Toward User-specific Tracking by Detection of Human Shapes in Multi-Cameras

Chun-Hao Huang[1], Edmond Boyer[2], Bibiana do Canto Angonese[1], Nassir Navab[1], Slobodan Ilic[3]
[1] Technische Universität München. [2] LJK-INRIA Grenoble Rhône-Alpes. [3] Siemens AG.

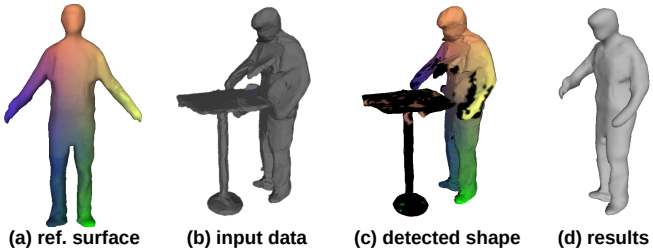

Figure 1: Given a reference surface (a), our method establishes reliable correspondences (c) between (a) and the input data (b). Correspondences guide the deformation of (a) toward the results in (d). Note that instead of tracking, our strategy detects user-specific shapes frame independently.

Human shape tracking consists in fitting a template model to temporal sequences of visual observations. It usually comprises an association step, that finds correspondences between the reference model and the input data, and a deformation step, that fits the model to the observations given the correspondences. Most current approaches find their common ground with the Iterative-Closest-Point (ICP) algorithm, which facilitates the association step with local distance considerations. However, when large deformations or outliers such as in Fig. 1(b) occur, discovering associations by only local distances is particularly difficult. Ambiguous correspondences result in erroneous solutions; the subsequent new associations are unreliable; errors propagate, and eventually break the tracking process. In this paper, we explore a discriminative alternative that leverages *random forests* to infer correspondences in *one shot* [5]. As demonstrated in Fig. 1, our framework 'detects' rather than tracks the subject, preventing errors from accumulation.

More formally, let $\mathcal{M} = (\mathbf{M}, \mathcal{T}_\mathcal{M})$ denotes a 3D reference mesh, where $\mathbf{M} = \{\mathbf{x}_v\}_{v=1}^{N_v} \subset \mathbb{R}^3$ are the locations of vertices v, and $\mathcal{T}_\mathcal{M}$ defines the triangles. Evolving $\mathcal{M}$ typically amounts to parameterizing $\mathbf{M}$ as a function of shape parameters Θ, namely, $\mathbf{M}(\Theta)$. We adopt a surface deformation framework that groups vertices into patches [1], and assign each of them a rigid body motion. Thus, Θ is the collection of rigid body motion of all patches, encoding the global shape of the surface. Given an observed visual hull $\mathcal{Y}^t = (\mathbf{Y}^t, \mathcal{T}_\mathcal{Y}^t)$, where $\mathbf{Y}^t = \{\mathbf{y}_i\}_{i=1}^{N_y} \subset \mathbb{R}^3$, the goal is to determine the optimal $\hat{\Theta}^t$ such that $\mathbf{M}^t = \mathbf{M}(\hat{\Theta}^t)$ resembles $\mathbf{Y}^t$ as much as possible. It typically boils down to two sub-problems:

1. finding correspondence pairs $\mathcal{C} = \{(i,v)\}$ between the vertex sets of $\mathcal{Y}$ and the vertex sets of $\mathcal{M}$, and

2. minimizing an energy E that describes the discrepancies between vertices in $\mathcal{C}$: $\hat{\Theta} = \arg\min_\Theta E(\Theta; \mathcal{C})$.

Our primary objective in this paper is to improve the first part. ICP-based generative approaches [1, 2] alternate between these two steps, refining $\mathcal{C}^t$ and Θ^t iteratively. The drawback however, is the requirement of close initializations ($\mathbf{M}^{t-1}$ has to be close to $\mathbf{Y}^t$), and the slow convergence.

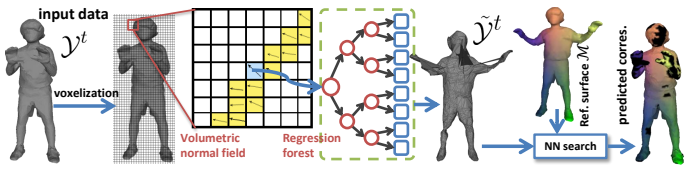

Figure 2: Pipeline of our framework. Correspondences are visualized in the same color. Black means no correspondence for that data point.

This is an extended abstract. The full paper is available at the Computer Vision Foundation webpage.

