Dynamic FAUST: Registering Human Bodies in Motion

Federica Bogo[1,*] Javier Romero[2,*] Gerard Pons-Moll[3] Michael J. Black[3]
[1]Microsoft, Cambridge, UK [2]Body Labs Inc., New York, NY
[3]MPI for Intelligent Systems, Tübingen, Germany
febogo@microsoft.com, javier.romero@bodylabs.com, {gpons, black}@tuebingen.mpg.de

Figure 1: **Dynamic FAUST**. We present a new 4D dataset containing more than one hundred dynamic performances of 10 subjects. We provide raw 3D scans (meshes) at 60 frames per second and dense ground-truth correspondences between them, obtained with a novel technique that combines shape and appearance to obtain accurate temporal mesh registration.

Abstract

While the ready availability of 3D scan data has influenced research throughout computer vision, less attention has focused on 4D data; that is 3D scans of moving non-rigid objects, captured over time. To be useful for vision research, such 4D scans need to be registered, or aligned, to a common topology. Consequently, extending mesh registration methods to 4D is important. Unfortunately, no ground-truth datasets are available for quantitative evaluation and comparison of 4D registration methods. To address this we create a novel dataset of high-resolution 4D scans of human subjects in motion, captured at 60 fps. We propose a new mesh registration method that uses both 3D geometry and texture information to register all scans in a sequence to a common reference topology. The approach exploits consistency in texture over both short and long time intervals and deals with temporal offsets between shape and texture capture. We show how using geometry alone results in significant errors in alignment when the motions are fast and non-rigid. We evaluate the accuracy of our registration and provide a dataset of 40,000 raw and aligned meshes. Dynamic FAUST extends the popular FAUST dataset to dynamic 4D data, and is available for research purposes at http://dfaust.is.tue.mpg.de.

1. Introduction

We inhabit a 4D world of 3D shapes in motion and the number of range sensing devices that can capture this world is growing rapidly. There is already extensive work on registering, or aligning, 3D scans of static scenes to create rich 3D mesh representations. The story is quite different, however, for dynamic scenes, containing articulated and non-rigid objects, where the problem is much harder and there are many fewer methods. Moreover there exist no ground-truth datasets for evaluating algorithms for the registration

*The work was performed at the MPI for Intelligent Systems.

of non-rigid 3D shapes over time. We address this here with a new dataset called Dynamic FAUST (D-FAUST), containing sequences with thousands of 3D scans of humans in motion, together with precise ground-truth correspondence.

Many algorithms, like deep learning methods [10, 11], require 3D meshes to be registered to a common reference topology. Such learning methods require large amounts of data, whereas existing 3D datasets tend to be small. One option is to generate synthetic data to learn the correspondences to a common template [40, 51] but that is not as rich as real data. The problem of aligning, or registering, 3D meshes, however, is challenging due to variations in shape, articulation, noise, missing data, and the size of high-resolution 3D scans. Consequently, there is a need for 1) methods to align 3D meshes accurately, 2) sequences of 3D scans containing non-rigid and articulated motion, and 3) a dataset together with ground-truth correspondence.

The FAUST dataset [8] is an example in which 3D body shapes, in a variety of poses, are precisely registered using a combination of 3D shape and surface texture. The dataset is challenging because it contains scans of real people, including high-resolution, missing data, noise, self-contact, articulation, and shape variation. It is widely used to develop, train, and test algorithms for 3D mesh alignment and processing. Despite its success, the dataset is still limited, with only 100 ground-truth alignments. The dataset contains only static scans, whereas many objects, like people, move and deform over time. We seek a dataset that is orders of magnitude bigger and contains temporal shape variation.

The Dyna dataset [39] is one possible candidate. This contains 40,000 3D meshes created by registering a common template mesh to sequences of 3D scans. The meshes are of people, with varying body shapes, performing a range of actions. The scans are captured by a 4D scanner at 60 fps, and then a template mesh is aligned to them using only geometric information. The aligned meshes exhibit noticeable soft tissue motion. The lesson of FAUST however is that geometry-based alignment is inaccurate and cannot be relied upon to establish ground-truth correspondence between 3D meshes. Additionally, the Dyna dataset only contains the registered meshes at a lower resolution than the original scans. This makes it impossible to evaluate new mesh registration algorithms for dynamic data.

Here we go beyond these previous datasets to develop a new dataset of the 40,000 Dyna meshes registered using both geometry and texture information. In doing so we show that geometry alone, as expected, does not accurately capture all the soft tissue motions. This is difficult to visualize in this paper but can be seen quite dramatically in a companion video [1]. Consequently we develop a novel method for registering 4D data.

Texture-based alignment of 3D meshes of highly dynamic sequences over long time frames is challenging due to variations in illumination caused by self shadowing, changes in shape due to deformation, and occlusion. Standard texture-based registration methods fail due to such difficulties. Hence, we go beyond FAUST to define a temporal alignment method that exploits both short-range motion and long-range matches between each frame and a reference frame. Our solution also deals with the fact that the scanning system captures shape and texture slightly out of phase with each other. The result is a highly accurate registration despite all the challenges mentioned earlier.

We define ground-truth points similarly to FAUST, by considering both 3D shape accuracy and the optical flow between a reference texture and each frame. Small flow vectors suggest that the sequence is well registered. We find that, in D-FAUST, 82% of scan points (out of more than 5 billion) are aligned with an accuracy within 1-2mm.

D-FAUST is available for research purposes [1]. We release raw scans, aligned templates, and masks of points with ground-truth accuracy. As with FAUST, this is likely to stimulate research on 3D mesh registration while enabling the community to explore new topics in non-rigid and articulated registration and deep learning for mesh registration.

2. Related Work

The history of 3D mesh registration is extensive; Chen and Koltun [20] provide a good recent review.

4D registration. There are many 3D acquisition systems ranging from depth sensors to multi-view stereo setups, which output scans in the form of unstructured point clouds or noisy meshes. There is an extensive literature on registering such data across time. For example, there are many non-rigid tracking methods, either model-based [3, 7, 22, 25, 27, 34, 44, 49] or model-free [2, 16, 21, 24, 37, 50, 54]. These methods focus on tracking single sequences: they adopt a sequential frame-to-frame registration approach, assuming relatively small non-rigid deformations. Frame-to-frame approaches can suffer from accumulation of errors, resulting in drift over time [19].

Other work focuses on non-sequential alignment [30, 48], tackling the problem of registering data from multiple sequences. This is important, for example, for constructing motion graphs to synthesize new motions from existing ones [15, 19, 31, 41]. As recognized in [12, 41], it is challenging to register very different motions. Hence, many approaches seek only locally consistent connectivity (*e.g.* looking for similar subsequences and matching them). In contrast, our method registers a unique reference mesh to scan data from thousands of frames and hundreds of different sequences.

Texture has been used for aligning isolated body parts such as faces and hands [6, 5, 17, 23, 43]. Full-body capture is significantly more difficult, since body deformations are a combination of articulated and non-rigid motion. For

the full body, [18] introduces the concept of 4D model flow to capture surface appearance changes over time. This is a substantially different problem from ours: the goal of model flow is to minimize the visual discrepancy between two 4D models, to synthesize a new (and sharp) textural appearance. Geometry is used as a proxy to extract texture from images, but texture is not used to improve the geometry.

Gall et al. [26] compute matches between a textured model and RGB images to stabilize drift in tracking. Theobalt et al. [46] use 3D motion fields to improve motion capture. Tsiminaki et al. [47] do something similar, using optical flow to register (and super-resolve) texture maps from different frames in a sequence. They consider only short sequences with very limited non-rigid deformations. Boukhayma et al. [12] seek a global appearance model spanning multiple sequences of a subject. No geometry correction based on color is used and evaluation is shown only on limited datasets. None of these methods use a combination of short and long range correspondences in texture space together with a body model to achieve highly accurate registration. In addition, the results of these methods are not accurate enough to be ground truth.

Datasets. In 3D registration, the FAUST dataset [8] filled a gap since other datasets were either synthetic [13, 14, 32] or without ground truth [4, 28, 42]. From this dataset, researchers have used both real scans (for benchmarking registration techniques [20, 51, 55]) and aligned templates (*e.g.* for training Convolutional Neural Networks [9, 10, 11, 33, 53]).

There exist previous datasets for 4D registration [21, 39, 45]. Collet et al. [21] release data captured with a multi-view stereo system using RGB and IR cameras. Starck and Hilton [45] propose 3D surfaces reconstructed from a multi-view RGB setup. In both cases, the amount of data released is limited to a few sequences, containing quite low-resolution meshes. The Dyna dataset [39] includes 40,000 registered meshes with consistent topology, but it provides only geometry-based aligned meshes and not the original scans. We show that the geometry-based alignment of Dyna can be significantly improved using texture information.

3. Data Acquisition

The 4D data was captured with a custom-built multi-camera active stereo system (3dMD LLC, Atlanta, GA). The scanner captures temporal sequences of full-body 3D scans at 60 frames per second (fps) using 22 pairs of stereo cameras, 22 color cameras, 34 speckle projectors and arrays of white-light LED panels. The speckle projectors and LEDs flash at 120 fps to alternate between stereo capture and color capture. The delay between stereo and color capture is approximately 4 milliseconds (ms). The stereo pairs are arranged to give full-body capture for a wide range of motions. The system outputs 3D meshes with approximately $150,000$ vertices on average.

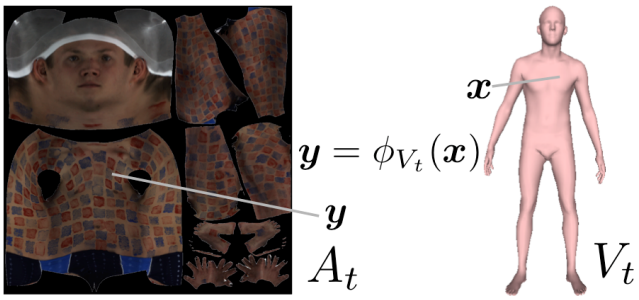

Figure 2: **Texture map**. A texture map A_t, computed at frame t, and the corresponding mesh V_t. A function ϕ_{V_t} maps any 3D point $\boldsymbol{x}$ on the surface of V_t to a pixel $\boldsymbol{y}$ in A_t.

The dataset includes dynamic performances of 10 subjects (5 men and 5 women) of various shapes and ages. We consider 129 sequences. This gives more than $40,000$ frames, with corresponding scans. All subjects were professional models working under a modeling contract, and they gave their informed written consent for the analysis and publication of their 4D scan data. During the scan sessions they all wore identical, minimal clothing: tight fitting swimwear bottoms for men and women and a sports bra top for women. As in [8], the skin of each subject was painted in order to provide high-frequency information across most of the body surface.

4. Methods

For each frame t, the acquisition system outputs a 3D scan S_t and 22 color images $I_{t,k}$, $1 \leq k \leq 22$. Calibration parameters are known for both color and stereo cameras. The goal of our approach is to bring all temporal 3D scans into correspondence by registering a 3D body template to all of them. The template, T, is a watertight triangulated mesh with $6,890$ vertices and $13,776$ triangles. The template comes with a UV map created by an artist. The UV map is an un-warping of the template surface onto an image, A_t, which is a texture map at frame t (Fig. 2). The texture map is simply an image with "foreground" regions that correspond to the surface and undefined regions that can be ignored (black regions in Fig. 2). Given a 3D mesh with vertices V_t, we denote by ϕ_{V_t} the function mapping a 3D point on the surface of V_t to a pixel in A_t, and by $\phi_{V_t}^{-1}$ its inverse. Note that the mapping between UV pixels and mesh surface coordinates is constant and independent of changes in 3D surface geometry. We call a *registration*, V_t, the template T deformed to fit a scan S_t. Registrations can be projected into the camera images at each time instant. Because registered meshes share the same topology, they provide correspondence between image pixels across time. The same surface point on two registered meshes projects to

