VolumeDeform: Real-time Volumetric Non-rigid Reconstruction

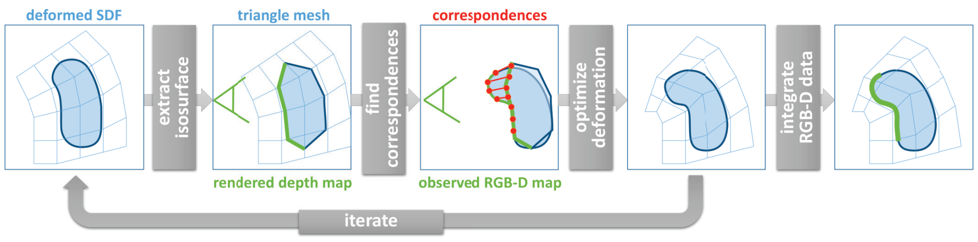
Matthias Innmann[1], Michael Zollhöfer[2], Matthias Nießner[3], Christian Theobalt[2], Marc Stamminger[1]

[1]University of Erlangen-Nuremberg, [2]Max-Planck-Institute for Informatics, [3]Stanford University

ABSTRACT

We present a novel approach for the reconstruction of **dynamic geometric shapes** using a single hand-held consumer-grade RGB-D sensor at real-time rates. Our method builds up the scene model **from scratch** during the scanning process, thus it does not require a pre-defined shape template to start with. Geometry and motion are parameterized in a unified manner by a **volumetric representation** that encodes a distance field of the surface geometry as well as the non-rigid space deformation.
The problem is tackled in **real-time** at the camera's capture rate using a data-parallel flip-flop optimization strategy. Our results demonstrate robust tracking even for fast motion and scenes that lack geometric features.

METHOD OVERVIEW



First, a deformed 3D mesh is extracted from the signed distance field using **Marching Cubes**. The mesh is rendered to obtain a depth map, which is used to generate **dense depth correspondences**. Next, we match **SIFT features** of the current frame with those of all previous frames. Based on all correspondences, we **optimize the deformation field** such that the resulting model explains the current depth and color observation. Finally, we **integrate** the RGB-D data of the current frame.

DEFORMATION ENERGY

To update the deformation field, **two distinct and complementary types of correspondences** between the current deformed shape and the new color and depth input are searched: for depth-image alignment, we perform a fast data-parallel projective lookup to obtain **dense depth correspondences**. Since in many situations depth features are not sufficient for robust tracking, we also use color information, and extract a **sparse set of robust color feature correspondences**. These also serve as global anchor points, since their descriptors are not modified over time. To reconstruct non-rigid surfaces in real time, we have to update the space deformation at sensor rate.

For simplicity of notation, we stack all unknowns of local deformations in a single vector:

$$\mathbf{X} = (\underbrace{\cdots, \mathbf{t}_i^T, \cdots}_{3|\mathcal{G}|\ coordinates} \,|\, \underbrace{\cdots, \mathbf{R}_i^T, \cdots}_{3|\mathcal{G}|\ angles})^T .$$

To achieve real-time performance, even for high-resolution grids, we cast finding the best parameters as a non-linear variational optimization problem. Based on these definitions, we define the following highly non-linear registration objective:

$$E_{total}(\mathbf{X}) = \underbrace{w_s E_{sparse}(\mathbf{X}) + w_d E_{dense}(\mathbf{X})}_{data\ \ term} + \underbrace{w_r E_{reg}(\mathbf{X})}_{prior\ \ term} ,$$

where

$$E_{dense}(\mathbf{X}) = \sum_{c=1}^{C} w_c \cdot \left[(\mathcal{S}(\hat{\mathbf{p}}_c) - \mathbf{p}_c^a)^T \cdot \mathbf{n}_c^a \right]^2 , \qquad E_{sparse}(\mathbf{X}) = \sum_{s=1}^{S} \| \mathcal{S}(\hat{\mathbf{f}}_s) - \mathbf{f}_s \|_2^2 ,$$

$$E_{reg}(\mathbf{X}) = \sum_{i \in \mathcal{M}} \sum_{j \in \mathcal{N}_i} \| (\mathbf{t}_i - \mathbf{t}_j) - \mathbf{R}_i (\hat{\mathbf{t}}_i - \hat{\mathbf{t}}_j) \|_2^2 .$$

