Generation of Virtual Humans for Virtual Reality, Medicine, and Domestic Assistance

Dissertation

zur Erlangung des akademischen Grades des
Doktors der Naturwissenschaften (Dr. rer. nat.)
an der Technischen Fakultät der Universität Bielefeld

vorgelegt von
Jascha Achenbach

Bielefeld 2019

Versicherung

Hiermit versichere ich,

- dass mir die geltende Promotionsordnung der Fakultät bekannt ist,
- dass ich die Dissertation selbst angefertigt habe, keine Textabschnitte von Dritten oder eigener Prüfungsarbeiten ohne Kennzeichnung übernommen und alle benutzten Hilfsmittel und Quellen in meiner Arbeit angegeben habe,
- dass Dritte weder unmittelbar noch mittelbar geldwerte Leistungen von mir für Vermittlungstätigkeiten oder für Arbeiten erhalten haben, die im Zusammenhang mit dem Inhalt der vorgelegten Dissertation stehen,
- dass ich die Dissertation noch nicht als Prüfungsarbeit für eine staatliche oder andere wissenschaftliche Prüfung eingereicht habe und
- dass ich keine gleiche, oder in wesentlichen Teilen ähnliche oder eine andere Abhandlung bei einer anderen Hochschule als Dissertation eingereicht habe.

Bielefeld, 2019

Jascha Achenbach

The supplemental material can be found here:
http://doi.org/10.4119/unibi/2936166.

Printed on non-aging paper in compliance with DIN-ISO 9706.

Acknowledgments

First and foremost, I am deeply grateful to my supervisor Prof. Dr. Mario Botsch for arousing my interest in Computer Graphics and Geometry Processing. In my master studies, his enthusiasm and passion for Computer Graphics during his lectures inspired me and became the main motivation for me to start my own research in this field. I am very thankful for his guidance, support, as well as for the innumerable discussions we have had throughout these years during my time as a doctoral candidate. Furthermore, I thank Prof. Dr. Mark Pauly for his valuable time and for agreeing to be my Ph.D. reviewer.

Special thanks go to my colleagues from the Bielefeld Graphics & Geometry Processing group for their highly valuable feedback after research group meetings, during coffee breaks, and for such a friendly atmosphere in general. In particular, I thank Eduard Zell and Thomas Waltemate with whom it was fun to publish, from whom I learned a lot, and who advanced my research by their contributions.

I also want to thank my esteemed collaborators Prof. Dr. Ulrich Schwanecke, Thomas Gietzen, and Robert Brylka for their close cooperation in the highly interesting medical domain as well as Prof. Dr. Marc Erich Latoschik for his insights into Virtual Reality. Furthermore, I gratefully acknowledge the Department of Diagnostic and Interventional Radiology, University Medical Center of the Johannes Gutenberg University Mainz, Germany for providing us with the DICOM data of the human heads. I am grateful to Prof. Dr. Ralf Wagner, Prof. Dr. Friederike Eyssel, Dr. Charlotte Diehl, Dr. Birte Schiffhauer, and Prof. Dr. Stefan Kopp for being open to interdisciplinary collaborations. In this context, I also thank Regina Stodden, Henning Mayer, Hannah Koppenrade, and Jana Blankertz for their help with preparing and running study. Special thanks go to all scanned subjects.

Finally, I am deeply thankful to my family. This thesis would not have been possible without the tremendous support of my parents throughout this time. Special thanks must go to my beloved wife Teresa—for her permanent support, patience, understanding, and for being my photographic model during my work. I appreciate it more than I can say.

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).

Abstract

Virtual humans are employed in various applications including computer games, special effects in movies, virtual try-ons, medical surgery planning, and virtual assistance. This thesis deals with virtual humans and their computer-aided generation for different purposes.

In a first step, we derive a technique to digitally clone the face of a scanned person. 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 eyelids improves the reconstruction of the eye region considerably. Next, we extend this work to full bodies and present a complete pipeline to create animatable virtual humans by fitting a holistic template character. Due to the careful selection of techniques and technology, our reconstructed humans are quite realistic in terms of both geometry and texture. Since we represent our models as single-layer triangle meshes and animate them through standard skeleton-based skinning and facial blendshapes, our characters can be used in standard VR engines out of the box. By optimizing computation time and minimizing manual intervention, our reconstruction pipeline is capable of processing entire characters in less than ten minutes.

In a following part of this thesis, we build on our template fitting method and deal with the problem of inferring the skin surface of a head from a given skull and vice versa. Starting with a method for automated estimation of a human face from a given skull remain, we extend this approach to bidirectional facial reconstruction in order to also estimate the skull from a given scan of the skin surface. This is based on a multilinear model that describes the correlation between the skull and the facial soft tissue thickness on the one hand and the head/face surface geometry on the other hand. We demonstrate the versatility of our novel multilinear model by estimating faces from given skulls as well as skulls from given faces within just a couple of seconds. To foster further research in this direction, we made our multilinear model publicly available.

In a last part, we generate assistive virtual humans that are employed as stimuli for an interdisciplinary study. In the study, we shed light on user preferences for visual attributes of virtual assistants in a variety of smart home contexts.

List of Abbreviations

ACBC	Adaptive choice-based conjoint analysis
CBCT	Cone Beam Computed Tomography
CT	Computed Tomography
DSLR	Digital single-lens reflex camera
FSTT	Facial soft tissue thickness
ICP	Iterative closest point
LM	Linear model
MANCOVA	Multivariate analysis of covariance
MLM	Multilinear model
MRI	Magnetic Resonance Imaging
ns	Not significant
PCA	Principal component analysis
RGB	Red, green, blue
RGB-D	red, green, blue, depth
RMS	Root Mean Square
SVD	Singular value decomposition
TPS	Thin plate spline
VR	Virtual Reality

Contents

1 Introduction

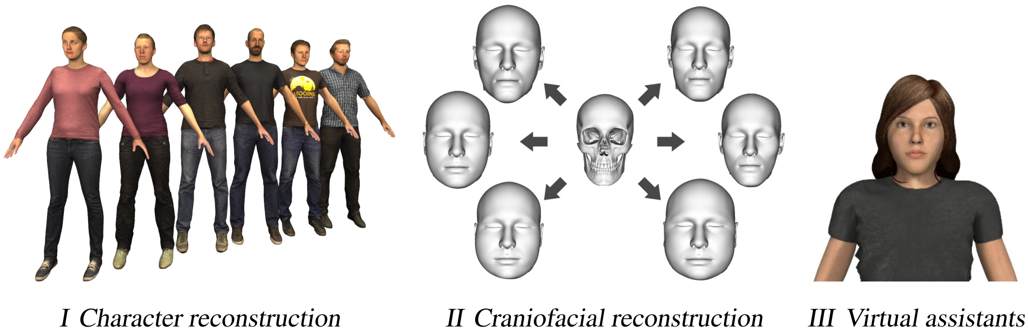
Today, virtual humans are widely used in innumerable contexts including computer games, special effects in movies, virtual try-ons, medical surgery planning, and virtual assistance. This thesis deals with virtual humans and their computer-aided generation for different purposes. The thesis is divided up into three different parts (Figure 1.1). The first part is about *character reconstruction* and focuses on techniques that can be used to generate virtual humans that resemble the appearance of a scanned person. In the second part, based on the derived techniques, we show how to efficiently deal with *craniofacial reconstruction in medicine* by inferring the skin surface of a head from a given skull and vice versa. The third part is about an interdisciplinary study in which we generate virtual humans that are employed as stimuli. Here, we analyze *preferences for virtual assistants* that can be used in smart home contexts. In the following, the three parts will be explained in more detail.

Character Reconstruction

In the context of Virtual Reality (VR), virtual humans are typically deployed as virtual agents simulated by artificial intelligence or as avatars, the digital alter-egos of the users in the virtual worlds. These days, high-resolution 3D-scanning technology is becoming more and more affordable and makes it possible to generate virtual humans by scanning a real person.

Such virtual clones of real persons can be full-body “3D-selfies” [LVG$^+$13] or head models for interactive facial puppetry [WBLP11, CHZ14]. For a robust reconstruction, a suitable template model is typically incorporated to the reconstruction process as it enables disambiguation of insufficient data and provides a reasonable surface completion in regions of missing data.

Such approaches are summarized under the term *template fitting* and can be found in different contexts. Besides being used for reconstructing head scans [BV99, WBLP11, CWZ$^+$14] or body models [ACP03, LVG$^+$13], they are further deployed for cross-parameterization [ZB13] or to enable statistical shape analysis [BV99, ACP03, CHZ14]. Consequently, a large variety of template fitting methods have been proposed. Although the approaches are conceptually very similar and share many algorithmic components, a structured evaluation of these components is still missing. In Chapter 2, we perform a thorough analysis and comparison of the individual design choices. This way, we are able to derive a template fitting method that provides more accurate reconstructions as opposed to typically employed algorithmic components. However, it is still challenging to achieve



I Character reconstruction *II Craniofacial reconstruction* *III Virtual assistants*

Figure 1.1: This thesis deals with virtual humans and their generation for different purposes. First, we focus on techniques that can be used to generate virtual humans that resemble a scanned person (I). Secondly, we show how to efficiently deal with craniofacial reconstruction (II). Finally, we investigate virtual humans that can be used in domestic contexts (III).

a faithful reconstruction of the eye region which is of high importance for the perception of virtual faces. We therefore extend our method to improve the reconstruction of the eye region and to more faithfully reconstruct strongly curved facial details. Overall, this leads to an accurate face reconstruction algorithm from multi-view stereo data.

Building upon this work, we then derive a pipeline to reconstruct full bodies that fulfill the requirements described as follows. A faithful and realistic human-like appearance requires detailed textures and geometrically accurate meshes. Furthermore, the application of the resulting models in interactive scenarios requires the characters to be animated. To be widely employable, the resulting character models should be compatible with standard game engines or VR frameworks. Finally, the overall avatar creation should ideally be fast enough to be performed during rapid prototyping or empirical studies.

However, creating believable and animatable virtual humans in a short amount of time is still a challenging problem. Approaches for the fast computer-aided generation of characters with all required animation controls are mostly lacking. Additionally, many approaches neglect the generation of high-quality textures from scanner input and focus on geometry reconstruction only. In Chapter 3, we present a complete character generation pipeline that is able to digitally clone a real person into a realistic, high-quality virtual human by fitting a holistic template model. The resulting characters can then be used for animation and visualization in any standard graphics or VR engine. The whole reconstruction process requires only a minimum amount of user interaction and takes less than ten minutes on a desktop PC. Overall, our contributions enable the generation of realistic and fully animatable virtual humans in just a couple of minutes. This makes them accessible to a wide range of VR experiments where they can be used as avatars or conversational agents.

Craniofacial Reconstruction in Medicine

An important topic in forensic medicine and archaeology is facial reconstruction from skeleton remains. By providing a human skull and several options of facial soft tissue thickness (FSTT), the goal is to reconstruct plausible facial appearances in order to enable recognition of the unknown subject. Measurements of FSTT provide important quantitative information [SS08] and are crucial for facial approximation and craniofacial superimposition methods. However, measurements based on a few distinct landmark points provide a few discrete thickness values only though a *dense* soft tissue map is to be preferred.

In Chapter 4, we build on our accurate template fitting method and present a method that fits a statistical head model to such a dense soft tissue profile. Thereby, we are able to estimate the visual appearance of the person to be identified. In contrast to most previous methods [TBK$^+$05, TBL$^+$07, RME$^+$14, SZD$^+$16, SZM$^+$17], our approach is fully automated and thus does not require any manual interaction.

Deriving the skull from the face also has high-potential applications in medical contexts. For example, given a 3D face scan, this technology can estimate the skull of a person *without* the need for X-ray radiation or other expensive medical imaging methods. A reasonably accurate, radiation-free alternative would be beneficial, e.g., for patients with craniofacial malformations. Computed Tomography is currently the standard imaging procedure for such patients [CHP03]. Another application is radiation-free bony cephalometric skull assessment in orthodontics. For such an assessment, both the skull and face shape are often of interest and a high radiation dose is prohibitive due to the typically young age of the patients [ECSS04].

In Chapter 5, we build on our work on forensic facial reconstruction to generate proper training data that is subsequently used to compute a multilinear model. This model maps from low-dimensional parameter spaces for skull shape and FSTT to high-resolution triangle meshes of the skull and the head/facial skin. In particular, we show how our model can be evaluated as well as fitted in just a couple of seconds. This allows us to produce skull and skin variations from given skull shape parameters and FSTT parameters, or to determine these parameters by fitting the multilinear model to a given skull or skin measured, e.g., by medical imaging or a face scanner. Moreover, we made our combined statistical model publicly available for research purposes.

Preferences for Virtual Assistants

Conversational virtual agents that serve as assistive technologies have already made their way into users' homes [Gmb18, ANA10, DvM00]. Today, such virtual assistants can facilitate the users' lives by providing information services, e.g., by obtaining information from the Internet [YKPK13]. In the future, they may soon be used by demographically diverse target groups with personal needs and preferences for activities like cooking, planning leisure time, or physical rehabilitation. However, people still exhibit rather negative attitudes toward service robots and show little willingness to integrate them into their everyday lives [Gmb18, RE13, SBE$^+$16]. Thus, to increase users' acceptance, user preferences have to be taken into account [SPC$^+$16, Nie94]. However, previous work lacks a differentiated analysis of demographically diverse users' preferences.

Accordingly, we conduct a corresponding laboratory study in Chapter 6. More specifically, we analyze preferences of users who differ in terms of age, gender, and even hair color with regard to visual attributes of virtual assistants. To this end, we semi-automatically generate virtual assistants that systematically vary in gender, age, hair color, hair length, and clothing. Subsequently, we use these virtual assistants as stimuli.

Based on user preferences and a determined virtual assistant for each participant, we examine the evaluation of the virtual assistant with respect to its appearance and as a function of task domain for different smart home contexts. Additionally, we examine the evaluation of the virtual assistant with respect to the similarity users perceive between them and the virtual assistants. Furthermore, we examine how the preferred virtual assistants are perceived in terms of their warmth and competence. Moreover, we investigate the effect of openness toward this assistive technology on the evaluation of the preferred virtual assistants.

Finally, we conclude this thesis in Chapter 7 by summarizing our results and describing limitations as well as possible directions of future work.

Contributions and Publications

In summary, the main contributions divided into the previously introduced parts are:

Character Reconstruction

- A structured analysis of individual design choices for template fitting methods. By combining the most promising techniques, we derive an accurate template fitting method. Extending our method by both an anisotropic bending model and an improved reconstruction of the eye region, we derive an accurate face reconstruction method.

  Corresponding publication:
  Jascha Achenbach, Eduard Zell, and Mario Botsch. Accurate Face Reconstruction through Anisotropic Fitting and Eye Correction. In *Proceedings of Vision, Modeling and Visualization*, pages 1–8, 2015.

- A complete character generation pipeline that is able to digitally clone a real person into a realistic, high-quality virtual human that can then be used for animation and visualization in any standard graphics or VR engine. The whole reconstruction process requires only a minimum amount of user interaction and takes less than ten minutes on a desktop PC. This makes such virtual humans accessible to a wide range of VR experiments in which they can be used as avatars or conversational agents.

  Corresponding publication:
  Jascha Achenbach, Thomas Waltemate, Marc Erich Latoschik, and Mario Botsch. Fast Generation of Realistic Virtual Humans. In *Proceedings of ACM Symposium on Virtual Reality Software and Technology*, pages 1–10, 2017.

Craniofacial Reconstruction in Medicine

- A method for fully automatic forensic facial reconstruction based on dense statistics of soft tissue thickness.

  Corresponding publication:
  Thomas Gietzen, Robert Brylka, Jascha Achenbach, Katja zum Hebel, Elmar Schömer, Mario Botsch, Ulrich Schwanecke, and Ralf Schulze. A method for automatic forensic facial reconstruction based on dense statistics of soft tissue thickness. In *PLOS ONE*, 14(1), pages 1–19, 2019.

- A multilinear model for bidirectional craniofacial reconstruction that can be evaluated as well as fitted in just a couple of seconds. To foster further research in this direction, we made our multilinear model publicly available.

  Corresponding publication:

  Jascha Achenbach, Robert Brylka, Thomas Gietzen, Katja zum Hebel, Elmar Schömer, Ralf Schulze, Mario Botsch, and Ulrich Schwanecke. A Multilinear Model for Bidirectional Craniofacial Reconstruction. In *Proceedings of Eurographics Workshop on Visual Computing for Biology and Medicine*, pages 67–76, 2018.

Preferences for Virtual Assistants

- A thorough analysis of different user groups' preferences for the visual attributes of virtual assistants in various smart home contexts. We further investigate how the perceived similarity between users and the virtual assistant's appearance effects the evaluation of virtual assistants. Additionally, we examine how preferred virtual assistants are perceived in terms of their warmth and competence. Also, we investigate the effect of openness toward this technology on the evaluation of preferred virtual assistants.

  Corresponding submission:

  Jascha Achenbach, Friederike Eyssel, Charlotte Diehl, Birte Schiffhauer, Ralf Wagner, Stefan Kopp, and Mario Botsch. Preferences of different user groups for the visual attributes of virtual assistants. In *ACM Transactions on Applied Perception*, 2019, *under submission*.

Other publications in which I was involved but which are not directly relevant to this thesis are:

Daniel Sieger, Sergius Gaulik, Jascha Achenbach, Stefan Menzel, and Mario Botsch. Constrained Space Deformation Techniques for Design Optimization. In *Computer Aided Design 72*, pages 40–51, 2016.

Andreas Richter, Jascha Achenbach, Stefan Menzel, and Mario Botsch. Evolvability as a Quality Criterion for Linear Deformation Representations in Evolutionary Optimization. In *Proceedings of IEEE Congress on Evolutionary Computation*, pages 901–910, 2016.

Andreas Richter, Jascha Achenbach, Stefan Menzel, and Mario Botsch. Multi-objective Representation Setups for Deformation-based Design Optimization. In *Proceedings of International Conference on Evolutionary Multi-Criterion Optimization*, pages 514–528, 2017.

Charlotte Diehl, Birte Schiffhauer, Friederike Eyssel, Jascha Achenbach, Sören Klett, Mario Botsch, and Stefan Kopp. Get One or Create One: The Impact of Graded Involvement in a Selection Procedure for a Virtual Agent on Satisfaction and Suitability Ratings. In *Proceedings of International Conference on Intelligent Virtual Agents*, pages 109–118, 2017.

Marc Erich Latoschik, Daniel Roth, Dominik Gall, Jascha Achenbach, Thomas Waltemate, and Mario Botsch. The Effect of Avatar Realism in Immersive Social Virtual Realities. In *Proceedings of ACM Symposium on Virtual Reality Software and Technology*, pages 1–10, 2017.

Part I

CHARACTER RECONSTRUCTION

2 Accurate Face Reconstruction

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 that can be full-body "3D-selfies" [LVG$^+$13] or head models for interactive facial puppetry [WBLP11, CHZ14]. However, although surface reconstruction is a rather advanced and mature field of research [BTS$^+$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 statistical shape analysis [BV99, ACP03, CHZ14] or cross-parameterization [ZB13]. Consequently, a large variety of template fitting methods that are conceptually very similar and that share many algorithmic components have been proposed. To date, a structured evaluation of these components is lacking.

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 eyelids, 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 eyeballs and eyelids. The combination of these contributions leads to accurate reconstructions from multi-view stereo data. We demonstrate this on a range of examples.

My Contribution *My contribution is a structured analysis of individual design choices for template fitting methods and the derivation of an accurate template fitting method. In addition, my contribution is the extension by an anisotropic bending model and the improvement of the reconstruction of the eye region. The face reconstructions in Figure 2.13 were rendered by Eduard Zell.*
Corresponding publication:

> *[AZB15] Accurate Face Reconstruction through Anisotropic Fitting and Eye Correction, VMV, 2015*

2.1 Related Work

There is a lot of work dedicated to face reconstruction from images, video, RGB-D data, laser scans, and multi-view stereo. Many approaches use an RGB-D sensor to reconstruct face models [CWZ+14, LKS14] and/or to animate them based on captured performance data [BWP13, HMYL15, TZN+15]. However, their face reconstructions suffer from low-quality in geometry and texture due to the inherent limitations of current RGB-D sensors. High-quality face reconstructions can be achieved through multi-camera rigs and multi-view stereo reconstruction [BBB+10, GFT+11, FGT+16]. However, these approaches aim at a *static* high-quality reconstruction and do not provide models that can be animated. Other works use video input to generate *dynamic* face models, which are subsequently animated based on the video stream [SWTC14, CBZB15, WBGB16, TZS+16, GZC+16, OLY+17]. Very recently, methods have been suggested that are based on neural networks and that are able to reconstruct 3D human faces [TZK+17, TZG+18, YSN+18] or complete heads [HSW+17] from a single image. For a comprehensive overview of 3D face reconstruction methods from monocular 2D data, we refer the reader to the state-of-the-art report of Zollhöfer et al. [ZTG+18].

Surface registration is a fundamental technique for most face reconstruction approaches. It 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 engineering in computer-aided geometric design. Early approaches considered *rigid* alignment of range scans only. These approaches were 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 thesis, we refer the reader to Tam et al.'s [TCL+13] survey paper and to Chang et al.'s [CLM+10] and Bouaziz et al.'s [BTP14] course notes.

In this chapter, 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 on 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$^+$14] generate an extensive database of animatable face models (shape and pose variations) from Kinect scans of 150 individuals, by deforming a facial template model to fit both the depth data and facial features detected in the color image. Once a PCA model has been generated, it can be used as a prior to increase the robustness of facial performance tracking (see, e.g., [WBLP11, CHZ14]). Since then, more comprehensive face models have been proposed that are built from thousands of 3D scans and combine a linear shape space with an articulated jaw, neck, eyeballs, and blendshapes [LBB$^+$17]. Ranjan et al. [RBSB18] suggest a versatile model with an hierarchical mesh representation that captures nonlinear variations in shape and expression.

Recently, Ichim et al. [IBP15] proposed a method for creating a textured 3D face rig from picture and video input taken on a cell-phone. In contrast to them, we focus on high-quality reconstruction of a neutral face from accurate 3D scanner data. Another approach by Ichim et al. [IKNDP16] builds a user-specific volumetric face rig and employs it for physics-based animation. This approach was extended in [IKKP17] to include a novel muscle activation model that separates active and passive soft tissue layers. Finally, Berard et al. [BBN$^+$14, BBK$^+$15, BBGB16] 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 a robust *space deformation* should be used as regularization (e.g., [SSP07] in [LAGP09]) for registration of (incomplete) range images, we can employ a *surface-based deformation* for the fitting of a (clean and complete) template model. 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 2.2). We then propose an anisotropic bending model for the regularization (Section 2.3) and a simultaneous fitting of eyeballs and eyelids (Section 2.4).

2.2 Template Fitting Framework

Our input data was acquired through multi-view reconstruction: From seven high-resolution digital single-lens reflex camera (DSLR) images (Figure 2.1) we reconstruct a 3D point cloud using the commercial software Agisoft PhotoScan [Agi17], resulting in about 1 million points (Figure 2.2(a)). We denote these n input points by $\mathcal{P}_F = (\mathbf{p}_1, \ldots, \mathbf{p}_n)$, their normal vectors by $\mathbf{n}_j$, and their RGB colors by $\mathbf{c}_j$. The camera images feature 18 Megapixels and are taken from mid-range consumer DSLR cameras of type Canon 550D with $50\,\mathrm{mm}$ lenses attached, respectively.

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 $12\,\mathrm{k}$ vertices, as shown in Figure 2.2(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 not only outliers (e.g., due to scanning hairs), but also sample points corresponding to beards or eyebrows so that these regions will be filled by the template data.

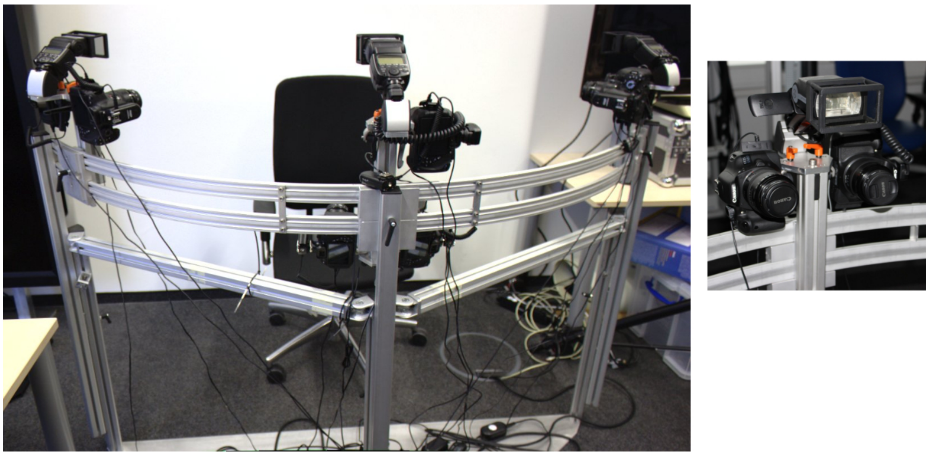
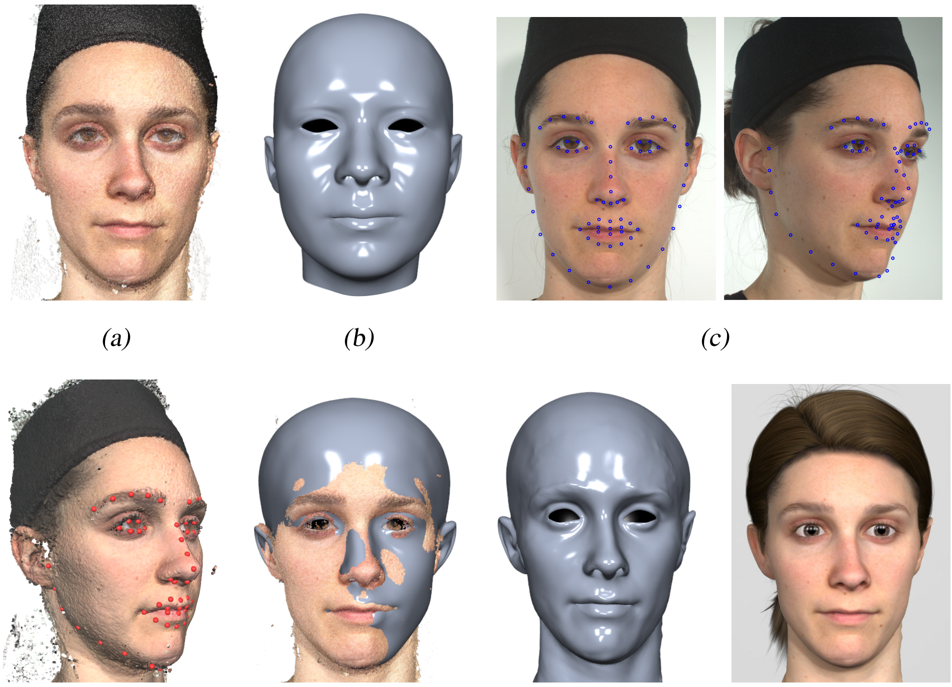


Figure 2.1: Our custom-built face scanner is based on multi-view stereo and consists of 7 DSLR cameras.



(a) *(b)* *(c)*

(d) *(e)* *(f)* *(g)*

Figure 2.2: Template fitting pipeline: Input point cloud, consisting of $1.4\,\mathrm{M}$ sample points (a), template mesh from FaceWarehouse with about $12\,\mathrm{k}$ vertices (b), 66 automatically detected facial features (c), corresponding 3D facial features (d), initial feature-based alignment (e), final fit after non-rigid registration (f), rendering with additional hair $\&$ eyes (g).

If instead facial hair is to be reconstructed accurately, the method of Beeler et al. [BBN$^+$12] could be used.

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 66 facial landmarks $\mathcal{L}$ in the input RGB images (using [AZCP13]) and fitting the template model to them, as also proposed, e.g., by Cao et al. [CWZ$^+$14]. In contrast to them, we do not have to distinguish between interior and contour features since we obtain reliable 3D-positions for all 2D-landmarks by detecting and reconstructing the facial features around eyes, nose, and mouth from the frontal image, while the other features are reconstructed from the side views (Figure 2.2, (c)). We generate a PCA model based on 150 reconstructed heads in neutral expression taken from the FaceWarehouse data [CWZ$^+$14]. Similarly to Cao et al. [CWZ$^+$14], we fit our template PCA model to the detected facial landmarks by determining the global position, orientation, and scaling, as well as the PCA weights, in order to best match the landmark positions in a (Tikhonov-regularized) least-squares sense (Figure 2.2(e)).

Concretely, we first optimize scaling s, rotation $\mathbf{R}$, and translation $\mathbf{t}$ of the point cloud $\mathcal{P}_F$ to align it to the template model by minimizing the sum of squared distances between facial landmarks $\mathbf{p}_l$, $l \in \mathcal{L}$, on the point cloud $\mathcal{P}_F$ and their counterpart vertices $\mathbf{x}_l$ on the template mesh $\mathcal{M}$, i.e., we solve

$$\operatorname*{arg\,min}_{\mathbf{R},\mathbf{t},s} \sum_{l\in\mathcal{L}} \left\|\mathbf{x}_l - s(\mathbf{R}\mathbf{p}_l + \mathbf{t})\right\|^2 .$$

According to Horn [Hor87], this can be computed in closed-form. In a nutshell, the procedure is to first mean-center the two point clouds involved, i.e., to compute

$$\begin{aligned} \bar{\mathbf{p}} &:= \frac{1}{|\mathcal{L}|}\sum_{l\in\mathcal{L}}\mathbf{p}_l & \qquad \bar{\mathbf{x}} &:= \frac{1}{|\mathcal{L}|}\sum_{l\in\mathcal{L}}\mathbf{x}_l \\ \hat{\mathbf{p}}_l &:= \mathbf{p}_l - \bar{\mathbf{p}} & \qquad \hat{\mathbf{x}}_l &:= \mathbf{x}_l - \bar{\mathbf{x}}\,. \end{aligned}$$

Next, a special 4×4 matrix

$$\begin{pmatrix} S_{xx}+S_{yy}+S_{zz} & S_{yz}-S_{zy} & S_{zx}-S_{xz} & S_{xy}-S_{yx} \\ S_{yz}-S_{zy} & S_{xx}-S_{yy}-S_{zz} & S_{xy}+S_{yx} & S_{zx}+S_{xz} \\ S_{zx}-S_{xz} & S_{xy}+S_{yx} & -S_{xx}+S_{yy}-S_{zz} & S_{yz}+S_{zy} \\ S_{xy}-S_{yx} & S_{zx}+S_{xz} & S_{yz}+S_{zy} & -S_{xx}-S_{yy}+S_{zz} \end{pmatrix},$$

where

$$S_{xx} = \sum_{l\in\mathcal{L}}(\hat{\mathbf{x}}_l)_x(\hat{\mathbf{p}}_l)_x, \qquad S_{xy} = \sum_{l\in\mathcal{L}}(\hat{\mathbf{x}}_l)_x(\hat{\mathbf{p}}_l)_y, \qquad \text{and so on}$$

can easily be computed. Its (unit) eigenvector w.r.t. the largest eigenvalue is the quaternion representing the optimal rotation $\mathbf{R}$. The optimal scaling s is then given by

$$s \;=\; \sum_{l\in\mathcal{L}} \hat{\mathbf{x}}_l^T \mathbf{R}\hat{\mathbf{p}}_l \Big/ \sum_{l\in\mathcal{L}} \|\hat{\mathbf{p}}_l\|^2 \;. \tag{2.1}$$

The optimal translation vector $\mathbf{t}$ is the difference between the centroid $\bar{\mathbf{x}}$ and the scaled and rotated centroid $\bar{\mathbf{p}}$

$$\mathbf{t} \;=\; \bar{\mathbf{x}} - s\mathbf{R}\bar{\mathbf{p}}\,. \tag{2.2}$$

In a next step we optimize for PCA weights. Note that the PCA model is of dimension d ($d = 10$ in our case) and can be written as

$$H(\mathbf{b}) \;=\; \bar{\mathbf{h}} + \mathbf{H}\mathbf{b}\,,$$

where $\bar{\mathbf{h}}$ is the mean head, $\mathbf{H}$ is the matrix containing the principal components in its d columns, and $\mathbf{b} = (b_1, \ldots, b_d)$ contains the PCA parameters representing a head $H(\mathbf{b})$. Similarly to Cao et al. [CWZ$^+$14], we then fit the template PCA model to the facial landmarks $\mathbf{p}_l$ by choosing its PCA weights $\mathbf{b}$ to minimize

$$E_{\text{PCA}}(\mathbf{b}) \;=\; \frac{1}{|\mathcal{L}|}\sum_{l\in\mathcal{L}} \left\|\mathbf{H}_l\mathbf{b} + \bar{\mathbf{h}}_l - \mathbf{p}_l\right\|^2 \;+\; \frac{\lambda_{\text{tik}}}{d}\sum_{k=1}^{d}\left(\frac{b_k}{\sigma_k}\right)^2 \;, \tag{2.3}$$

which leads to solving a linear least-squares problem. In the first term, $\mathbf{H}_l$ and $\bar{\mathbf{h}}_l$ are the rows of $\mathbf{H}$ and $\bar{\mathbf{h}}$ representing the point $\mathbf{h}_l$ corresponding to $\mathbf{p}_l$, that is $\mathbf{h}_l = \bar{\mathbf{h}}_l + \mathbf{H}_l\mathbf{b}$. We use $\lambda_{\text{tik}} = 0.002$ for the regularization term, where σ_k^2 is the variance of the kth principal component.

After initialization, the deformable registration updates the vertex positions $\mathcal{X}$, such that the template model better fits the scanner points $\mathcal{P}_F$ (Figure 2.2(f)). This is achieved by minimizing an objective function $E_{\text{face}}(\mathcal{X})$ that consists of a fitting and a regularization term:

$$E_{\text{face}}(\mathcal{X}) \;=\; E_{\text{fit}}(\mathcal{X}, \mathcal{P}_F) \;+\; \lambda_{\text{reg}} E_{\text{reg}}\big(\mathcal{X}, \bar{\mathcal{X}}\big) \;. \tag{2.4}$$

The *fitting energy* E_{fit} penalizes the distance between the template $\mathcal{X}$ and the point-cloud $\mathcal{P}_F$ (Section 2.2.1), and the *regularization energy* E_{reg} penalizes the physical distortion from the undeformed state $\bar{\mathcal{X}}$ after initial alignment to the deformed state $\mathcal{X}$ (Section 2.2.2). The minimization of (2.4) finds a compromise between a small alignment error and low physical distortion, which is controlled by the parameter λ_{reg}.

The deformable registration algorithm is summarized in Algorithm 1. In the spirit of non-rigid ICP [ARV07, LSP08], we alternatingly compute correspondences and minimize (2.4), starting with a rather stiff surface ($\lambda_{\text{reg}} = 1$) that is subsequently softened until $\lambda_{\text{reg}} = 10^{-7}$ to allow for more and more accurate fits.

Algorithm 1: Template Fitting

```
Initial alignment
while not done do
    while not done do
        compute correspondences
        deform model to minimize $E_{\text{fit}}(\mathcal{X}) + \lambda_{\text{reg}} E_{\text{reg}}\big(\mathcal{X}, \bar{\mathcal{X}}\big)$
    end
    relax surface stiffness: $\lambda_{\text{reg}} \leftarrow 0.1 \cdot \lambda_{\text{reg}}$
    plastic deformation: $\bar{\mathcal{X}} \leftarrow \mathcal{X}$
end
```

The main design decisions for ICP algorithms are: (i) how to compute correspondences, (ii) how to measure the fitting error, and (iii) how to formulate the regularization energy. In the following two subsections, we analyze different options for each subproblem in order to find the method most suitable for our task.

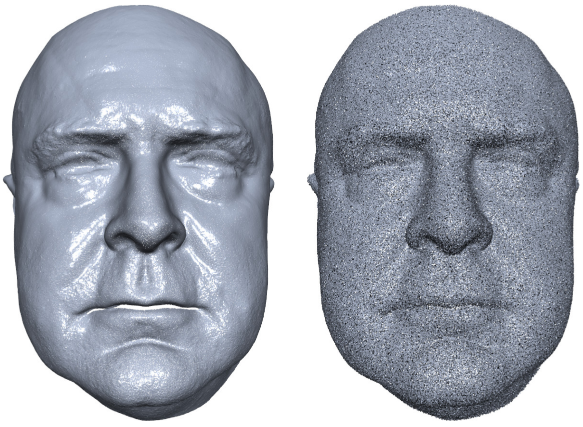
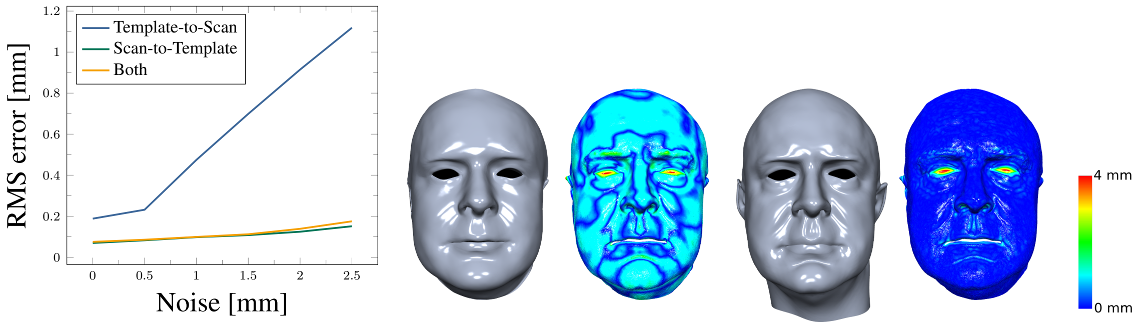


Figure 2.3: Synthetic "scan" by taking the vertices of a high-resolution face model from [BHB$^+$11] (left) and adding varying amounts of noise (right: $\pm 2\,\mathrm{mm}$).

The analysis requires a synthetic dataset with a known ground truth. We use a high-resolution face model from [BHB$^+$11] and use vertices as sample points ($n \approx 1\,\mathrm{M}$). To analyze robustness with respect to varying amounts of noise, we add uniformly distributed noise up to an amplitude of $\pm 2.5\,\mathrm{mm}$ to the model's vertices and recompute (noisy) normal vectors from this data (Figure 2.3). Fitting accuracy is measured as root-mean-square (RMS) error

$$\mathrm{rms}(\mathcal{X}, \mathcal{P}_F) \;=\; \sqrt{\frac{1}{n}\sum_{j=1}^{n} \|\tilde{\mathbf{x}}_j - \mathbf{p}_j\|^2}$$

from the clean *non-noisy* ground-truth points $\mathbf{p}_j$ to the closest points $\tilde{\mathbf{x}}_j$ on the deformed template.



(a) RMS Comparison *(b) Template-to-Scan* *(c) Scan-to-Template*

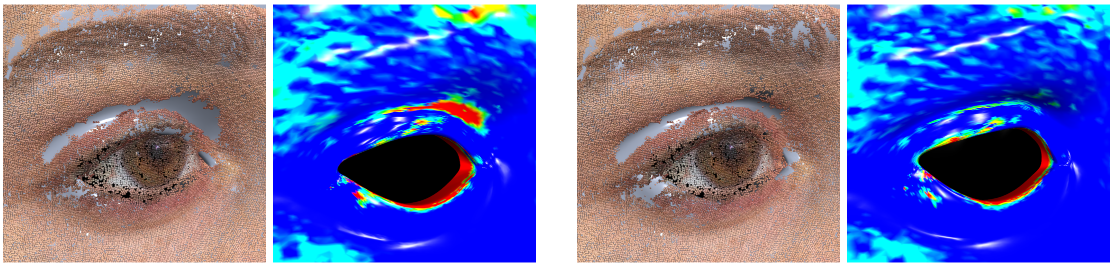
Figure 2.4: Comparison of template-to-scan and scan-to-template correspondences for varying amounts of noise plotted as RMS error to non-noisy ground truth data (a). Scan-to-template correspondences are clearly superior for noisy input data. The fitting results (b) and (c) correspond to noise of ± $1.5\,\mathrm{mm}$. Their RMS errors are $0.7\,\mathrm{mm}$ and $0.1\,\mathrm{mm}$, respectively.

2.2.1 Fitting Energy

The fitting energy penalizes the distance between corresponding point pairs from $\mathcal{X}$ and $\mathcal{P}_F$, 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 the closest point in $\mathcal{P}_F$ for each of the N template vertices $\mathbf{x} \in \mathcal{X}$, whereas the latter finds the closest neighbor on the template mesh $\mathcal{M}$ for each of the n points $\mathbf{p} \in \mathcal{P}_F$. This closest point is usually located within a triangle of the template mesh, which is expressed in terms of barycentric coordinates.

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 the high-resolution synthetic face scan reveals that scan-to-template correspondences lead to a more accurate reconstruction, in particular for noisy data (Figure 2.4). 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 2.5). 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.

Since the quality of the alignment strongly depends on the choice of good correspondences, many heuristics for pruning bad correspondences exist [RL01]. We also employ the typical pruning strategies, i.e., we discard correspondences that are on the boundary,



(a) *Template-to-Scan*

(b) *Scan-to-Template*

Figure 2.5: For high-resolution scanner data, our scan-to-template correspondences (b) yield more accurate reconstruction than the typically employed template-to-scan correspondences (a). The color-coding visualizes the two-sided Hausdorff distance of scan and template.

that have a distance above a certain threshold, or that have a normal deviation above a certain threshold (typically $5\,\mathrm{mm}$ and $30°$).

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 measured using a linear combination of the two. If we denote the correspondences as a set of 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 the two. Note that, due to correspondence pruning, the number of valid correspondences is, in practice, smaller than n and should replace n in the above equation.

Most recent non-rigid registration approaches (e.g., [LSP08, LAGP09, BTP14]) suggest using 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

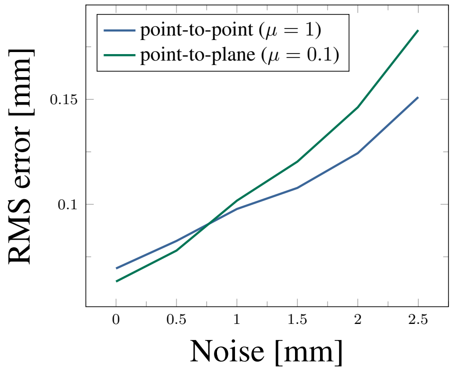


Figure 2.6: Comparison of pure point-to-point and combined point-to-point/point-to-plane distance for the fitting energy for varying amounts of noise, plotted as RMS error to non-noisy ground truth data.

distance ($\mu = 1$) with the combined distance ($\mu = 0.1$) using several high-resolution scans shown in this chapter. 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 because 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, resulting in one $3N \times 3N$ system. For the complete fitting process, the point-to-point fitting took about one third of the computational time of the point-to-plane variant on average. Since both methods converge to comparable fits (Figure 2.6), we decide to use the faster option.

2.2.2 Regularization Energy

During the fitting process, the regularization energy E_{reg} is responsible for ensuring the 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:

$$E_{\text{reg}}(\mathcal{X}, \bar{\mathcal{X}}) = \frac{1}{\sum_i A_i} \sum_{i=1}^{N} A_i \left\| \Delta \mathbf{x}_i - \mathbf{R}_i \Delta \bar{\mathbf{x}}_i \right\|^2 . \tag{2.5}$$

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 (2.5) 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 (2.4) requires solving 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). This method, as proposed in [SA07] and sketched in Algorithm 2, alternatingly fixes the vertex position $\mathbf{x}_i$ and computes the best rotation matrices $\mathbf{R}_i$, and then fixes the matrices $\mathbf{R}_i$ and solves for the vertex positions $\mathbf{x}_i$.

Algorithm 2: Alternating Optimization

```
while not done do
    Fix vertices $\mathcal{X}$ and find optimal rotations $\mathcal{R}$
    Fix matrices $\mathcal{R}$ and solve for vertex positions $\mathcal{X}$
end
```

While the best rotation matrices $\mathbf{R}_i$ could be found by SVD polar decomposition [SA07], following [BML$^+$14], an easier closed-form solution can be found by replacing $\mathbf{R}_i \Delta\bar{\mathbf{x}}_i$ in (2.5) with $\frac{\Delta \mathbf{x}_i \|\Delta\bar{\mathbf{x}}_i\|}{\|\Delta \mathbf{x}_i\|}$. The latter sub-problem is quadratic in the vertex positions $\mathbf{x}_i$ and hence amounts to solving a sparse weighted linear least-squares problem.

Given an overdetermined system of linear equations $\mathbf{A}\mathbf{x} = \mathbf{b}$ with a matrix $\mathbf{A} \in \mathbb{R}^{p\times q}$, $(p > q)$, a vector $\mathbf{b} \in \mathbb{R}^p$, and the residual $\mathbf{r} = \mathbf{b} - \mathbf{A}\mathbf{x}$, the linear least-squares problem is finding $\mathbf{x}^* \in \mathbb{R}^q$ so that

$$E(\mathbf{x}) \;=\; \sum_{i=1}^{p} \left(b_i - \sum_{j=1}^{q} A_{ij} x_j \right)^2 \;=\; \sum_{i=1}^{p} r_i(\mathbf{x})^2 \;=\; \|\mathbf{b} - \mathbf{A}\mathbf{x}\|^2$$

is minimized. It is well known that the minimizer is given by the solution of the normal equations

$$\mathbf{A}^T\mathbf{A}\mathbf{x} \;=\; \mathbf{A}^T\mathbf{b}\,.$$

The weighted linear least-squares problem [Bjo96]

$$E(\mathbf{x}) \;=\; \sum_{i=1}^{p} w_i \left(b_i - \sum_{j=1}^{q} A_{ij} x_j \right)^2 \;=\; \sum_{i=1}^{p} w_i r_i(\mathbf{x})^2 \;=\; \left\|\mathbf{W}^{1/2}(\mathbf{b} - \mathbf{A}\mathbf{x})\right\|^2$$

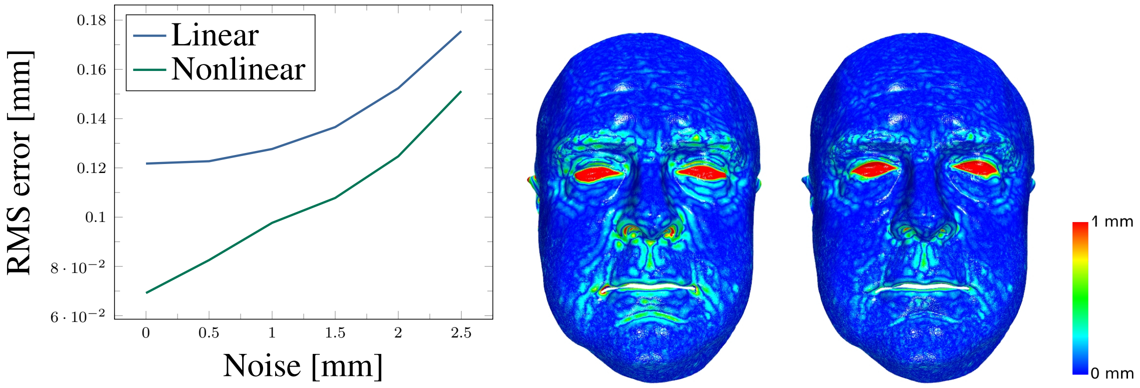
weights the (squared) residuals, while $\mathbf{W} = \mathrm{diag}(\ldots, w_i, \ldots)$. The normal equations are then given by

$$\mathbf{A}^T\mathbf{W}\mathbf{A}\mathbf{x} \;=\; \mathbf{A}^T\mathbf{W}\mathbf{b}\,.$$

Our sub-problem, i.e. minimizing (2.4) for vertex positions $\mathbf{x}_i$ given matrices $\mathbf{R}_i$, is a weighted linear least-squares problem since it is of the general form

$$E(\mathbf{x}) \;=\; \frac{\mu_1}{\sum_i w_{i,1}} \sum_{i=1}^{p_1} w_{i,1} r_i(\mathbf{x})^2 \;+\; \ldots \;+\; \frac{\mu_T}{\sum_i w_{i,T}} \sum_{i=1}^{p_T} w_{i,T} r_i(\mathbf{x})^2\,,$$

with $p = \sum_{t=1}^{T} p_t$ being the overall number of constraints and with additional weights $\frac{\mu_t}{\sum_i w_{i,t}}$ for the different energy terms.



(a) RMS Comparison *(b) Linear* *(c) Nonlinear*

Figure 2.7: Comparison of linear and nonlinear regularization energy for varying amounts of noise plotted as RMS error to non-noisy ground truth data (a). A nonlinear regularization energy is more accurate. The color-coded distances (b) and (c) correspond to fits without noise. Their RMS errors are $0.12\,\mathrm{mm}$ and $0.07\,\mathrm{mm}$, respectively.

Solving our nonlinear model using alternating optimization is easy to implement, the constant system matrix can be pre-factorized, and one can solve for x/y/z using three $N \times N$ systems. The overall process has to be iterated until convergence is reached. In in our experiments the process required about 2 – 3 iterations only.

The comparisons on the synthetic dataset of Figure 2.3 revealed that the RMS error of the linear model is about twice as large as that of the nonlinear model ($0.12\,\mathrm{mm}$ vs. $0.07\,\mathrm{mm}$). The difference is concentrated around mouth, nose, and eyes (Figure 2.7). 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.

2.2.3 Hierarchical Optimization

To improve computational performance while simultaneously providing an accurate high-resolution template fit, we employ a hierarchical optimization inspired by [ZNI+14]. Our simple two-level hierarchy starts with the original template resolution of about $12\,\mathrm{k}$ vertices [CWZ+14] on which we run the fitting algorithm from stiff ($\lambda_{\text{reg}} = 1$) to soft ($\lambda_{\text{reg}} = 10^{-7}$). After convergence, we apply one step of Loop subdivision to the template model, resulting in about $46\,\mathrm{k}$ vertices, and perform one more inner loop with stiffness $\lambda_{\text{reg}} = 10^{-7}$.

To reduce costly correspondence computations, we follow Bonarrigo et al. [BSB14] and sub-sample the point cloud $\mathcal{P}_F$ 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 a voxel size that is 1/4 of the template's mean edge length. When we subdivide the template, the point subsampling is updated accordingly. In this context, we verify Bonarrigo's statement that using more points does not noticeably improve fitting accuracy. This simple two-level hierarchy improved the performance from $> 12\,\mathrm{min}$ to $< 2\,\mathrm{min}$ for our examples while accuracy remained unaffected.

2.3 Anisotropic Refinement

In the (typical) case of noisy input data, the stiffness parameter λ_{reg} 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 (2.5) represents.

We therefore propose switching 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 2.8(a))

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

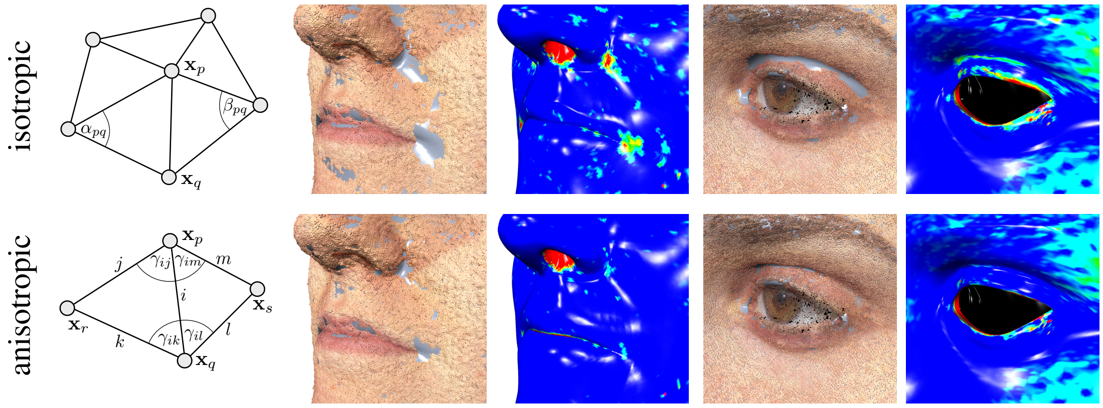
can be decomposed into a sum of discrete, edge-based Laplacians of all edges i incident to vertex p:

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

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 2.8(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].



(a) Laplacian notation

(b) Fitting results on real data

Figure 2.8: 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.

To achieve the desired anisotropic fitting, we re-formulate the regularization energy (2.5) in terms of edge Laplacians

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

where we use the robust Huber norm $\|\cdot\|_h$. $\mathbf{R}_e$ are per-edge rotations to best-fit deformed and undeformed Laplacians. This metric behaves like an ℓ^2-norm below a certain threshold h and like an ℓ^1-norm above (see Equation (2.6)), 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.