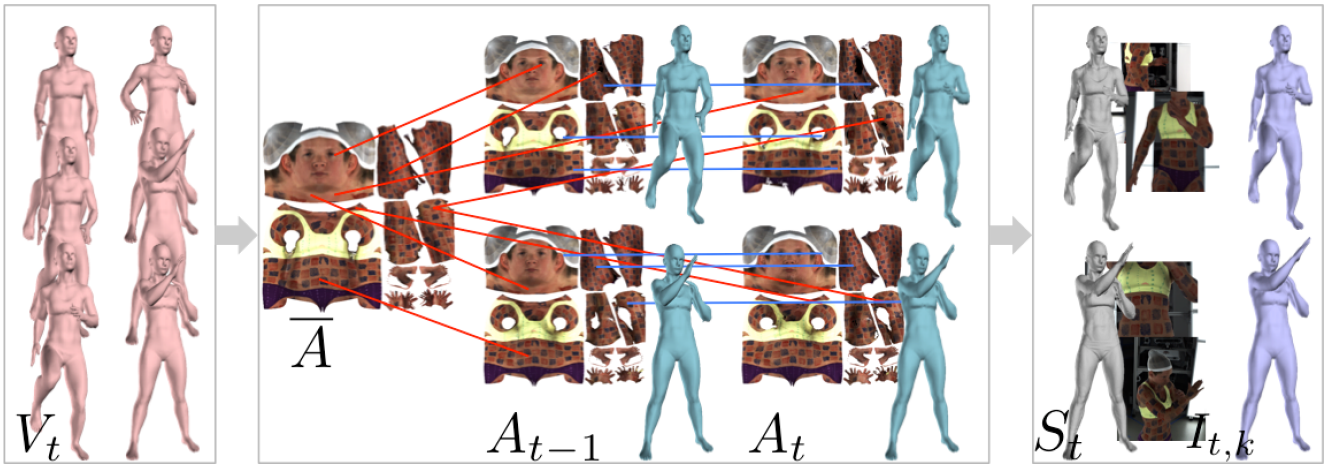

Figure 3: **Overview.** Input to our approach is a set of geometry-based registrations, V_t (in pink, 2 example sequences). Our approach proceeds in 2 stages. In stage 1, after obtaining a texture map A_t from each registration V_t, we compute dense matches between each A_t and a reference map $\overline{A}$ (red lines), and between maps at subsequent frames (blue lines). These matches are mapped to 3D and interpolated to obtain a set of rectified registrations (in blue). In stage 2 we refine such registrations by fitting a 3D model to un-synchronized streams of geometry (scans, in gray) and image data. This gives a set of highly accurate registrations (in purple).

pixels in two different images, establishing the correspondence between those pixels. These registrations also allow us to extract the texture map A_t at each frame [8].

Our approach is model-based. It builds on the methods in [8, 39] introducing several novelties. We extend [39] to deal with both geometry and color. We extend FAUST [8] to deal with temporal sequences. The FAUST method registered a template to 30 scans of a subject in a variety of poses. Here we register $\approx 4,000$ scans of each subject (2 orders of magnitude more data per subject). This poses a series of technical challenges, as described in Sec. 4.2 and 4.3. Details about the body model used are provided in Sec. 4.1.

Figure 3 provides an overview of our approach. We start with the set of *Dyna registrations* obtained from [39] (V_t meshes shown in pink for 2 example sequences). The objective function used in [39] penalizes distance in 3D between scan and template surfaces. Because it does not use image texture, the method does not prevent the template from *sliding* over the scan. As a result, Dyna registrations closely fit the scan surface but are actually inaccurate (see Sec. 5). The texture-based registration method of FAUST [8] should fix this sliding, but simply initializing a FAUST-style registration with Dyna registrations fails. Since our dataset contains large, highly non-rigid motions, the FAUST method quickly becomes trapped in local optima.

Our approach improves Dyna registrations by exploiting both geometry and color data, in 2 stages. In stage 1 we compute a texture map A_t from each registration V_t. Ideally, we would like to establish correspondences (*i.e.* compute matches) between *all* the texture maps obtained for a subject and a single reference map (red lines in Fig. 3, exemplified for 2 sequences). We compute dense matches [52] in 2D between each map A_t and a reference map $\overline{A}$ (red lines). Texture maps computed from different sequences, however, may differ due to changes in pose and illumination, and occlusions. We therefore combine such *long-range* matches with *short-range* ones (blue lines), computed between subsequent frames in a sequence. Short-range matches can better track small variations, while long-range ones prevent drifting. These matches are mapped to 3D and interpolated to obtain a set of *rectified registrations* (blue meshes).

Such registrations exhibit much less sliding than Dyna ones and provide a better initialization. However, they exhibit small geometric artifacts (due to inaccurate matches) and do not capture the delay between geometry (scans, in gray) and color capture (Sec. 3). Hence, in stage 2 we use these rectified registrations to learn a subject-specific shape and appearance model, and refine them using a novel model-based registration approach that can deal with temporally-offset streams of geometry and texture data (Sec. 4.3). The final result is a set of highly accurate registrations (shown in purple in Fig. 3).

4.1. Body Model

Whereas the authors of [8, 39] use BlendSCAPE [29], here we use the SMPL body model [36], which has several advantages in terms of simplicity, accuracy, and portability. SMPL defines a skinning function, $M(\boldsymbol{\theta}, T^p; \Phi)$, parameterized by pose $\boldsymbol{\theta}$, a 3D mesh T^p, and learned model param-

eters Φ. Output of the function is a triangulated, watertight mesh with $N = 6,890$ vertices, with the same topology as the template used for registration. The mesh is segmented into parts; pose parameters $\boldsymbol{\theta}$ are the axis-angle representation of the relative rotation between parts. T^p captures the personalized shape of person p, and is the mesh in a neutral pose $\boldsymbol{\theta}^*$, before applying pose-dependent deformations (see [36] for details).

4.2. Stage 1: Match-based Rectification

The goal of Stage 1 is to mitigate the sliding problems exhibited by geometry-only registrations, in order to get a better initialization for our model-based approach (Sec. 4.3). We use Dyna registrations to compute a texture map A_t for each frame t (Fig. 3). We compute A_t from a set of images $I_{t,k}$ with the same technique described in [8].

Consider the set of K^p frame-wise texture maps of subject p, $\mathcal{A}^p = \{A_i\}_{i=1}^{K^p}$. The texture of a particular location on the body should not change much over time. Therefore, if registrations were perfect, *all the maps in $\mathcal{A}^p$ should look almost identical*. The residual differences should be due to illumination changes, shadows, slight skin color difference due to blood flow variation, and subtle facial expressions. However, we observe that the Dyna registrations exhibit significant sliding; *i.e.*, the per-subject frame-wise texture maps vary significantly. Since the Dyna mesh-to-scan distances are small, motion in the texture maps implies that vertices are sliding along the surface. Our goal then is to solve for the registrations such that the resulting texture maps are as similar as possible.

Long-range matching. To that end, we compute dense matches between texture maps using DeepMatching [52]. This works very well, in part because the Dyna subjects have a texture pattern painted on their bodies. To establish correspondences across frames and sequences, we compute matches between a unique per-subject reference texture map, $\overline{A} \in \mathcal{A}^p$, (corresponding to registration $\overline{V}$) and all the texture maps in $\mathcal{A}^p$. The choice of $\overline{A}$ is arbitrary; we pick the first frame of one of the sequences for each subject. Note that these texture maps are computed from different sequences (thousands of frames), all relative to subject p.

Such matches establish correspondences in 2D. We map them into a set of 3D displacements, that we can apply to registration vertices; Fig. 4 illustrates the process. Consider a match computed between $\overline{A}$ and $A_t \in \mathcal{A}^p$, obtained from registration V_t. It establishes a correspondence between pixels $\boldsymbol{x}$ in $\overline{A}$ and $\boldsymbol{y}$ in A_t. This means that, to make the two maps coherent, pixel $\boldsymbol{x}$ in A_t should store the color now at $\boldsymbol{y}$. Hence, the 3D point $\phi_{V_t}^{-1}(\boldsymbol{x})$ should move to the point $\phi_{V_t}^{-1}(\boldsymbol{y})$, according to the (3D) displacement $\boldsymbol{d}_x = \phi_{V_t}^{-1}(\boldsymbol{y}) - \phi_{V_t}^{-1}(\boldsymbol{x})$. In practice, we compute displacements only at vertex locations V_t: given $\boldsymbol{v} \in V_t$

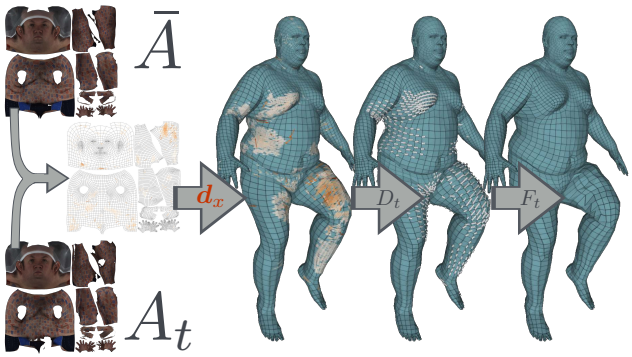

Figure 4: **Match-based rectification.** Given a reference texture map $\overline{A}$ and a texture map A_t at time t, we compute dense 2D matches between them (orange arrows, middle left). These 2D matches are translated into 3D correspondences (left-most mesh). We average them per vertex (middle mesh), and optimize a final set of displacements, applied to the mesh on the right.

we collect all matches originating at triangles sharing $\boldsymbol{v}$ as a vertex, compute the corresponding displacements in 3D and take their average. This gives us a set of per-vertex displacements, $D_t \in \mathbb{R}^{N\times 3}$, for registration V_t. Based on them, we optimize a match-based, per-vertex set of 3D displacements $F_t \in \mathbb{R}^{N\times 3}$ and impose a smoothness term enforcing similar displacements on adjacent vertices:

$$E_F(F_t) = E_{\text{long}}(F_t) + \lambda_{\text{sm}} E_{\text{sm}}(F_t). \tag{1}$$

E_{long} simply penalizes discrepancy between D_t and F_t with the squared Frobenius norm $E_{\text{long}}(F_t) = \|D_t - F_t\|^2$, λ_{sm} is the weight for the smoothness term and

$$E_{\text{sm}}(F_t) = \sum_{(v,v')\in\mathcal{E}} \|F_{t,v} - F_{t,v'}\|^2. \tag{2}$$

where $\mathcal{E}$ is the set of edges of our template.

Short-range matching. With highly accurate matches, this would be enough to align all the meshes V_t to a common reference mesh $\overline{V}$. However, when considering frames far apart in time, changes in pose and illumination can make matches inaccurate, and therefore produce significant errors in the displacements (Fig. 5). To account for this, we introduce a second error term based on matches computed between subsequent frames. Computing matches between subsequent, and therefore similar, frames is less error-prone and helps make our algorithm more robust. More precisely, after optimizing $E_F(F_0)$, we compute matches between A_1 and the rectified map A_0^f obtained after applying F_0 to V_0. This gives a set of displacements $\widetilde{D}_t$, computed as above. We then optimize $E_F(F_t) =$

$$E_{\text{long}}(F_t) + \lambda_{\text{short}} E_{\text{short}}(F_t) + \lambda_{\text{sm}} E_{\text{sm}}(F_t) \tag{3}$$

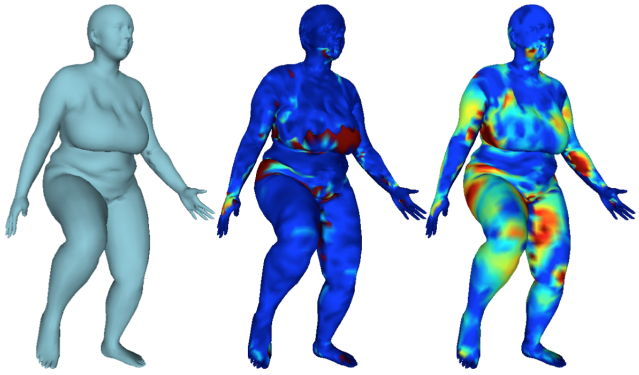

Figure 5: **Importance of E_{short} and E_{long}.** Left: Rectified mesh V_t^f optimized as in Eq. (3). Middle: Vertex-to-vertex distance between V_t^f and the mesh obtained from Eq. (3) after dropping E_{short}: long-range matches are unreliable in the presence of shadows and occlusions (armpits, chin) and clothing movement (chest). Right: Vertex-to-vertex distance between V_t^f and the mesh obtained from Eq. (3) after dropping E_{long}: relying only on short-range matches produces drifting. Red denotes a distance $\geq$ 1cm.

where $E_{\text{short}}(F_t) = \|\widetilde{D}_t - F_t\|^2$. We initialize F_t to $\widetilde{D}_t$, and discard as unreliable the vertex displacements $D_{t,v}$ if $||D_{t,v} - \widetilde{D}_{t,v}|| > 1$ centimeter. We optimize Eq. (3) sequentially, starting from the initial frame of each sequence. E_{long} prevents drifting over time, while E_{short} corrects for correspondence errors from far apart frames by tracking small changes over time (Fig. 5). The result of this stage is a set of rectified registrations, V_t^f, one per frame, where the superscript f indicates that the vertices come from the match-rectification phase, which optimizes F_t.

Note that, since we start by optimizing Eq. (3) for frame 0, using $\overline{A}$, all frames in all sequences are eventually aligned to a unique per-subject registration $\overline{V}$. Recall that the choice of $\overline{V}$ is arbitrary (to get $\overline{A}$, we picked the first frame of one of the sequences).

Matches near the boundaries of the texture map and occlusion boundaries (between the defined and undefined areas of the texture map) are unreliable. We filter out matches originating or mapping to these unreliable regions.

4.3. Stage 2: Appearance-based Registration

Match-based rectification in Stage 1 helps dramatically reduce sliding. Registrations, however, may still suffer from artifacts and slight inaccuracies and, more importantly, do not model the temporal offset between geometry and color capture. We address this in the second stage.

First, we use registrations $\{V_t^f\}$ from Stage 1 to learn a per-subject model of shape T^p and appearance A^p. We choose uniformly at random approximately 100 registrations per subject. To learn T^p, we put each registration in the neutral pose $\boldsymbol{\theta}^*$ (by "undoing" its pose-dependent deformations [36]), and then average the vertices across templates. For the appearance model A^p, we compute the corresponding texture maps and again simply average them.

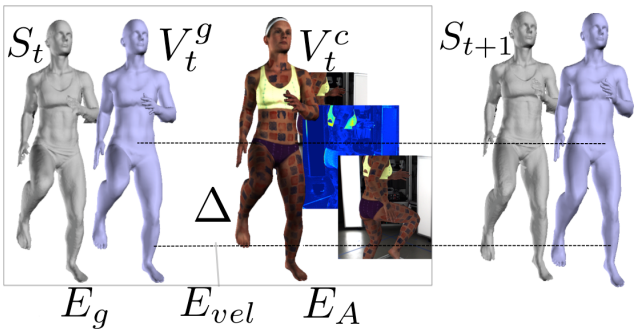

Figure 6: **Appearance-based registration**. We deal with the temporal offset between geometry and color streams by optimizing two registrations (V_t^g and V_t^c) per frame t. Our objective penalizes the Euclidean distance between scan S_t and V_t^g, and discrepancy between real images $I_{t,k}$ and synthetic ones rendered from V_t^c. E_{vel} enforces a constant velocity model between V_t^g and V_t^c.

Then, we refine the rectified registrations, minimizing an objective that takes into account both geometry and color information, matching the model to scans and RGB images. We explicitly model the delay between geometry and color capture (Sec. 3) as a soft constraint. In fact, we optimize two registrations per frame – one relative to the geometry (g) frame, V_t^g, and one relative to the color (c) frame, V_t^c (Fig. 6).

For each frame t, we minimize an objective E given by the sum of 4 error terms:

$$E(V_t^g, V_t^c, \boldsymbol{\theta}_t) = E_g(V_t^g) + \lambda_{cpl} E_{cpl}(V_t^c, \boldsymbol{\theta}_t) + \lambda_{vel} E_{vel}(V_t^g, V_t^c) + \lambda_A E_A(V_t^c) \quad (4)$$

where λ_{cpl}, λ_{vel} and λ_A are the weights for the different terms. As in [39], E_g penalizes distance in 3D between scan and registration surface; E_{cpl} penalizes discrepancy between V_t^c and the SMPL model with shape T^p and pose $\boldsymbol{\theta}_t$ (cf. [39]). As in [8], the appearance-based error term E_A penalizes discrepancy between real images $I_{t,k}$ and synthetic images $\widetilde{I}(V_t^c, A^p)$ rendered from the model:

$$E_A(V_t^c) = \sum_{\text{camera } k} ||\Gamma(I_{t,k}) - \Gamma(\widetilde{I}(V_t^c, A^p))||^2. \quad (5)$$

where Γ denotes a Ratio-of-Gaussians contrast-normalization term [8]. While in [8] Eq. (5) was computed only over foreground image pixels, here we sum over both foreground and background.

Finally, E_{vel} enforces a constant velocity model between

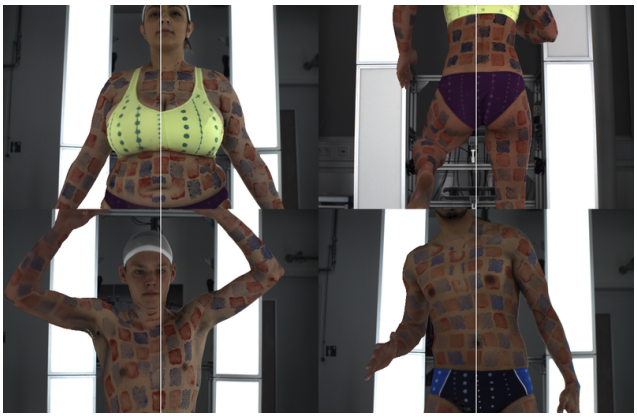

Figure 7: **Comparison between real and rendered images**. Four example frames: the right half shows the real image and the left half is the synthetic image rendered from the model. They look very similar.

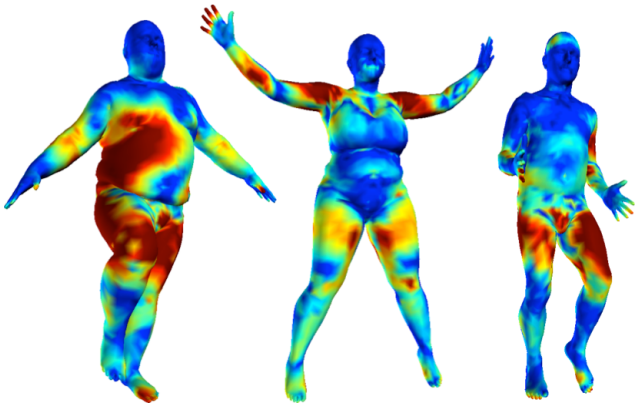

Figure 8: **Vertex-to-vertex distance between Dyna and D-FAUST**. Heat maps show the vertex-to-vertex distance between three Dyna registrations and the corresponding registrations in D-FAUST. Red denotes a distance ≥ 1.5cm.

registrations V_t^g and V_t^c:

$$E_{vel}(V_t^g, V_t^c) = ||V_t^c - V_t^g - \Delta||^2. \quad (6)$$

Here $\Delta = (\frac{60 \cdot 4}{1000}) \cdot (V_{t+1}^f - V_t^f)$, where V_{t+1}^f and V_t^f are the rectified registrations relative to frames $t+1$ and t; recall the delay between geometry and color is (roughly) equal to 4 ms, and sequences are captured at 60 fps.

4.4. Optimization

The objective functions in Eq. (3) and (4) are minimized using a gradient-based dogleg minimization [38]. Gradients are computed with automatic differentiation using the Chumpy framework and the differentiable renderer [35]. Minimizing Eq. (3) takes less than one minute per frame; minimizing Eq. (4) takes approximately 10 minutes.

5. Evaluation

The registration of our template to each scan brings all the scans into correspondence. Evaluating registration accuracy is difficult due to the lack of ground truth. As in [8], we quantitatively evaluate registration accuracy so that it can be considered ground truth. We label as ground truth the vertices that satisfy the following three criteria:

Geometric error: In the spirit of [8], we discard all scan vertices that are further away than 1mm from the corresponding registration surface. We find that 93% of scan points are closer than 1mm (by contrast, in FAUST they report 90% of points are within 2mm).

Image error: We can take the registered meshes V_t^c and the per-subject average appearance model computed from them, project them into any camera view, and compare them with actual observed images. If the geometric distance between the scan and the registration is small, then the optical flow between the synthetic and real image provides a measure of misalignment tangential to the surface (sliding).

We compute optical flow [52] between real and synthetic images (cf. [8]). A 3D point is discarded if the optical flow magnitude in the corresponding pixel $\boldsymbol{x}$ is bigger than one pixel in at least one camera where $\boldsymbol{x}$ is visible. One pixel corresponds roughly to an error of 2mm in the 3D surface. In this evaluation, we do not consider pixels where the dot product between camera axis and the surface normal is smaller than 0.5, since optical flow is highly unreliable due to the large grazing angle and the corresponding point is likely to be better covered from a different view.

Since we seek to label accuracy of the scan vertices, not the template vertices, for every scan point in S_t we find the closest registration point in the geometry registration V_t^g (expressed in barycentric coordinates). Using these coordinates we can find the corresponding 3D point in the color registration V_t^c and evaluate the flow and viewing angle. We find 94% of the scan points satisfy this criterion for D-FAUST registrations; the synthetic images from D-FAUST registrations match quite well the real images, see Fig. 7.

Motion consistency: Scan points whose corresponding registration points deviate too much from constant velocity are discarded. We compute the velocity from adjacent geometry registrations $\Delta = V_{t+1}^g - V_t^g$ and evaluate consistency in the color frame V_t^c using Eq. (6). Points that deviate more than 1mm are discarded. Note that at the short time interval of 4ms this simple motion model suffices. We find that 92% of scan points satisfy this criterion, indicating that the geometry and color registrations are coherent.

Combined: 82% of scan points satisfy all three criteria.

Comparison with Dyna: Overall, we observe that D-FAUST registrations are significantly more accurate than

Figure 9: **Comparison between computed and synthetic flow.** From top to bottom: real images at time t for three example frames; optical flow [52] computed between them and real images at time $t+1$ (not shown); synthetic flow generated from the corresponding D-FAUST registrations; synthetic flow generated from Dyna registrations. Dyna-synthesized flow looks over-smooth and is too piece-wise constant. Results from D-FAUST are more realistic and better approximate the computed optical flow.

the original Dyna registrations. Figure 8 shows vertex-to-vertex Euclidean distance between D-FAUST and Dyna registrations, for three example frames. Significant differences are visible in areas like the belly, arms and thighs. The average vertex-to-vertex distance between Dyna and D-FAUST registrations, computed over the entire dataset, is 6mm.

In particular, D-FAUST captures non-rigid soft tissue deformations with higher accuracy. This can be seen in Fig. 9, where we compare the optical flow computed between consecutive real images, versus the flow synthesized using the D-FAUST and Dyna registrations. Optical flow can be trivially synthesized from registrations as they are in correspondence. One can observe how the D-FAUST synthetic flow is much more realistic than Dyna flow, which lacks detail. In the companion video [1] we show video sequences comparing D-FAUST with Dyna registrations; it is clear that D-FAUST registrations capture much more non-rigid soft tissue motion and do not suffer from tangential sliding along the surface. We also show the resulting frame-wise texture maps of D-FAUST registrations and we compare them with texture maps computed from Dyna registrations. One can observe in the video how the texture maps of D-FAUST are much more stable, showing almost not changes except those due to illumination and shadows. This is also a strong indicator of good registration quality.

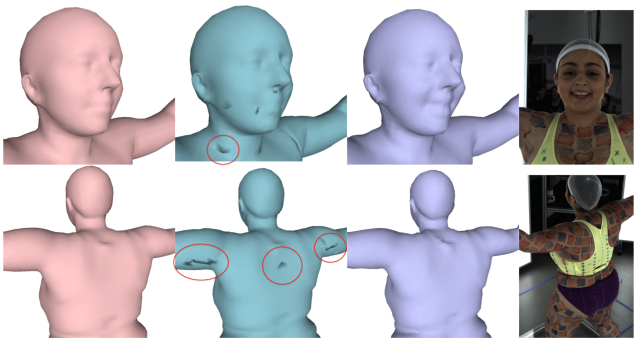

Figure 10: **Importance of appearance-based registration.** From left to right: Dyna registration (pink), rectified registration (blue), final D-FAUST result (purple), real image, for two frames. Match-based rectification can introduce artifacts in the meshes (red circles), that are corrected during appearance-based registration. Also, details like facial expressions are captured in the last stage.

Importance of appearance-based registration: Our technique works well even in areas like the face, where the painted texture pattern was not applied. Figure 10 (top row) compares a registration from Dyna (pink), after match-based rectification (blue) and the final D-FAUST result (purple). Facial expressions are captured in the last stage, thanks to the robust matching between model and image data. Figure 10 also shows the role played by each stage of the technique. Match-based correction helps remove gross misalignment, but can produce artifacts in the mesh (red circles in the image). Highly accurate alignment is achieved through a combination of all the stages.

6. Conclusion

We presented D-FAUST, the first 4D dataset providing both real scans and dense ground-truth correspondences between them. D-FAUST collects over $40,000$ real meshes, capturing 129 dynamic performances from 10 subjects. We registered all the scans to a common template by introducing a novel approach that combines computation of 2D correspondences in texture space with a model-based registration approach dealing with temporally-offset streams of geometry and texture data. All the scans and registrations will be made publicly available for research purposes [1].

Acknowledgments. We thank S. Pujades for helpful discussions and A. Osman for help with data registration.

References

[1] http://dfaust.is.tue.mpg.de. 2, 8

[2] N. Ahmed, C. Theobalt, C. Roessl, S. Thrun, and H.-P. Seidel. Dense correspondences finding for parameterization-free animation reconstruction from video. In *CVPR*, pages 1–8, 2008. 2

[3] B. Allain, J. Franco, and E. Boyer. An efficient volumetric framework for shape tracking. In *CVPR*, pages 268–276, 2015. 2

[4] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis. SCAPE: Shape Completion and Animation of PEople. *ACM Transactions on Graphics*, 24(3):408–416, 2005. 3

[5] T. Beeler, F. Hahn, D. Bradley, B. Bickel, P. Bearsdley, C. Gotsman, R. Sumner, and M. Gross. High-quality passive facial performance capture using anchor frames. *ACM Transactions on Graphics, (Proc. SIGGRAPH)*, 30(4):75:1–75:10, 2011. 2

[6] V. Blanz and T. Vetter. A morphable model for the synthesis of 3D faces. In *ACM SIGGRAPH*, pages 187–194, 1999. 2

[7] F. Bogo, M. J. Black, M. Loper, and J. Romero. Detailed full-body reconstructions of moving people from monocular RGB-D sequences. In *ICCV*, pages 2300–2308, 2015. 2

[8] F. Bogo, J. Romero, M. Loper, and M. J. Black. FAUST: Dataset and evaluation for 3D mesh registration. In *CVPR*, pages 3794–3801, 2014. 2, 3, 4, 5, 6, 7

[9] D. Boscaini, D. Eynard, D. Kourounis, and M. Bronstein. Shape-From-Operator: Recovering shapes from intrinsic operators. *Computer Graphics Forum*, 34(3):265–274, 2015. 3

[10] D. Boscaini, J. Masci, S. Melzi, M. Bronstein, U. Castellani, and P. Vandergheynst. Learning class-specific descriptors for deformable shapes using localized spectral convolutional networks. *Computer Graphics Forum*, 34(5):12–23, 2015. 2, 3

[11] D. Boscaini, J. Masci, E. Rodolà, and M. Bronstein. Learning shape correspondence with anisotropic convolutional neural networks. In *NIPS*, pages 3189–3197, 2016. 2, 3

[12] A. Boukhayma, V. Tsiminaki, J. Franco, and E. Boyer. Eigen appearance maps of dynamic shapes. In *ECCV*, pages 230–245, 2016. 2, 3

[13] A. Bronstein, M. Bronstein, U. Castellani, A. Dubrovina, L. Guibas, R. Horaud, R. Kimmel, D. Knossow, E. von Lavante, D. Mateus, M. Ovsjanikov, and A. Sharma. SHREC 2010: Robust correspondence benchmark. In *3DOR*, 2010. 3

[14] A. Bronstein, M. Bronstein, and R. Kimmel. *Numerical geometry of non-rigid shapes*. Springer, 2008. 3

[15] C. Budd, P. Huang, M. Klaudiny, and A. Hilton. Global non-rigid alignment of surface sequences. *International Journal of Computer Vision*, 102(1-3):256–270, 2013. 2

[16] C. Cagniart, E. Boyer, and S. Ilic. Probabilistic deformable surface tracking from multiple videos. In *ECCV*, pages 326–339, 2010. 2

[17] C. Cao, D. Bradley, K. Zhou, and T. Beeler. Real-time high-fidelity facial performance capture. *ACM Transactions on Graphics, (Proc. SIGGRAPH)*, 34(4):46:1–46:9, 2015. 2

[18] D. Casas, C. Richardt, J. Collomosse, C. Theobalt, and A. Hilton. 4D model flow: Precomputed appearance alignment for real-time 4D video interpolation. *Computer Graphics Forum*, 34(7):173–182, 2015. 3

[19] D. Casas, M. Tejera, J. Guillemaut, and A. Hilton. 4D parametric motion graphs for interactive animation. In *Symp. on Interactive 3D Graphics and Games*, pages 102–110, 2012. 2

[20] Q. Chen and V. Koltun. Robust nonrigid registration by convex optimization. In *ICCV*, pages 2039–2047, 2015. 2, 3

[21] A. Collet, M. Chuang, P. Sweeney, D. Gillet, D. Evseev, D. Calabrese, H. Hoppe, A. Kirk, and S. Sullivan. High-quality streamable free-viewpoint video. *ACM Transactions on Graphics, (Proc. SIGGRAPH)*, 34(4):69:1–69:13, 2015. 2, 3

[22] E. De Aguiar, C. Stoll, C. Theobalt, N. Ahmed, H.-P. Seidel, and S. Thrun. Performance capture from sparse multi-view video. *ACM Transactions on Graphics, (Proc. SIGGRAPH)*, 27(3):98:1–75:10, 2008. 2

[23] M. de la Gorce, D. Fleet, and N. Paragios. Model-based 3D hand pose estimation from monocular video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(9):1793–1805, 2011. 2

[24] M. Dou, J. Taylor, H. Fuchs, A. Fitzgibbon, and S. Izadi. 3D scanning deformable objects with a single RGBD sensor. In *CVPR*, pages 493–501, 2015. 2