PARALLEL ENERGY OPTIMIZATION

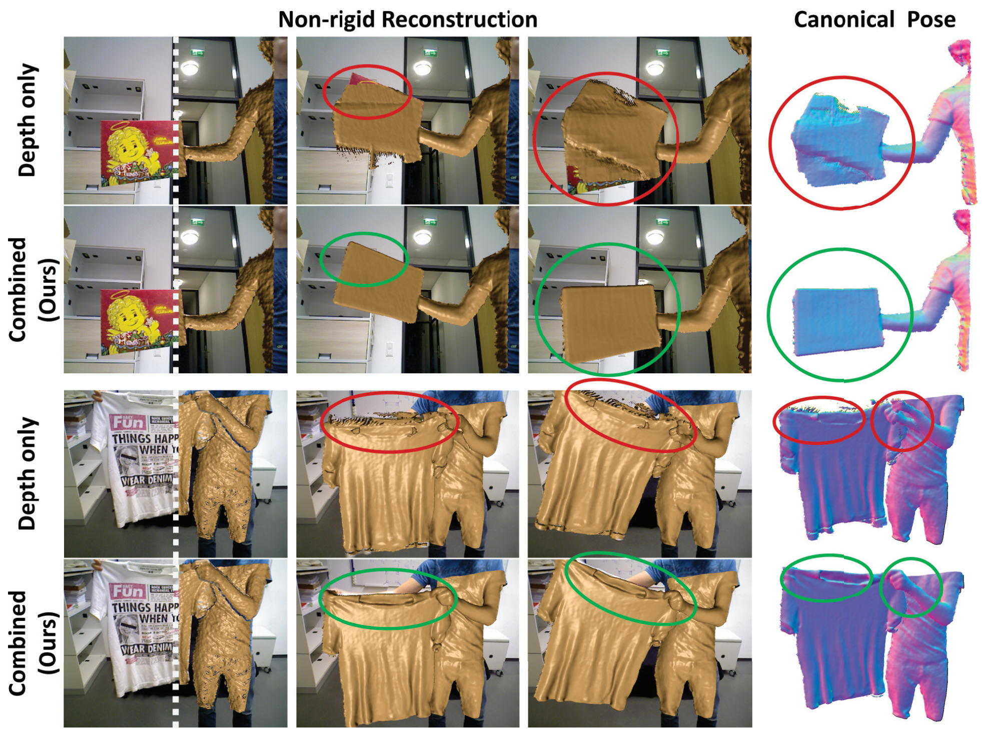
The non-linear optimization objective E_{total} can be split into **two independent subproblems** by employing an iterative flip-flop optimization strategy: first, the rotations $\mathbf{R}_i$ are fixed and we optimize for the best positions $\mathbf{t}_i$. Second, the positions $\mathbf{t}_i$ are considered constant and the rotations $\mathbf{R}_i$ are updated. These two step are iterated until convergence. The two resulting subproblems can both be solved in a highly efficient data-parallel manner:

1. We find the **optimal positions** by solving the linear system $(\mathbf{L} + \mathbf{B}^T\mathbf{B}) \cdot \mathbf{t} = \mathbf{b}$. Here, $\mathbf{L}$ is the Laplacian matrix, $\mathbf{B}$ encodes the point-point and point-plane constraints. The right-hand side $\mathbf{b}$ encodes the fixed rotations and the target points of the constraints. We solve the linear system of equations using a preconditioned conjugate gradient (PCG) solver. Since the matrix $\mathbf{L}$ is sparse, we compute it on-the-fly in each iteration step. In contrast, $\mathbf{B}^T\mathbf{B}$ has many non-zero entries. Thus, we pre-compute and cache $\mathbf{B}^T\mathbf{B}$.
2. We obtain the **best fitting rotation** based on Procrustes analysis with respect to the canonical pose. With our implementation, we can compute the best rotations for 400K voxels in 1.9ms.

RESULTS

COMPARISON TO STATE-OF-THE-ART

The following figure demonstrates the importance of our **sparse color tracker**:



We compare our method to a state-of-the-art **template based approach**. The results are comparable – without the necessity of a pre-scanned template: