Progressive Acquisition of SVBRDF and Shape in Motion

Hyunho Ha[†] Seung-Hwan Baek[‡] Giljoo Nam[§] Min H. Kim[¶]

KAIST

1. Details of SVBRDF-Aware Motion Estimation

We formulate the SVBRDF-aware motion estimator as follows:

$$E_{\text{motion}}\left(\mathcal{W}^t\right) = E_{\text{depth}} + \lambda_{\text{dreg}} E_{\text{dreg}} + \lambda_{\text{pcolor}} E_{\text{pcolor}}, \quad (1)$$

where E_{depth} and E_{dreg} are the data term and its regularizer for geometry, E_{pcolor} is our novel data term for SVBRDF. λ_{dreg} and λ_{pcolor} are the corresponding weights.

Geometric Energy Similar to [NFS15], we formulate the conventional geometric energy term E_{depth} to ensure that the result of the optimization is consistent with the current frame depth image:

$$E_{\text{depth}}(\mathcal{W}^t) = \sum_{u \in \mathcal{P}^t_{\mathcal{D}}} \left([\tilde{\mathbf{N}}^t_{\mathcal{D}}(u)]^\intercal (\tilde{\mathbf{V}}^t_{\mathcal{D}}(u) - \mathbf{V}^t_{\mathcal{D}}(\tilde{u}_{\mathcal{D}}))\right)^2, \quad (2)$$

where $\mathcal{P}^t_{\mathcal{D}}$ is a set of visible pixels u obtained by rendering the warped static model to the current depth camera frame $\mathcal{D}^t$, $\tilde{\mathbf{V}}^t_{\mathcal{D}}:\mathbb{N}^2 \to \mathbb{R}^3$ is the vertex map of the warped mesh $\tilde{\mathcal{V}}^t_{\mathcal{K}}$ transformed by $\mathbf{T}^t_{\mathcal{K}\to\mathcal{D}}$ from $\mathcal{K}$ to current $\mathcal{D}^t$, $\tilde{\mathbf{N}}^t_{\mathcal{D}}:\mathbb{N}^2 \to \mathbb{R}^3$ is the normal map of $\tilde{\mathcal{V}}^t_{\mathcal{K}}$ transformed by $\mathbf{T}^t_{\mathcal{K}\to\mathcal{D}}$. $\tilde{u}_{\mathcal{D}}$=$P(\mathbf{K}_{\mathcal{D}}\tilde{\mathbf{V}}^t_{\mathcal{D}}(u))$ is a pixel in the current depth image $\mathbf{D}^t$ that corresponds to the rendered pixel u, $\mathbf{V}^t_{\mathcal{D}}(\tilde{u}_{\mathcal{D}})$=$\mathbf{K}^{-1}_{\mathcal{D}}\mathbf{D}^t(\tilde{u}_{\mathcal{D}})\left[\tilde{u}^\intercal_{\mathcal{D}}, 1\right]^\intercal$ is the vertex map of $\mathbf{D}^t$, $P(\cdot)$ is perspective projection, and $\mathbf{K}_{\mathcal{D}}$ is the intrinsic matrix of the depth camera.

Geometric Regularizer The regularization term E_{dreg} enforces local smoothness of motion and to prevent overfitting:

$$E_{\text{dreg}}(\mathcal{W}^t) = \sum_{i=1}^{n} \sum_{j=N(i)} \left\|\mathbf{T}^t_i \mathbf{q}_i - \mathbf{T}^t_j \mathbf{q}_i\right\|_2^2, \quad (3)$$

where $N(i)$ is the k-nearest neighbor of the ith node.

Color Energy Our motion estimation has a per-pixel color term E_{pcolor} that accounts for SVBRDF to enforce photometric consistency at the ith node in the camera space $\mathcal{C}$ as follows:

$$E_{\text{pcolor}}(\mathcal{W}^t) = \sum_{u \in \mathcal{P}^t_{\mathcal{C}}} \left\|\mathbf{C}^t(\tilde{u}_{\mathcal{C}}) - L^t\left(\tilde{\mathbf{O}}^t_{\mathcal{C}}(u); \tilde{\mathbf{N}}^t_{\mathcal{C}}(u), \tilde{\mathbf{V}}^t_{\mathcal{C}}(u)\right)\right\|_2^2, \quad (4)$$

