Vision, Modeling, and Visualization (2015)
D. Bommes, T. Ritschel and T. Schultz (Eds.)

Accurate Face Reconstruction through Anisotropic Fitting and Eye Correction

Jascha Achenbach Eduard Zell Mario Botsch

Computer Graphics Group, Bielefeld University

Abstract

Fitting a facial template model to 3D-scanner data is a powerful technique for generating face avatars, in particular in the presence of noisy and incomplete measurements. Consequently, there are many approaches for the underlying non-rigid registration task, and these are typically composed from very similar algorithmic building blocks. By providing a thorough analysis of the different design choices, we derive a face matching technique tailored to high quality reconstructions from high resolution scanner data. We then extend this approach in two ways: An anisotropic bending model allows us to more accurately reconstruct facial details. A simultaneous constrained fitting of eyes and eye lids improves the reconstruction of the eye region considerably.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Hierarchy and geometric transformations

1. Introduction

Thanks to the steady advance in acquisition technology, high resolution 3D-scanning is becoming more and more affordable, being based on either laser scanning, structured light scanning, or multi-view stereo. The Kinect sensor and follow-up RGB-D cameras have made 3D-scanning available even to everyday novice users. These technologies have increased the desire to generate virtual clones of real persons, which can be a full-body "3D-selfie" [LVG*13] or a head model for interactive facial puppetry [WBLP11, CHZ14]. However, although surface reconstruction is a rather advanced and mature field of research [BTS*14], reconstructing a complete and high quality surface from noisy and incomplete data is still a challenging task. Incorporating a suitable template model to the reconstruction process enables disambiguation of insufficient data and provides a reasonable surface completion in regions of missing data.

Template fitting is not only used for reconstructing human body scans [ACP03, LVG*13] or head models [BV99, WBLP11, CWZ*14], but also to enable cross-parameterization [ZB13] or statistical shape analysis [BV99, ACP03, CHZ14]. Consequently, a large variety of template fitting methods have been proposed, which are conceptually very similar and share many algorithmic components. A structured evaluation of these components is missing though.

We analyze and compare the individual design choices, and by combining the most promising techniques we derive a template fitting method that provides more accurate reconstructions compared to the typically employed algorithmic components. Nevertheless, a faithful reconstruction of the eye region, which is of high importance for the perception of virtual faces, is still challenging. This is mostly due to scanning artifacts (noise, occlusions) caused by eye lashes or because of highly curved folds around eye lids, which are problematic for template fitting.

We therefore extend our method by an *anisotropic bending model* that more faithfully reconstructs strongly curved facial details. In addition we further improve the reconstruction of the eye region by a *simultaneous constrained fitting* of eye balls and eye lids. The combination of these contributions leads to accurate reconstructions from multi-view stereo data, which we demonstrate on a range of examples.

2. Related Work

Surface registration aligns overlapping components of multiple scans of an object that have been captured from different viewpoints, in order to eventually obtain a complete model of the scanned object. It is a fundamental research topic for computer graphics, computer vision, and reverse engineer-

DOI: 10.2312/vmv.20151251

ing in computer aided geometric design. Early approaches considered *rigid* alignment of range scans only, with approaches being variations of the classic iterative closest point (ICP) algorithm [BM92, CM92, RL01]. In the last decade, *non-rigid* registration of scans captured from deforming objects has been investigated intensively. Since a detailed discussion of general non-rigid registration is out of scope for this article, we refer the reader to the survey paper of Tam et al. [TCL*13] and the course notes of Chang et al. [CLM*10] and Bouaziz et al. [BTP14].

In the following we focus on template fitting, i.e., the non-rigid, deformable registration of a given surface mesh to noisy and incomplete scanner data. Moreover, we focus on 3D-scans or RGB-D as input data, and on general deformable registration of facial models, rather than skeleton-based articulated templates of full human bodies.

Several approaches successfully employ template fitting for reconstructing a consistently triangulated animated mesh from a sequence of measured point clouds for successive time-frames of an actor's performance [WJH*07, LSP08, LAGP09, ZNI*14]. These methods typically compute a template mesh for the first frame, which is then deformed in order to track the following frames.

Blanz and Vetter first proposed a PCA-based statistical face model for reconstructing models from 3D scanner data or even from a single photograph [BV99, BSS07]. Similar face fitting approaches have been proposed since then [THHI06, PB11, YMYK14], some of which are based on piecewise PCA sub-models. In [LKS14] a 3D face is reconstructed from a single RGB-D frame of a person's face, by dividing the input depth frame into semantically meaningful regions and searching the parts individually in a database. Our work uses a PCA-model as well, but only as a prior for initialization.

In their FaceWarehouse project, Cao et al. [CWZ*14] generate an extensive database of animatable face models (shape and pose variations) from Kinect scans of 150 individuals, by deforming a facial template model to fit both the depth data and facial features detected in the color image. Once a PCA-model has been generated, it can be used as a prior to increase the robustness of facial performance tracking (see e.g., [WBLP11, CHZ14]). Recently, Ichim et al. [IBP15] proposed a method for creating a 3D face rig from hand-held video input. In contrast to them, we focus on high-quality reconstruction of a neutral face from accurate 3D scanner data. Finally, Bernard et al. [BBN*14, BBK*15] reconstruct high-quality models of eyes and eyelids using (among other techniques) a non-rigid deformation approach.

