shows a significantly lower error.

Next, in Tab. 2, we evaluate our design choices concerning the novel displacement network architecture and the proposed supervision strategy. To this end, we first replace the *DisplaceNet* architecture with a ResNet50 that has a fully connected backbone predicting

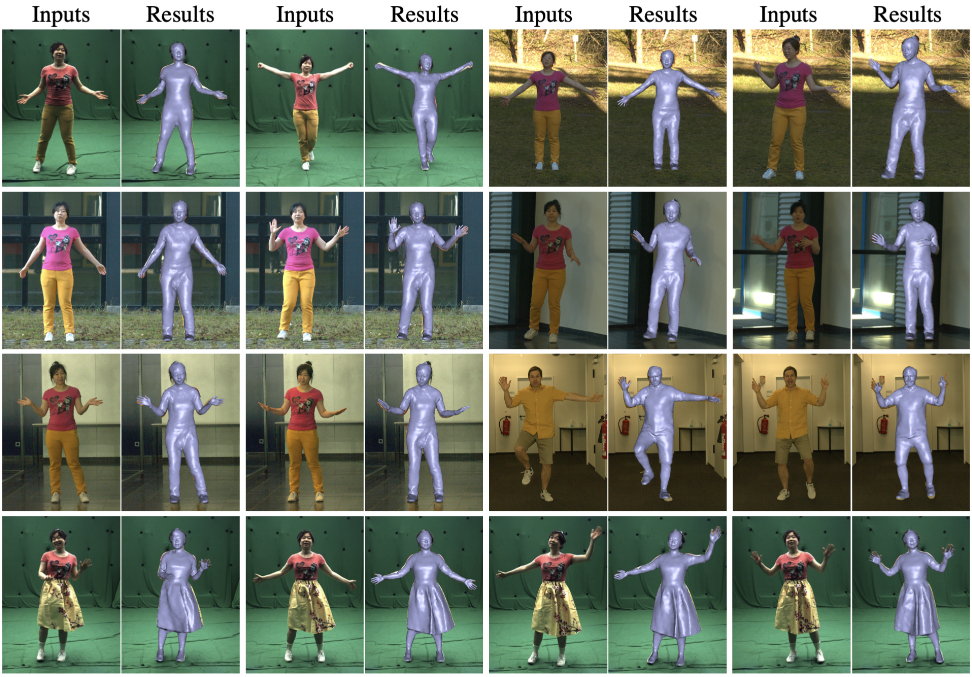

Figure 1: Qualitative results. We show results for subjects with different types of apparel, poses, and backgrounds. We show the overlay of our reconstruction onto the input frames. Note that our method not only precisely overlays onto the input images, but also captures the wrinkle patterns nicely.

the displacement field directly (second result row). The result shows that our proposed architecture improves the quality of deformation. This is due to the fact, that the receptive field of our proposed architecture is more local and, thus, the deep features attached to the graph are better suited for predicting these local displacement vectors compared to the ResNet50 architecture, which in its last fully connected layer creates a fully global dependency between the spatial image features.

To show the importance of the different loss terms and the proposed visibility-aware and rigidity-aware vertex displacement network *DisplaceNet*, we conduct an ablation study where we remove each of these components from the non-rigid training process. We report the average Chamfer distance and the average symmetric Hausdorff distance and visualize their per-vertex errors. We can see that removing any component from our carefully designed combination reduces the quality of the estimated deformations, which confirms that our design choice indeed provides the best performance. Furthermore, we show an error comparison of how the visibility-modulated feature map improves the overall reconstruction performance in Fig. 4.

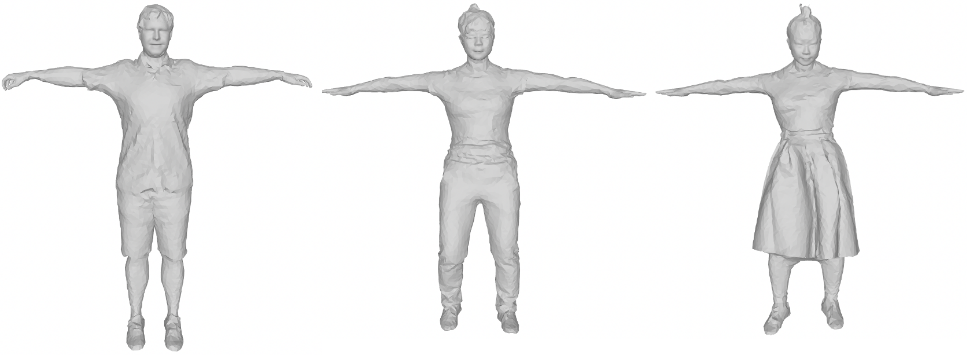

Figure 2: Examples of the final template meshes with hands and face.

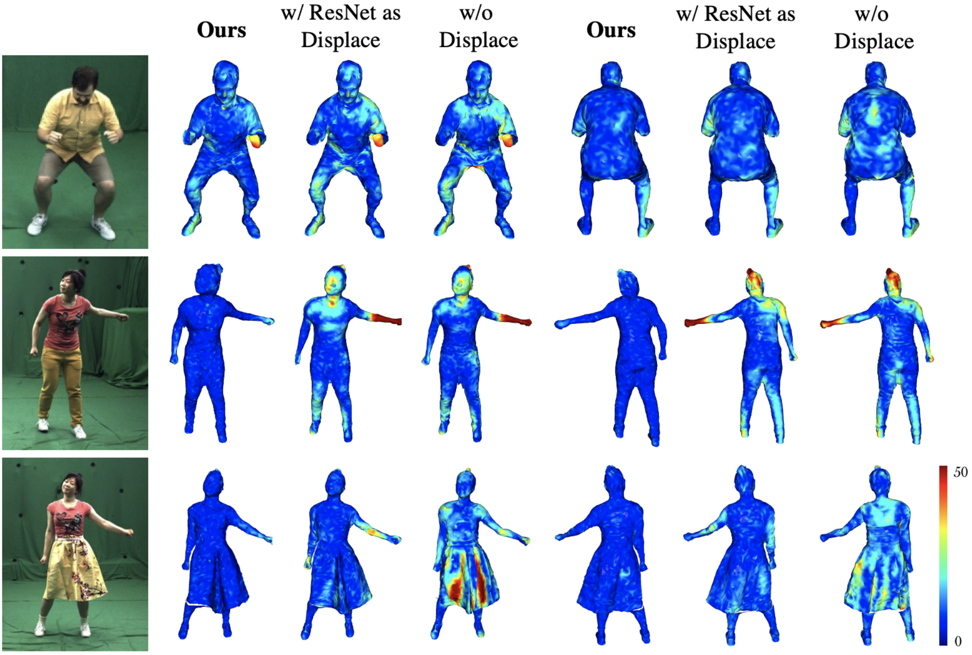

Figure 3: Ablation studies. Visualization of the per-vertex error (MSE) of our method, our method without *DisplaceNet*, and another baseline where we replace our proposed architecture with a pure image-convolutional one. Our method has significantly lower error than the baselines, which validates our design choices.

3 Details about the Loss Terms

3.1 Silhouette Loss

We define the multi-view silhouette loss for the input frame f as

$$\mathcal{L}_{\text{sil}}(\mathbf{V}) = \sum_{c=1}^{C} \sum_{i \in \mathcal{B}_c} d_{c,i} \|\mathcal{D}_c(\pi_c(\mathbf{V}_i))\|^2. \tag{1}$$

Method			Sequence 1		Sequence 2		Sequence 3	
Rend.	Chamfer	Displ.	**Chamfer↓**	**Hausd.↓**	**Chamfer↓**	**Hausd.↓**	**Chamfer↓**	**Hausd.↓**
✓	✓	✓	**7.25**	**39.72**	**9.21**	**40.07**	**7.26**	**32.69**
✓	✓	+	9.17	41.82	9.62	51.70	8.75	38.91
✗	✓	✓	15.66	51.21	10.88	71.53	9.80	46.82
✓	✗	✓	18.21	65.45	12.23	93.64	15.43	89.77
✓	✓	✗	18.34	65.85	12.70	88.78	16.32	75.47
✗	✗	✓	18.47	72.81	12.92	99.16	16.73	93.58
✗	✓	✗	18.76	69.19	13.17	97.79	17.52	80.71
✓	✗	✗	20.15	75.46	13.45	105.62	17.81	96.71
✗	✗	✗	21.09	77.83	14.32	107.21	17.88	98.49

Table 1: Quantitative ablation study. We evaluate the necessity of the different loss terms and the visibility-aware and rigidity-aware displacement network in terms of Chamfer distance and Hausdorff distance compared to the ground truth meshes. "+" indicates that we use a ResNet architecture instead of our proposed *DisplaceNet* architecture. Note that our specific design choices provide the best results in terms of surface tracking accuracy compared to the baselines.

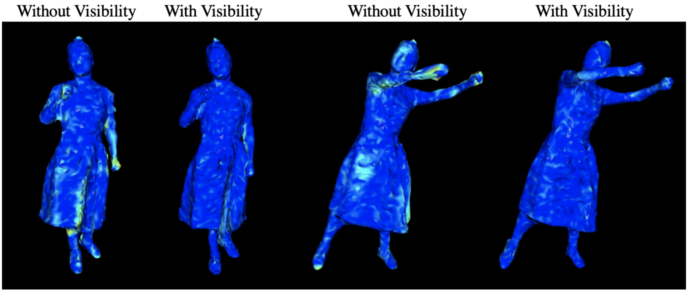

Figure 4: Error comparison showing how the visibility-modulated feature map improves the overall reconstruction performance.

The silhouette loss $\mathcal{L}_{\text{sil}}$ encourages the deformed mesh to fit the multi-view image silhouettes from all the cameras. $\mathbf{V}_i$ is the position of the i-th vertex of the posed and the deformed template mesh obtained by the deformation layer. π_c, the perspective camera projection of the camera c, projects $\mathbf{V}_i$ onto the image plane. $d_{c,i} \in \{-1,+1\}$ is a directional weight to encourage the optimization to follow the right direction in the distance field [2] and $\mathcal{B}_c$ is the set of boundary vertices computed by rendering the depth maps and detecting whether the projected vertex $\pi_c(\mathbf{V}_i)$ is near the intersection of background and foreground pixels. Here, $\mathcal{D}_c$ is the distance transform image of camera c.

3.2 Marker Loss

Similar to the 2D keypoint loss for *PoseNet*, we define the marker loss as the weighted squared loss of the difference between the projected 3D marker on the posed mesh from the camera c and the detected 2D marker on the input image. The loss is defines as

$$\mathcal{L}_{\text{mk}} = \sum_c \sum_m \beta_{c,j} \|\pi_c(\mathbf{M}_j) - \mathbf{m}_{c,j}\|^2, \tag{2}$$

where $\pi_c(\mathbf{M}_j)$ is the projected 3D marker j on the posed mesh from the camera c, $\mathbf{m}_{c,j}$ is the detected 2D marker on the input image, and $\beta_{c,j}$ is its corresponding confidence value.

3.3 As-Rigid-As-Possible Deformation Loss

To ensure the smoothness of the mesh surface, we apply as-rigid-as-possible deformation loss $\mathcal{L}_{\text{arap}}$ [8] and use material-aware weights [2] to define the different levels of rigidity for embedded graph nodes (*e.g.,* clothing is assigned to a lower rigidity weight than the skin, and, thus, it has more freedom to deform than skin).

3.4 Differentiable Rendering Loss

The multi-view silhouette-based loss can be used to get the posed and coarsely deformed mesh. However, it does not allow to capture fine non-rigid deformations such as surface folds. Thus, we deploy a dense rendering loss

$$\mathcal{L}_{\text{dr}}(\mathbf{V}, L) = \sum_c \|R_c(\mathbf{V}, L_c, \mathcal{T}) - \mathcal{I}_c\|^2, \tag{3}$$

which takes the posed and deformed mesh and the static texture, renders it from various camera views, and compares the rendered images $R_c(\mathbf{V}, L, \mathcal{T})$ under the camera's lighting condition L with the corresponding input frame $\mathcal{I}_c$: To deal with different light conditions, camera optics and scene reflections, we optimize the lighting parameters for all the cameras used in the multi-camera sequences. Under the assumption of Lambertian material and smooth lighting environment, we apply the spherical harmonics (*SH*) lighting representation [6] to model the lighting condition L_c of each camera based on 27 lighting coefficients $\mathbf{l}_c \in \mathbb{R}^{9\times 3}$. There are nine SH basis functions and for each SH basis function SH_j, and we have $\mathbf{l}_{c,j} \in \mathbb{R}^3$ for each color channel. Then, the lighting condition for each camera $L_c(\mathbf{V}, \mathbf{l}_c)$ can be computed as

$$L_c(\mathbf{V}, \mathbf{l}_c) = \sum_{j=1}^{9} \mathbf{l}_{c,j} SH_j(n_c(\mathbf{V})), \tag{4}$$

where $n_c(\mathbf{V})$ represents the image pixel normal based on the underlying geometry. Then, the rendering function can be defined as

$$R_c(\mathbf{V}, \mathbf{l}_c, \mathcal{T}) = A_c(\mathbf{V}, \mathcal{T}) \cdot L_c(\mathbf{V}, \mathbf{l}_c). \tag{5}$$

