1 Motivation: SMPL does not work for infants

Infants and adults have different body proportions. Thus, simply scaling the SMPL [6] model - which was learned from adult subjects - to infant size does not provide satisfactory results. This becomes obvious by processing an rgb image of an infant with the publicly available Keep it SMPL [2] method (see Fig. 1a). As our new SMIL model is compatible with SMPL, we can replace the SMPL model in [2] with our SMIL model - Keep it SMIL - thus obtaining the results shown in Fig. 1b.

2 Registration Process

In this section we provide implementation details for the registration process described in Sec. 3 of the main paper. We first detail the optimized energy, and then detail the optimization process.

2.1 Registration Energy

The main energy being optimized w.r.t. shape $\beta$ and pose $\theta$ parameters is

$$E(\beta, \theta) = E_{\text{data}} + E_{\text{lim}} + E_{\text{table}} + E_{\text{sm}} + E_{\text{sc}} + E_{\beta} + E_{\theta}. \quad (1)$$

We note the scan points as $P$. Using the method described in [9], $P$ is segmented into the scan points belonging to the skin ($P_{\text{skin}}$) and the ones belonging to the onesie or the diaper ($P_{\text{cloth}}$).
Supplementary Material

Fig. 1: Comparison of (a) Keep it SMPL and (b) Keep it SMIL.

Data term.

The data term $E_{data}$ consists of two different terms:

$$E_{data} = E_{s2m} + \lambda_{m2s}E_{m2s}.$$  \hspace{1cm} (2)

$E_{s2m}$ accounts for the distance of the scan points to the model mesh and $E_{m2s}$ accounts for the distance of the visible model points to the scan points.

$E_{m2s}$ can be written as

$$E_{m2s}(M, P) = \sum_{m_i \in \text{vis}(M)} \rho(\min_{v \in P} ||(m_i, v)||),$$  \hspace{1cm} (3)

where $M$ denotes the model surface and $\rho$ is the robust GemanMcClure function $[4]$. The function vis$(M)$ selects the visible model vertices. The visibility is computed using the Kinect V1 camera calibration.

$E_{s2m}$ consists also of two terms,

$$E_{s2m} = \lambda_{\text{skin}}E_{\text{skin}} + \lambda_{\text{cloth}}E_{\text{cloth}}.$$  \hspace{1cm} (4)

$E_{\text{skin}}$ enforces the skin points to be close to the model mesh and $E_{\text{cloth}}$ enforces the cloth points to be outside the model mesh.

The skin term can be written as

$$E_{\text{skin}}(M, P_{\text{skin}}, W) = \sum_{v_i \in P_{\text{skin}}} W_i \rho(\text{dist}(v_i, M)),$$  \hspace{1cm} (5)

where $W$ is the weight of each skin vertex and $\rho$ is the robust GemanMcClure function.
where \( W \) are the skin weights. For their computation as well as for the details of \( E_{\text{cloth}} \) we refer the reader to [13].

The term \( E_{c2m} \) used for the evaluation in the main paper does not use the GemanMcClure function, as we are interested in the actual euclidean distances. Moreover, all scan points are considered to be labeled as skin.

### Landmark term.

The landmark term \( E_{\text{lm}} \) is similar to Eq. 2 from [2]. Instead of skeleton joints, we use estimated 2D face landmarks (nose, eyes outlines and mouth outline) [12] as well as hand landmarks (knuckles) [11]. Of the estimated body pose [3], we only use eye and ear landmarks in this term, which help for correcting head rotation for extreme profile faces where facial landmark estimation fails. We note the set of all markers as \( L \).

Hand landmarks are used for aligning coarse hand rotation, since the sensor accuracy doesn’t allow fitting finger details. Notice that the estimated body joints positions are only used for initialization in Sec. 3.

The 3D model points corresponding to the above landmarks were manually selected through visual inspection. They are projected into the image domain using the camera calibration in order to compute the final 2D distances.