Since all these methods for fitting a template model to scanner data can be considered as generalizations of the rigid ICP algorithm [BM92] to non-rigid registration [ARV07, BR07], they naturally share many algorithmic components. Their objective function to be minimized is typically composed of a fitting term, which attracts the template model to the measured point cloud, and a regularization term, which prevents physically implausible deformations. The various approaches mainly differ in how these two components are formulated and computed.

For the fitting term, correspondences between the point cloud and the template model are typically found by simple closest point queries, but these might be computed in the direction of either *scan-to-template* (e.g., [ZB13]) or *template-to-scan* (e.g., [LAGP09]). The fitting energy can then be computed based on Euclidean distances between corresponding points (*point-to-point*) [BM92], distances from tangent planes (*point-to-plane*) [CM92, RL01], or combinations thereof (e.g., [LAGP09]).

While for registration of (incomplete) range images a robust *space deformation* should be used as regularization (e.g., [SSP07] in [LAGP09]), for the fitting of a (clean and complete) template model we can employ a *surface-based deformation*. This regularization term might be based on a *linearly* elastic model (e.g., [SKR*06, BR07, ARV07, THHI06]) or a *nonlinear* measure of geometric distortion (e.g., [LSP08, LAGP09, HAWG08, WJH*07, BTP14, CWZ*14, ZNI*14]).

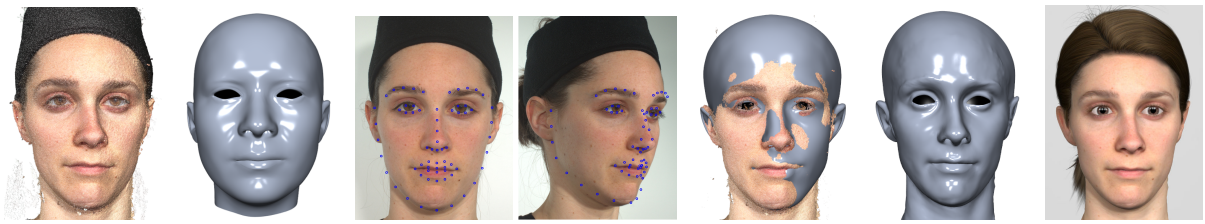
In the following we first analyze the different design options for the fitting and regularization term with respect to reconstruction accuracy and computational performance (Section 3), before proposing an anisotropic bending model for the regularization (Section 4) and a simultaneous fitting of eye balls and eye lids (Section 5).

3. Template Fitting Framework

Our input data was acquired through multi-view reconstruction: From seven high-resolution DSLR camera images we reconstruct a 3D point cloud using the commercial software Agisoft PhotoScan, resulting in about 1 million points (Figure 1(a)). We denote these N input points by $\mathcal{P} = (\mathbf{p}_1, \ldots, \mathbf{p}_N)$, their normal vectors by $\mathbf{n}_j$, and their RGB colors by $\mathbf{c}_j$.

Our goal is to deform a template head model to fit the given scanner data. The template mesh $\mathcal{M}$ consists of n vertices, whose positions are $\mathcal{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$. During the optimization we denote the current (deformed) vertex positions by $\mathbf{x}_i$ and the original (undeformed) positions by $\bar{\mathbf{x}}_i$. Our template model is based on the FaceWarehouse database [CWZ*14] and consists of about 12k vertices, as shown in Figure 1(b).

In order to remove outliers caused by erroneous hair samples, we initially perform a simple skin detection in RGB color space [KPS03] to prune any non-skin points. This effectively removes outliers (e.g., due to scanning hairs), but also removes sample points corresponding to beards or eye brows, such that these regions will be filled by the template data. If instead facial hair is to be reconstructed accurately, the method of Beeler et al. [BBN*12] could be used.



(a) (b) (c) (d) (e) (f)

Figure 1: *Template fitting pipeline: (a) Input point cloud, consisting of 1.4M sample points, (b) template mesh from FaceWarehouse with 12k vertices, (c) automatically detected facial features, (d) initial feature-based alignment, (e) final fit after non-rigid registration, (f) rendering with additional hair and eyes.*

Like all rigid or non-rigid ICP-based approaches [BM92], our face matching technique requires a coarse initial alignment to converge to a meaningful result. We obtain a robust and fully automatic initial alignment by detecting facial landmarks in the input RGB images (using [AZCP13]) and fitting the template model to them, as also proposed, e.g., by Cao et al. [CWZ*14]. In contrast to them, we do not have to distinguish between interior and contour features, since we obtain reliable 3D-positions for all 2D-landmarks by detecting and reconstructing the facial features around eyes, nose, and mouth from the frontal image, while the other features are reconstructed from the side views (Figure 1(c)). Similar to Cao et al. [CWZ*14], we fit the template PCA-model to the detected facial landmarks by determining the global position and orientation, as well as the PCA weights, in order to best match the landmark positions in a (Tikhonov-regularized) least squares sense (Figure 1(d)).

After initialization, the deformable registration updates the vertex positions $\mathcal{X}$, such that the template model better fits the scanner points $\mathcal{P}$ (Figure 1(e)). This is achieved by minimizing an objective function $E(\mathcal{X})$ that consists of a fitting and a regularization term:

$$E(\mathcal{X}) = E_{\text{fit}}(\mathcal{X},\mathcal{P}) + \lambda E_{\text{reg}}(\mathcal{X},\tilde{\mathcal{X}}). \qquad (1)$$

The *fitting energy* E_{fit} penalizes the distance between the template $\mathcal{X}$ and the point-cloud $\mathcal{P}$ (Section 3.1), and the *regularization energy* E_{reg} penalizes the distortion from the undeformed state $\tilde{\mathcal{X}}$ to the deformed state $\mathcal{X}$ (Section 3.2). In the spirit of non-rigid ICP [ARV07, LSP08] we alternatingly compute correspondences and minimize (1), starting with a rather stiff surface ($\lambda = 1$) that is subsequently softened until $\lambda = 10^{-7}$ to allow for more and more accurate fits.

3.1. Fitting Energy

The fitting energy penalizes the distance between corresponding point pairs from $\mathcal{X}$ and $\mathcal{P}$, which we compute as simple closest point correspondences due to simplicity and speed. These correspondences can be constructed either *from template to scan* or *from scan to template*. The former finds for each of the n template vertices $\mathbf{x} \in \mathcal{X}$ the closest point in $\mathcal{P}$, whereas the latter finds for each of the N points $\mathbf{p} \in \mathcal{P}$ its closest neighbor on the template mesh $\mathcal{M}$. This closest point is usually located within a triangle of the template mesh, which is expressed in terms of barycentric coordinates.

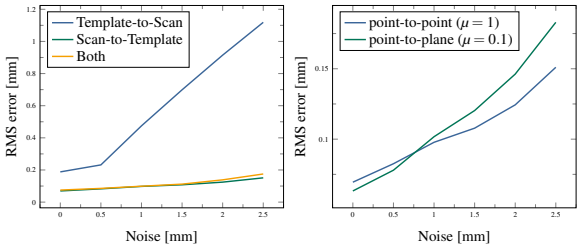
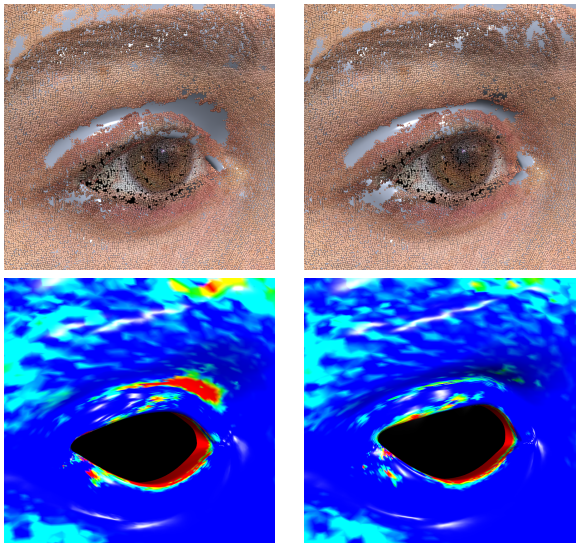


Figure 2: *Comparison of correspondence directions (template-to-scan vs. scan-to-template, left) and correspondence distance (point-to-point vs. point-to-plane, right) on a high-resolution face scan with synthetically added noise (uniformly distributed, zero mean).*

The lower computational complexity ($\mathcal{O}(n \log N)$ vs. $\mathcal{O}(N \log n)$ for $N \gg n$) and the simpler implementation is the reason that most approaches choose template-to-scan correspondences (e.g., [LAGP09, WBLP11, BTP14]). However, a direct comparison on a high-resolution synthetic face scan reveals that scan-to-template correspondences lead to a more accurate reconstruction, in particular for noisy data (Figure 2, left). Although the employed uniform noise does not model the real noise characteristics of our/any scanner, comparisons on real data also show improved fits for scan-to-template correspondences (Figure 3). Although the overall fitting process is about 3–4 times slower using scan-to-template correspondences (for our n and N), we chose this option since we prefer an accurate over a fast reconstruction.

Once correspondences are found, the fitting energy penalizes their (squared) deviation, which is measured either in a *point-to-point* or *point-to-plane* manner, or a linear combination thereof. If we denote the correspondences as a set of



template-to-scan scan-to-template

Figure 3: *For high-resolution scanner data, our scan-to-template correspondences (right) yield more accurate reconstruction than the typically employed template-to-scan correspondences (left). The bottom row shows a color-coding of the two-sided Hausdorff distance of scan and template.*

pairs $\{(\mathbf{p}_j, \tilde{\mathbf{x}}_j)\}_j$, with $\tilde{\mathbf{x}}_j$ being the point on $\mathcal{M}$ closest to $\mathbf{p}_j$, the combined fitting energy can be written as

$$E_{\text{fit}}(\mathcal{X}) = \frac{1}{N}\sum_{j=1}^{N} \mu \left\|\tilde{\mathbf{x}}_j - \mathbf{p}_j\right\|^2 + (1-\mu)\left(\mathbf{n}_j^T\left(\tilde{\mathbf{x}}_j - \mathbf{p}_j\right)\right)^2.$$

The first term measures point-to-point distances, the second point-to-plane distances, and μ blends between the two.

Most recent non-rigid registration approaches (e.g., [LSP08, LAGP09, BTP14]) suggest to use a combination of point-to-point and point-to-plane metric ($\mu = 0.1$), since this allows the template to "slide" along the point cloud and requires fewer iterations.

To analyze the performance of both approaches, we compare a pure point-to-point distance ($\mu = 1$) and the combined distance ($\mu = 0.1$) using several high-resolution scans shown in this paper. Our experiments confirm that the point-to-point distance requires about 30 % more iterations than the combined distance measure. However, the point-to-point distance is computationally faster, since it results in three linear systems of size $n \times n$ (the problem is separable in x/y/z). In contrast, the point-to-plane distance couples the coordinates, which results in one $3n \times 3n$ system. For the complete fitting process, the point-to-point fitting in average took about one third of the computational time of the point-to-plane variant. Since both methods converge to comparable fits (Figure 2, right), we decide for the faster option.

3.2. Regularization Energy

During the fitting process, the regularization energy E_{reg} is responsible for ensuring physical validity of the deformed model by penalizing unwanted types of deformations, typically by trying to keep the surface locally rigid. The two design options are (i) whether to use a surface-based or space-based deformation and (ii) whether to use a linear or a nonlinear deformation model.

Since we fit a clean template model to scanner data, we can safely employ a *surface-based deformation*, which in turn allows us to employ well-established discrete bending models for the deformation energy.

In order to decide between a linear and nonlinear deformation model, we compare two representative techniques on a synthetic head dataset with known solution. Our regularization energy minimizes a discrete bending model by penalizing the Laplacian of the deformation:

$$E_{\text{reg}}(\mathcal{X}) = \frac{1}{\sum_i A_i}\sum_{i=1}^{n} A_i \left\|\Delta\mathbf{x}_i - \mathbf{R}_i\Delta\bar{\mathbf{x}}_i\right\|^2. \qquad (2)$$

The Laplacian $\Delta\mathbf{x}_i$ is discretized using the cotangent weights and A_i is the local Voronoi area of vertex i [BKP*10]. The per-vertex best-fitting rotations $\mathbf{R}_i$ cancel out local rigid transformation, such that the model can deal with large deformations [SA07].

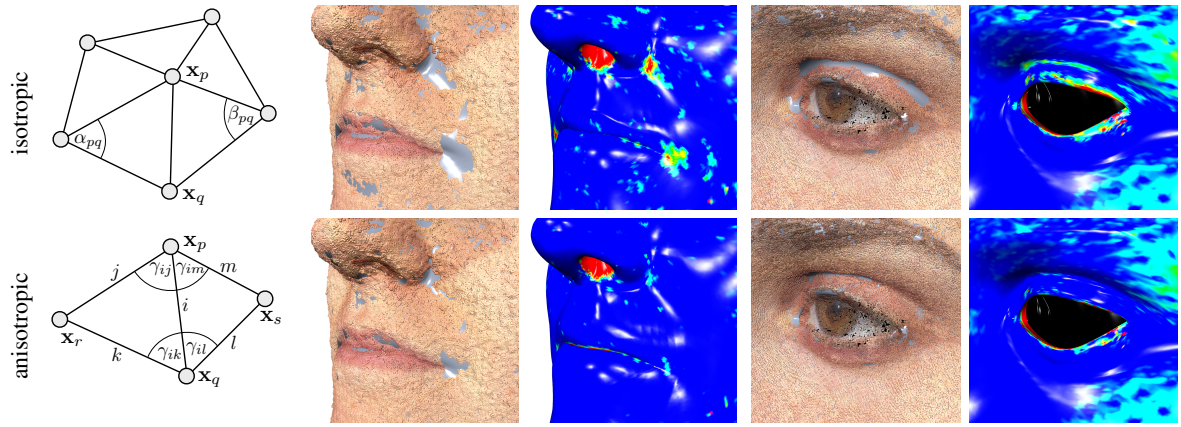
The linear deformation omits the rotations $\mathbf{R}_i$, which turns (2) into a linear thin shell model [BS08]. Since the point-to-point fitting energy is also quadratic in the unknown vertex positions, minimizing the combined energy (1) requires to solve three $n \times n$ systems, which is very efficient. However, the linear model erroneously penalizes locally rigid transformations, which might prevent an accurate fit.

Our nonlinear model solves for vertex positions $\mathbf{x}_i$ and local rotations $\mathbf{R}_i$ using alternating optimization (or block coordinate descent), similar to [SA07]. This method is easy to implement, can pre-factorize its constant system matrix, and can solve for x/y/z using three $n \times n$ systems. However, the overall process has to be iterated until convergence, which in our experiments required about 2–3 iterations only.

The comparisons on the synthetic datasets revealed that the RMS error of the linear model is about twice as large as that of the nonlinear model, with the difference being concentrated around mouth, nose, and eyes. The increased accuracy of the nonlinear model comes at the price of a factor of about 10 in computational cost. Since our primary goal is a precise reconstruction, we (like most recent approaches) choose the nonlinear deformation model.

3.3. Hierarchical Optimization

To improve computational performance while at the same time providing an accurate high resolution template fit, we employ a hierarchical optimization, inspired by [ZNI*14].



(a) Laplacian notation (b) Fitting results on real data

Figure 4: *Notation for discrete Laplacians (a) and close-ups of fitting results (b) for isotropic (top) and anisotropic (bottom) bending energies. The anisotropic bending, using the Huber norm of edge Laplacians, yields more accurate fits of local facial features. The color coding visualizes the two-sided Hausdorff distance between the mesh and the point cloud.*

Our simple two-level hierarchy starts with the original template resolution of 12k vertices [CWZ*14], on which we run the fitting algorithm from stiff ($\lambda = 1$) to soft ($\lambda = 10^{-7}$). After convergence, we apply one step of Loop subdivision to the template model, resulting in about 46k vertices, and perform one more inner loop with stiffness $\lambda = 10^{-7}$.

To reduce costly correspondence computations we follow Bonarrigo et al. [BSB14] and subsample the point cloud $\mathcal{P}$ to a density that is four times higher than the vertex density of the template mesh. We perform this subsampling using an efficient voxelization approach [RC11], with voxel-size being 1/4 of the template's mean edge length. When we subdivide the template, the point subsampling is updated accordingly. We verified Bonarrigo's statement that using more points does not noticeably improve fitting accuracy. This simple two-level hierarchy improved the performance from >12 min to <2 min for our examples, while not affecting accuracy.

4. Anisotropic Refinement

In the (typical) case of noisy input data, the stiffness parameter λ has to be chosen carefully in order to balance between underfitting (surface too stiff, imprecise fit) and overfitting (surface too soft, reconstruction of noise). A sufficiently high surface stiffness yields a smooth fit even for noisy data, but unfortunately also prevents the development of mid-scale facial wrinkles and other high-curvature facial features. Those, however, are typically anisotropically bent, with a high maximum principal curvature and a rather small minimum curvature. This is inherently difficult to fit with an isotropic bending model, which the discrete Laplacian energy (2) is.

We therefore propose to switch to an anisotropic bending model in order to improve the fitting for anisotropic facial features. Due to Polthier [Pol02], the discrete Laplacian of vertex p (Figure 4(a))

$$\Delta\mathbf{x}(p) \;=\; \sum_{(p,q)\in\mathcal{E}} (\cot\alpha_{pq} + \cot\beta_{pq})\,(\mathbf{x}_q - \mathbf{x}_p)$$

can be decomposed into a sum of discrete edge-based Laplacians of all edges i incident to vertex p:

$$\Delta\mathbf{x}(p) \;=\; \sum_{i=(p,*)} \Delta^e\mathbf{x}(i).$$

While the Laplacian $\Delta^e\mathbf{x}(i)$ of edge i is typically defined in the edge-based linear Crouzeix-Raviart basis, it can be reformulated in terms of the vertex-based linear Lagrange basis [WBH*07], yielding the discrete edge Laplacian

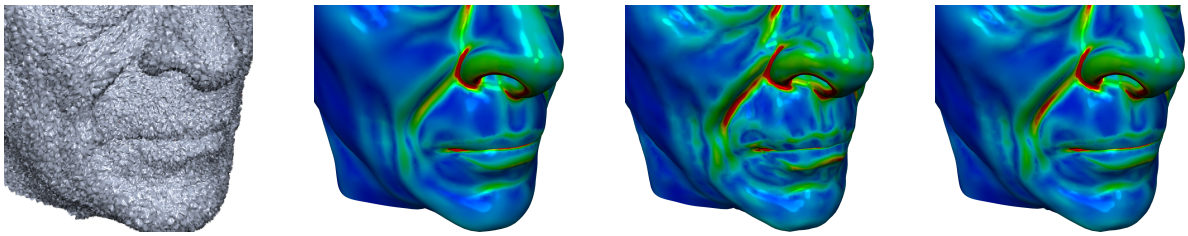
$$\begin{aligned}\Delta^e\mathbf{x}(i) \;=\; &(\cot\gamma_{il} + \cot\gamma_{im})\,\mathbf{x}_s - (\cot\gamma_{ik} + \cot\gamma_{il})\,\mathbf{x}_p + \\ &(\cot\gamma_{ij} + \cot\gamma_{ik})\,\mathbf{x}_r - (\cot\gamma_{ij} + \cot\gamma_{im})\,\mathbf{x}_q,\end{aligned}$$

where γ_* are the incident angles of edge i (Figure 4(a)). The edge Laplacian should be normalized by the edge area A_e, which is 1/3 of the sum of the areas of its two incident triangles. Interestingly, this formulation is identical to the differential edge operator proposed by He and Schaefer [HS13].

To achieve the desired anisotropic fitting, we re-formulate the regularization energy (2) in terms of edge Laplacians

$$E_{\text{reg}}(\mathcal{X}) \;=\; \frac{1}{\sum_e A_e} \sum_{e\in\mathcal{E}} A_e \left\|\Delta^e\mathbf{x}(e) - \mathbf{R}_e\Delta^e\bar{\mathbf{x}}(e)\right\|_h, \qquad (3)$$

where we use the robust Huber norm $\|\cdot\|_h$. This metric behaves like an ℓ^2-norm below a certain threshold and like an ℓ^1-norm above, thereby allowing for stronger local bending for some edges. The minimization of the Huber norm can be implemented as an iteratively re-weighted ℓ^2 minimization [MB93], requiring 2–5 iterations until convergence. For



(a) $\pm 1.5\,mm$ noise added (b) L2 norm, $\lambda = 10^{-7}$ (c) L2 norm, $\lambda = 10^{-8}$ (d) Huber norm, $\lambda = 10^{-7}$

Figure 5: *Comparison of isotropic and anisotropic bending on a synthetic model with added noise: The isotropic bending either does not fit the wrinkle well (b) or overfits the noisy input (c, see mouth region). The anisotropic model does not suffer from overfitting and reconstructs the wrinkle better. The RMS errors for (b), (c), and (d) are 0.36 mm, 0.43 mm, and 0.28 mm.*

all examples we used a Huber threshold of $h = 10^{-6}$. Note that our anisotropic *bending* is similar to anisotropic fairing [HP04], where certain edge Laplacians are weighted down to concentrate curvature (instead of bending).

Figure 4 compares the isotropic and anisotropic bending models, and shows the anisotropic model to more accurately reconstruct facial details at the nose, mouth and eyelids. Figure 5 shows further results on a synthetic noisy model with facial wrinkles. It can be seen that the isotropic model has problems with either under- or overfitting, while the anisotropic model yields a better fit.

5. Eye Lids Correction

The eye region is perceptually one of the most crucial parts of a virtual face. Unfortunately, in scanner data it is typically very noisy, such that the above fitting strategies would typically fail around the eye lid (see Figure 7). Due to the amount of noise in this region, manually picked 3D correspondences between the template model and the point cloud (e.g., [ARV07, WBLP11]) can cause either jaggy eye contours for low stiffness values or inaccurate matching for high stiffness values. We solve a combined 2D/3D fitting in order to correct for these problems.

In a first step we fit 3D eyeballs. To this end we detect both eyes in the frontal image (Figure 1(c)), which can be done robustly using several CV algorithms. From the eye pixels we discard all that are not white/bright enough (belonging to the cornea) or that are classified as skin. This effectively keeps only the pixels corresponding to the sclera, whose corresponding 3D positions (known from the scanning) constitute two point sets that approximately lie on two spheres (the eyeballs). After the initial PCA alignment (Figure 1(d)) we initialize two eyeball meshes (spheres of radius 1.25 cm) at the eye position of the template model. We then iteratively fit these two spheres to the sclera point clouds in an ICP manner, by adjusting positions and (coupled) radii.

Given the precise fit of the eyeballs, it is possible to accurately define the contour of the eye lids. We use seven

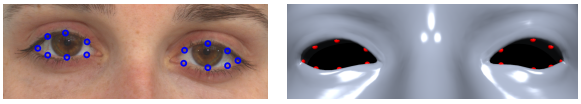


Figure 6: *We detect 2D features on the eye contour (left) and use them as 3D fitting constraints on the template (right).*

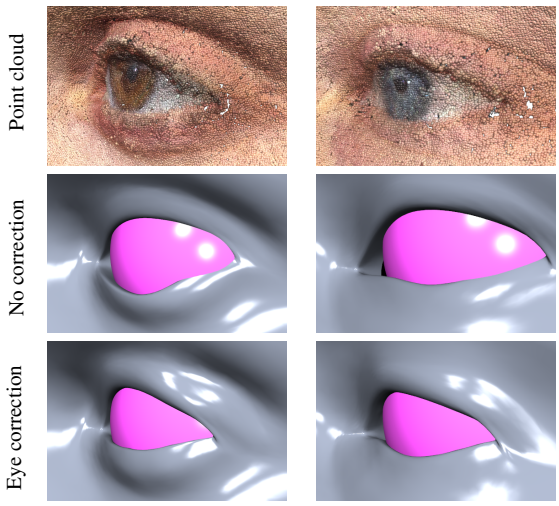


Figure 7: *Fitting an eyeball (pink) to sclera points of the scan and using it to define target positions on the eye contours improves the reconstruction of the eye lids, as shown for two models (left and right).*

feature points on each eye's contour from the frontal photograph, as shown in Figure 6, left. We mark those feature points manually, since the automatically detected facial features are not precise enough. However, using more advanced computer vision algorithms this step can probably be automated. For each of these 2D feature points, the camera calibration yields a viewing ray, which we intersect with the fitted eyeball to get a 3D feature point.

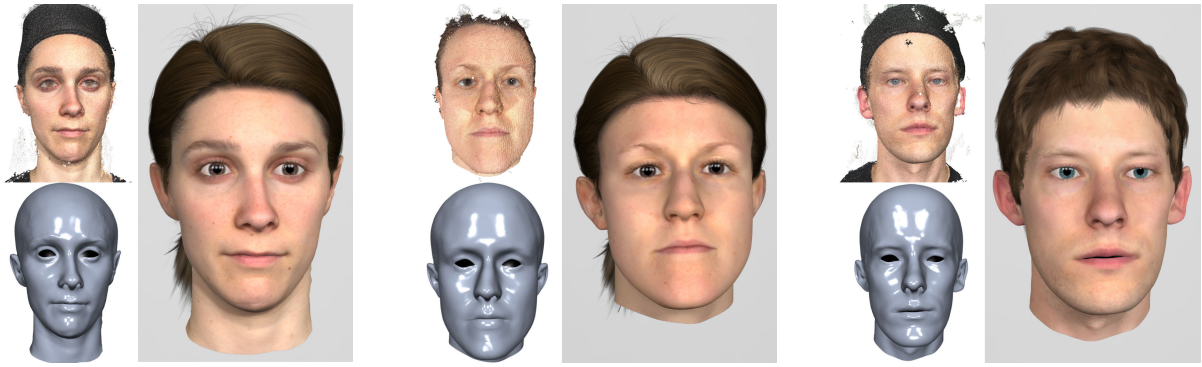


Figure 8: *Three different examples obtained with our proposed non-rigid registration technique, by using anisotropic refinement and contour correction for the eye lids. Each example shows the original scan, the fitted model, and a final rendering.*

The resulting 14 feature points $\mathbf{f}_i$ act as point-to-point constraints for the corresponding vertices $\mathbf{x}_i$ on the template model (Figure 6, right). A corresponding point-to-point fitting term is added to the global energy (1) with weight 0.1 and is used throughout the template fitting process. To further improve the eye lid reconstruction, we constrain all vertices on the *interior* of the template's eye lids to lie exactly on the eyeball spheres, using the projective constraints of Shape-Up [BDS*12]. The results in Figure 7 show how our combined eyeball-eyelid fitting considerably improves the reconstruction of the eye region.

6. Results and Conclusion

Our template fitting framework is based on a structured analysis of the different algorithmic building blocks for non-rigid registration, of which we combine the most promising design choices. When fitting accuracy is the primary goal, our evaluation shows that the fitting energy should use *scan-to-template correspondences*. Moreover, simple *point-to-point distances* are fully sufficient in terms of fitting accuracy, and provide performance benefits compared to point-to-plane distances. Regularizing the fitting with a *nonlinear deformation model* leads to a more precise fit. Combined with the anisotropic refinement and the eyeball/eyelid correction, our method yields accurate and detailed face reconstructions in a couple of minutes (<5 for all our examples). Figure 8 shows more results obtained with our method, based on multi-view stereo reconstruction. The image shows for each example the reconstructed point cloud, the obtained template fit, and a final rendering with additional textures, eyes, and hair.

An interesting direction for future work is the reconstruction of non-neutral facial expressions, where the transfer of our constrained eyeball/eyelid fitting toward the combined reconstruction of teeth and lips should help to produce more realistic results. Moreover, a more precise detection of eye contours (Figure 6) would avoid manual interaction and make our approach fully automatic.

Acknowledgments

We thank Kun Zhou for providing the FaceWarehouse data [CHZ14]. The scanner data in Figure 5 is from http://www.3dscanstore.com. This work was supported by the Cluster of Excellence Cognitive Interaction Technology "CITEC" (EXC 277) at Bielefeld University, which is funded by the German Research Foundation (DFG), and through the project "KogniHome" funded by the Federal Ministry of Education and Research (BMBF).

References

[ACP03] ALLEN B., CURLESS B., POPOVIĆ Z.: The space of human body shapes: reconstruction and parameterization from range scans. *ACM Transactions on Graphics 22*, 3 (2003). 1

[ARV07] AMBERG B., ROMDHANI S., VETTER T.: Optimal step nonrigid ICP algorithms for surface registration. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition* (2007). 2, 3, 6

[AZCP13] ASTHANA A., ZAFEIRIOU S., CHENG S., PANTIC M.: Robust discriminative response map fitting with constrained local models. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition* (2013). 2

[BBK*15] BERMANO A., BEELER T., KOZLOV Y., BRADLEY D., BICKEL B., GROSS M.: Detailed spatio-temporal reconstruction of eyelids. *ACM Transactions on Graphics 34*, 4 (2015). 2

[BBN*12] BEELER T., BICKEL B., NORIS G., MARSCHNER S., BEARDSLEY P., SUMNER R. W., GROSS M.: Coupled 3D reconstruction of sparse facial hair and skin. *ACM Transactions on Graphics 31*, 4 (2012). 2

[BBN*14] BÉRARD P., BRADLEY D., NITTI M., BEELER T., GROSS M.: High-quality capture of eyes. *ACM Transactions on Graphics 33*, 6 (2014). 2

[BDS*12] BOUAZIZ S., DEUSS M., SCHWARTZBURG Y., WEISE T., PAULY M.: Shape-up: Shaping discrete geometry with projections. *Computer Graphics Forum 31*, 5 (2012). 7

[BKP*10] BOTSCH M., KOBBELT L., PAULY M., ALLIEZ P., LEVY B.: *Polygon Mesh Processing*. AK Peters, 2010. 4

[BM92] BESL P. J., MCKAY N. D.: A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence 14*, 2 (1992). 2

[BR07] BROWN B. J., RUSINKIEWICZ S.: Global non-rigid alignment of 3-D scans. *ACM Transactions on Graphics 26*, 3 (2007). 2

[BS08] BOTSCH M., SORKINE O.: On linear variational surface deformation methods. *IEEE Transactions on Visualization and Computer Graphics 14*, 1 (2008). 4

[BSB14] BONARRIGO F., SIGNORONI A., BOTSCH M.: Deformable registration using patch-wise shape matching. *Graphical Models 76*, 5 (2014). 5

[BSS07] BLANZ V., SCHERBAUM K., SEIDEL H.-P.: Fitting a morphable model to 3D scans of faces. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition* (2007). 2

[BTP14] BOUAZIZ S., TAGLIASACCHI A., PAULY M.: Dynamic 2D/3D registration. In *Eurographics Tutorials* (2014). 2, 3, 4

[BTS*14] BERGER M., TAGLIASACCHI A., SEVERSKY L., ALLIEZ P., LEVINE J., SHARF A., SILVA C.: State of the art in surface reconstruction from point clouds. In *Eurographics State of the Art Reports* (2014). 1

[BV99] BLANZ V., VETTER T.: A morphable model for the synthesis of 3D faces. In *Proc. of ACM SIGGRAPH* (1999). 1, 2

[CHZ14] CAO C., HOU Q., ZHOU K.: Displaced dynamic expression regression for real-time facial tracking and animation. *ACM Transactions on Graphics 33*, 4 (2014). 1, 2, 7

[CLM*10] CHANG W., LI H., MITRA N., PAULY M., WAND M.: Geometric registration for deformable shapes. In *Eurographics Tutorials* (2010). 2

[CM92] CHEN Y., MEDIONI G.: Object modelling by registration of multiple range images. *Image Vision Comput. 10*, 3 (1992). 2

[CWZ*14] CAO C., WENG Y., ZHOU S., TONG Y., ZHOU K.: FaceWarehouse: A 3D facial expression database for visual computing. *IEEE Transactions on Visualization and Computer Graphics 20*, 3 (2014). 1, 2, 3, 4

