User-Specific Hand Modeling from Monocular Depth Sequences

Jonathan Taylor[†], Richard Stebbing[†‡], Varun Ramakrishna[†♭], Cem Keskin[†],
Jamie Shotton[†] , Shahram Izadi[†], Aaron Hertzmann[⋆], Andrew Fitzgibbon[†]

Microsoft Research[†] University of Oxford[‡] Carnegie Mellon University[♭] Adobe[⋆]

1. Contents

This supplementary material includes some additional detail on the linear blend skinning model, the subdivision surface model, and the parameter values used for the experiments.

2. Parameter settings

The following parameter settings, unless otherwise stated, define the final energy that we optimize in the experiments described in the paper.

σ_x	0.3
λ_{norm}	0.1
λ_{core}	3
λ_{inst}	2.5
$\lambda_{\text{skeleton}}$	0.5
λ_{L2}	0.05
λ_{prior}	10
λ_{motion}	300
λ_{scale}	1000

3. Linear Blend Skinning

In this section we specify the details of our chosen skinning function $\mathcal{P}(V;\theta,\kappa)$. The skeleton that we use consists of B bones organized into a tree structure in which the first bone is the root and $\pi(b)$ indicates the parent of each node $b \in \{2,...,B\}$. Each bone b has an attached local coordinate system related to its parent's (or the world's in the case of the root) by a transformation $T_b(\theta,\kappa)$ consisting of a rotation and a 3D translation. The rotation is specified using three exponential map coordinates contained in θ. The translation is simply a scaling $\beta_b(\kappa)$ of the translation $\hat{t}_b$ in our template.

To define the function $G_b(\theta,\kappa)$, we then simply compose the transformations in a recursive manner up the skeleton as

$$G_1(\theta,\kappa) = T_1(\theta,\kappa) \tag{1}$$

$$G_b(\theta,\kappa) = G_{\pi(b)}(\theta,\kappa) * T_b(\theta,\kappa) \tag{2}$$

where the $*$ operator indicates the composition of transformations. For our purposes, the global transformation $G_{\text{glob}}(\theta,\kappa)$ is composed of an isotropic scaling encoded in κ and a rigid transform encoded in θ.

4. Subdivision Surfaces

A modified Loop subdivision surface is used to *explicitly* model the surface of the hand. The advantages of an explicit surface representation are that the surface topology is fixed and that the surface is completely defined by a fixed number of control vertices $\{\mathbf{v}_m\}_{m=1}^M$. The advantages of subdivision surfaces are that they are continuous and have smoothly-varying normals.

A Loop subdivision surface is defined completely by a set of control vertices configured in a triangular mesh. The surface is described in sections by *patches*, where each patch is defined by a subset of the control vertices and the local patch topology. For patch p the surface patch is given by the set of points $\{\mathcal{S}_p(\mathbf{u};\mathbf{v}_{i_1^p},\ldots,\mathbf{v}_{i_{I_p}^p})\colon \mathbf{u}\in\triangle\}$, where $\mathcal{S}_p$ is the patch position function, $\triangle = \{(u,v)\colon 0\le u\le 1, 0\le v\le 1-u\}$ is the unit triangle, and i^p is the list of I_p vertex indices which contribute to patch p.

The local surface map $\mathcal{S}_p$ depends on the patch topology. For a *regular* patch, $\mathcal{S}_p(\mathbf{u};\cdot) = \mathbf{b}(\mathbf{u})^\top V$, where $\mathbf{b}(\mathbf{u})$ are the basis functions for a regular triangular spline [4] and $V \in \mathbb{R}^{12\times 3} = [\mathbf{v}_{i_1^p}\ldots\mathbf{v}_{i_{12}^p}]^\top$. For the purposes of optimization, the essential property of $\mathcal{S}$ is that it is linear in V and polynomial in $\mathbf{u}$.

For irregular or *extraordinary* patches the basis functions $\mathbf{b}(\mathbf{u})$ cannot be applied directly. Instead, Loop subdivision [2] is used which defines $\mathcal{S}_p(\mathbf{u};\cdot)$ as a piecewise smooth function consisting of an infinite number of regular triangular patches [4]. The process of evaluating $\mathcal{S}_p(\mathbf{u};\cdot)$ can be understood by considering the extraordinary patch which contains a single extraordinary vertex with valency $N=5$. By subdividing the control mesh, four child patches are created, three of which are regular and one of which has the same topology as the original patch. With reference to [4], $\mathcal{S}_p(\mathbf{u};\cdot)$ is given by:

$$\mathcal{S}_p(\mathbf{u};\cdot) = \mathbf{b}(t_{k,n}(\mathbf{u}))^\top P_k \bar{A} A^{n-1} V \tag{3}$$

where $\mathbf{u} = (u, v)$ and $n = \lfloor -\log_2(u + v) + 1 \rfloor$ is the required level of subdivision, $k \in \{0, 1, 2\}$ is the regular child patch index, $t_{k,n}$ transforms $\mathbf{u}$ to the child patch domain, and A, $\bar{A}$ and P_k are subdivision and "picking" matrices which are defined in [4]. In our implementation, code is automatically generated for up to 5 levels up subdivision, so this operation is no more expensive than the evaluation of ordinary patches.

A problem with (3) is that as $\mathbf{u} \to \mathbf{0}$ first derivatives either vanish ($N < 6$) or diverge ($N > 6$) and are numerically unstable, which is problematic for continuous optimisation over $\mathbf{u}$. Similar behaviour has been noted for Catmull-Clark subdivision surfaces, but reparameterisation is computationally expensive [1]. Instead, we closely approximate extraordinary patches with quartic Bezier triangles. Similar approximations have been performed for Catmull-Clark subdivision surfaces using bicubic B-splines [3]. While the resulting surface is no longer $\mathcal{C}^1$ continuous between extraordinary patches, the discontinuities are minor and neglible in practice.

References

[1] I. Boier-Martin and D. Zorin. Differentiable parameterization of Catmull-Clark subdivision surfaces. In *Proc. Eurographics Symp. on Geometry processing*, 2004.

[2] C. Loop. Smooth Subdivision Surfaces Based on Triangles. Master's thesis, The University of Utah, Aug. 1987.

[3] C. Loop and S. Schaefer. Approximating catmull-clark subdivision surfaces with bicubic patches. *ACM Trans. Graph.*, 27(1), 2008.

[4] J. Stam. Evaluation of Loop subdivision surfaces. 1998.