The landmark term is then

\[
E_{\text{lm}} = \lambda_{\text{lm}} \sum_{l \in L} c_l \rho(l_M - l_{\text{est}}),
\]

where \( c_l \) denotes the confidence of an estimated landmark 2D location \( l_{\text{est}} \), and \( l_M \) is the model landmark location projected in 2D using the camera calibration.

### Table term.

We note the table plane as \( \Pi \). The table energy has two terms: \( E_{\text{in}} \) prevents the model vertices \( M \) from lying inside the table (i.e. behind the estimated table plane), by applying a quadratic error term on points lying inside the table. \( E_{\text{close}} \) acts as a gravity term, by pulling the model vertices \( M \) which are close to the table towards the table, by applying a robust GemanMcClure penalty function to the model points which are close to the table.

We write the table energy term as

\[
E_{\text{table}} = \lambda_{\text{in}} E_{\text{in}} + \lambda_{\text{close}} E_{\text{close}},
\]

with

\[
E_{\text{in}}(M) = \sum_{x_i \in M} \delta_i^\text{in}(x_i) \text{dist}(x_i, \Pi)^2,
\]

and

\[
E_{\text{close}}(M) = \sum_{x_i \in M} \delta_i^{\text{close}}(x_i) \rho(\text{dist}(x_i, \Pi)),
\]
where $\delta_i^{in}$ is an indicator function, returning 1 if $x_i$ lies inside the table (behind the estimated table plane), or 0 otherwise. $\delta_i^{close}$ is an indicator function, returning 1 if $x_i$ is close to the table (dist less than 3 cm) and faces away from the camera, or 0 otherwise.

To account for soft tissue deformations of the back, which SMIL does not model, we allow the model to virtually penetrate the table. We effectively enforce this by translating the table plane by 0.5 cm, i.e. pushing the virtual table back.

Other terms.

The temporal pose smoothness term $E_{sm}$ is the same as in Eq. 21 in [10] and penalizes large differences between the current pose $\theta$ and the pose from the last processed frame $\theta'$.

The penalty for model self intersections $E_{sc}$ and the shape prior term $E_\beta$ are the same as in Eq. 6 and Eq. 7 in [2] respectively.

The SMIL pose prior consists of mean and covariance that were learned from 37K sample poses. $E_\theta$ penalizes the squared Mahalanobis distance between $\theta$ and the pose prior, as described in [1].

### 2.2 Registration Optimization

To compute the registrations of a sequence we start by computing an initial shape using 5 frames. In this first step we only optimize for the shape parameters $\beta$. This shape will be kept fixed and used later on as a regularizer. Experiments showed that otherwise the shape excessively deforms in order to explain occlusions in the optimization process.

With the initial shape fixed, we compute the poses for all the frames in the sequence, i.e. we optimize the following energy w.r.t. pose parameters $\theta$ and the global translation $t$:

$$E(\theta, t) = E_{data} + E_{lm} + E_{sm} + E_{sc} + E_\theta.$$  \hspace{1cm} (10)

Notice that this energy is equal to Eq. 1 without $E_{table}$ and $E_{beta}$. We note the computed posed shape at frame $f$ as $S_f$.

In the last step we compute the registration meshes $R_f$ and allow the model vertices $v \in R_f$ to freely deform to best explain the input data. We optimize w.r.t. $v$ the energy

$$E(v) = E_{data} + E_{lm} + E_{table} + E_{cpl},$$ \hspace{1cm} (11)

where $E_{cpl}$ is used to keep the registration edges close to the edges of the initial shape. We use the same energy term as Eq. 8 from [1]

$$E_{cpl}(R_f, S_f) = \lambda_{cpl} \sum_{e \in V'} ||(AR)_e - (AS)_e||_F^2,$$ \hspace{1cm} (12)
where $V'$ denotes the edges of the model mesh. $AR$ and $AS$ are edge vectors of the triangles of $R_f$ and $S_f$, and $e$ indexes the edges. The results of these optimizations are the final registrations.