[HAWG08] HUANG Q.-X., ADAMS B., WICKE M., GUIBAS L. J.: Non-rigid registration under isometric deformations. *Computer Graphics Forum 27*, 5 (2008). 2

[HP04] HILDEBRANDT K., POLTHIER K.: Anisotropic filtering of non-linear surface features. *Computer Graphics Forum 23*, 3 (2004). 5

[HS13] HE L., SCHAEFER S.: Mesh denoising via L0 minimization. *ACM Transactions on Graphics 32*, 4 (2013). 5

[IBP15] ICHIM A. E., BOUAZIZ S., PAULY M.: Dynamic 3D avatar creation from hand-held video input. *ACM Transactions on Graphics 34*, 4 (2015). 2

[KPS03] KOVAC J., PEER P., SOLINA F.: Human skin color clustering for face detection. In *Proc. of EUROCON 2003. Computer as a Tool. The IEEE Region 8* (2003). 2

[LAGP09] LI H., ADAMS B., GUIBAS L. J., PAULY M.: Robust single-view geometry and motion reconstruction. *ACM Transactions on Graphics 28*, 5 (2009). 2, 3, 4

[LKS14] LIANG S., KEMELMACHER-SHLIZERMAN I., SHAPIRO L. G.: 3D face hallucination from a single depth frame. In *Proc. of International Conference on 3D Vision* (2014). 2

[LSP08] LI H., SUMNER R. W., PAULY M.: Global correspondence optimization for non-rigid registration of depth scans. *Computer Graphics Forum 27*, 5 (2008). 2, 3, 4

[LVG*13] LI H., VOUGA E., GUDYM A., LUO L., BARRON J. T., GUSEV G.: 3D self-portraits. *ACM Transactions on Graphics 32*, 6 (2013). 1

[MB93] MIRZA M. J., BOYER K. L.: Performance evaluation of a class of M-estimators for surface parameter estimation in noisy range data. *IEEE Transactions on Robotics and Automation 9*, 1 (1993). 5

[PB11] PAPAZOV C., BURSCHKA D.: Deformable 3D shape registration based on local similarity transforms. *Computer Graphics Forum 30* (2011). 2

[Pol02] POLTHIER K.: Polyhedral surfaces of constant mean curvature. Habilitation Thesis, Berlin Technical University, 2002. 5

[RC11] RUSU R. B., COUSINS S.: 3D is here: Point cloud library (PCL). In *Proc. of IEEE International Conference on Robotics and Automation* (2011). 5

[RL01] RUSINKIEWICZ S., LEVOY M.: Efficient variants of the ICP algorithm. In *Proc. of International Conference on 3D Digital Imaging and Modeling* (2001). 2

[SA07] SORKINE O., ALEXA M.: As-rigid-as-possible surface modeling. In *Proc. of Symposium on Geometry Processing* (2007). 4

[SKR*06] STOLL C., KARNI Z., RÖSSL C., YAMAUCHI H., SEIDEL H.-P.: Template deformation for point cloud fitting. In *Proc. of Symposium on Point-Based Graphics* (2006). 2

[SSP07] SUMNER R. W., SCHMID J., PAULY M.: Embedded deformation for shape manipulation. *ACM Transactions on Graphics 26*, 3 (2007). 2

[TCL*13] TAM G. K. L., CHENG Z.-Q., LAI Y.-K., LANGBEIN F. C., LIU Y., MARSHALL D., MARTIN R. R., SUN X.-F., ROSIN P. L.: Registration of 3D point clouds and meshes: A survey from rigid to non-rigid. *IEEE Transactions on Visualization and Computer Graphics 19*, 7 (2013). 2

[THHI06] TENA J. R., HAMOUZ M., HILTON A., ILLINGWORTH J.: A validated method for dense non-rigid 3D face registration. In *Proc. of IEEE International Conference on Video and Signal Based Surveillance* (2006). 2

[WBH*07] WARDETZKY M., BERGOU M., HARMON D., ZORIN D., GRINSPUN E.: Discrete quadratic curvature energies. *Computer Aided Geometric Design 24*, 8–9 (2007). 5

[WBLP11] WEISE T., BOUAZIZ S., LI H., PAULY M.: Real-time performance-based facial animation. *ACM Transactions on Graphics 30*, 4 (2011). 1, 2, 3, 6

[WJH*07] WAND M., JENKE P., HUANG Q., BOKELOH M., GUIBAS L., SCHILLING A.: Reconstruction of deforming geometry from time-varying point clouds. In *Proc. of Symposium on Geometric Processing* (2007). 2

[YMYK14] YOSHIYASU Y., MA W.-C., YOSHIDA E., KANEHIRO F.: As-conformal-as-possible surface registration. *Computer Graphics Forum 33*, 5 (2014). 2

[ZB13] ZELL E., BOTSCH M.: ElastiFace: Matching and blending textured faces. In *Proc. of International Symposium on Non-Photorealistic Animation and Rendering* (2013). 1, 2

[ZNI*14] ZOLLHÖFER M., NIESSNER M., IZADI S., REHMANN C., ZACH C., FISHER M., WU C., FITZGIBBON A., LOOP C., THEOBALT C., STAMMINGER M.: Real-time non-rigid reconstruction using an RGB-D camera. *ACM Transactions on Graphics 33*, 4 (2014). 2, 4