Using regression forests, we develop a different strategy that warps the input data $\mathcal{Y}$ to the reference mesh $\mathcal{M}$, denoted as $\tilde{\mathcal{Y}} = (\tilde{\mathbf{Y}}, \mathcal{T}_\mathcal{Y})$ and visualized as a triangular mesh in Fig. 2. If the warping is perfect, this mesh will look clean and resemble $\mathcal{M}$ as much as possible. Incorrect mapping results in huge edges between wrong correspondences. Vertex positions $\tilde{\mathbf{Y}}$ represent the locations of potential matches between $\mathcal{Y}$ and $\mathcal{M}$. Therefore, $\mathcal{C}$ can be built directly by nearest neighbor search between $\tilde{\mathbf{Y}}$ and $\mathbf{M}$, as illustrated in the pipeline in Fig. 2. Unlike those ICP-based methods, this process depends little on the proximity of successive frames and, therefore, it is more robust to drifting than pure ICP-based approaches.

Specifically, we consider this $\mathbb{R}^3 \to \mathbb{R}^3$ mapping as a composite one: $\mathbb{R}^3 \to \Omega_3 \to \mathbb{R}^3$. The former mapping is voxelization, while the latter is regression. Voxelization gives a volumetric field, where each voxel $\mathbf{v}$ either stores the surface normals, if it is occupied by the mesh, or stores the indicators to distinguish the internal/external empty space. This representation shares a similar spirit with implicit surface, e.g., truncated signed distance field, and we refer to it as volumetric normal field (VNF).

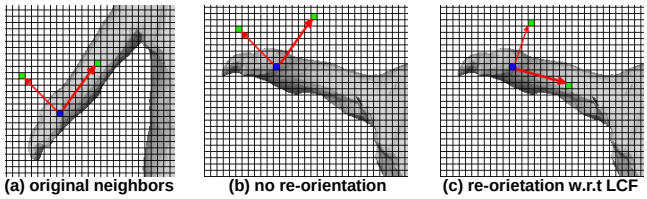

Figure 3: Without re-orientations, different types of neighbors might be chosen(c.f. (a) and (b)), and hence cause different feature responses, despite the fact that the current voxels are located on the same position on the body.

A user-specific forest is trained with many voxelized meshes off-line. Fig. 3 depicts our volumetric feature. Given two randomly chosen neighbors (green) of the current voxel $\mathbf{v}$ (blue), we consider the dot product of their normals, and the difference of VNF within local cuboids. A local coordinate system is attached to select neighboring voxels adaptively in order to achieve pose invariance, as in Fig. 3. The dataset for learning is many sample-label pairs $\mathcal{S} = \{(\mathbf{v}, \mathbf{x}_v)\}$. The entropy is the variance of labels: $H(\mathcal{S}) = \sigma^2(\mathcal{S})$. If the models of outliers are available, one can also include them in the training data. In this case, the entropy is augmented by a classification measure, and the forests do simultaneous classification and regression as in [3].

During training, the feature parameters that maximizes the information gain are stored at each branch node. During testing, an input $\mathbf{y}_i$ is first mapped to a voxel $\mathbf{v}_i$, regressed to a 3D point $\tilde{\mathbf{y}}_i \in \tilde{\mathbf{Y}}$, and attains a closest vertex $\hat{v}_i$ in the reference vertex set $\mathcal{V}_\mathcal{M}$: $\hat{v}_i = \arg\min_{v \in \mathcal{V}_\mathcal{M}} \|\tilde{\mathbf{y}}_i - \mathbf{x}_v\|_2$.

Given the properly estimated correspondence $\mathcal{C}$, poses and shapes are then jointly recovered as in [4]. When combined with ICP, we confirm that this discriminative association yields better accuracy in registration, more stability when tracking over time, and faster convergence. Evaluations on existing datasets demonstrate the benefits with respect to the state-of-the-art.

[1] C. Cagniart, E. Boyer, and S. Ilic. Probabilistic deformable surface tracking from multiple videos. In *ECCV*. Springer, 2010.

[2] S. Corazza, L. Mündermann, E. Gambaretto, G. Ferrigno, and Thomas P. Andriacchi. Markerless motion capture through visual hull, articulated icp and subject specific model generation. *IJCV*, 2010.

[3] G. Fanelli, M. Dantone, J. Gall, A. Fossati, and L. Van Gool. Random forests for real time 3d face analysis. *IJCV*, 2013.

[4] C.-H. Huang, E. Boyer, and S. Ilic. Robust human body shape and pose tracking. In *3DV*, 2013.

[5] J. Taylor, J. Shotton, T. Sharp, and A. Fitzgibbon. The vitruvian manifold: Inferring dense correspondences for one-shot human pose estimation. In *CVPR*, 2012.