All energies are minimized using a gradient-based dogleg minimization method [8] with OpenDR [7] and Chumpy [5].

**Energy weights:**

For each fit we use the same energy weights for all sequences. For Eq. 1 and Eq. 10 we use the weight values: $\lambda_{\text{skin}} = 800$, $\lambda_{\text{cloth}} = 300$, $\lambda_{\text{m2s}} = 400$, $\lambda_{\text{lm}} = 1$, $\lambda_{\text{table}} = 10000$, $\lambda_{\text{sm}} = 800$, $\lambda_{\text{sc}} = 1$, $\lambda_{\beta} = 1$ and $\lambda_{\theta} = 0.15$.

For Eq. 11 we use the weight values: $\lambda_{\text{skin}} = 1000$, $\lambda_{\text{cloth}} = 500$, $\lambda_{\text{m2s}} = 1000$, $\lambda_{\text{lm}} = 0.03$, $\lambda_{\text{table}} = 10000$ and $\lambda_{\text{cpl}} = 1$.

### 3 Initialization

The initialization energy $E_{\text{init}}$ is used for a coarse estimation of shape and pose which is refined afterwards. It is

$$E_{\text{init}} = \lambda_{\text{j2d}} E_{\text{j2d}} + \lambda_{\theta} E_{\theta} + \lambda_{a} E_{a} + \lambda_{\beta} E_{\beta} + \lambda_{\text{sm}2m} E_{\text{sm}2m}$$  \hspace{1cm} (13)$$

where $E_{\text{j2d}}$ is similar to $E_{\text{lm}}$ with landmarks being 2D body joint positions. $E_{\theta}$ is a strong pose prior, $E_{a}(\theta) = \sum_{i} \exp(\theta_{i})$ is an angle limit term for knees and elbows and $E_{\beta}$ a shape prior. In contrast to Eq. 1 in [2], we omit the self intersection term, and add a scan-to-mesh distance term $E_{\text{sm}2m}$.

**Energy weights:** $\lambda_{\text{j2d}} = 6$, $\lambda_{\theta} = 10$, $\lambda_{a} = 30$, $\lambda_{\beta} = 1000$, $\lambda_{\text{sm}2m} = 30000$.

### 4 Personalized Shape

To compute the personalized shape we uniformly random sample 1 million points from the fusion cloud and proceed in two stages. In the first stage, we optimize $E = E_{\text{data}} + E_{\beta}$ w.r.t. the shape parameters $\beta$, and keep the pose $\theta$ fixed in the zero pose of the model (T-pose with legs and arms extended). We obtain an initial shape estimate that lies in the original shape space. In the second stage, we allow the model vertices to deviate from the shape space, but tie them to the shape from the first stage with a coupling term. We optimize $E = E_{\text{data}} + E_{\text{cpl}}$ w.r.t. the vertices.

**Energy weights:** $\lambda_{\text{skin}} = 100$, $\lambda_{\text{cloth}} = 100$, $\lambda_{\beta} = 0.5$ and $\lambda_{\text{cpl}} = 0.4$. 
5 Learning parameters

To learn the SMIL model we use the EMPCA algorithm provided in https://github.com/jakevdp/wpca, which computes weighted PCA with an iterative expectation-maximization approach.

The weights we use to train the model are: 3 for the scan points labeled as skin ($P_{\text{skin}}$), 1 for the scan points labeled as clothing ($P_{\text{cloth}}$), and we compute smooth transition weights for the scan points near the cloth boundaries using the skin weights $W$ computed using the method in [13]. In Fig. 2 we display the weights used for the weighted PCA on a sample frame.