[25] A. Elhayek, C. Stoll, N. Hasler, K. I. Kim, H. P. Seidel, and C. Theobalt. Spatio-temporal motion tracking with unsynchronized cameras. In *CVPR*, pages 1870–1877, 2012. 2

[26] J. Gall, B. Rosenhahn, and H.-P. Seidel. Drift-free tracking of rigid and articulated objects. In *CVPR*, pages 1–8, 2008. 3

[27] J. Gall, C. Stoll, E. De Aguiar, C. Theobalt, B. Rosenhahn, and H.-P. Seidel. Motion capture using joint skeleton tracking and surface estimation. In *CVPR*, pages 1746–1753, 2009. 2

[28] N. Hasler, C. Stoll, M. Sunkel, B. Rosenhahn, and H. P. Seidel. A statistical model of human pose and body shape. *Computer Graphics Forum*, 28(2):337–346, 2009. 3

[29] D. A. Hirshberg, M. Loper, E. Rachlin, and M. J. Black. Coregistration: Simultaneous alignment and modeling of articulated 3D shape. In *ECCV*, pages 242–255, 2012. 4

[30] P. Huang, C. Budd, and A. Hilton. Global temporal registration of multiple non-rigid surface sequences. In *CVPR*, pages 3473–3480, 2011. 2

[31] P. Huang, A. Hilton, and J. Starck. Human motion synthesis from 3D video. In *CVPR*, pages 1478–1485, 2009. 2

[32] Z. Lähner, E. Rodolà, M. Bronstein, D. Cremers, O. Burghard, L. Cosmo, A. Dieckmann, R. Klein, and Y. Sahillioglu. SHREC16: Matching of deformable shapes with topological noise. In *3DOR*, 2016. 3

[33] Z. Lähner, E. Rodolà, F. R. Schmidt, M. Bronstein, and D. Cremers. Efficient globally optimal 2D-to-3D deformable shape matching. In *CVPR*, pages 2185–2193, 2016. 3

[34] H. Li, B. Adams, L. J. Guibas, and M. Pauly. Robust single-view geometry and motion reconstruction. *ACM Transactions on Graphics*, 28(5):175:1–175:10, 2009. 2

[35] M. Loper and M. J. Black. OpenDR: An approximate differentiable renderer. In *ECCV*, pages 154–169, 2014. 7

[36] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black. SMPL: A skinned multi-person linear model. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, 2015. 4, 5, 6

[37] R. A. Newcombe, D. Fox, and S. M. Seitz. DynamicFusion: Reconstruction and tracking of non-rigid scenes in real-time. In *CVPR*, pages 343–352, 2015. 2

[38] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, 2006. 7

[39] G. Pons-Moll, J. Romero, N. Mahmood, and M. J. Black. Dyna: A model of dynamic human shape in motion. *ACM Transactions on Graphics, (Proc. SIGGRAPH)*, 34(4):120:1–120:14, 2015. 2, 3, 4, 6

[40] G. Pons-Moll, J. Taylor, J. Shotton, A. Hertzmann, and A. Fitzgibbon. Metric regression forests for correspondence estimation. *International Journal of Computer Vision*, pages 1–13, 2015. 2

[41] F. Prada, M. Kazhdan, M. Chaung, A. Collet, and H. Hoppe. Motion graphs for unstructured textured meshes. *ACM Transactions on Graphics, (Proc. SIGGRAPH)*, 35(4):108:1–108:14, 2016. 2

[42] K. Robinette, H. Dannen, and E. Paquet. The CAESAR project: A 3-D surface anthropometry survey. In *Conference on 3D Digital Imaging and Modeling*, pages 380–386, 1999. 3

[43] S. Saito, T. Li, and H. Li. Real-time facial segmentation and performance capture from RGB input. In *ECCV*, pages 244–261, 2016. 2

[44] J. Starck and A. Hilton. Model-based multiple view reconstruction of people. In *ICCV*, pages 915–922, 2003. 2

[45] J. Starck and A. Hilton. Surface capture for performance-based animation. *Computer Graphics and Applications*, 27:21–31, 2007. 3

[46] C. Theobalt, J. Carranza, and M. A. Magnor. Enhancing silhouette-based human motion capture with 3D motion fields. In *Computer Graphics and Applications*, pages 185–193, 2003. 3

[47] V. Tsiminaki, J. Franco, and E. Boyer. High resolution 3D shape texture from multiple videos. In *CVPR*, pages 1502–1509, 2014. 3

[48] T. Tung and T. Matsuyama. Dynamic surface matching by geodesic mapping for animation transfer. In *CVPR*, pages 1402–1409, 2010. 2

[49] D. Vlasic, I. Baran, W. Matusik, and J. Popovic. Articulated mesh animation from multi-view silhouettes. *ACM Transactions on Graphics, (Proc. SIGGRAPH)*, 27(3):97:1–97:9, 2008. 2

[50] R. Wang, L. Wei, E. Vouga, Q. Huang, D. Ceylan, G. Medioni, and H. Li. Capturing dynamic textured surfaces of moving targets. In *ECCV*, pages 271–288, 2016. 2

[51] L. Wei, Q. Huang, D. Ceylan, E. Vouga, and H. Li. Dense human body correspondences using convolutional networks. In *CVPR*, pages 1544–1553, 2016. 2, 3

[52] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid. DeepFlow: Large displacement optical flow with deep matching. In *ICCV*, pages 1385–1392, 2013. 4, 5, 7, 8

[53] C. Zhang, B. Heeren, M. Rumpf, and W. Smith. Shell PCA: Statistical shape modelling in shell space. In *ICCV*, pages 1671–1679, 2015. 3

[54] M. Zollhöfer, M. Nießner, S. Izadi, C. Rehmann, C. Zach, M. Fisher, C. Wu, A. Fitzgibbon, C. Loop, C. Theobalt, and M. Stamminger. Real-time non-rigid reconstruction using an RGB-D camera. *ACM Transactions on Graphics*, 33(4):156:1–156:12, 2014. 2

[55] S. Zuffi and M. J. Black. The stitched puppet: A graphical model of 3D human shape and pose. In *CVPR*, pages 3537–3546, 2015. 3