Given the vertex positions $\mathbf{V}$, the lighting coefficients $\mathbf{l}_c$ the texture $\mathcal{T}$—and under the assumption of Lambertian surface—the rendering function equals to the dot product of the albedo of the projected surface $A_c(\mathbf{V}, \mathcal{T})$ and the lighting condition $L_c(\mathbf{V}, \mathbf{l}_c)$. The optimized lighting condition $\mathbf{l}_c$ for each camera c is then

$$\text{argmin}_{\mathbf{l}_c} \|R_c(\mathbf{V}, \mathbf{l}_c, \mathcal{T}) - \mathcal{I}_c\|^2. \tag{6}$$

To optimize the lighting coefficients across all the frames in the training sequence, we apply the Adam optimizer [4] to minimize $\|R_c(\mathbf{V}, \mathbf{l}_c, \mathcal{T}) - \mathcal{I}_c\|^2$ by iteratively sampling the training frames.

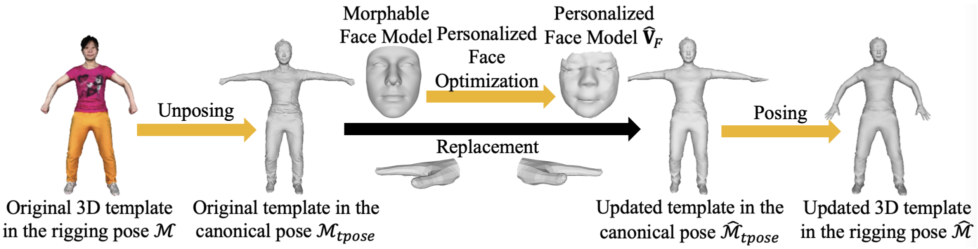
Morphable
Face Model
Personalized
Face
Optimization
Personalized
Face Model $\hat{\mathbf{V}}_F$
Unposing
Replacement
Posing
Original 3D template
in the rigging pose $\mathcal{M}$
Original template in the
canonical pose $\mathcal{M}_{tpose}$
Updated template in the
canonical pose $\widehat{\mathcal{M}}_{tpose}$
Updated 3D template
in the rigging pose $\widehat{\mathcal{M}}$

Figure 5: We apply Dual Quaternion Skinning to unpose the original template mesh to T-pose. We then replace the face and hands of the original template with the optimized models to obtain the updated template mesh that contains the controllable hand and face models.

3.5 Isometry Loss

We employ an Isometry loss

$$\mathcal{L}_{\text{iso}}(\mathbf{V}) = \sum_{i} \sum_{(i,j)\in E} \frac{r(\mathbf{V}_i,\mathbf{V}_j)}{|(i,j)\in E|} |||\mathbf{V}_i - \mathbf{V}_j|| - ||\mathbf{V}_{\text{Relax},i} - \mathbf{V}_{\text{Relax},j}|||^2 \tag{7}$$

on the mesh geometry to encourage consistent edge lengths with respect to the undeformed template mesh. E represents all the edges on the mesh, $\mathbf{V}_i$ is the vertex position of the i-th vertex on the posed and deformed mesh and $\mathbf{V}_j$ where $(i,j)\in E$, is any vertex connecting to $\mathbf{V}_i$. $\mathbf{V}_{\text{Relax},i}$ and $\mathbf{V}_{\text{Relax},j}$ are their corresponding vertices on the original (unposed and undeformed) template mesh. $r(\mathbf{V}_i,\mathbf{V}_j)$ are the predefined per-edge rigidity weights based on the rigidity of body parts similar to the per-vertex rigidity weights of the template. Lower rigidity weights are assigned to materials with less non-rigid deformations. For example, the skin has less deformation ability, so we assign weight 200.0 to the face and 50.0 to other parts of the skin. For more deformable parts, we assign lower weights, *e.g.,* we assign weight 1.0 to dresses, 2.0 to upper clothes, and 2.5 to pants. The isometry loss encourages that the length of every edge in the posed and deformed mesh is similar to the corresponding edge in the original unposed and undeformed mesh to penalize large stretching of the surface.