![Image](a) Original rgb image. b) Smooth weights used for weighted PCA. White points have a weight value of 3 (high weight), red point have a weight value of 1 (low weight). The smooth transition is computed using the skin weights $W$. 
6 GMA Case Study Rating Results

In our case-study, we observe that $R_1$’s and $R_2$’s ratings only agree on $\approx 65\%$ of the original RGB videos $V_{rgb}$ although both are very experienced. In Fig. 3 we present the histogram of signed differences between the ratings of all raters. In the first row, we show the differences between $R_1$ and $R_2$. We can see that the peak of what looks like a normal distribution is centered at one instead of zero. Our interpretation is that $R_2$ has the same tendency as $R_1$, but $R_2$’s ratings are shifted by 1. The comparison of the more experienced raters with $R_3$ shows a more diffuse distribution of differences, indicating that $R_3$’s ratings are more inconsistent.

![Histogram of signed differences](image)

Fig. 3: Histogram of the signed differences for the original RGB videos $V_{rgb}$ between: Top: $R_1$ and $R_2$, Middle: $R_1$ and $R_3$, Bottom: $R_2$ and $R_3$.

We further analyze the signed distances between the ratings of the original RGB videos $V_{rgb}$ and the different synthetic sequences of $R_1$ (Fig. 4) and $R_2$ (Fig. 5). Interestingly, we observe that for $R_2$, $V_{reg}$, $V_{other}$ and $V_{mean}$ have in general healthier ratings, whereas $V_{large}$ have less healthy ratings, as shown in Fig. 5. Future work will study which non-motion related factors (body shape, texture, lighting) most affect the GMA ratings.
Fig. 4: Histograms of signed differences between $R_1$’s ratings of $V_{rgb}$ and $R_1$’s ratings of $V_{reg}$, $V_{other}$, $V_{mean}$ and $V_{large}$.

7 Samples and failure cases

In this section we show further samples of the input data, as well as the pre-processing results and the final registrations.

7.1 Registration samples

In Fig. 6, we show input RGB images, 3D point clouds, and registration results for three sample frames.
Fig. 5: Histograms of signed differences between $R_2$’s ratings of $V_{rgb}$ and $R_2$’s ratings of $V_{reg}$, $V_{other}$, $V_{mean}$ and $V_{large}$.

7.2 Preprocessing sample

A sample of the preprocessing steps 2D pose estimation, background removal, and clothing segmentation is displayed in Fig. 7.
7.3 Failure cases

Our energy has the interpenetration penalty $E_{sc}$, but, despite it, we observed few cases where the legs interpenetrated, as in the first example presented in Fig. 8. The registration of all sequences is time consuming (between 10 and 30 seconds per frame), so rerunning the full 200K registrations many times to optimize the parameters was not feasible. Of course, that would require to split the data into a training, test and validation set. The parameters were manually selected in order to balance the different terms of the energy, and by visually inspecting the results of some sequences. Further manual adjustment of the $E_{sc}$ weight could fix these rare cases. In the second example, the right knee is twisted in an unnatural way after the right foot was completely occluded. When the foot is visible again, the pose recovers (knee twisted for 5-6 seconds). Similar to the first failure case, a higher weight on the pose prior would prevent such cases.
Fig. 7: Preprocessing. a) Pose estimation, with face and hand landmarks (left finger estimation fails due to low resolution and occluded fingers). b) Unsegmented input point cloud. c) Result of foreground segmentation based on estimated table plane. d) Result of diaper segmentation. Blue: skin, red: diaper.

but finding the perfect weight which completely forbids all illegal poses while allowing all legal poses is not an easy task.

Notice that although these impossible poses can (and possibly do) affect the agreement between the ratings of the original RGB images and the synthetic sequences, it does not explain the different ratings among the synthetic sequences with different shapes. The artifacts are present in all synthetic sequences.
Fig. 8: Failure cases: a) RGB image. b) Registered SMIL model to the point cloud. c) Textured point cloud (side view) d) Overlay of registration and textured point cloud (side view). e) Registration (side view).

References