where $\mathcal{P}^t_{\mathcal{C}}$ is a set of visible pixels u obtained by rendering the warped static model to the current color camera space $\mathcal{C}^t$, $\tilde{\mathbf{V}}^t_{\mathcal{C}}:\mathbb{N}^2 \to \mathbb{R}^3$ is the vertex map of the warped mesh $\tilde{\mathcal{V}}^t_{\mathcal{K}}$ transformed by $\mathbf{T}^t_{\mathcal{K}\to\mathcal{C}}$ from $\mathcal{K}$ to current $\mathcal{C}^t$, $\tilde{\mathbf{O}}^t_{\mathcal{C}}$ is the view direction of $\tilde{\mathbf{V}}^t_{\mathcal{C}}$ to the color camera, $\tilde{\mathbf{N}}^t_{\mathcal{C}}:\mathbb{N}^2 \to \mathbb{R}^3$ is the normal map of $\tilde{\mathcal{V}}^t_{\mathcal{K}}$ transformed by $\mathbf{T}^t_{\mathcal{K}\to\mathcal{C}}$, $\tilde{u}_{\mathcal{C}}$=$P(\mathbf{K}_{\mathcal{C}}\tilde{\mathbf{V}}^t_{\mathcal{C}}(u))$ is the pixel in the color image $\mathbf{C}^t$ that corresponds to u, $\mathbf{K}_{\mathcal{C}}$ is the intrinsic matrix of the color camera, and the reflected light L^t=B^t+S^t is rendered by Equation (2) in the main paper.

Shape Estimation Our shape estimation follows the traditional fusion method [NFS15]. We obtain a weighted average of the projective TSDF values for every voxel $\mathbf{x}$ using the estimated warp motion field. Given depth images $\mathbf{D}^t$, we transform voxel $\mathbf{x}$ to the depth camera space $\mathcal{D}$, yielding $\tilde{\mathbf{x}}^t_{\mathcal{D}}$. We then perform perspective projection to get corresponding depth pixel $\tilde{u}_{\mathbf{x}_{\mathcal{D}}}$, and its depth value $\mathbf{D}^t(\tilde{u}_{\mathbf{x}_{\mathcal{D}}})$. We calculate the TSDF distance $d_{\mathcal{T}} = \mathbf{D}^t(\tilde{u}_{\mathbf{x}_{\mathcal{D}}}) - \left[\tilde{\mathbf{x}}^t_{\mathcal{D}}\right]_z$ along the z-axis of $\mathcal{D}$ using depth and the z-axis value of $\tilde{\mathbf{x}}^t_{\mathcal{D}}$, denoted by $[\tilde{\mathbf{x}}^t_{\mathcal{D}}]_z$. When $d_{\mathcal{T}}$ is larger than the truncated value $-\tau$, we average the TSDF value $d^t_{\mathcal{T}}(\mathbf{x})$ with its weight $\omega^t_{\mathcal{T}}(\mathbf{x})$, which is proportional to distance between k-nearest nodes. Finally, we conduct the marching cube algorithm on the TSDF volume to create a polygonal mesh model per frame.

Implementation Details We set the resolution of the TSDF volume as 512×512×512, and each TSDF voxel is defined as a cube with a width of 2 mm. Each node in the deformation graph has a radius of 20 mm. For the ground truth data, we use 1.5 mm voxel size and 15 mm deformation graph radius. Truncated value for TSDF is 5 times bigger than voxel size. We precompute a discrete table of the BRDF function for predefined samples of parameters: The half-angle is sampled from 0 to 60 degrees with a step size of 1 degree. Then, the Ward BRDF model is precomputed with the values of α and ρ_s from 0.05 to 0.70 and 0.01 to 1 both with 0.01 intervals, respectively. For the simulation data, we use $m = 2$ number of cluster. For the real case, we use $m = 1$ number of cluster in the *Cloth* and the *Captain* scene, $m = 5$ for the *Bag* scene, $m = 7$ for the *Hoodie* scene. We use $k = 8$ for the k-nearest neighbor in the deformation graph for all results. We use $\lambda_{\text{dreg}} = 5$, $\lambda_{\text{pcolor}} = 0.00005$, $\lambda_{\text{treg}} = 100$, and $\lambda_{\text{sreg}} = 1$ for the regularizer in the optimization. We run 15 Gauss-Newton iterations for the SVBRDF-aware motion estimation with 10 iterations for the PCG. We run 8 Gauss-Newton iterations for the SVBRDF estimation with 5 iterations for PCG.

References

[NFS15] Newcombe R. A., Fox D., Seitz S. M.: Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Boston, Massachusetts, USA, 2015), pp. 343–352. 1