Let $\mathbf{A} \in \mathbb{R}^{p \times q}, \mathbf{b} \in \mathbb{R}^p, \mathbf{x} \in \mathbb{R}^q$ as before and $\mathbf{r} = \mathbf{b} - \mathbf{A}\mathbf{x}$ be the residual. An objective function

$$E(\mathbf{x}) \;=\; \sum_{i=1}^{p} \rho(r_i(\mathbf{x}))\,,$$

with

$$\rho(r_i) \;=\; \begin{cases} \frac{1}{2} r_i^2, & |r_i| \leq h, \\ h\,|r_i| - \frac{1}{2} h^2, & |r_i| > h, \end{cases} \tag{2.6}$$

being the Huber function, can be minimized by using an iteratively reweighted least-squares algorithm [MB93]. The procedure is based on iteratively solving a weighted least-squares problem and is given in Algorithm 3 [Bjo96]. When minimizing the Huber function, the weight function is

$$w(r_i) \; = \; \begin{cases} 1, & |r_i| \leq h, \\ \frac{h}{|r_i|}, & |r_i| > h. \end{cases}$$

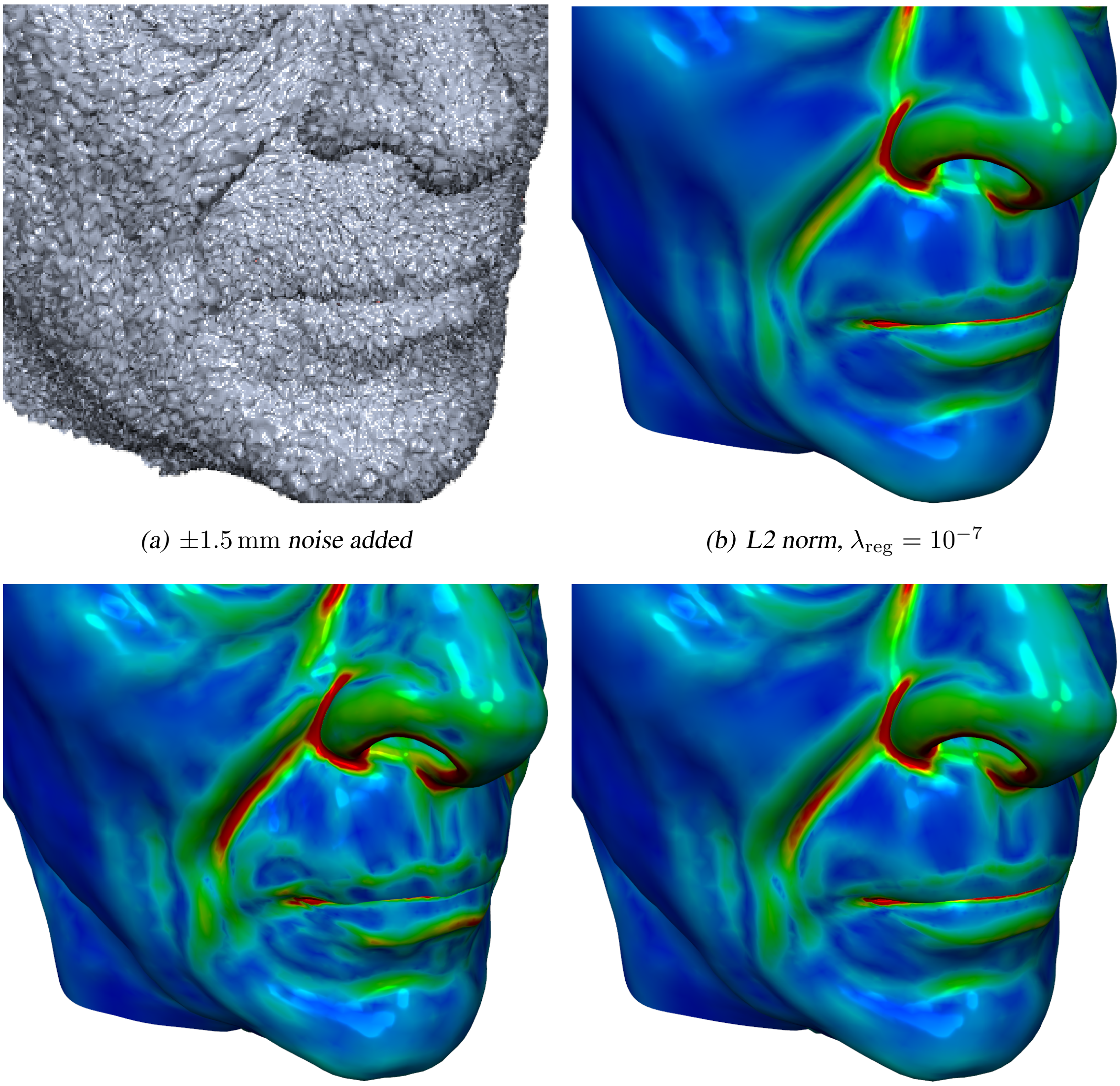
Algorithm 3: Iteratively reweighted least squares

```
Let $\mathbf{A} \in \mathbb{R}^{p \times q}, \mathbf{b} \in \mathbb{R}^p, \mathbf{x} \in \mathbb{R}^q$
Let $\mathbf{x}^{(0)} = \arg\min_{\mathbf{x}} \|\mathbf{b} - \mathbf{A}\mathbf{x}\|^2$ be an initial approximation
for $k = 0, 1, 2, \dots$ do
    $r_i^{(k)} = (\mathbf{b} - \mathbf{A}\mathbf{x}^{(k)})_i\,,$     with $i = 1, \dots, p$
    $\mathbf{W}_k = \mathrm{diag}(\dots, w(r_i^{(k)}), \dots)$
    solve $\delta\mathbf{x}^{(k)}$ from
        $\arg\min_{\delta\mathbf{x}} \left\|\mathbf{W}_k(\mathbf{r}^{(k)} - \mathbf{A}\delta\mathbf{x})\right\|^2$
    $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \delta\mathbf{x}^{(k)}$
end
```

For 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 2.8 compares the isotropic and anisotropic bending models and shows that the anisotropic model more accurately reconstructs facial details at the nose, mouth, and eyelids. Figure 2.9 shows further results on a synthetic noisy model with facial wrinkles[1]. It can be seen that the isotropic model has problems with either under- or overfitting, while the anisotropic model yields a better fit.

[1] The scanner data in Figure 2.9 is from http://www.3dscanstore.com.



(a) $\pm 1.5\,\mathrm{mm}$ *noise added*

(b) L2 norm, $\lambda_{\mathrm{reg}} = 10^{-7}$

(c) L2 norm, $\lambda_{\mathrm{reg}} = 10^{-8}$

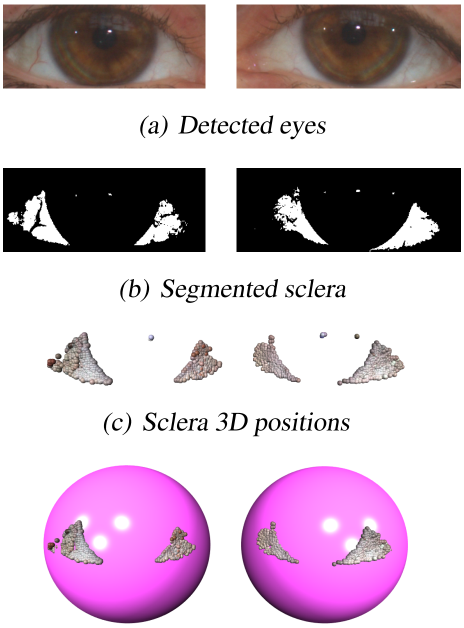
(d) Huber norm, $\lambda_{\mathrm{reg}} = 10^{-7}$

Figure 2.9: 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 $0.36\,\mathrm{mm}$, $0.43\,\mathrm{mm}$, and $0.28\,\mathrm{mm}$.

2.4 Eyelids 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 eyelid (Figure 2.12). 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 2.2(c)) which can be done robustly using several computer vision algorithms. We do so by considering the region around eyes given by the detected facial landmarks during initial alignment. 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 leaves us with only the pixels corresponding to the sclera, whose corresponding 3D positions (known from the scanning) constitute two point clouds that approximately lie on two spheres (the eyeballs). After the initial PCA alignment (Figure 2.2(e)), we initialize two eyeball meshes (spheres of radius $1.25\,\mathrm{cm}$) 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 (Figure 2.10).


(a) Detected eyes

(b) Segmented sclera

(c) Sclera 3D positions

(d) Fitted spheres

Figure 2.10: To fit 3D eyeballs we detect both eyes in the frontal image (a), segment the sclera (b) with corresponding 3D position (c) and fit two spheres to the sclera point clouds (d).

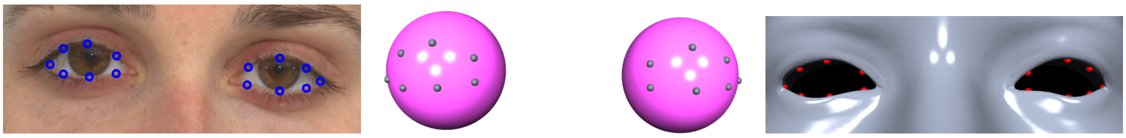


Figure 2.11: We detect 2D features on the eye contour (left), compute 3D feature points (middle), and use them as 3D fitting constraints on the template (right).

Given the precise fit of the eyeballs, it is possible to accurately define the contour of the eyelids. We use seven feature points on each eye's contour from the frontal photograph (Figure 2.11, left). We mark those feature points manually, since the automatically detected facial features are not precise enough. However, by 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 2.11, middle).

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 2.11, right). A corresponding point-to-point fitting term

$$E_{\text{eye}}(\mathcal{X}) = \frac{\lambda_{\text{eye}}}{14} \sum_{i=1}^{14} \|\mathbf{x}_i - \mathbf{f}_i\|^2$$

is added to the global energy (2.4) with weight $\lambda_{\text{eye}} = 0.1$ and is used throughout the template fitting process. To further improve the eyelid reconstruction, we constrain all vertices on the *interior* of the template's eyelids to lie exactly on the eyeball spheres using the projective constraints of Shape-Up [BDS$^+$12]. The results in Figure 2.12 show how our combined eyeball-eyelid fitting considerably improves the reconstruction of the eye region.

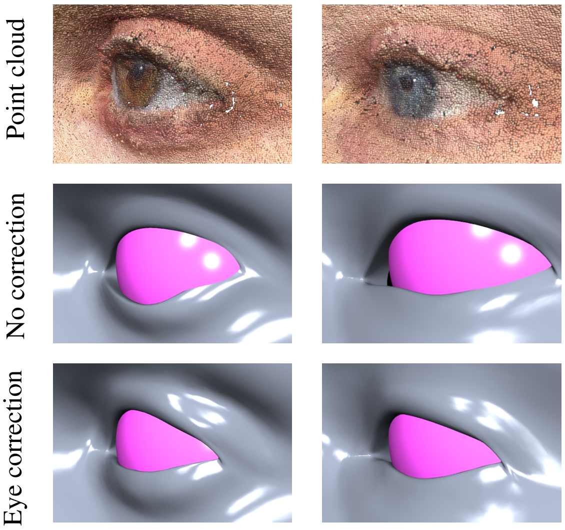


Figure 2.12: Fitting an eyeball (pink) to sclera points of the scan and using it to define target positions on the eye contours improves the reconstruction of the eyelids as shown for two models (left and right).

2.5 Results

Our template fitting framework is based on a structured analysis of the different algorithmic building blocks for non-rigid registration. For our framework, 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 when 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 ($< 5\,\mathrm{min}$ for all our examples) on a desktop PC with

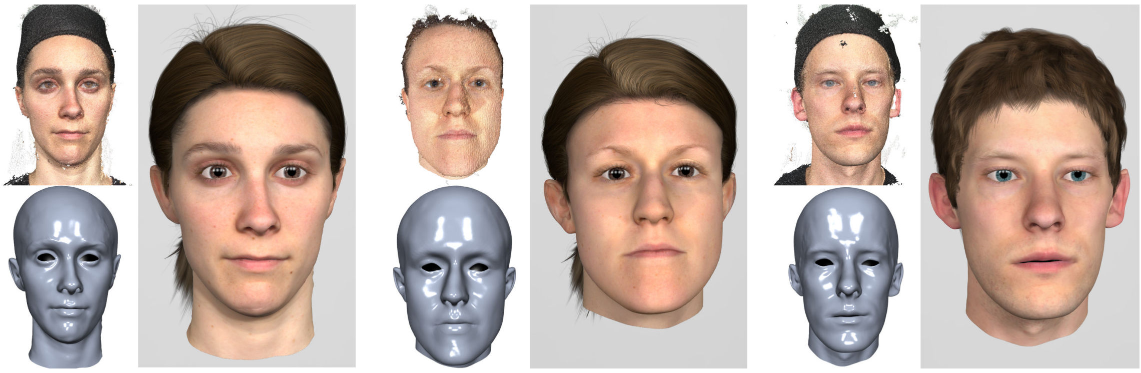


Figure 2.13: Three different examples obtained with our proposed non-rigid registration technique by using anisotropic refinement and contour correction for the eyelids. Each example shows the original scan, the fitted model, and a final rendering.

Intel Xeon CPU ($4 \times 3.6\,\mathrm{GHz}$). Figure 2.13 shows more results obtained with our method, which is based on multi-view stereo reconstruction. For each example, the image shows the reconstructed point cloud, the obtained template fit, and a final rendering with additional textures, eyes, and hair.

In this chapter, we derived an accurate template fitting method that provides accurate face reconstructions from multi-view stereo data. With this method at hand, we continue with presenting how to digitally clone a real person into a realistic high-quality virtual human.

3 Fast Generation of Realistic Virtual Humans

Today, virtual humans are widely used in innumerable contexts including computer games, special effects in movies, virtual try-ons, medical surgery planning, and virtual assistance. Virtual humans are especially important in the Virtual Reality (VR) context for both virtual agents simulated by artificial intelligence as well as for avatars, the digital alter-egos of the users in the virtual worlds. Immersive embodied scenarios provide ample possibilities for studying psychophysical effects caused by modifying avatar appearance. Hence, e.g., altering self-perception and body ownership [SSSVB10, GFPMSS10, PSAS13, BS14, LLL15, LLR16, RWS$^+$16, LRG$^+$17] are common and interesting topics in VR research.

Striving for realism and human-like appearance requires geometrically accurate meshes and detailed textures, and the application of the resulting models in interactive scenarios requires them to be animated: Their full-body posture, hand posture, eye gaze, and facial expressions have to be controllable through suitable skeletal rigs and blendshapes, respectively. To be widely employable, the resulting character models should be compatible with standard game engines or VR frameworks, and the overall avatar creation should ideally be fast enough to be performed during rapid prototyping or empirical studies.

However, despite the increasing availability of scanning technologies and the large body of research on 3D-scanning and mesh reconstruction in both computer vision and computer graphics, creating believable and animatable virtual humans in a short amount of time is still a challenging problem. Existing approaches reconstruct static full-body “selfies” [LVG$^+$13] without animation controls, full-body models without controls for hands or facial expressions [BRLB14, BBLR15], or head models for facial puppetry without a full body [WBLP11, CHZ14]. Approaches for the fast generation of characters with all required animation controls are mostly lacking. In addition, many approaches focus on geometry reconstruction only and neglect the generation of high-quality textures from scanner input.

In this chapter, we extend our approach for face reconstruction from Chapter 2 to full bodies and present a complete character generation pipeline that is able to digitally clone a real person into a realistic high-quality virtual human that can then be used for animation and visualization in any standard graphics or VR engine. The whole reconstruction process requires only a minimum amount of user interaction and takes less than ten minutes on a desktop PC.

For 3D-scanning, we employ a custom-built camera rig with 40 cameras for the body and our face scanner from Chapter 2 for the face. We extend the latter to 8 cameras for a better coverage of the face region and compute dense point clouds through multi-view stereo reconstruction. In order to robustly deal with noise and missing data, and to avoid character rigging in a post-process, we fit a generic human body model to the user's scanner data. More specifically, we build upon the template model from Autodesk Character Generator [Aut14] which is already equipped with a detailed skeleton and skinning weights, a rich set of blendshapes, as well as eyes and teeth. This template model is further enriched by statistical data on human body shapes and, thus, yields a prior for the template fitting process. By fitting the template geometry to the scanner data and transferring eyes, teeth, skeleton, and blendshapes to the morphed template, our reconstructed models are ready to be animated.

By construction, all our reconstructed characters share the tessellation of the template model. Hence they are in dense one-to-one correspondence, which allows transferring of properties between models. As one application example, we exploit this fact by scanning subjects with and without clothing and then storing the clothes, i.e., the difference between the two models. This allows us to easily and seamlessly transfer clothing from one character to another and largely reduces potential confounds caused by different cloths from different avatars used, e.g., in perception studies. To keep our models simple and compatible to any standard rendering engine and to enable highly efficient character animation, we represent our characters by a single-layer mesh and employ standard skinning and blendshapes for body and face animation, respectively.

Overall, our contributions enable the generation of realistic and fully animatable virtual humans in just a couple of minutes on a desktop PC requiring only a minimum amount of user interaction. This makes the virtual humans accessible to a wide range of VR experiments where they can be used as avatars or conversational agents.

My Contribution *The virtual human generation pipeline was developed in close cooperation with Thomas Waltemate. I developed the non-rigid registration framework from Chapter 2. The framework was extended jointly by Thomas Waltemate and me to a pipeline so that the fitting to full bodies became possible. Further, Thomas Waltemate worked on speeding up the pipeline to decrease the overall computing time to under* 10 *minutes. I worked on a faithful face reconstruction, including facial details and blendshapes. Additionally, I worked on the clothing transfer and implemented the texture processing techniques presented in this chapter. Moreover, we worked together on the pipeline so that it would work as automatically and reliably as possible. Finally, the full-body scanning rig was designed and built by both of us.*
Corresponding publication:

[AWLB17] Fast Generation of Realistic Virtual Humans, VRST, 2017

3.1 Related Work

Due to the increasing availability of 3D-scanning solutions and the growing demand for virtual human models, there is a huge body of literature on scanning, reconstructing, and animating virtual characters. In the following, we focus on the approaches most relevant to ours. We begin with techniques for reconstructing full body models. We then extend the related work on face capturing methods from the previous chapter. Finally, we discuss approaches for reconstructing animatable VR characters.

Full-Body Reconstruction

Several methods employ affordable RGB-D sensors (e.g., Kinect) for scanning and reconstructing human bodies [TZL$^+$12, LVG$^+$13, SBKC13, FSR$^+$14]. However, due to the coarse and noisy data delivered by these sensors, their character reconstructions are bound to a rather low quality. Very recently, methods have been proposed that obtain 3D body models and texture from monocular video [AMX$^+$18b, AMX$^+$18a] or a single image [KBJM18, OLPM$^+$18]. Since our goal is reconstructing realistic, high-quality virtual humans, we instead base our framework on a multi-camera rig that can capture a subject in a fraction of a second. Using multi-view stereo we then reconstruct a dense point cloud from the camera data.

This point cloud could then be fed into a surface reconstruction method, followed by an auto-rigging process for embedding a control skeleton and defining skinning weights [BP07, FCS15]. However, the surface reconstruction might fail to faithfully capture delicate features (e.g., fingers), causing the auto-rigging to fail. We therefore use a fully-rigged template model that we fit to the scanner data using non-rigid registration.

Fitting a template model to a large amount of training data allows the construction of a statistical model which can act as prior when fitting the template to scanner data. The SCAPE model [ASK+05] is one of the first, most prominent, and most frequently employed human body models. It has been extended in many ways [HLRB12, BRLB14, SBB07, SHRB12, PWH+17] and has been applied in different scenarios including breathing animation [TMB14], soft-tissue animation [LMB14], and estimation of shape and posture from either a single image [GWBB09] or from RGB-D sequences [WHB11, BBLR15].

Many other statistical human body models have been proposed that can be roughly classified as triangle-based or vertex-based methods, depending on how they model posture articulation and fine-scale deformation [ACP03, ACPH06, HSS+09, WPB+14, LMR+15]. Triangle-based methods have to solve a linear Poisson system to compute the deformed vertex positions and are therefore incompatible to standard graphics engines. In contrast, models based on per-vertex linear blend skinning, such as, e.g., SMPL [LMR+15] or S-SCAPE [PWH+17], can readily be used in such engines. We therefore also base our model on vertex-based linear blend skinning. However, in comparison to SMPL and S-SCAPE, our model has a higher geometric resolution and provides fine-scale details such as fingers, eyes, and teeth. Furthermore, it is equipped with a more detailed skeleton and allows for hand and face animation. Recently, Romero et al. [RTB17] presented a fully articulated body and hand model based on SMPL. In [KIL+16], a fully automated approach was presented for reconstructing personalized anatomical models ready for physics-based animation. Their work focuses on the reconstruction of large and medium anatomical details, leaving out parts like hands, toes, and the face, which are important in our context.

In order to place the skeleton within the model shape, SMPL learns a joint regressor from a large amount of data. The resulting regressor represents joint positions as a linear function of the model's shape. Since our skeleton is more detailed than that of SMPL and the training data is not available, we cannot use their regressor. Instead, we follow Feng et al. [FCS15] and represent the joint positions as generalized barycentric combinations of the template's vertex positions. This is also a linear function.

While the above methods work well for reconstructing the *geometry* of human bodies, they mostly neglect the texture reconstruction. This, however, is crucial for VR applications. Unlike the above methods, we reconstruct a high-quality texture from the reconstructed geometry and the individual camera images of our scanner.

Face Reconstruction

Related work on face reconstruction has already been discussed in Section 2.1, with a focus on deformable registration of *static* facial models to 3D-scans or RGB-D as input data. Since we aim not only at high-quality geometry and texture but also at short acquisition time, we employ multi-view face scanning based on our method presented in Chapter 2. In particular, we take the deformed template model, which was previously fit to the full-body scan, and refine its face region by fitting it to the point cloud resulting from the face scan.

In the following, we focus on *dynamic* face models since we are interested in fully animatable virtual humans. Dynamic facial animations are crucial for VR characters, e.g., for speech animation or emotional facial expressions. With the industry standard being linear blendshape models [LAR$^+$14], the character generation pipeline also has to construct the required set of FACS blendshapes [EF78]. For high-quality production without time constraints, these blendshapes are often created manually by artists or reconstructed by scanning real actors performing these expressions [ARL$^+$09]. A faster process is enabled by example-based facial rigging [LWP10], which generates personalized facial blendshapes from a small set of example expressions. Since we want to keep acquisition and processing time low, we scan the actor in neutral expression only and generate the full set of FACS blendshapes by adjusting the template's generic blendshapes to the deformed model using deformation transfer [SP04]. If acquisition and processing time is not that critical, reconstructing a few additional expressions and using example-based facial rigging would be a good compromise.

Avatar Reconstruction

While there are many approaches for reconstructing human body shapes *or* human faces *or* human hands, only few previous works aim at reconstructing a complete virtual human featuring animatable body, face, *and* hands.

Malleson et al. [MKK$^+$17] present a single snapshot system for rapid acquisition of animatable, full-body avatars based on a stereo RGB camera pair as well as a single RGB-D sensor. While the total processing time is in the order of seconds, the body is a stylized astronaut character that fits the body dimensions only roughly. Albeit face shape and texture are also considered, the results are of rather low quality and lack facial details as only a low-dimensional face space is considered for fitting.

Feng et al. [FRS17] present a system for generating virtual characters by scanning a human subject. Their model is equipped with a full-body skeleton rig and is capable of facial expressions and finger movements. In direct comparison, their reconstruction process takes about twice as long as ours and requires more manual effort. Blendshapes are generated by explicitly scanning the actor in five different expressions, restricting the model to a few, but nicely personalized blendshapes. In contrast, our method reconstructs the full set of FACS blendshapes from a single face scan in neutral pose. It is thus compatible with standard animation packages. On the downside, our blendshapes are more generic and not as actor-specific. The biggest drawback of Feng's method is that by construction each model has

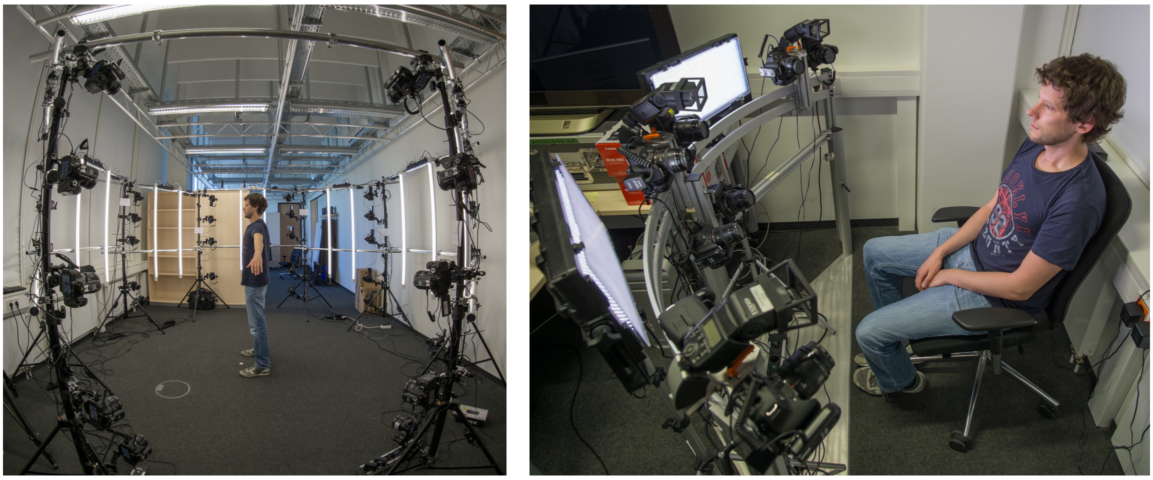


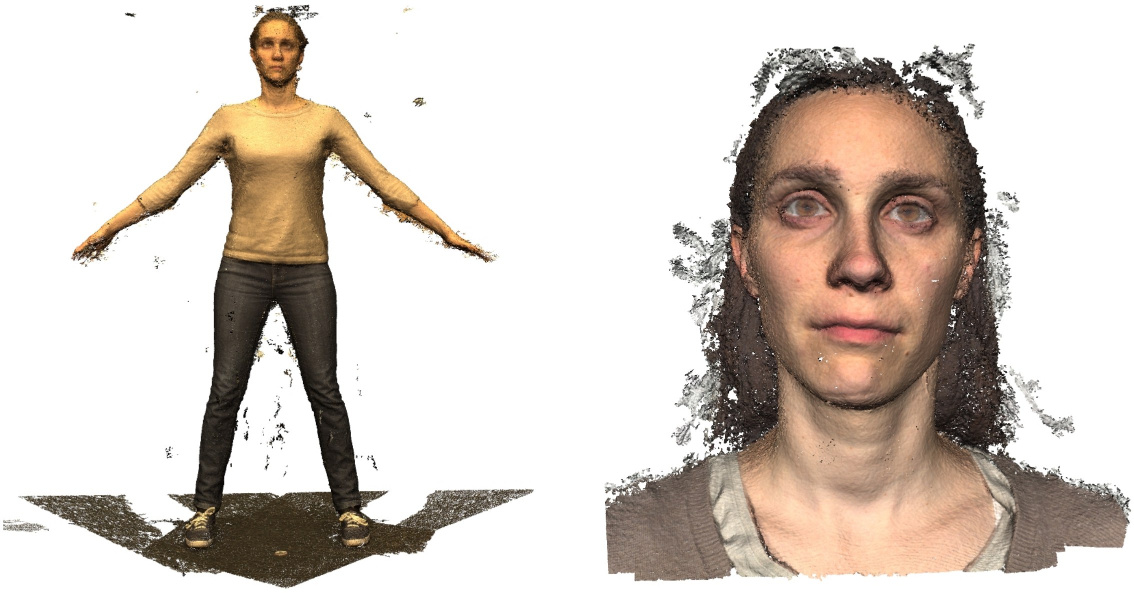
Figure 3.1: Our custom-built full-body scanner (left) and face scanner (right) are both based on multi-view stereo and consist of 40 and 8 DSLR cameras, respectively.

a different tessellation. This prevents statistical analysis and detail/cloth transfer between models. In contrast, all our models share the tessellation of the initial template mesh.

3.2 Input Data

Our 3D-scanning setup is based on multi-view stereo reconstruction using a single-shot multi-camera rig, since this minimizes acquisition time to a fraction of a second while at the same time providing high-quality results in terms of geometry and texture. We built a full-body scanner and a separate face scanner consisting of 40 and 8 mid-range consumer DSLR cameras of type Canon 700D and featuring 18 Megapixels, respectively, as shown in Figure 3.1. As already mentioned, we extend our face scanner from Chapter 2 to 8 cameras for a better coverage of the face region. As a good trade-off between low image distortion and a large field of view, there are $35\,\mathrm{mm}$ lenses attached to our full-body scanner and $50\,\mathrm{mm}$ lenses attached to our face scanner. However, due to space constraints and a limited field of view, body scans must still be performed in A-pose instead of T-pose (Figure 3.2(a)). The cameras of each scanner are triggered simultaneously, and the resulting pictures are subsequently downloaded from the cameras. We decided to use a separate face scanner instead of to augmenting the full-body scanner with more cameras aiming at the face region. Otherwise, the face cameras would have had to be manually adjusted to the individual subjects' heights.

The images of the 40 body cameras and of the 8 face cameras are automatically passed to the commercial software Agisoft PhotoScan [Agi17] which computes two high-resolution point clouds $\mathcal{P}_B$ of the body and $\mathcal{P}_F$ of the face, as well as camera calibration data (Fig-



(a) *(b)*

Figure 3.2: Computed point clouds from our full-body scanner, consisting of $3\,\mathrm{M}$ sample points (a), and from our face scanner, consisting of $1\,\mathrm{M}$ sample points (b).

ure 3.2). Face scans usually consist of about $1\,\mathrm{M}$ points, and body scans usually consist of about $3\,\mathrm{M}$ points. Since the template mesh has a limited resolution of $21\,\mathrm{k}$ vertices, we uniformly sub-sample the two point clouds to $40\,\mathrm{k}$ and $80\,\mathrm{k}$ points, respectively. This sampling resolution is chosen such that the resulting point density is still about twice as high as the vertex density of the template mesh. This speeds up the fitting process significantly without noticeably sacrificing geometric fidelity. When it is clear from the context, we omit the index B and F and just write $\mathcal{P} = (\mathbf{p}_1, \ldots, \mathbf{p}_n)$. Note that each point $\mathbf{p}_j$ is equipped with a normal vector $\mathbf{n}_j$ and RGB colors $\mathbf{c}_j$.

Since the bottoms of the feet are not visible for the full-body scanner, these regions cannot be captured properly. The missing points below the feet can easily result in an erroneous fitting of the feet regions. In contrast, the floor around the feet is usually scanned quite well. We exploit this by detecting the floor plane and removing its points from the point cloud $\mathcal{P}_B$. We then uniformly sample the detected floor plane underneath the feet region. This proved to be effective to capture the real extent of the feet and keep the feet on the floor during fitting without special treatment.

We picked a character from Autodesk Character Generator [Aut14] as a template model because these characters are already equipped with facial blendshapes, eyes, teeth, and a skeleton with corresponding skinning weights. However, any other template model with skeleton and blendshapes would work as well. The template mesh consists of $N \approx 21\,\mathrm{k}$

vertices with positions $\mathcal{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$. A bar denotes vertex positions in the undeformed state: $\bar{\mathcal{X}} = (\bar{\mathbf{x}}_1, \ldots, \bar{\mathbf{x}}_N)$.

In order to incorporate prior knowledge on human body shapes into the reconstruction process, we integrate shapes from multiple databases by fitting our template character to the databases' registered body models. We use 10 scans of different subjects standing in A-pose from the FAUST database [BRLB14], and we include 111 scans from [HSS$^+$09]. Moreover, we add 82 synthetic models with different shapes from Autodesk Character Generator. After fitting our template model to these models, they all share the same tessellation, allowing us to compute a ten-dimensional PCA subspace based on vertex positions of posture-normalized characters in T-pose. This PCA will act as initialization and regularization for the body fitting described in the next section.

3.3 Body Reconstruction

After computing and post-processing the point cloud $\mathcal{P}_B$ of the full-body scan, the next step is to align and fit the template model to this point cloud. As in most template fitting approaches, this fit is robustly performed in several steps: In the initialization phase, we optimize the alignment (scaling, rotation, translation), pose (skeleton joint angles), and PCA parameters for the ten-dimensional shape space. Afterwards, a fine-scale deformation fits the model to the data. Once the geometry fit is done, we have to compute texture, correct joint positions, and pose-normalize the model.

3.3.1 Initialization

Initially, the point cloud $\mathcal{P}_B$ and the template are in different coordinate systems and have different poses because the template is in T-pose and the body scan is performed in A-pose. To bootstrap the template fitting procedure, we manually select nine landmarks $\mathcal{L}$ on the point cloud $\mathcal{P}_B$. Their corresponding vertices on the template model have been preselected (Figure 3.3). The landmarks have been chosen to ensure that important body parts like head, hands, and feet are fitted properly.

In the first step, we optimize the alignment and pose of the template model in order to minimize the squared distances between these nine landmarks on the template model and their corresponding landmarks in the point cloud. To this end, we alternatingly (a) compute the optimal scaling, rotation, and translation [Hor87] as explained in Section 2.2 and further (b) optimize the joint angles using inverse kinematics based on linear blend skinning [Bus04]. This procedure is iterated until the relative change of the squared distances falls below 0.05. This initialization process is depicted in Figure 3.4, (a) and (b).

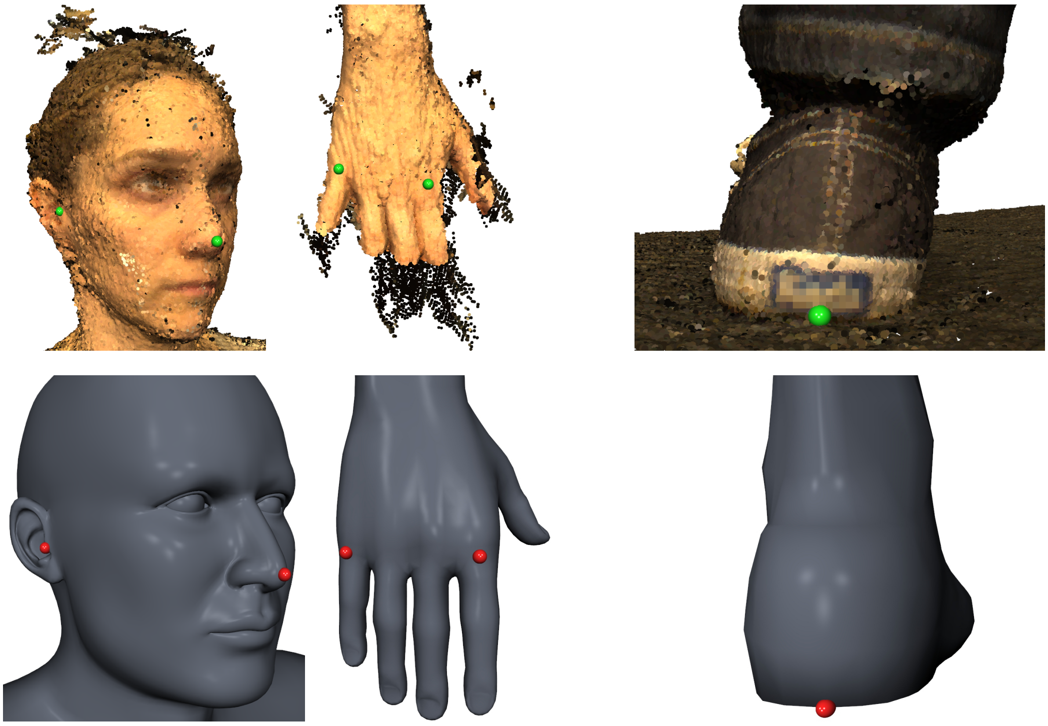


Figure 3.3: Nine landmarks are selected manually on the full-body point cloud (top). The landmarks' counterpart vertices on the template model have been pre-selected (bottom).

The landmark-based fit gives us a good estimate of scaling, rotation, translation, and joint angles. We further optimize these variables by additionally taking closest point correspondences into account. These are computed by finding, for each point in $\mathcal{P}_B$, its closest point on the template. We prefer these scan-to-model correspondences over model-to-scan correspondences, since they were shown to yield more accurate fits in Section 2.2.1. As usually done in ICP-based registrations, we prune unreliable correspondences based on distances and normal deviations (typically $10\,\mathrm{cm}$ and $50°$). We employ the same alternating optimization as before to optimize alignment and pose, this time minimizing squared distances of landmarks and of correspondences (Figure 3.4(c)).

After convergence of the alignment and pose optimization, we add the PCA weights to the active variables and thereby optimize the geometric shape in the ten-dimensional PCA space. We do this by, again, minimizing squared distances between landmarks and correspondences (see Sections 2.2 and 4.2.4). As our PCA model is pose-normalized in T-pose, the PCA-fitting is performed in T-pose. To this end, we first compute closest point correspondences to the template model in the (current) optimized pose. Since each corresponding point from the point cloud has a direct correlation with its nearest triangle from the template, we transform those to T-pose, just as the template model itself, using linear blend skinning. As a result, the correspondences as well as the template model are in T-

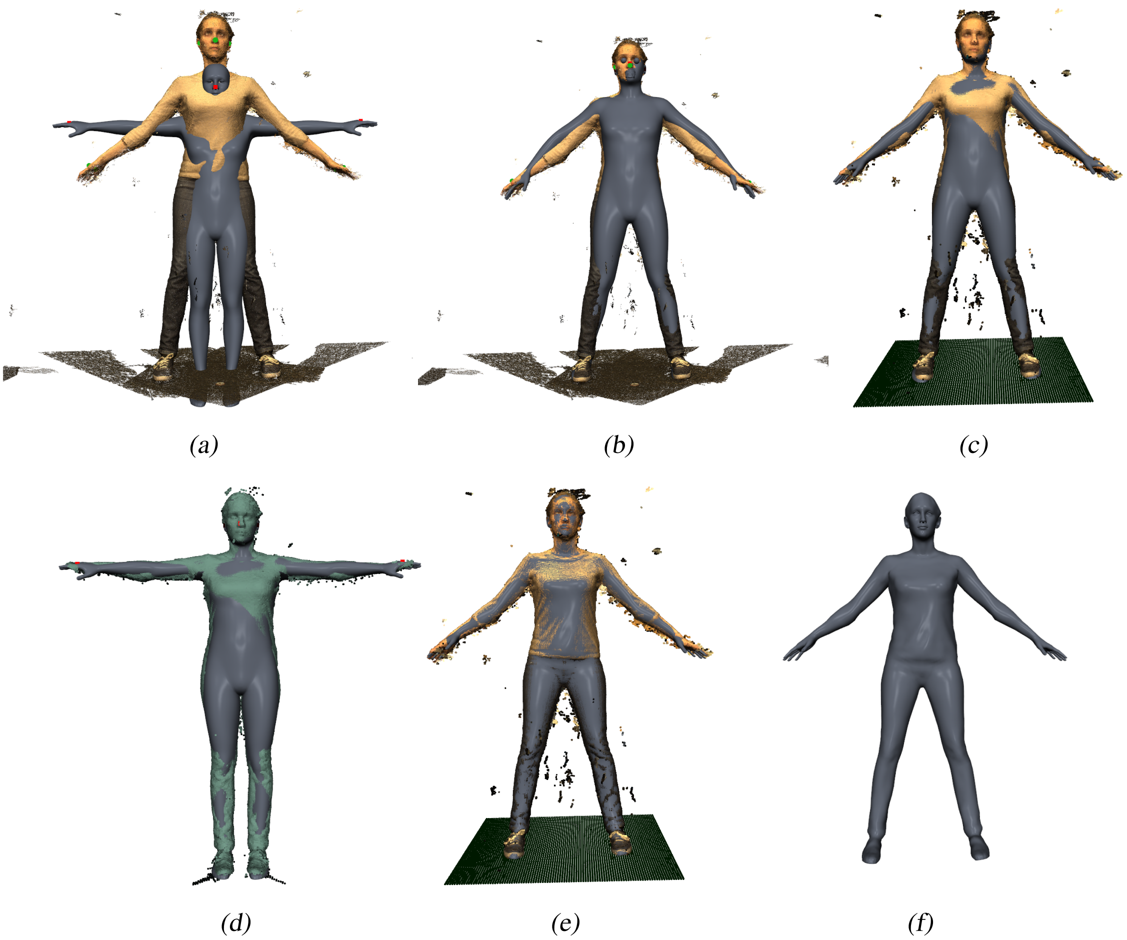


Figure 3.4: We first optimize alignment (scaling, rotation, translation: (a)) and pose (joint angles: (b)) based on nine manually selected landmarks. This fit is refined by incorporating closest point correspondences (c) and by alternating with PCA regularization in T-pose (d). After this initialization, we perform a fine-scale deformation to the point cloud (e–f).

pose so that we can perform the PCA-fitting (Figure 3.4(d)). The shape-change caused by adjusting PCA parameters requires adjusting the skeleton's joint positions. To this end, we represent joint positions by mean value coordinates [JSW05] with respect to the vertex positions of the template mesh. Joint positions are then a linear function of vertex positions and hence also a linear function of PCA parameters. After one PCA-fitting step, we reapply the (current) optimized pose to the template model and continue with optimizing alignment and pose parameters in an alternating fashion. Two iterations of this procedure are usually sufficient for a good initial fit of shape and pose.

3.3.2 Deformable Registration

With the point cloud and template model in good initial alignment, we perform a fine-scale non-rigid registration following the approach from Chapter 2. To this end, we minimize the energy

$$E_{\text{body}}(\mathcal{X}) \;=\; \lambda_{\text{lm}} E_{\text{lm}}(\mathcal{X}) \;+\; \lambda_{\text{fit}} E_{\text{fit}}(\mathcal{X}) \;+\; \lambda_{\text{reg}} E_{\text{reg}}\big(\mathcal{X}, \bar{\mathcal{X}}\big) \;. \tag{3.1}$$

The three energy terms are explained below.

The *landmark term* E_{lm} penalizes the squared distance between the nine manually selected landmarks $\mathbf{p}_l$, $l \in \mathcal{L}$, in the point cloud and their counterpart vertices $\mathbf{x}_l$ on the template model

$$E_{\text{lm}}(\mathcal{X}) \;=\; \frac{1}{|\mathcal{L}|} \sum_{l \in \mathcal{L}} \left\| \mathbf{x}_l - \mathbf{p}_l \right\|^2 \;.$$

The *fitting term* E_{fit} penalizes the squared distance between corresponding points $\mathbf{x}_c$ and $\mathbf{p}_c$

$$E_{\text{fit}}(\mathcal{X}) \;=\; \frac{1}{\sum_{c \in \mathcal{C}} w_c} \sum_{c \in \mathcal{C}} w_c \left\| \mathbf{x}_c - \mathbf{p}_c \right\|^2 \;, \tag{3.2}$$

where $\mathcal{C}$ is the set of closest point correspondences and $w_c \in [0, 1]$ are per-correspondence weights, as discussed below. The closest points $\mathbf{x}_c$ are expressed as barycentric combinations of the template vertices $\mathbf{x}_i$.

The *regularization term* E_{reg} penalizes the geometric distortion from the undeformed model $\bar{\mathcal{X}}$ (the result of the initialization phase of Section 3.3.1) to the deformed state $\mathcal{X}$, measured by the squared deviation of the per-edge Laplacians

$$E_{\text{reg}}\big(\mathcal{X}, \bar{\mathcal{X}}\big) \;=\; \frac{1}{\sum_e A_e} \sum_{e \in \mathcal{E}} A_e \left\| \Delta^e \mathbf{x}(e) - \mathbf{R}_e \Delta^e \bar{\mathbf{x}}(e) \right\|^2 \;. \tag{3.3}$$

Analog to Section 2.3, A_e is the area associated to edge e, and $\mathbf{R}_e$ are per-edge rotations to best-fit deformed and undeformed Laplacians. We prefer the edge-based Laplacian over the standard vertex-based Laplacian since, in our experiments, it converges slightly faster to very similar results. Note that we do not employ the anisotropic bending model of Section 2.3 since the template's body region is too coarse to benefit from the (computationally more expensive) anisotropic wrinkle reconstruction.

The three coefficients λ_{lm}, λ_{fit}, and λ_{reg} are used to guide the iterative fitting procedure. The surface stiffness is controlled by λ_{reg}. In the beginning, only the manually specified (hence quite reliable) landmarks are taken into account using $\lambda_{\text{reg}} = 1$, $\lambda_{\text{lm}} = 1$, and $\lambda_{\text{fit}} = 0$. We then gradually decrease λ_{reg} after each iteration until $\lambda_{\text{reg}} = 10^{-5}$. After these iterations, the template is sufficiently well aligned to yield reliable closest point correspondences. We therefore continue with $\lambda_{\text{reg}} = 10^{-5}$ and $\lambda_{\text{lm}} = 1$ and additionally set

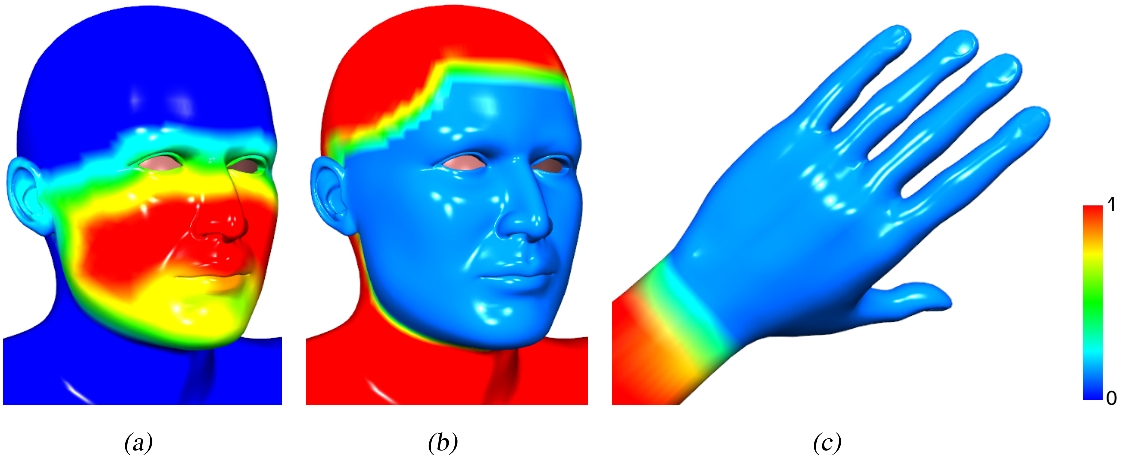


Figure 3.5: Per-correspondence weights in the fitting term allow fitting of only the face region (a) or mostly the body (b) and down-weighting of the (typically poorly scanned) hands (c).

$\lambda_{\text{fit}} = 1$ to also consider E_{fit}. Then, both λ_{lm} and λ_{reg} are gradually decreased ($\lambda_{\text{lm}} \leftarrow \frac{\lambda_{\text{lm}}}{2}$ and $\lambda_{\text{reg}} \leftarrow \frac{\lambda_{\text{reg}}}{10}$) until $\lambda_{\text{reg}} = 10^{-9}$.

During the fitting procedure, we weight down parts of the template using the per-correspondence weights w_c in E_{fit} in order to prevent unreliably scanned regions from being fitted too strongly (Figure 3.5). We weight down the hands, since they are usually not scanned well, and the face region to allow us to add more detail when combining with the face scan in Section 3.4.

The nonlinear objective function (3.1) is minimized by solving for vertex positions $\mathbf{x}_i$ and per-edge rotations $\mathbf{R}_e$ using alternating optimization (a.k.a. block-coordinate descent) [BTP14, AZB15], as presented in Section 2.2.2. Figure 3.4(f) shows the final result of the body fitting procedure.

3.3.3 Texture Reconstruction

After the coarse-scale initialization and the fine-scale non-rigid registration, the template has been accurately aligned and deformed to fit the point cloud of the body scan. We pass the deformed template model to Agisoft PhotoScan, which makes use of the existing texture layout from Autodesk Character Generator and computes a high-quality $4\,\text{k} \times 4\,\text{k}$ texture based on the 40 camera images and their calibration data (Figure 3.9(a)).

Since the camera images typically do not provide meaningful texture information for eyes and teeth, we use a pre-selected image mask to preserve the corresponding texture regions, i.e., to use eye and teeth texture from the generic template texture.

Due to occlusions and delicate geometric structures, scanning artifacts can easily occur for the fingers. This can result in an inaccurate template fit and then to misaligned textures for the fingers. We reconstruct a plausible hand texture by searching for the best-matching hand texture in Autodesk Character Generator and using this hand texture instead. We identify the best-matching texture based on the Euclidean distance between RGB values of the backs of both hands, the Autodesk texture, and the one of the scanned subject (the latter is fitted reliably due to the manually selected landmark on the hands). Here, it turned out to be beneficial to distinguish between male and female hand textures. The obtained hand texture area is then seamlessly merged into the reconstructed full-body texture using Poisson image editing [PGB03].

In a nutshell, Poisson image editing works as follows. Let S be one channel of the source image (Figure 3.9(b)) and T one channel of the target image (Figure 3.9(a)). We segment a common mask for S and T to outside, boundary $\partial\Omega$, and inside pixels Ω (Figure 3.9(c)). Then, the problem is finding (unknown) inner pixels I over Ω in T that we want to seamlessly combine with $\partial\Omega$ in T and that should resemble the appearance of Ω in S. Given a guidance field G, the mathematical problem boils down to solving a Poisson equation with Dirichlet boundary conditions

$$\Delta I = \operatorname{div} G \qquad \text{over } \Omega\,, \text{ with } I|_{\partial\Omega} = T|_{\partial\Omega}\,,$$

where

$$\Delta I(x,y) \approx -4I(x,y) + I(x+1,y) + I(x-1,y) + I(x,y+1) + I(x,y-1)\,.$$

Since we are looking for a symmetric positive definite matrix, we instead solve

$$-\Delta I = -\operatorname{div} G\,.$$

With $G = \nabla S$ being the gradient field taken from the source image, it remains to solve

$$-\Delta I = -\Delta S \qquad \text{over } \Omega\,, \text{ with } I|_{\partial\Omega} = T|_{\partial\Omega}\,,$$

for new pixel colors for each channel separately.

Finally, the texture area below the armpits is typically corrupt as the armpits are not sufficiently visible from our cameras. We smoothly fill these texture regions by harmonic color interpolation which we compute by solving a sparse linear Laplace system with suitable Dirichlet color boundary constraints, similar in concept to Poisson image editing [PGB03].

3.3.4 Pose Normalization

Due to the non-rigid shape deformation, the template's joints are not at their correct positions anymore. We again adjust the joint positions based on the pre-computed mean value coordinates, this time representing the joint positions as a linear function of vertex positions (instead of PCA parameters). Employing mean value coordinates for this mapping ensures that joints are placed at meaningful positions even for strong shape deformations.

After mapping the skeleton to the deformed template (in scan pose), we use it to undo the pose fitting, i.e., to put the model into T-pose, as it is usually required by animation tools. This is an important step, particularly for character animation via motion capturing since these systems usually rely on a standardized T-pose as initialization. To make sure that both feet of the resulting character are standing exactly on the floor after pose-normalization, we first rigidly translate the model to put the (pre-selected) sole vertices onto the floor. Then, we non-rigidly deform them onto the floor plane while allowing only the feet to slightly deform, regularized by the Laplacian energy (3.3).

3.4 Face Reconstruction

After fitting the template model to the full-body scan $\mathcal{P}_B$, we next improve the geometry and texture of its facial region by fitting it to the face scan $\mathcal{P}_F$ and exploiting its eight close-up camera images. We closely follow the face reconstruction approach of Chapter 2 but adjust it to the combined body-and-face reconstruction setup and extend it by blendshape reconstruction.

3.4.1 Initialization

Since the face scan and the body scan are not aligned to each other, the template model is not aligned to the face scan either.

Following our approach from Section 2.2, we automatically detect facial landmarks in the input camera images using [AZCP13]. The detected facial landmarks are then mapped to 3D points in $\mathcal{P}_F$ using the camera calibration data. Thereafter, in order to align the template (Figure 3.6(a)) to the face scan, we find an optimal similarity transformation (scaling, rotation, and translation) [Hor87] by minimizing squared distances between the detected 3D facial landmarks and their (pre-selected) counterparts on the template model. Afterwards, we refine scaling, rotation, and translation by iteratively finding closest point correspondences and computing the optimal similarity transformation in the usual ICP manner [BM92] (Figure 3.6(b)). Note that we transform the whole full-body template based on landmarks and correspondences of the face scan $\mathcal{P}_F$ only.

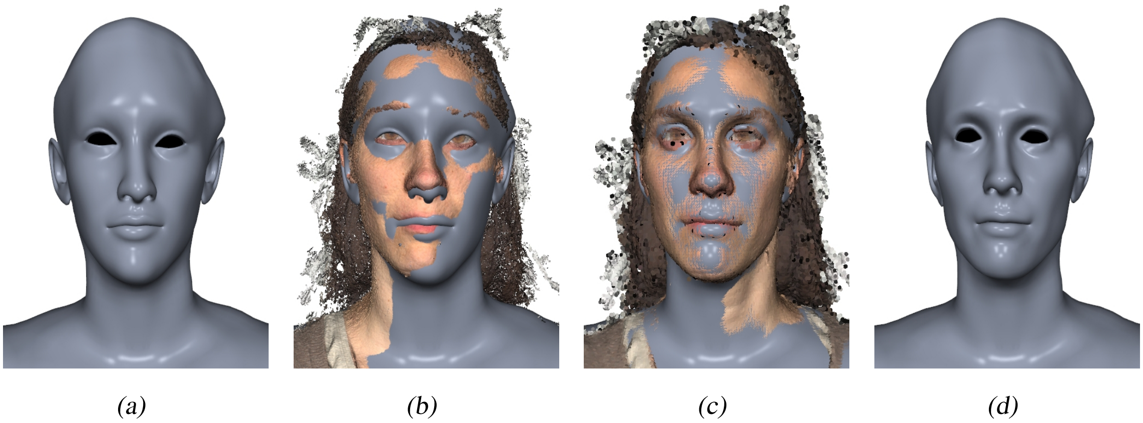


Figure 3.6: After our body reconstruction, we improve the geometry of the facial region by initially aligning the whole-body template (a) to the face scan (b) and performing a fine-scale non-rigid deformation. Final fit with (c) and without (d) face scan.

3.4.2 Deformable Registration

After the initialization, the template model and the facial point cloud $\mathcal{P}_F$ are sufficiently well aligned to start the fine-scale non-rigid deformation (Figure 3.6, (c) and (d)). To this end, we minimize the energy

$$E_{\text{face}}(\mathcal{X}) \;=\; \lambda_{\text{lm}} E_{\text{lm}}(\mathcal{X}) \;+\; \lambda_{\text{fit}} E_{\text{fit}}(\mathcal{X}) \;+\; \lambda_{\text{reg}} E_{\text{reg}}\big(\mathcal{X}, \bar{\mathcal{X}}\big) \;+\; \lambda_{\text{mouth}} E_{\text{mouth}}(\mathcal{X}) \;.$$

Here E_{fit} again represents closest point correspondences and is weighted by $\lambda_{\text{fit}} = 1$. We again employ per-correspondence weighting in the fitting term E_{fit}, such that only the face and ear vertices are dragged toward the face scan (Figure 3.5(a)).

E_{lm} represents a landmark term, weighted by $\lambda_{\text{lm}} = 1$, and includes three types of landmarks: Besides the automatically detected facial features, we manually pick two landmarks on each ear to more precisely fit the ears. Furthermore, we manually pick seven contour points for each eye in the frontal face picture and compute landmarks for eyelid reconstruction (see Section 2.4).

The regularization term E_{reg} is the same as for the body fitting. It is initially weighted by $\lambda_{\text{reg}} = 1$ and is gradually decreased to $\lambda_{\text{reg}} = 10^{-9}$ during the iterative fitting procedure.

We observed that it is not guaranteed that the mouth stays closed during fitting. We therefore add an energy term preventing contour points on the upper/lower lip from diverging

$$E_{\text{mouth}}(\mathcal{X}) \;=\; \frac{1}{11} \sum_{i=1}^{11} \|\mathbf{x}_i^{(u)} - \mathbf{x}_i^{(l)}\|^2 \,,$$

where $\left\{\mathbf{x}_i^{(u)}, \mathbf{x}_i^{(l)}\right\}$ are 11 pairs from upper and lower lip, respectively. The point pairs are pre-selected on the template mesh. This energy term is weighted by $\lambda_{\text{mouth}} = 0.5$.

Note that, at this stage, we optimize the vertices of the head region only while keeping all other vertices fixed by removing them from the linear systems. Analogous to the body fitting step we solve the nonlinear optimization using alternating optimization for vertex positions and edge rotations. Analog to our body reconstruction, we do not employ the anisotropic bending model of Section 2.3 because the template's face region is too coarse to benefit from the anisotropic wrinkle reconstruction.

3.4.3 Facial Details and Blendshapes

Similarly to [IBP15], we adjust the template's teeth by optimizing for anisotropic scaling, rotation, and translation based on the deformation of the mouth region vertices $\mathcal{V}$ from the undeformed template to the deformed and fitted mesh. The computation for anisotropic scaling in x-direction turns Equation (2.1) into

$$s_x \;=\; \sum_{l\in\mathcal{V}} \hat{x}_{l,x}(\mathbf{R}\hat{\mathbf{p}}_l)_x \Bigg/ \sum_{l\in\mathcal{V}} \left|\hat{p}_{l,x}\right|^2 \;,$$

with s_y and s_z being computed similarly. Moreover, Equation (2.2) turns into

$$\mathbf{t} \;=\; \bar{\mathbf{x}} - \begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & s_z \end{pmatrix} \mathbf{R}\bar{\mathbf{p}} \,.$$

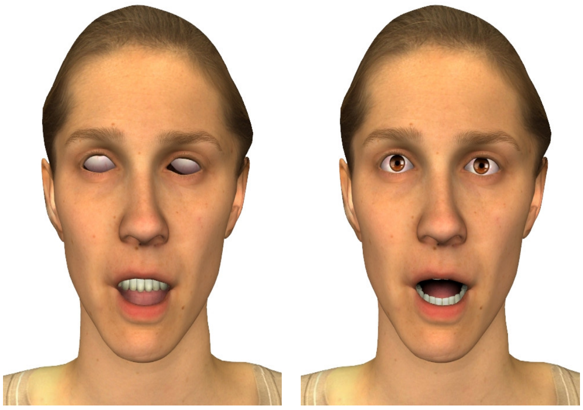
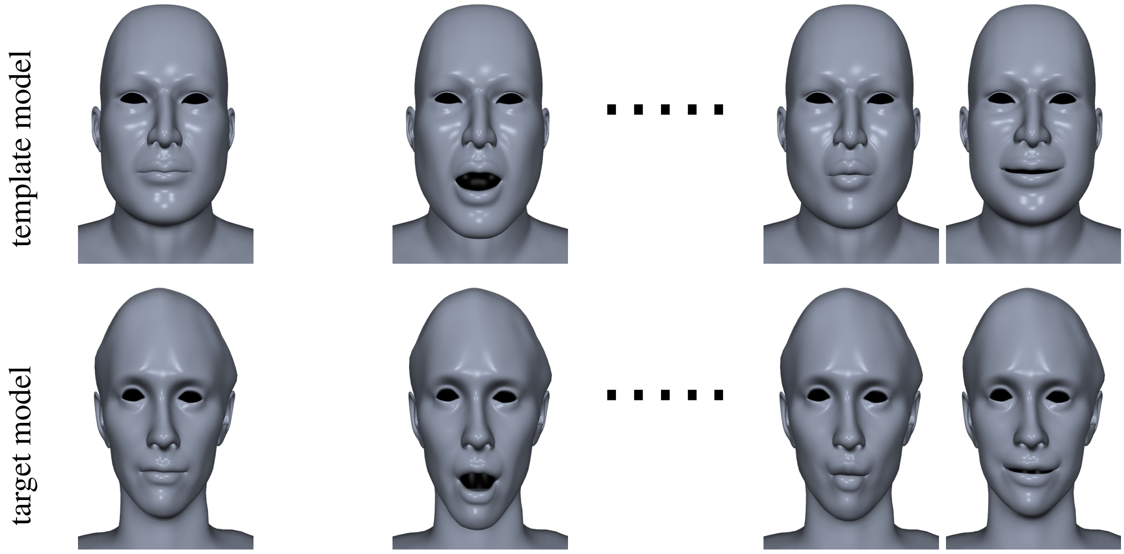


Figure 3.7: We adjust the template's teeth and eyes to fit the deformed mesh. The mouth is opened to show the teeth.

We also transform the eyes by optimizing for isotropic scaling, rotation, and translation for each eye individually. Again, this transformation is based on the deformation of the individual eye region from the undeformed to the deformed mesh (Figure 3.7).

Face animation requires a suitable set of blendshapes, which represent the face in different expressions, typically consisting of the FACS blendshapes [EF78] and of visemes for speech animation. Since we only scan the actor in a neutral facial expression, we have to "invent" a proper set of blendshapes. Since facial expressions are similar across different individuals, we transfer all blendshapes from our generic template model to the fitted model using deformation transfer [SP04], similarly to [WBLP11]. This transfers the deformation from the template model (generic neutral $\mapsto$ generic expression) to the target model (Figure 3.8).



(a) Neutral

(b) Expressions

Figure 3.8: For our fitted model (a, bottom), we compute a set of blendshapes (b, bottom) by deformation transfer. To this end, we transfer the deformation from the template model (top, a ↦ b) to the target model (a, bottom) for all expressions.

Deformation transfer computes the deformation between an undeformed source mesh $\bar{\mathcal{S}}$ and its deformed state $\mathcal{S}$ and transfers it to another target mesh $\bar{\mathcal{T}}$ to get a deformed version $\mathcal{T}$.

To this end, an affine source deformation $\mathbf{S}_j \in \mathbb{R}^{3\times3}$ (called *deformation gradient*) is computed for each triangle $\bar{t}_j \in \bar{\mathcal{S}}$ that maps triangle edges to $t_j \in \mathcal{S}$ and consists of the rotational, stretch, and shear parts of the deformation.

Given a target mesh $\bar{\mathcal{T}}$, the goal is to find new vertex positions for $\mathcal{T}$ such that the triangles' deformation gradients $\mathbf{T}_j$ for $\bar{\mathcal{T}}$ match those of the source deformations $\mathbf{S}_j$ in a least-squares sense. Mathematically, this boils down to solving a linear least-squares problem for $\mathcal{T}$'s unknown vertex positions [SP04].

Note that our blendshapes are rather generic since they transfer the template's expression to the scanned person. Feng et al. [FRS17] instead scan additional expressions and use those as (highly personalized) blendshapes, but they do not generate additional ones. A good compromise would be to add a small number of scanned example expressions to the deformation transfer process as is done by example-based facial rigging [LWP10]. This, however, increases the acquisition time.

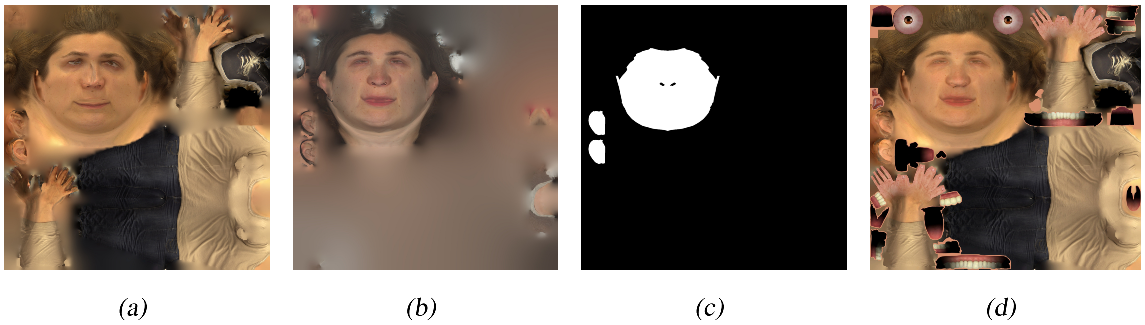


Figure 3.9: Textures computed from the camera images of the body scan (a) and the face scan (b). Since the face region is more accurately represented in the latter, it is extracted using a pre-computed image mask (c) and seamlessly copied into the body texture through Poisson image editing (d).

3.4.4 Texture Reconstruction

Analogous to the body fitting step, we generate a $4\,\mathrm{k} \times 4\,\mathrm{k}$ texture from the eight camera images of the face scanning session using Agisoft PhotoScan. This yields an accurate high-quality texture, but only for the face region. We therefore extract the face region using a pre-selected image mask and then seamlessly copy it into the full-body texture using Poisson image editing [PGB03] (see Figure 3.9 and Section 3.3.3). As mentioned before, we keep the texture for eyes and teeth from the original texture. The luminances of these regions are adjusted so that their mean luminances coincide with the mean luminance of the face. This adapts the texture of teeth and eyes to the lighting conditions of the scan.

3.5 Results

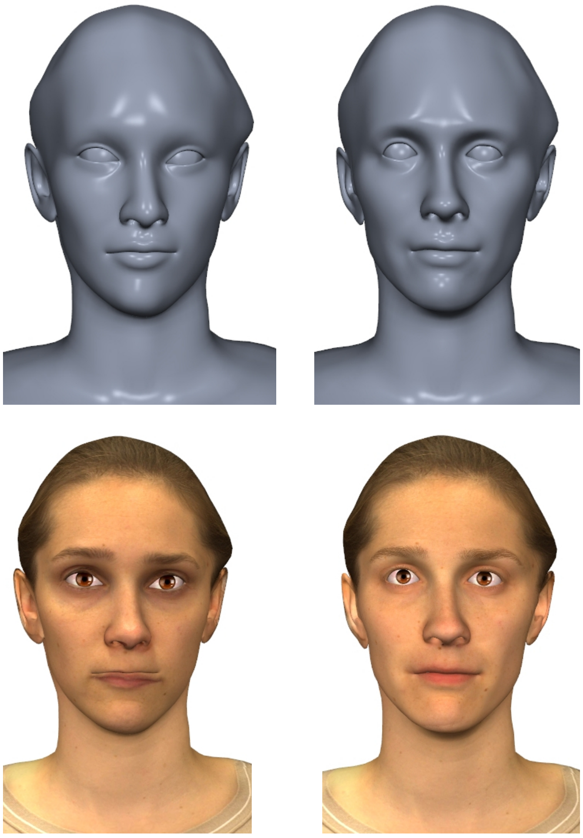


Figure 3.10: Illustration of the face region reconstructed from the full-body scan only (left) compared to the face region reconstructed by additionally incorporating the dedicated face scanning (right). For a fair comparison, we did not down-weight the face region when fitting to the full-body scan only (cf. Figure 3.5(b)).

We tested our virtual human generation pipeline on a large set of subjects, and our approach reliably produced convincing results for all of them. A representative subset can be seen in Figure 3.11 and in the accompanying video.

The use of multi-view stereo reconstruction allows us to reconstruct both accurate geometries as well as high-quality textures. As can be seen in Figure 3.10, additionally incorporating our dedicated face scanner significantly improves the visual quality of the face region, since it was scanned at a higher resolution. A comparison of a captured image from the body scanning session with the personalized virtual human is depicted in Figure 3.12.

Our reconstructed characters can readily be animated in any standard graphics or VR engine since they feature a standard skeleton for full-body and hand animation as well as a standard set of blendshapes for face animation. The accompanying video demonstrates that our characters can efficiently be animated and rendered in a real-time scenario. Figure 3.13 and the accompanying video show one of our scanned characters used as a conversational virtual agent, where face and body animation are crucial to enable the agent to talk, perform gestures, and show facial expressions.

Our method also has some limitations: Texture artifacts may still occur in regions that are not visible from more than one camera, as is the case for all photogrammetry approaches. The most critical areas are the armpits and the hands, but the crotch and the inner parts of the arms can also be problematic. These issues can be overcome by using more cameras. More cameras would lead to a better coverage for texture data at the expense of longer computation times. Furthermore, we do not remove the scene lighting during scanning from the albedo textures, as done, e.g., in [BRLB14].

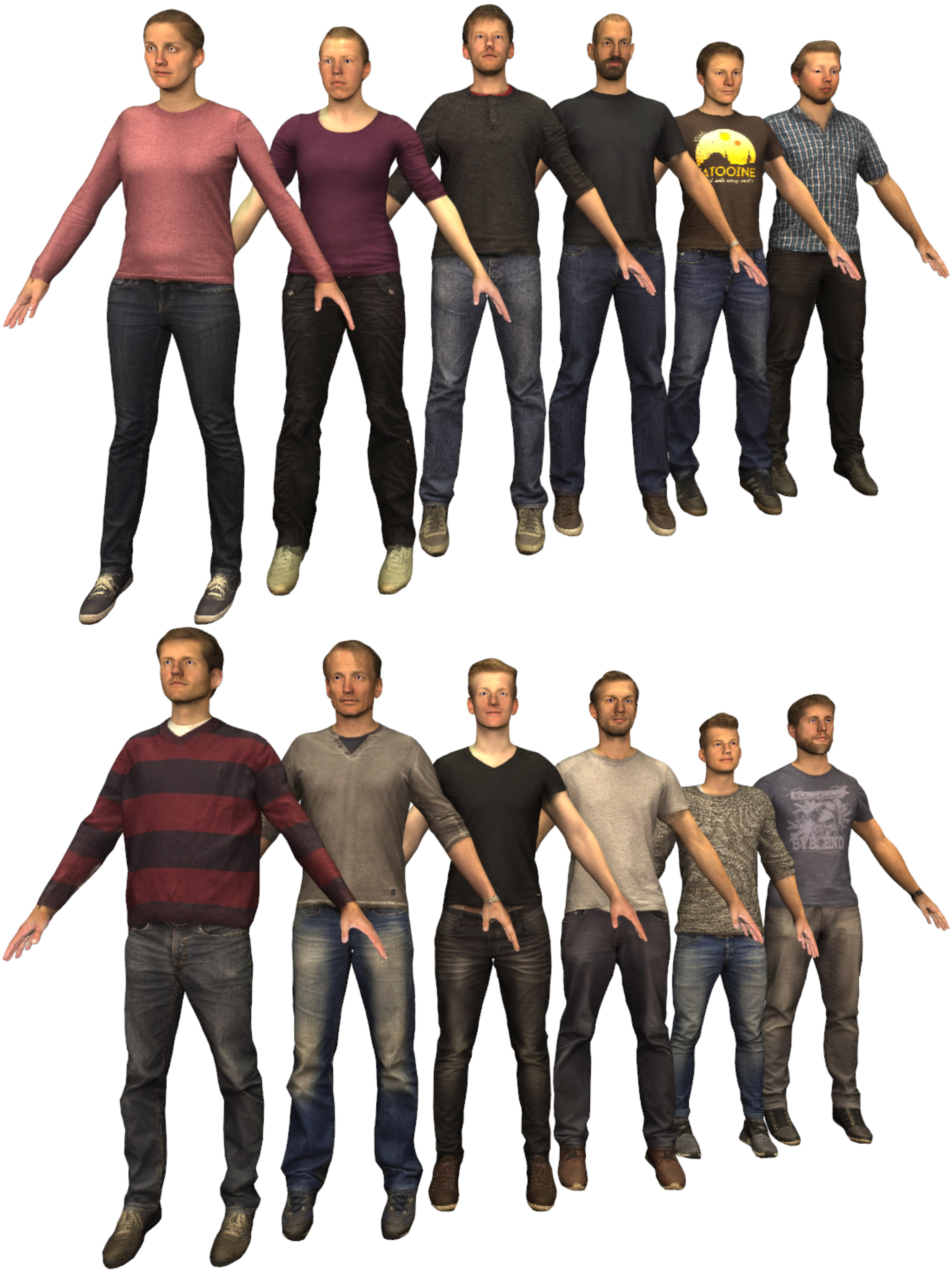


Figure 3.11: We generate realistic virtual humans from real persons through multi-view stereo scanning. The resulting characters are ready to be animated through a skeletal rig and facial blendshapes and are compatible with standard graphics and VR engines. The whole reconstruction process requires only minimal user input and takes less than ten minutes.

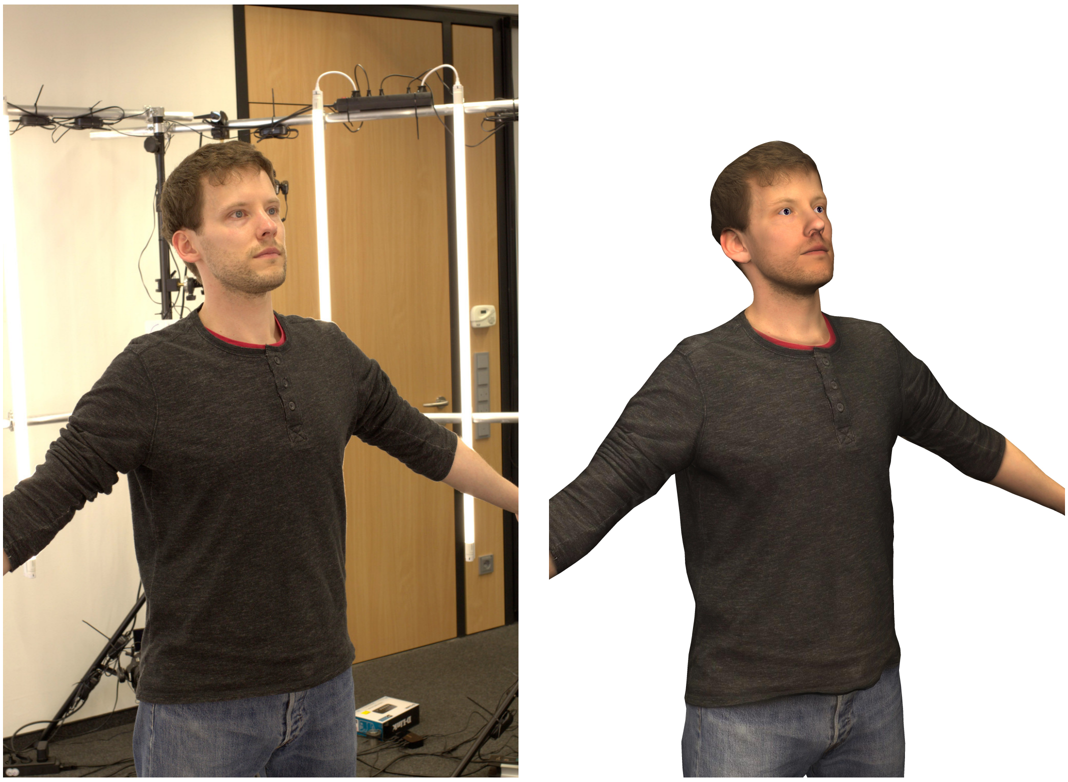


Figure 3.12: Comparison of a photo from the body scanning session (left) with a rendering from the generated virtual human (right).

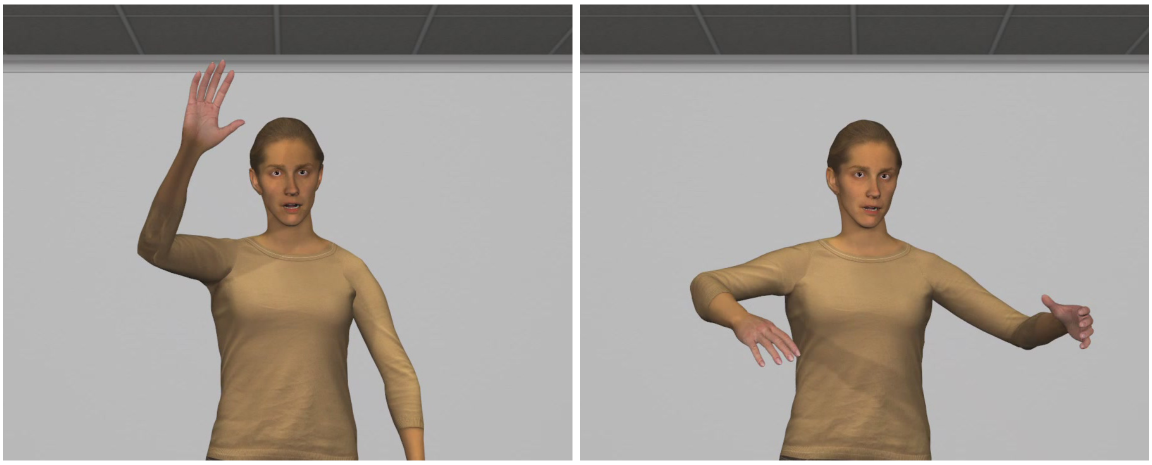


Figure 3.13: Our virtual humans can be directly used as expressive conversational agents; They are able to gesture, talk, and to show facial expressions and emotions.

3.5.1 Performance

On average, the processing of a single character takes about ten minutes from scan to a complete animatable avatar. See Table 3.1 for detailed information about the computation times needed for our sub-processes. The times were measured on a desktop PC with Intel Xeon CPU (6×3.5 GHz) and a Nvidia GTX 980 GPU.

The computationally most expensive part of our template fitting procedure is the computation of the closest point correspondences in each fitting iteration. While this can be accelerated by using a kD-tree or a similar space partitioning technique, we found that a simple linear search implemented on the GPU provides a much higher speed-up for the model complexities in our application. In comparison to a CPU-based kD-tree, our straightforward GPU implementation of a brute-force search is about 12 times faster. A GPU-based implementation of a spatial hierarchy would probably lead to an even greater speed-up, but would also require a considerably more complex implementation.

Process	Approx. time
Face scanning	1/10 s
Transfer images from face scanner	15 s
Full-body scanning	1/10 s
Transfer images from body scanner	80 s
Compute face point cloud $\mathcal{P}_F$	15 s
Compute body point cloud $\mathcal{P}_B$	75 s
Manual selection of landmarks	120 s
Automatic selection of facial features	60 s
Fit face geometry	20 s
Fit body geometry	35 s
Compute face texture	45 s
Compute and merge body texture	100 s
Compute facial blendshapes	5 s
Overall	$\sim$ 10 min

Table 3.1: Time needed for the sub-processes of our pipeline.

3.5.2 Clothing Transfer

Due to their construction by fitting the same generic template model to scanner data, all our models share the same tessellation and hence are in one-to-one correspondence. This allows transferring of arbitrary per-vertex or per-texel properties between models. We exploit this for transferring clothing.

Similar to [PMPHB17], we extract and store clothing as the difference (in geometry and texture) between a character wearing minimal clothing and the same character wearing a desired set of clothes. This clothing can then be transferred to another character, as shown in Figure 3.14.

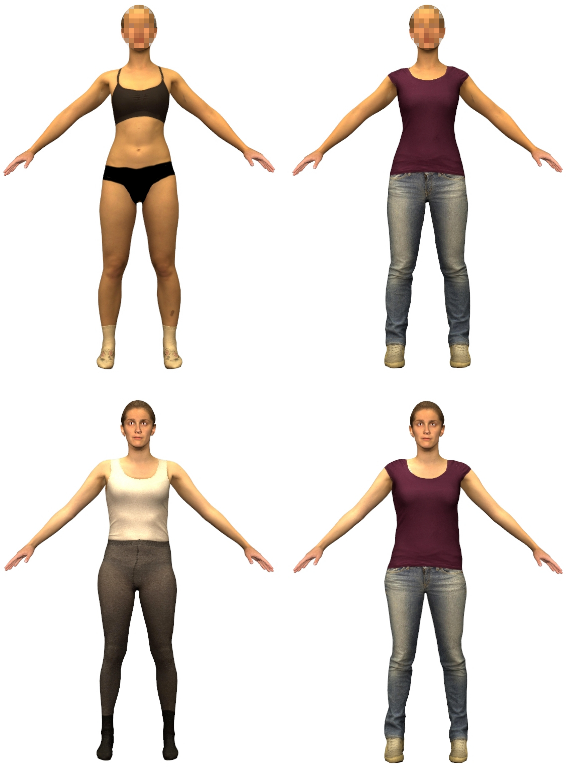


Figure 3.14: After reconstructing Subject A with both minimal clothing (top left) and clothing-of-interest (top right), we transfer this clothing to another Subject B with minimal clothing (bottom left) in order to get Subject B with the clothing from Subject A (bottom right). All models are visualized in A-pose, while the transfer is performed in T-pose.

Concretely, the regions of the texture from the source character that represent the clothing-of-interest are just copied to our target character's textures. In our pipeline, we segment clothing either manually or automatically by wearing a green suit. For transferring shape, the difference between a model with clothing-of-interest and the same model without clothing is computed and subsequently added to our character that is to be dressed. In our experiments, this difference was either based on simple subtraction of vertex coordinates [PMPHB17] or deformation transfer—comparable to our generation of facial blendshapes as presented in Section 3.4.3. While both approaches give visually very similar results, we computed the models in Figure 3.14 by deformation transfer.

Additionally, Figure 3.15 shows clothing transferred to reconstructed subjects of very different shapes. Moreover, Figure 3.16 demonstrates layered clothing transfer done by gradually adding clothing in an arbitrary sequence.

Note that, in contrast to [PMPHB17], we still represent our character models as single-layer meshes, i.e., we bake the clothing into the model's geometry and texture. While this leads to less realistic cloth animations, it preserves the computational efficiency and compatibility with standard graphics engines.

While being a comparatively simple application, the ability to control the clothing of virtual humans is crucial in experiments with scanned virtual characters as it allows factoring out perceptional effects caused by different clothing styles of the scanned subjects.

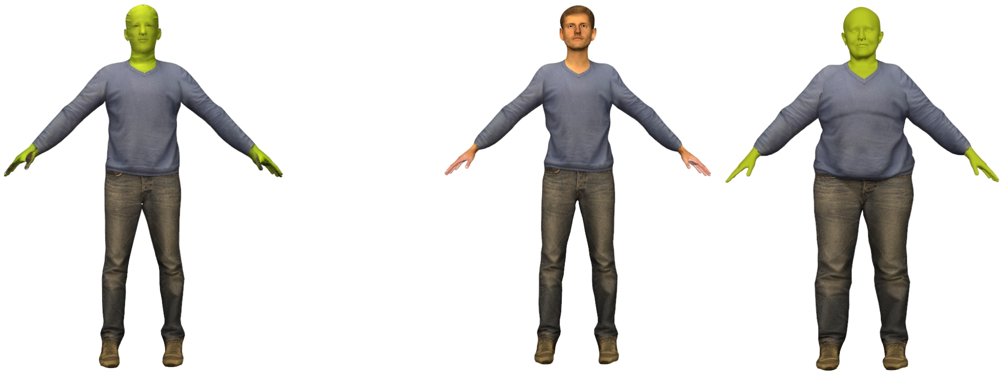
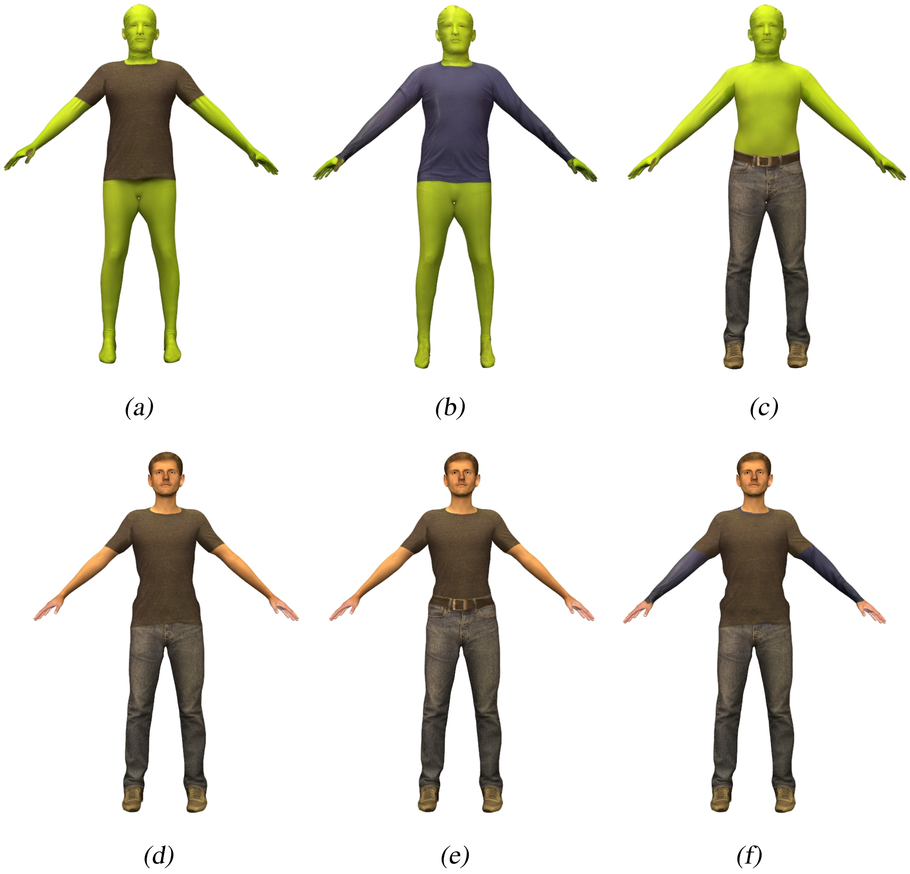


Figure 3.15: Examples of transferred clothing from a reconstructed subject wearing a green suit with the desired clothing (left) to two reconstructed subjects of different shapes (middle and right).



(a) *(b)* *(c)*

(d) *(e)* *(f)*

Figure 3.16: Example of layered clothing transfer. We scanned a subject wearing only a green suit, and we scanned the same subject wearing the green suit in combination with individual pieces of desired clothing (a–c). This clothing can be transferred to a target character in a layered fashion by gradually adding clothing in an arbitrary sequence (d–f).

In this chapter, we presented a fast and reliable pipeline to digitally clone real persons into realistic virtual humans. Our pipeline has already been used successfully in several works [WGR$^+$18, LRG$^+$17]. This finalizes the first part of this thesis. In the following chapter, we build on our accurate template fitting method and explore virtual humans in the medical context.

Part II

CRANIOFACIAL RECONSTRUCTION IN MEDICINE

4 Automatic Forensic Facial Reconstruction

Facial reconstruction is mainly used in two principal branches of science: forensic science and archaeology. Remains of a human skull act as input to reconstruct the most likely corresponding facial appearance of the dead individual to enable recognition.

Traditional methods for facial reconstruction in forensic science and archaeology rely on manually sculpturing a moldable substance onto the replica of the unknown skull using anatomic clues and reference data. Claes et al. [CVDG$^+$06] consider this a highly subjective procedure requiring a great deal of anatomical and artistic modeling expertise. The result is often limited to a single reconstruction because it is very time consuming.

Computer-based methods can provide consistent and objective results and also allow multiple reconstructions using different meta-information, such as age or weight because a reconstruction can be accomplished in a short time [CVDG$^+$06]. In her comprehensive review, Wilkinson [Wil10] reports that there is a lot of criticism on facial reconstruction techniques from scientists. Wilkinson concludes that achieving anatomical accuracy should be reproducible and reliable; however, both manual and computer-based techniques involve some degree of artistic interpretation.

Computer-aided facial reconstruction methods have been previously proposed in other publications [TBK$^+$05, TBL$^+$07, RME$^+$14, SZD$^+$16, SZM$^+$17]. Related work uses different techniques for the underlying registration as well as for the subsequent facial reconstruction [TBK$^+$05, TBL$^+$07, RME$^+$14, SZD$^+$16, SZM$^+$17]. Although not standardized, facial soft tissue thickness (FSTT) measurements play an important role both in facial approximation and craniofacial superimposition methods due to the quantitative information provided [SS08]. A wide variety of different techniques, such as needle probing, caliper or radiographic measurements, or ultrasonographic assessments, are used to determine the FSTT, leading to different results in the FSTT statistics. In addition, 3D imaging techniques such as Computed Tomography (CT) or Magnetic Resonance Imaging (MRI) are employed for this purpose. Driven by the generally lower radiation dose when compared to medical CT, Cone Beam Computed Tomography (CBCT) has recently also been used [HCH$^+$15]. In general, it is difficult to compare FSTT studies based on CT and CBCT scans. CT scans are taken in supine position, whereby CBCT scans can be taken in various positions (sitting, lying down, standing up). The positioning possibilities have different gravity effects on the FSTT. CBCT also has the inherent drawback that some landmarks cannot be found in the data sets because it is normally limited to the craniofacial region. Although not backed by numerical data, measurements on living individuals are generally preferred and advocated over measurements on cadavers [SS08]. In [SS08], Stephan and Simpson conclude that, regardless of the applied technique, the measurement error for FSTT assessment is rather high (relative error of around $10\,\%$). They argue that no method so far can be considered superior to any other. In addition, the authors stated that the small sample sizes for most of the studies also compromise the degree to which the results from such studies can be generalized.

Generally spoken, measurements based on a few distinct landmark points yield the inherent drawback of providing only a few discrete thickness values. Areas between these distinct measurement points need to be interpolated. A *dense* soft tissue map would yield important information for facial reconstruction.

In this chapter, we present a method for forensic facial reconstruction where a statistical head model is fitted to a dense soft tissue profile thereby providing an estimate of the visual appearance of the person to be identified. Our approach is divided into two parts: model generation and forensic facial reconstruction. Only the initial model generation (preprocessing or training phase) requires a few manual steps. Otherwise, unlike most previous methods [TBK+05, TBL+07, RME+14, SZD+16, SZM+17], our approach is fully automated, from the initial skull registration up to the final face reconstruction, and thus does not require any manual interaction. After discussing related work in the following section, we describe the generation of the three models required for our automated facial reconstruction approach: The parametric skull model, the FSTT statistics, and the parametric head model. Thereafter, the automated facial reconstruction process is presented, including the modeling of variants of plausible FSTT distributions for a given skull. In the final section of this chapter, we represent the FSTT-offset by a sphere-mesh (Section 4.4.1) as proposed in our follow-up work [ABG+18] and thus improve the quality of the resulting facial reconstructions.

My Contribution *The proposed method [GBA+19] for automatic forensic facial reconstruction was developed in close cooperation with Thomas Gietzen, Robert Brylka, and Ulrich Schwanecke from RheinMain University of Applied Sciences in Wiesbaden. It was further developed in cooperation with Katja zum Hebel, Elmar Schömer, and Ralf Schulze from the Johannes Gutenberg University Mainz. The colleagues from Wiesbaden and Mainz prepared the CT data that were used for our method. Further, Thomas Gietzen and Robert Brylka worked on the generation of our parametric skull model as well as on our (initial) statistics of facial soft tissue thickness. In addition, they developed the methods for skull fitting and adding FSTT by utilizing union-of-spheres. I worked on the generation of our parametric head model and its evaluation. This also includes a manual selection of* 70 *landmarks for* 82 *head scans each. Furthermore, I developed the head fitting approach to union-of-spheres. Moreover, in follow-up work [ABG+18], I improved our (previous) FSTT*

by utilizing sphere-meshes, fitted our head models to sphere-meshes, gave a comparison to union-of-spheres, and, thereby, improved the quality of the resulting facial reconstructions considerably. Finally, plausible head variants and facial reconstructions were presented by Thomas Gietzen, Robert Brylka, and me.
Corresponding publications:

[GBA+19]	*A method for automatic forensic facial reconstruction based on dense statistics of soft tissue thickness, PLOS ONE, 2019*
[ABG+18]	*A Multilinear Model for Bidirectional Craniofacial Reconstruction, VCBM, 2018*

4.1 Related Work

Turner et al. [TBK+05] introduced a method for automated skull registration and craniofacial reconstruction based on extracted surfaces from CT data that was applied to a large CT data base consisting of 280 individuals in [TBL+07]. For registration of a known skull to a skull in question, the authors use a heuristic method to find crest lines in combination with a two-step ICP registration followed by a thin plate spline (TPS) warping process. The same warping function is applied to the extracted skin of the known skull. Subsequently, from a collection of 50 to 150 warped skin surfaces, they use PCA to construct a “face-space” with a mean face for the skull in question. Using the linear combination of the eigenvectors with some a-priori knowledge, such as age and sex, they are able to generate a subset of most likely appropriate appearances for the subject in question. To this end, both the skull in question and the known skull are represented as polygonal meshes and are reduced to their single, outer surface. By disregarding the volumetric nature of the bony structure, this leads to poor fitting results in some cases.

The utilization of a deformable template mesh for forensic facial reconstruction was presented by Romeiro et al. [RME+14]. Their computerized method depends on manually identifying 57 landmarks placed on the skull. Based on these pre-selected landmarks and a corresponding FSTT (obtained from other studies) an implicit surface is generated using Hermite radial basis functions. To improve the quality of the result, they use several anatomical rules, such as the location of the anatomical planes and anatomical regressions related to the shape of the ears, nose, or mouth. Hence, as with our method, the quality of their results strongly depends on an appropriate template that properly takes age, sex, and ethnicity into account.

Shui et al. [SZD+16] presented an approach for craniofacial reconstruction based on dense FSTT statistics utilizing CT data. Their method depends on 78 manually selected landmarks placed on the skull. The landmarks guide the coarse registration of a skull template to each individual skull. This is followed by a fine-scale registration using ICP and TPS. The FSTT measurement is performed for each vertex of the deformed skull in the direction defined by the geometric coordinate. A coarse reconstruction of a face from an unidentified skull is achieved by translating each skull vertex in the defined direction by the length of the FSTT measured at this position. To achieve a smooth appearance, six additional points have to be marked manually for guiding a TPS deformation of a face template to the coarse reconstruction. Finally, the recovery of mouth, eyes, and nose has to be performed by a forensic expert. Thus, the method is not fully automatic.

Shui et al. [SZM$^+$17] proposed a method for determining the craniofacial relationship and sexual dimorphism of facial shapes derived from CT scans. Their approach employs a method for registering a reference skull and face to a target skull and face as presented in [SZD$^+$16]. Applying a PCA to the sets of registered skull and skin templates, they derive a parametric skull and skin model. By analyzing the skull- and skin-based principal component scores, they establish the craniofacial relationship between the scores and therefore reconstruct the face of an unidentified subject. Although the visual comparison of the estimated face with the real one shows good results, these results appear to be due to over-fitting. Moreover, the geometric deviations, especially in the frontal part of the face, are mostly around 2.5 – $5\,\mathrm{mm}$, indicating rather inaccurate reconstruction results.

4.2 Model Generation

In this section, we present the proposed model generation processes as outlined in Figure 4.1. We use volumetric CT scans and optical 3D surface scans as input and distinguish between two input types: skulls and heads. In the following, the outer skin surface of a head is referred to as *head* and the bony skull structure is referred to as *skull*. In order to obtain a uniform data basis, a *preprocessing* step is performed to extract the skull and the head as triangular surface meshes from each CT scan. In the next step, we need to establish the relationship between different skulls as well as between different heads. For this purpose, in a *fitting process*, we register an appropriate template model to each given mesh of a specific input type. After that, we are able to utilize the fitted templates to determine the geometric variability of the skulls, respectively heads, performing a *PCA*. As a result, we derive two parametric models: a parametric skull model and a parametric head model. Based on corresponding skulls and heads extracted from CT scans, we additionally build a dense FSTT map in the *statistical evaluation* step.

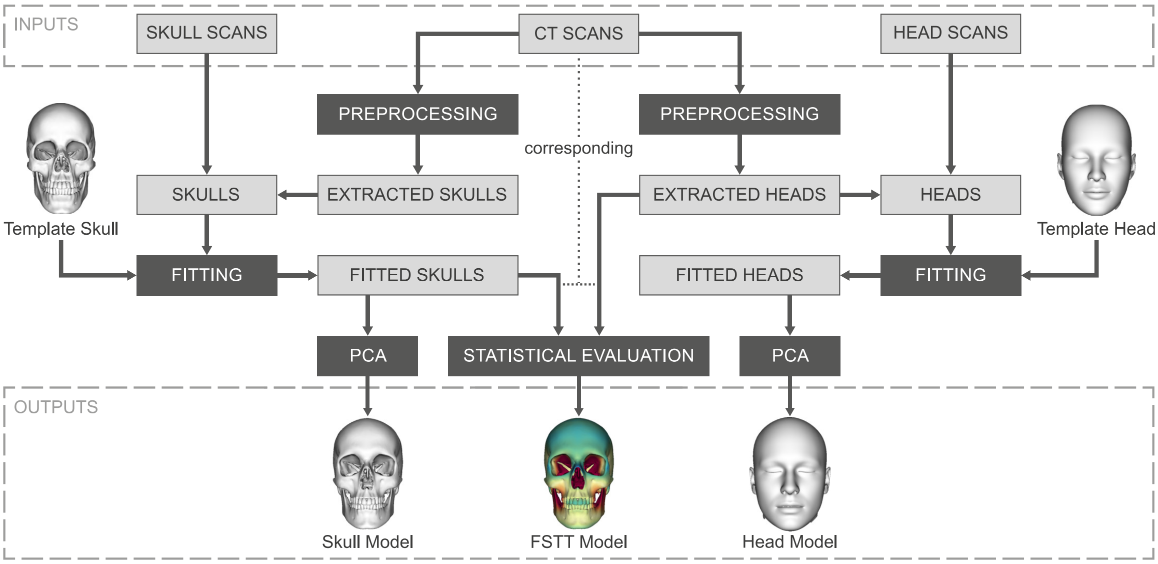


Figure 4.1: Overview of our model generation processes. Generation of a skull and a head model as well as dense FSTT statistics from multimodal input data.

4.2.1 Database

Following internal ethical review board approval, head CT scans were collected from the PACS system of the University Medical Center Mainz. We only used existing CT data (from four different CT devices) for our database. No subject was exposed to ionizing radiation for this research. The local ethical approval board (Landesärztekammer Rheinland-Pfalz, Deutschhausplatz 2, 55116 Mainz) has approved the processing of the pseudonymized existing CTs (from the DICOM database of the University Medical Center Mainz) to generate the statistical models under the approval number No 837.244.15 (10012) (date: August 5, 2015). In our study, we include CT scans that meet the following criteria:

1. The facial skull of the patient is *completely imaged*.
2. The *slice thickness* is less than or equal to $1\,\mathrm{mm}$.
3. The subject has no significant oral and maxillofacial deformations or missing parts.

From several hundred CT scans that we analyzed, a total number of 60 were suitable for our purpose. However, only 43 of these scans could be used for generating the parametric head model and the FSTT statistics, since in the remaining 17 CT scans external forces (e.g. frontal extending neck stabilizers, nasogastric tubes, etc.) compressed the soft tissue. In a *preprocessing* step, every CT scan was cropped so that we could obtain a consistent volume of interest limited to the head area. For this purpose, the most posterior point of the mandibular bone was determined automatically in the 2D slice images, and the volume was trimmed with an offset below this detected position. After this cropping step, bone and skin surface meshes were extracted using the Marching Cubes algorithm [LC87]. We used the Hounsfield units -200 and 600 as iso-values for skin and bone surface extraction, respectively. To remove unwanted parts, such as the spine or internal bone structures, a connectivity filter was applied to the bone mesh, leaving only the skull. Finally, all extracted meshes were decimated to obtain a uniform point density for all data sets [GH97]. The meshes extracted from CT data were supplemented by triangle meshes from 3D surface skull and head scans[1] of real subjects in order to fill up the database for our model generation processes. The 3D surface scans are of high quality, do not suffer from artifacts or strong noise, and consist of about $500\,\mathrm{k}$ vertices for the head and about $400\,\mathrm{k}$ vertices for the skull. In summary, the following data sets were included in the study:

1. A total number of 62 skulls (60 extracted skulls from CT scans and 2 skulls from 3D surface scans) were used to generate a skull model.

2. A total number of 82 heads (43 extracted skin surfaces from CT scans and 39 heads from 3D surface scans) were used to generate a head model.

3. A total number of 43 corresponding skulls and skin surfaces extracted from CT scans were used to build the FSTT statistics.

4.2.2 Generating a Parametric Skull Model

In order to generate a parametric skull model, we need to establish the relationship between the different skulls from our database. For this purpose, we register a single skull template to each skull individually. This template model has to be a volumetric tetrahedral mesh in order to accurately represent the solid nature of a bony skull. We therefore converted a surface triangle mesh of a skull[2] to a volumetric Delaunay tetrahedral mesh by using TetGen [Si15]. Our skull template model, which is shown in Figure 4.1, consists of $M \approx 69\,\mathrm{k}$ vertices, whose positions we denote by $\mathcal{S} = (\mathbf{s}_1, \ldots, \mathbf{s}_M)$. Tetrahedra are denoted by $T(\mathcal{S})$ and the set of all tetrahedra is denoted by $\mathcal{T} = \mathcal{T}(\mathcal{S})$. The vertices $\mathcal{S}$ and tetrahedra $\mathcal{T}$ constitute the tetrahedral mesh of our skull template.

The *fitting process* comprises the following two main stages for an input skull with vertex positions $\mathcal{P}_S = (\mathbf{p}_1, \ldots, \mathbf{p}_n)$:

1. A global rigid transformation that coarsely aligns the input skull to the skull template. The registration starts with the fast global registration approach presented in [ZPK16]

[1] From `www.3dscanstore.com`

[2] Based on `www.turbosquid.com/3d-models/3d-human-skull/691781`

followed by a refinement step using the well-known Iterative Closest Point (ICP) algorithm [BM92].

2. A fine-scale registration of the skull template to the input skull. The registration consists of several non-rigid transformation steps computed by minimizing the energy (inspired by [DDB$^+$15])

$$E_{\text{skull}}(\mathcal{S}) \;=\; E_{\text{fit}}(\mathcal{S}) \;+\; \lambda_{\text{reg}} E_{\text{reg}}\big(\mathcal{S}, \bar{\mathcal{S}}\big) \;, \tag{4.1}$$

consisting of a fitting term E_{fit} and a regularization term E_{reg}.

In the non-rigid step, the *fitting term*

$$E_{\text{fit}}(\mathcal{S}) \;=\; \frac{1}{\sum_{c\in\mathcal{C}} w_c} \sum_{c\in\mathcal{C}} w_c \left\|\mathbf{s}_c - \mathbf{p}_c\right\|^2$$

penalizes the squared distance between a vertex on the skull template $\mathbf{s}_c$ and its corresponding point $\mathbf{p}_c$, weighted by per-correspondence weights $w_c \in [0,1]$ (explained below).

The *regularization term*

$$E_{\text{reg}}\big(\mathcal{S}, \bar{\mathcal{S}}\big) \;=\; \sum_{T\in\mathcal{T}} \big(\text{vol}(T(\mathcal{S})) - \text{vol}(T(\bar{\mathcal{S}}))\big)^2$$

penalizes geometric distortion of the skull template during the fitting. $\bar{\mathcal{S}}$ represents the vertex positions of the previous deformation state, while $\mathcal{S}$ stands for the current (to-be-optimized) positions. The function $\text{vol}(T)$ denotes the volume of tetrahedron T. Thus, the regularization term penalizes the change of volume of tetrahedra. The non-rigid deformation starts with rather stiff material settings and successively softens the material during the registration process (by reducing λ_{reg}).

During the various non-rigid transformation steps, we use different strategies to define the correspondences $\mathcal{C}$. First, correspondences are determined (and weighted) by the *hierarchical ICP* approach described in [GBSS17] where we register hierarchically subdivided parts of the skull template to the input skull using individual similarity transformations. This results in several small pieces (e.g., the eye orbit) that are well aligned with the input skull. Based on the correspondences found in this step, the whole skull template is registered toward the input skull. In subsequent deformation steps, we estimate the correspondences in a closest vertex-to-vertex manner; we only consider vertices lying in high curvature regions, additionally pruning unreliable correspondences based on distance and normal deviation [GBSS17]. In the final non-rigid transformation steps when the meshes are already in good alignment, we use vertex-to-surface-point correspondences. These correspondences are determined considering all vertices employing a two-step search. First, we search for vertex-to-vertex correspondences from the input skull to the skull template, pruning unreliable correspondences based on distance and normal deviation. Second, we search for correspondences from the computed corresponding vertices on the template toward the input skull. This second step is computed in vertex-to-surface-point manner and large deviations between the vertex and surface normal are pruned.

The described two-way correspondence search prevents tangential distortions of the fitted skull template and can handle artifacts in the input skulls, e.g., artifacts in the teeth region due to metallic restorations. Additionally, it makes our registration process robust against the porous bony structure caused by low resolution of the CT scan or the age of the subject. To further prevent mesh distortions, we additionally use a release step in which the undeformed template is deformed toward the current deformed state using only pre-selected points of interest [GBSS17].

In order to analyze the accuracy of our skull registration process, we evaluated the fitting error by computing the distance for all vertices of an input skull's facial area (which covers all predefined landmarks) toward the fitted template model. The mean RMS fitting error for all 62 fitted skulls is below $0.5\,\mathrm{mm}$.

Stacking the vertex coordinates of each fitted skull into column vectors $\mathbf{s} = (x_1, y_1, z_1, \ldots, x_m, y_m, z_m)^\mathsf{T}$, we can apply PCA to the set of fitted skulls (after mean-centering them by subtracting their mean $\bar{\mathbf{s}}$). This results in a matrix $\mathbf{S} = [\mathbf{s}_1, \ldots, \mathbf{s}_d]$ containing the d ($d = 61$ in our case) principal components $\mathbf{s}_i$ in its columns. A particular skull S in the PCA space spanned by $\mathbf{S}$ can be represented as

$$S(\mathbf{a}) \;=\; \bar{\mathbf{s}} + \mathbf{S}\mathbf{a}\,, \tag{4.2}$$

where $\mathbf{a} = (a_1, \ldots, a_d)^\mathsf{T}$ contains the individual weights of the principal components of $\mathbf{S}$. The parametric skull model (4.2) can be used to generate plausible skull variants as a linear combination of the principal components. This is depicted exemplarily for the first two main principal components in Figure 4.2.

We finally pre-select 10 landmarks on the parametric skull model (Figure 4.10, left). These are used to guide the head fitting process in the automatic forensic facial reconstruction (see detailed explanation in Section 4.3.3).

4.2.3 Generating Statistics of Facial Soft Tissue Thickness

In a *statistical evaluation process*, the distances between 43 corresponding skulls and heads extracted from the CT scans are measured. To this end, for each vertex of a fitted skull we determine the shortest distance to the surface of the extracted skin surface [ASCE02]. Finally, the mean and standard deviation of the FSTT are computed per vertex. Figure 4.3 shows the mean skull $\bar{\mathbf{s}}$ with color-coded mean and standard deviation of the obtained FSTT.

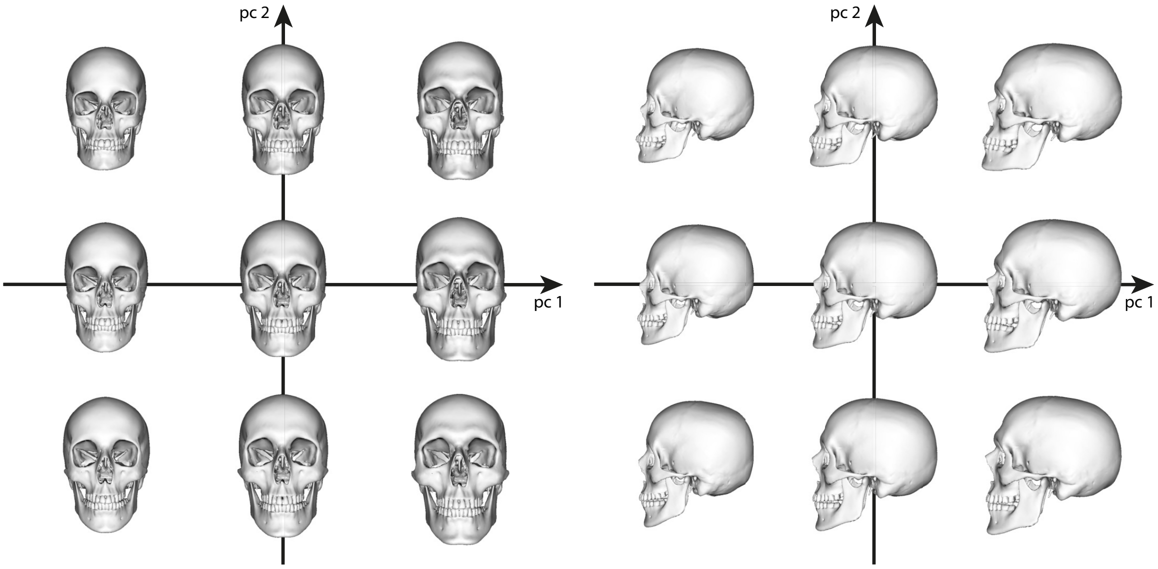


Figure 4.2: Skull variants along the two principal components with the largest eigenvalues. We visualize $\bar{\mathbf{s}} + a_1\mathbf{s}_1 + a_2\mathbf{s}_2$, where $a_i = \alpha_i \cdot \sigma_i$, $i = 1, 2$, is the weight containing the standard deviation σ_i to the corresponding eigenvector $\mathbf{s}_i$, and the factor $\alpha_i \in \{-2, 0, 2\}$.

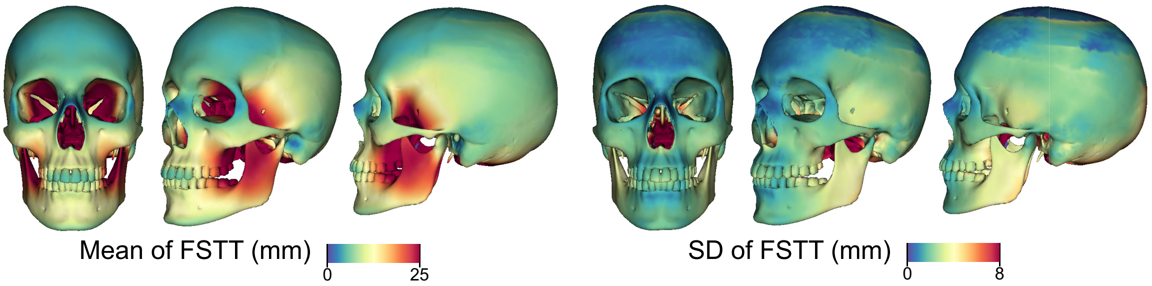


Figure 4.3: Statistics of the FSTT on a mean skull. Mean and standard deviation of FSTT computed from the 43 CT scans.

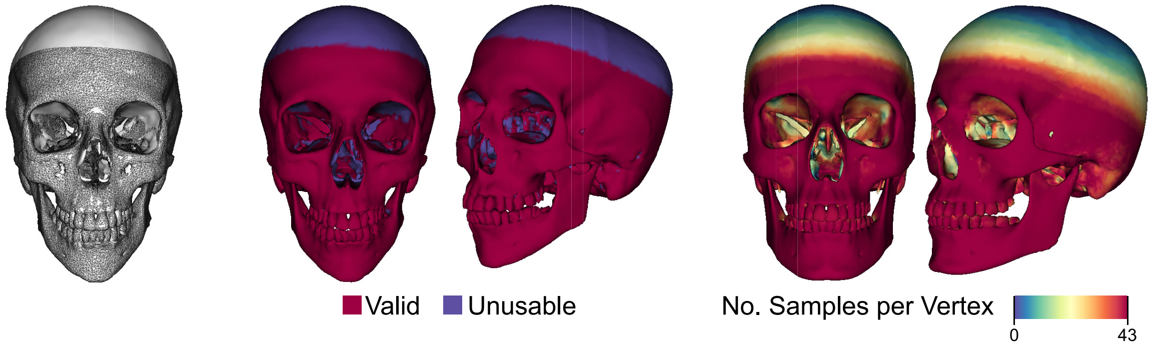


Figure 4.4: Basis for the statistical evaluation of the FSTT. From left to right: Example of a fitted skull (white) and corresponding extracted skull (black wireframe), validation mask (corresponding to left), number of samples used for all vertices in the FSTT statistics in Figure 4.3.

To obtain the FSTT data, we often register our complete skull template to *partial* input skulls, which, for instance, have holes in the bony structure or a missing upper part of the calvaria. Figure 4.4 (left) shows an example of our skull template fitted to a partial skull extracted from CT data. To avoid bias caused by false FSTT measurements, we validate if a vertex of a fitted skull corresponds to a surface point on the corresponding extracted partial skull. We exclude all vertices of the former whose distance to the latter is larger than a given threshold ($2\,\mathrm{mm}$ in our implementation). This results in the validation mask used for the statistical evaluation, depicted in Figure 4.4 (center). The number of FSTT measurements used for a particular vertex in our statistics are visualized in Figure 4.4 (right). The facial skull is covered predominantly by all 43 samples, whereas the upper part of the calvaria is covered by a few samples only.

The generated FSTT statistics are based on 43 different subjects (26 males and 17 females) with a mean age of 28 years. Figure 4.5 presents the computed FSTT (see Figure 4.3) at some landmarks commonly used in forensic reconstruction [CS16]. Our results for these landmarks fit well into the range presented in [Ste17].

4.2.4 Generating a Parametric Head Model

Similarly to the skull model, we generate the parametric head model by fitting a head template to head scans of real subjects. This establishes a correspondence between them. We then perform statistical analysis using PCA. Note that we do not use the head PCA model from Chapter 2 because it is based on low-quality Kinect scans. Furthermore, the scans for generating that PCA model were taken from the face only. Thus, the PCA model from Chapter 2 is implausible in a statistical sense for the other parts of the head. For model generation we instead employ the skin surfaces extracted from the 43 CT scans used for building the FSTT statistics (26 male, 17 female). However, since the nose tip or the upper part of the calvaria are cropped in some CT scans, we bootstrap the model generation by first fitting the head template to a set of 39 optical surface scans (20 male, 19 female) that represent complete heads. We generate a preliminary PCA model from these complete surface scans and use it to fit to the incomplete CT scans. The preliminary PCA model essentially fills the missing regions from the incomplete CT scans in a realistic manner. The final PCA model is then built from the template fits to all 82 scans.

In the following, a head scan (extracted from CT or generated through optical scan) is represented by its point set $\mathcal{P}_H = (\mathbf{p}_1, \ldots, \mathbf{p}_n)$. Since the head models are skin surfaces

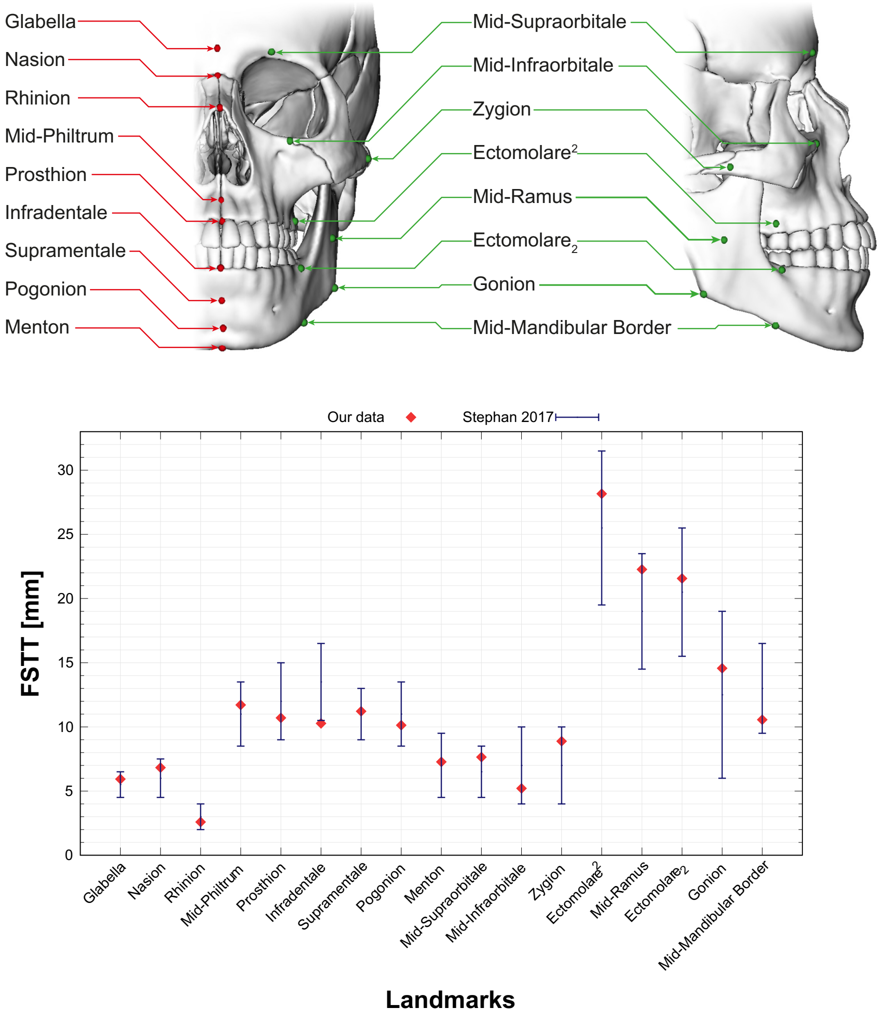


Figure 4.5: FSTT for commonly used midline and bilateral landmarks. Landmarks defined by [CS16] as produced by our method (red dots) in relation to pooled data from a recent meta-analysis [Ste17] (mean ± standard deviation as blue error bars).

only, our head template is a surface triangle mesh consisting of $N \approx 6\,\mathrm{k}$ vertices with positions $\mathcal{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$, as shown in Figure 4.1. Similarly to the skull fitting process, the template fitting process consists of two stages:

1. We first optimize scaling, rotation, and translation (see Section 2.2) of the template model to align it to the point set $\mathcal{P}_H$ by minimizing the sum of squared distances between points $\mathbf{p}_c$ on the point set $\mathcal{P}_H$ and their corresponding points $\mathbf{x}_c$ on the template model $\mathcal{X}$ using ICP [BM92].

2. After this coarse initialization, we perform a fine-scale non-rigid registration to update the vertex positions $\mathcal{X}$ such that the template model better fits the points $\mathcal{P}_H$. Following our approach from Chapter 3, we minimize a nonlinear objective function

$$E_{\text{head}}(\mathcal{X}) \;=\; E_{\text{fit}}(\mathcal{X}) \;+\; \lambda_{\text{reg}} E_{\text{reg}}\big(\mathcal{X}, \bar{\mathcal{X}}\big)\ . \tag{4.3}$$

The *fitting term* E_{fit} is defined in Equation (3.2) and penalizes squared distances between points $\mathbf{p}_c$ on the point set $\mathcal{P}_H$ and corresponding points $\mathbf{x}_c$ on the template model $\mathcal{X}$. To allow for more precise fits, we extend these closest point correspondences by 70 *facial landmarks* in the face region, on the ears, and on the lower jaw. These landmarks are manually selected on the template model and on all scans to be fitted (note that this manual work is necessary during model generation only). The per-correspondence weights w_c are used to give the landmarks a higher weight than the closest point correspondences and to assign a lower weight to surface regions that are not supposed to be fitted closely (e.g., hairs for surface scans or CT artifacts due to teeth restorations).

Analog to Equation (3.3), the *regularization term* E_{reg} penalizes the geometric distortion of the undeformed model $\bar{\mathcal{X}}$ (the result of the previous rigid/similarity transformation) to the deformed state $\mathcal{X}$.

From the 39 fits to the complete optical surface scans we construct a preliminary parametric head model. Similarly to the skull model generation, we stack the vertex positions of each fitted head $\mathbf{h} = (x_1, y_1, z_1, \ldots, x_n, y_n, z_n)^{\mathsf{T}}$ and compute a PCA model of dimension d ($d = 30$ in our case) so that we can write

$$H(\mathbf{b}) \;=\; \bar{\mathbf{h}} + \mathbf{H}\mathbf{b}\,,$$

where $\bar{\mathbf{h}}$ is the mean head, $\mathbf{H}$ is the matrix containing the principal components in its d columns, and $\mathbf{b} = (b_1, \ldots, b_d)$ contains the PCA parameters representing the head.

With the preliminary PCA model at hand, we can now fit the head template to the incomplete skin surfaces extracted from CT scans. Regions of missing data are filled realistically by the PCA model. Similarly to Section 2.2, fitting to a point set $\mathcal{P}_H$ amounts to additionally optimizing the PCA parameters $\mathbf{b}$ during the initial rigid/similarity transformation step. To this end, we minimize squared distances of corresponding points with a Tikhonov regularization ensuring plausible weights:

$$E_{\mathrm{PCA}}(\mathbf{b}) \;=\; \frac{1}{\sum_{c\in\mathcal{C}} w_c} \sum_{c\in\mathcal{C}} w_c \left\|\bar{\mathbf{h}}_c + \mathbf{H}_c\mathbf{b} - \mathbf{p}_c\right\|^2 \;+\; \frac{\lambda_{\mathrm{tik}}}{d} \sum_{k=1}^{d} \left(\frac{b_k}{\sigma_k}\right)^2 . \tag{4.4}$$

In the fitting term, $\mathbf{H}_c$ and $\bar{\mathbf{h}}_c$ are the rows of $\mathbf{H}$ and $\bar{\mathbf{h}}$ representing the point $\mathbf{h}_c$ corresponding to $\mathbf{p}_c$, that is $\mathbf{h}_c = \bar{\mathbf{h}}_c + \mathbf{H}_c\mathbf{b}$. We use $\lambda_{\mathrm{tik}} = 1 \cdot 10^{-4}$ for the regularization term, where σ_k^2 is the variance of the kth principal component. The optimal weights $\mathbf{b}$ are found by minimizing (4.4) and thus solving a linear least-squares problem. In step (1) of the head fitting process, we optimize for alignment (scaling, rotation, translation) and for shape (PCA weights) in an alternating manner until convergence is reached. Step (2), the non-rigid registration, is then performed in the same way as without the PCA model.

We finally combine the fits to the 43 CT scans and to the 39 surface scans into a single parametric PCA head model. The variation of this model along the first two principal directions is shown in Figure 4.6. While the first principal component basically characterizes head size, the second principal component describes strong variation of head shape within our training data. The strong variation in head size is due to some optical surface scans that tend to be larger than the CT scans. However, since they were specified as providing the correct scale unit, we decided to include them in our database.

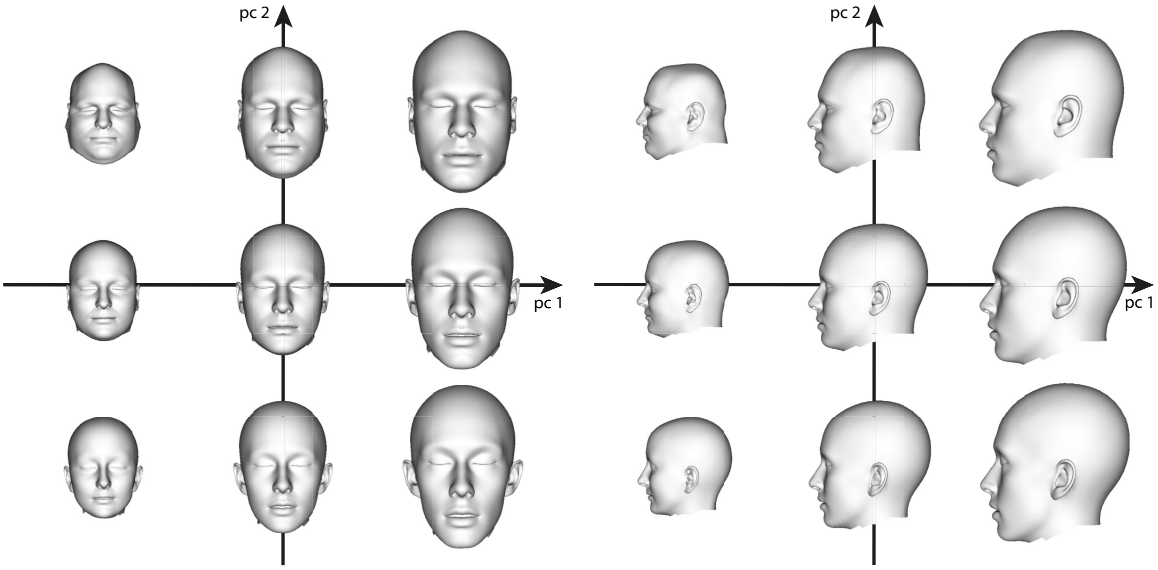


Figure 4.6: Head variants along the two principal components with the largest eigenvalues. We visualize $\bar{\mathbf{h}} + b_1\mathbf{h}_1 + b_2\mathbf{h}_2$, where $b_i = \beta_i \cdot \sigma_i$, $i = 1, 2$, is the weight containing the standard deviation σ_i to the corresponding eigenvector $\mathbf{h}_i$, and the factor $\beta_i \in \{-2, 0, 2\}$.

In order to analyze the accuracy of our head fitting process, we evaluate the RMS error for all 82 head scans:

$$\mathrm{rms}(\mathcal{X}, \mathcal{P}_H) \;=\; \sqrt{\frac{1}{\sum_{c \in \mathcal{C}} w_c} \sum_{c \in \mathcal{C}} w_c \left\| \mathbf{x}_c - \mathbf{p}_c \right\|^2}\,.$$

This is similar to (3.2) and measures the distance between corresponding point pairs from $\mathcal{X}$ and $\mathcal{P}_H$. Depending on our input data, we weight down regions that should not be fitted closely (hairs, CT artifacts) to prevent these regions from influencing the error measure too much. Averaging this error over all 82 scans gives an overall fitting error of $0.19\,\mathrm{mm}$. Note that we prune unreliable correspondences above a distance threshold of $2\,\mathrm{mm}$. These are therefore not considered for error evaluation. However, since the overall fitting error is an order of magnitude smaller, it is not significantly influenced by this pruning.

As done before for the parametric skull model, we also manually pre-select 10 corresponding landmarks on the parametric head model (Figure 4.10, right). These landmarks are used for the automatic forensic facial reconstruction.

4.3 Automatic Forensic Facial Reconstruction

Our automatic forensic facial reconstruction process is based on the generated parametric skull model, the FSTT statistics, and the parametric head model described in the previous sections. In the following, we use an anonymized CT scan of a female subject with an age of 21 years to demonstrate the quality of our forensic facial reconstruction. This CT scan was not used for constructing the parametric skull model, head model, or FSTT statistics. The reconstruction process runs in three steps, as shown in Figure 4.7, and is explained in the following sections.

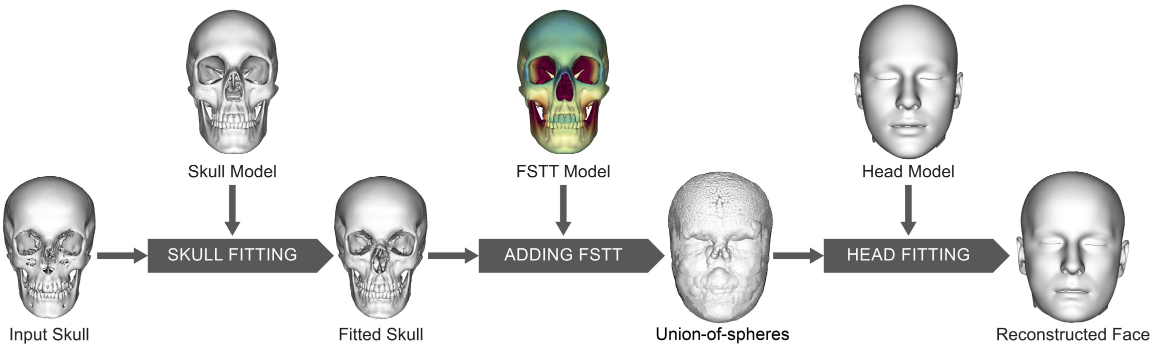


Figure 4.7: Processing steps for the automatic forensic facial reconstruction. The reconstruction of a face from a given input skull utilizing the generated parametric skull model, the FSTT statistics, and the parametric head model.

4.3.1 Skull Fitting

Given scanned skull remains as input (Figure 4.8, left), the *skull fitting* process is very similar to the registration process described in the section about the generation of the parametric skull model (Section 4.2.2). The main difference is that we are finally able to utilize the generated parametric skull model (4.2) as a starting point for the subsequent deformation steps. First, we align the parametric skull model to the given skull by using the global registration approach presented in [ZPK16]. To further optimize the alignment, we search for reliable point correspondences $\mathcal{C}$ between the given skull and the parametric skull model and compute the optimal scaling, rotation, and translation in closed-form [Hor87], as explained in Section 2.2. After optimizing the alignment, we continue with optimizing the shape. Similarly to the PCA fitting of heads (4.4), we look for the coefficient vector $\mathbf{a}$ of the parametric skull model (4.2) with

$$E_{\text{PCA}}(\mathbf{a}) \;=\; \frac{1}{|\mathcal{C}|}\sum_{c\in\mathcal{C}} \|\bar{\mathbf{s}}_c + \mathbf{S}_c\mathbf{a} - \mathbf{p}_c\|^2 \;+\; \frac{\lambda_{\text{tik}}}{d}\sum_{k=1}^{d}\left(\frac{a_k}{\sigma_k}\right)^2 ,$$

where $\lambda_{\text{tik}} = 1 \cdot 10^{-3}$, σ_k^2 is the variance of the kth principal component of the skull model, and d ($d = 61$ in our case) is the number of employed PCA components. Optimization for alignment and shape is alternated until convergence. Before each optimization (alignment or shape) we recompute the point correspondences $\mathcal{C}$. After this initialization, we continue with non-rigid registration by minimizing (4.1) (Figure 4.8, right).

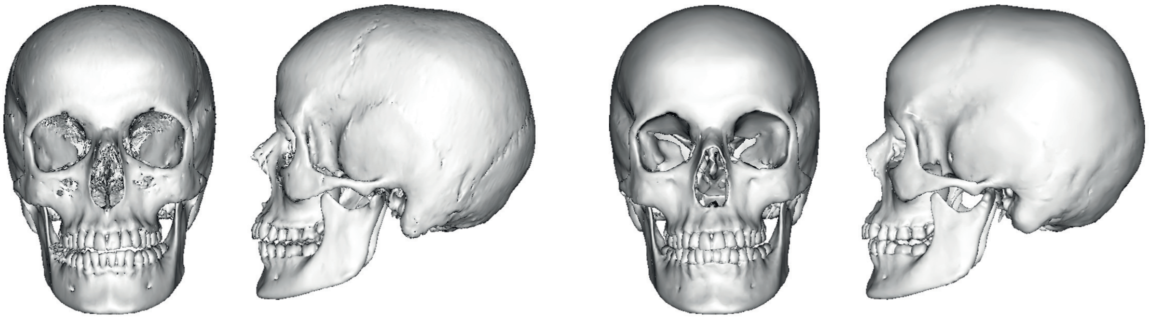


Figure 4.8: Skull fitting results for a given skull. Extracted skull from CT (left) and non-rigidly fitted skull (right).

4.3.2 Adding Facial Soft Tissue Thickness

Next we assign FSTT values based on our FSTT statistics to the fitting result of a given skull. An important advantage of our approach is that our FSTT statistics only contain *scalar* FSTT values without a particular measurement direction, such as skull normal or skin normal, since these directions are hard to determine in a robust manner due to noise or fitting errors. In our case, the measured skin position which is the closest point on the skin surface for a vertex of the skull is located on a sphere centered at the skull vertex with the radius being the corresponding FSTT value. Figure 4.9 (left) shows a side view of the FSTT measurement results for a few pre-selected points on the midline.