3.6 Laplacian Loss

To avoid geometry distortion, we regularize the mesh with a Laplacian regularization term

$$\mathcal{L}_{\text{lap}}(V) = \sum_{i} w_i || |(i,j)\in E| (\mathbf{V}_i - \mathbf{V}_{\text{Relax},i}) - \sum_{(i,j)\in E} (\mathbf{V}_j - \mathbf{V}_{\text{Relax},j}) ||^2 \tag{8}$$

where w_i represents the spatially varying regularization weights for each vertex on the mesh. The Laplacian loss term ensures that adding the vertex displacements does not largely change the Laplacian of the mesh so that the mesh has a smooth surface.

3.7 Tracking of Hands and Face

Fig. 5 shows the process of replacing the face and hand regions on the template mesh with a parametric 3D face model [1] and hand model [7] to enable the joint tracking of hands and face.

4 Training Details

We train and test our approach on an NVIDIA Quadro RTX8000 GPUs with 48GB of memory. Our training process includes 4 different stages, *i.e.,*

1. Training *PoseNet* with 2D keypoint Loss;
2. Training *EDefNet* with silhouette loss and the regularization terms;
3. Training *EDefNet* with all the loss terms;
4. Training *DisplaceNet* with all the loss terms.

We use the Adam optimizer [4] for all the training stages. All the network architectures are implemented in the Tensorflow framework. We train each stage for 120k iterations with a learning rate of 10^{-5}. The training data are multi-view videos captured in studio with around 100 cameras. Due to the limited memory and training time, instead of training on all the multi-view inputs, for each iteration, we randomly sample 30 camera views for all the multi-view loss terms. The entire training process takes about 4 days in total if we train on 4 GPUs in parallel with a batch size of 8. At training time, the system requires multi-view videos, however, at test time, it only takes single-view videos. Thus, at test time, our method takes about 0.3 seconds on a single GPU.

5 Discussion and Future Work

Although our approach is the first to jointly track the human pose, facial expressions, hand gestures, and non-rigid clothing solely from a monocular video, it still has some limitations that open up future work in this direction. Hands and face capture still has some room to improve. Our input video captures the entire body without any additional information about the hands and face, and we further localize the hands and face to capture these parts. Thus, existing face-only and hand-only methods cannot be directly applied to our setting as our method requires cropping and alignments of the human's hands, face, and body. Concerning full-body methods, we show the superior performance of hands and face capture results compared to previous work as shown in our qualitative results (Zhou et al. [9] and VIBE [5]). It is hard to compare face and hand results quantitatively because it is very hard to obtain the ground truth meshes under this setting (*i.e.,* marker-less multi-view full-body capture). Also, extreme body poses and hand gestures that are too different from the training motions can lead to erroneous pose estimation and deformation. To tackle this problem, we envision that in the future the overall robustness of our method can be further improved by jointly training on, both, in-studio multi-view data and in-the-wild monocular data. Moreover, tighter integration of physics into the capture process could also be interesting, i.e. physically more accurate skeletal pose estimation and surface deformations. For now, our method is a per-frame approach and we mainly focused on the expressiveness of our output, *i.e.,* capturing all aspects of the human. However, there are many more directions to explore in this field, one of them being the temporal aspect. Thus, future work could involve modeling the temporal domain more explicitly in the neural network architecture. Finally, we believe that our method has the potential to run in real-time, but for this, the foreground segmentation and 2D keypoint detection have to be more tightly linked to the regression of the character parameters, i.e. skeletal pose, surface deformations, hand gestures, and facial expressions.

References

[1] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *ACM Transactions on Graphics, (Proc. SIGGRAPH)*, 1999.

[2] Marc Habermann, Weipeng Xu, Michael Zollhöfer, Gerard Pons-Moll, and Christian Theobalt. Livecap: Real-time human performance capture from monocular video. *ACM Transactions on Graphics*, 2019.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[4] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, 2015.

[5] Muhammed Kocabas, Nikos Athanasiou, and Michael J. Black. VIBE: Video inference for human body pose and shape estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[6] Claus Müller. *Spherical harmonics*. 2006.

[7] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 2017.

[8] Olga Sorkine and Marc Alexa. As-rigid-as-possible surface modeling. In *Proceedings of the Fifth Eurographics Symposium on Geometry Processing (SGP)*, 2007.

[9] Yuxiao Zhou, Marc Habermann, Ikhsanul Habibie, Ayush Tewari, Christian Theobalt, and Feng Xu. Monocular real-time full body capture with inter-part correlations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.