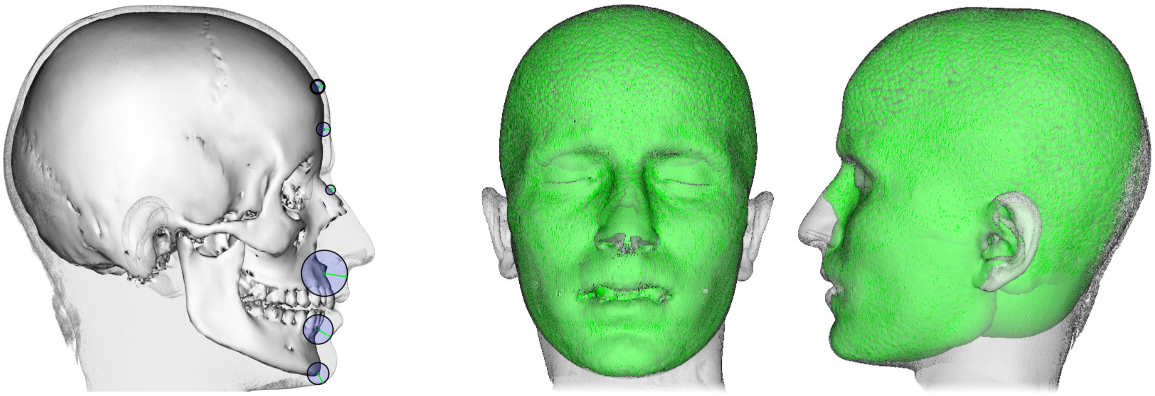


Figure 4.9: FSTT for a given individual visualized as union-of-spheres. At each skull vertex, a sphere with a radius corresponding to the actual FSTT value from the ground truth data set is drawn. From left to right: Some example spheres for points on the midline, union of all spheres (in green) with original skin surface as overlay.

Knowing both the skull and the skin surface for a subject allows the computation of the *actual* FSTT. Figure 4.9 (center and right) shows an overlay of the extracted skin surface and the union of all spheres centered at the skull vertices. The spheres have the appropriate FSTT values as radii. We call this the *union-of-spheres*. The depicted union-of-spheres is based on the exact FSTT of this subject and provides a visually good approximation of the real skin surface. Certainly, since nose and ears do not have a directly underlying bony structure, the method does not provide this kind of information. Approaches for prediction of nasal morphology, such as [KHS03, RWP10], give some hints about the nose, e.g., the approximated position of the nose tip, but they do not really create an individual nose shape for a particular subject. In a real application scenario, the age, sex, and ancestry of the individual are derived from its skeleton remains and disaggregated FSTT statistics are used for reconstruction. In our case, the sample size is too small to build specific FSTT statistics. As an approximation, we simply build the union-of-spheres based on the mean or a sample of our parametric FSTT model (Section 4.3.4).

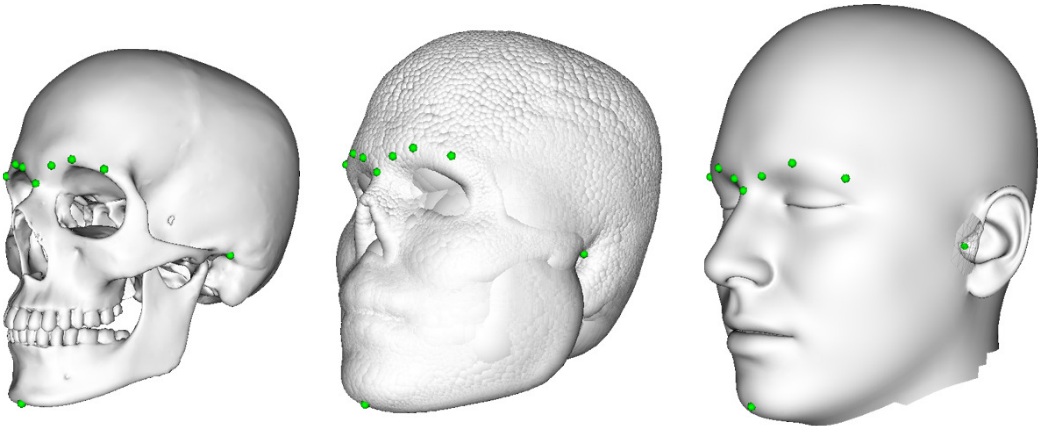


Figure 4.10: Landmarks for the automatic facial reconstruction. From left to right: Mean skull with pre-selected landmarks, union-of-spheres based on mean FSTT with projected landmarks, and mean head with pre-selected landmarks. The landmarks consist of two midline landmarks and four bilateral landmarks which are selected once on the parametric skull and head model after model generation. The landmarks are based on the proposed nomenclature of [CS16]: *nasion* and *menton* (from craniometry) and *mid-supraorbitale* and *porion* (from craniometry) as well as *ciliare lateralis* and *ciliare medialis* (from capulometric) and their corresponding counterparts on the skull or skin surface, respectively.

4.3.3 Head Fitting

Given a specific union-of-spheres, the next step is to derive a facial profile from this data. For this purpose, we deform our parametric head model to the (under-specified) union-of-spheres. The fitting procedure is very similar to the generation of our parametric head model. Similarly to the above approach, we initially align the union-of-spheres with the parametric head model. This time, however, the landmarks on the fitted skull, which have been selected during the skull model generation, are projected automatically onto the surface of the union-of-spheres as depicted in Figure 4.10. More precisely, each individual landmark on the fitted skull is shifted along its skull normal by the corresponding FSTT value.

The projected landmarks give us robust correspondences on the parametric head model. They are automatically determined and replace the manually selected landmarks used during model generation. We start by optimizing scaling, rotation, and translation, as well as PCA parameters based on the set of landmarks. This initialization is followed by a fine-scale non-rigid registration based on landmarks and closest point correspondences between the parametric head model and the given union-of-spheres.

While this process is very similar to the model generation phase, it differs from it in the following point: We use the per-correspondence weights w_c in the fitting energy E_{fit} to give points on the outer surface of the union-of-spheres more influence than points in the interior, since the former can be considered as an approximation to the skin surface that we intend to fit. To this end, we first identify if a point $\mathbf{q}_c$ on the union-of-spheres is outside of its corresponding point $\mathbf{h}_c$ on the head template by checking if $\mathbf{n}_c^{\mathsf{T}}(\mathbf{q}_c - \mathbf{h}_c) \geq 0$, where $\mathbf{n}_c$ is the normal vector of $\mathbf{h}_c$. For such correspondences, we set

$$w_c \;=\; 1 + 10^8 \cdot \|\mathbf{h}_c - \mathbf{q}_c\| \,/B\,, \tag{4.5}$$

where B is the bounding box size of the model.

As mentioned before, nose and ears do not have a directly underlying bony structure. Thus, the union-of-spheres does not provide any data for such regions. Utilizing a parametric head model allows the reconstruction of nose and ears in a statistical sense, i.e., as an element related to the underlying PCA space.

4.3.4 Generating plausible Head Variants

The simplest method for facial reconstruction is to fit the head template to a union-of-spheres based on the *mean* of the FSTT statistics. However, this approximation will rarely match a specific subject. To get a reliable FSTT diversification for an individual, we again adopt the PCA approach creating a parametric FSTT model

$$\text{FSTT}(\mathbf{c}) \;=\; \bar{\mathbf{f}} + \mathbf{F}\mathbf{c}\,, \tag{4.6}$$

where $\bar{\mathbf{f}}$ is the mean FSTT, $\mathbf{F}$ contains the principal components of the FSTT, and $\mathbf{c} = (c_1, \ldots, c_d)$ contains the d ($d = 42$ in our case) PCA parameters. Using this parametric FSTT model, we can create plausible FSTT variants for the given input skull. Since the CT scans used for the FSTT statistics are mostly missing the upper part of the calvaria, the FSTT values obtained in this area are mainly very large and invalid. Thus, we omit this area for the construction of our parametric FSTT model (4.6). This results in partial union-of-spheres. Figure 4.11 (top) depicts a subset of the partial union-of-spheres along the two principal components with the largest eigenvalues for the given input skull.

Our head fitting process described above can be applied to partial sphere models without special adjustments. As depicted in Figure 4.11 (bottom), our approach is able to generate plausible head variants based on the corresponding union-of-spheres in Figure 4.11 (top). As we are using a parametric model of the complete head, the missing parts like nose, ears, and especially the skin surface above the calvaria are reconstructed in a statistical sense, i.e., as an element related to the underlying PCA space.

The automated technique suggested in this chapter aids recognition of unknown skull remains by providing statistical estimates derived from a CT head database and 3D surface scans. By creating a range of plausible heads in the sense of statistical estimates, a “visual guess” of likely heads can be used for recognition of the individual represented by the unknown skull. Note that due to privacy reasons the extracted or reconstructed skin surface can only be shown for one single subject. Compared to clay-based sculpturing, which depends on the ability of the operator, our method provides a good approximation of the facial skin surface in a statistical sense (Figure 4.12). In the following section, we present our follow-up work that shows how to further improve our FSTT.

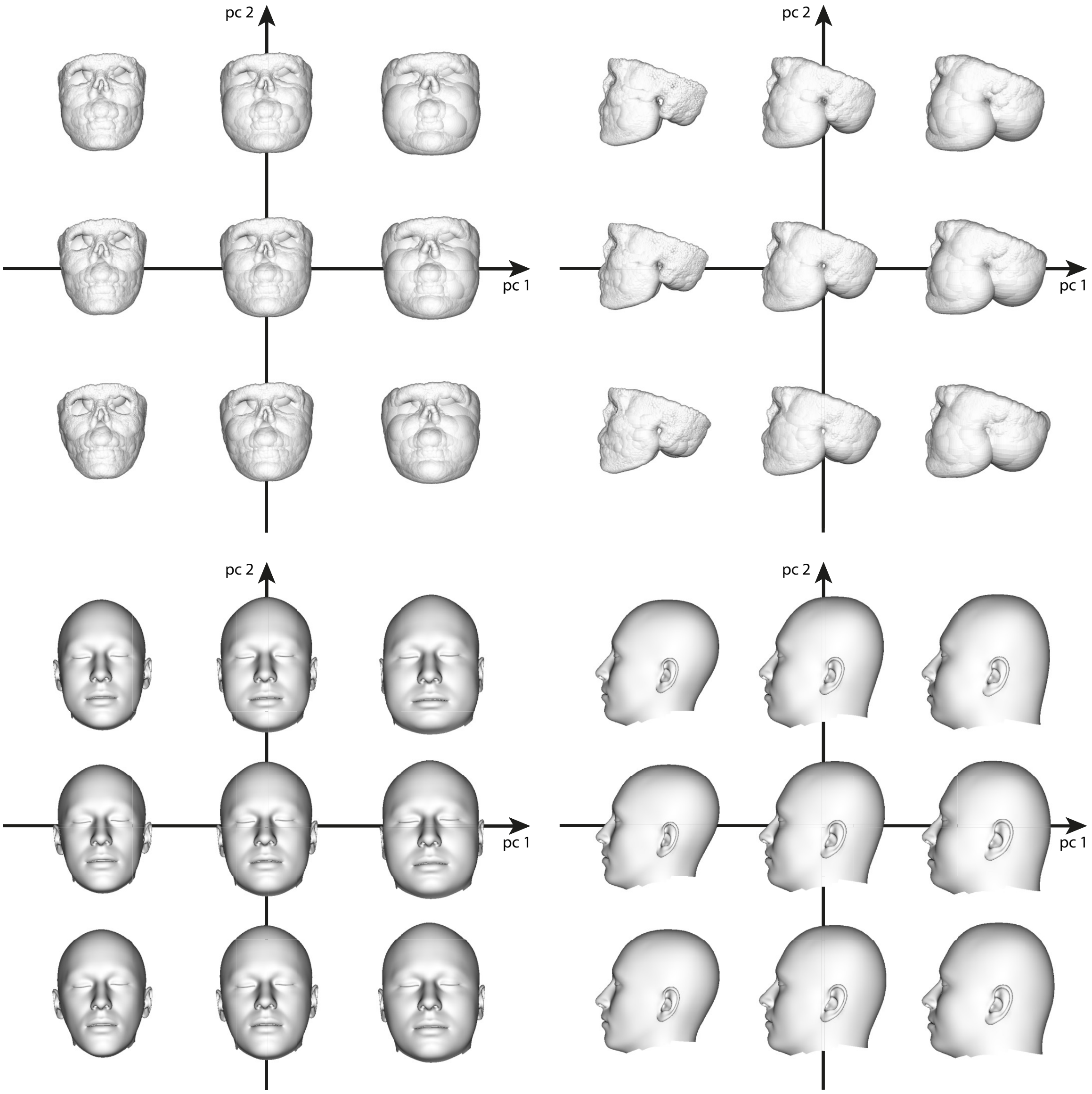


Figure 4.11: Variants of plausible FSTT distributions for the anonymized given skull. Top: Partial union-of-spheres variants along the two principal components with the largest eigenvalues. We visualize $\bar{\mathbf{f}} + c_1\mathbf{f}_1 + c_2\mathbf{f}_2$, where $c_i = \gamma_i \cdot \sigma_i$, $i = 1, 2$, is the weight containing the standard deviation σ_i to the corresponding eigenvector $\mathbf{f}_i$, and the factor $\gamma_i \in \{-2, 0, 2\}$. Bottom: Head model fitted to these partial union-of-spheres.

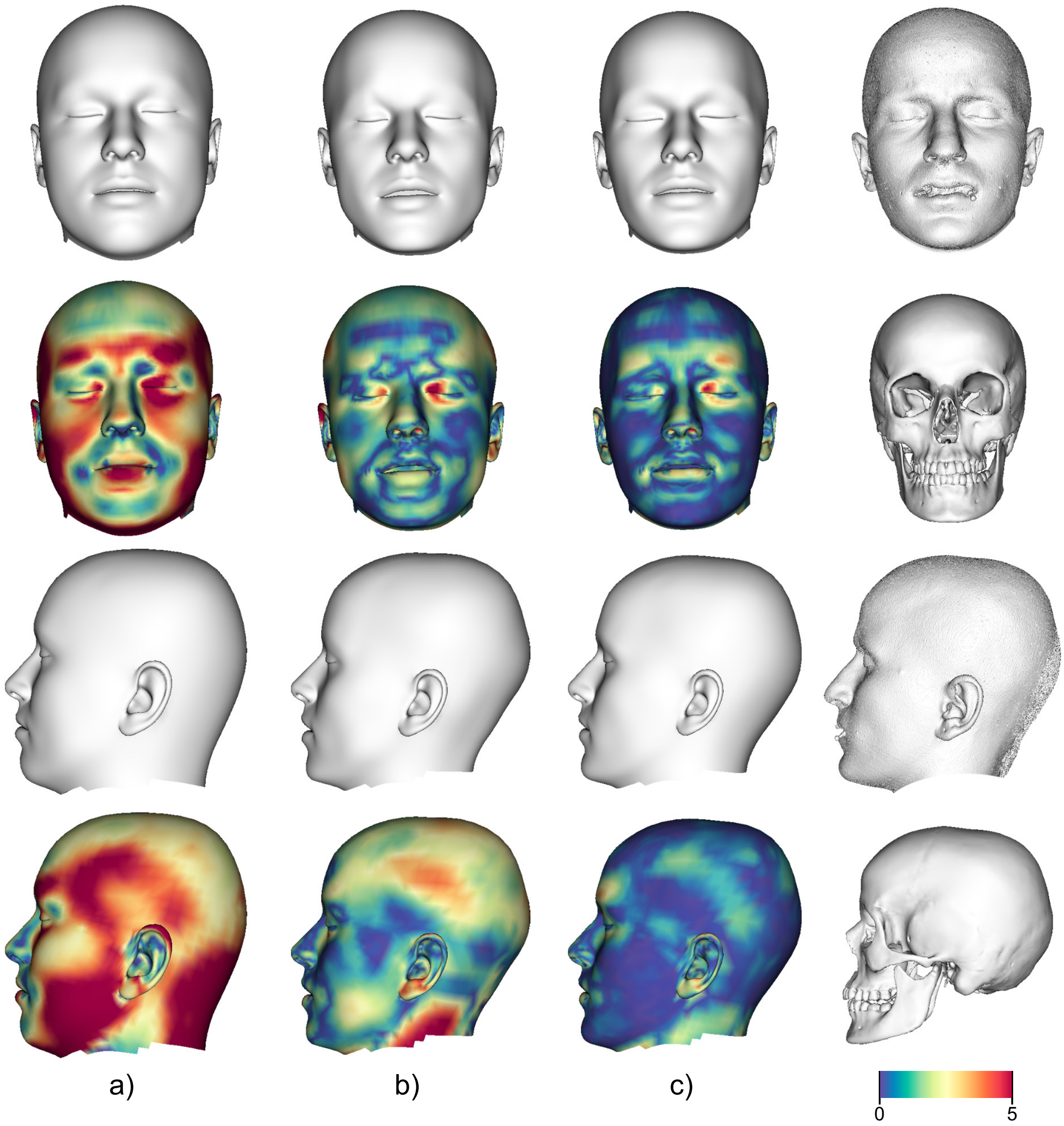


Figure 4.12: Head fittings with color coded distances (in mm) to original skin surface extracted from CT (last column). First three columns from left to right: Fitted head to union-of-spheres based on a) mean FSTT (RMS error $4.04\,\mathrm{mm}$), b) best fit in PCA space (RMS error $1.99\,\mathrm{mm}$), and c) original FSTT (RMS error $1.32\,\mathrm{mm}$).

4.4 Accurate Facial Soft Tissue Thickness

The FSTT is defined by a scalar thickness radius r_i for each vertex on the outside of the skull model, i.e., where a meaningful tissue thickness between the skull bone and the skin surface can be determined. The set of these radii is denoted by $\mathcal{R} = (r_1, \ldots, r_m)$, where $m \approx 16.5\,\mathrm{k}$ is the number of outer skull vertices (from the overall $M \approx 69\,\mathrm{k}$ skull vertices).

So far, the geometric representation of an FSTT-specified offset from a given skull was constructed as the union-of-spheres centered at each outer skull vertex $\mathbf{c}_i$ with its corresponding FSTT radius r_i, as shown in Figure 4.13(a). In the following, we replace the discontinuous, non-smooth union-of-spheres by a *sphere-mesh* [TGB13]. This leads to a continuous surface envelope around the skull representing the FSTT offset (Figure 4.13(b)).

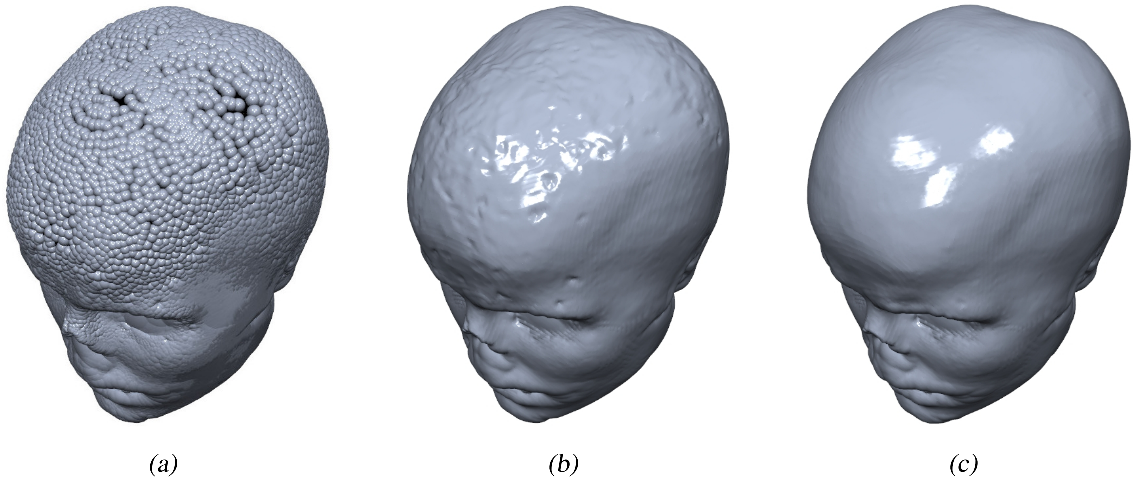


Figure 4.13: Adding facial tissue, specified by FSTT distribution, onto a given skull: Union-of-spheres (a), sphere-mesh based on unoptimized radii with dent-like artifacts (b), smooth sphere-mesh based on optimized radii (c).

4.4.1 Sphere-Mesh Representation

Sphere-meshes are a variant of convolution surfaces [BS91] and were originally used for shape approximation [TGB13]. Recently, they have also been employed for hand modeling and tracking [TPT16, TTR$^+$17]. For representing the FSTT-offset from a skull through sphere-meshes, we consider all triangles on the outer skull surface where each vertex $\mathbf{c}_i$ has an associated FSTT thickness radius r_i. Each such triangle $(\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3)$ is convolved with a sphere whose spatially varying radius is determined by barycentric interpolation of the FSTT thicknesses r_1, r_2, r_3, leading to a *triangle wedge* as shown in Figure 4.14. With

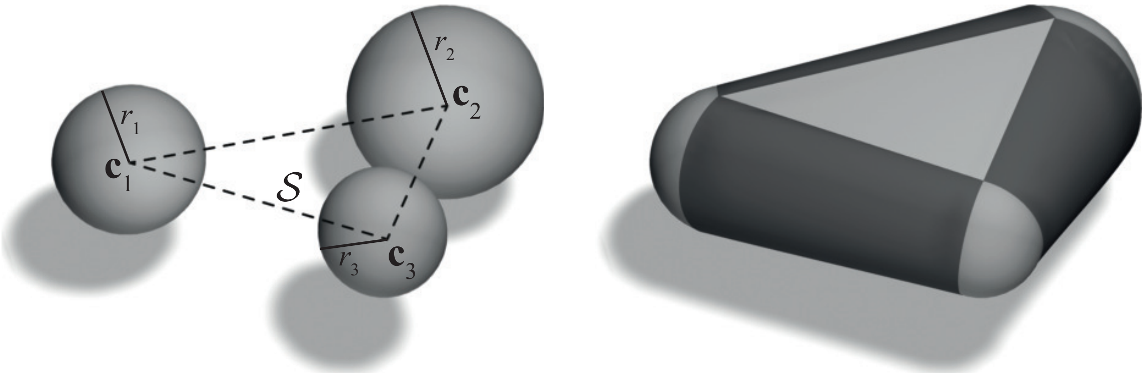


Figure 4.14: Left: Skeleton triangle $\mathcal{S}$ with centers $\mathbf{c}_i$ and radii r_i. Right: The resulting sphere-mesh wedge (image from [TPT16]).

$\mathrm{B}(\mathbf{x}, \mathbf{c}, r) = \|\mathbf{x} - \mathbf{c}\| - r$ denoting the signed distance from a sphere of radius r centered at $\mathbf{c}$, the triangle wedge is implicitly defined as the zero-set of

$$\min_{\substack{\alpha,\beta,\gamma\geq 0\\ \alpha+\beta+\gamma=1}} \mathrm{B}(\mathbf{x},\, \alpha\mathbf{c}_1 + \beta\mathbf{c}_2 + \gamma\mathbf{c}_3,\, \alpha r_1 + \beta r_2 + \gamma r_3) \ ,$$

where α, β, and γ are the barycentric coordinates.

If we denote the set of all wedges as $\mathcal{W}$, the FSTT, as the union of all wedges, is defined as the zero-set of its signed distance

$$\mathrm{dist}(\mathbf{x}) \;=\; \min_{t\in\mathcal{W}} \min_{\substack{\alpha,\beta,\gamma\geq 0\\ \alpha+\beta+\gamma=1}} \mathrm{B}\big(\mathbf{x},\, \alpha\mathbf{c}_1^t + \beta\mathbf{c}_2^t + \gamma\mathbf{c}_3^t,\, \alpha r_1^t + \beta r_2^t + \gamma r_3^t\big) \ . \tag{4.7}$$

From the above *implicit* representation, we can extract an *explicit* triangle mesh through the Marching Cubes algorithm. In our experiments, a voxel size of $2\,\mathrm{mm}$ turned out to provide a good trade-off between precision and computing time. Thus, we use this voxel size for all reconstructed sphere-meshes.

The Marching Cubes algorithm requires evaluation of the signed distance to the sphere-mesh, i.e., Equation (4.7), for each point from the volumetric grid. Despite parallelizing this operation over multiple CPU cores using OpenMP, it remains a computational bottleneck. We therefore employ bounding spheres for each triangle wedge to quickly select potential wedges or prune wedges that are too far away. This simple strategy reduced the average time required for Marching Cubes from $19.5\,\mathrm{min}$ to $67\,\mathrm{s}$ on a desktop PC with Intel Xeon CPU ($4 \times 3.6\,\mathrm{GHz}$).

As shown in Figure 4.13(b), the resulting FSTT-offset is a continuous surface, as opposed to the discontinuous union-of-spheres shown in Figure 4.13(a). However, it suffers from dent-like artifacts due to wrong FSTT values, which we correct in the following.

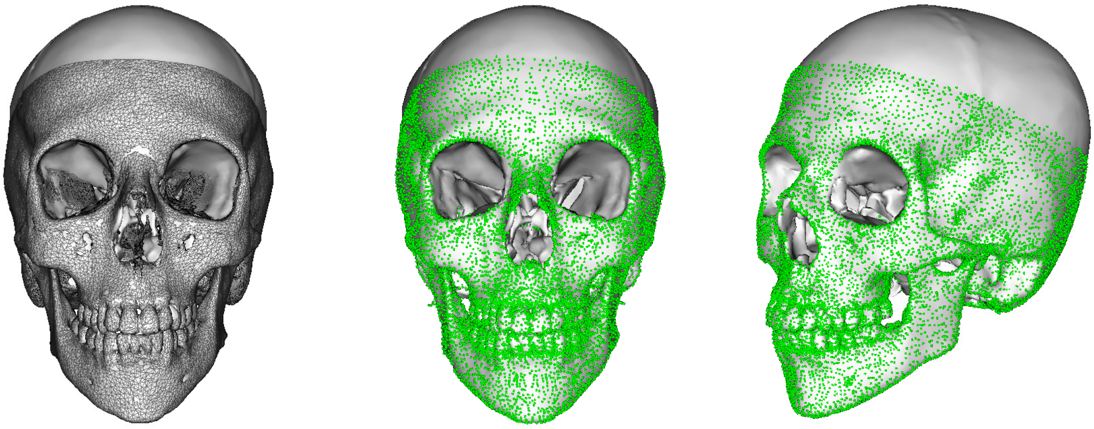


Figure 4.15: The parametric skull model (solid white) fitted to the incomplete skull extracted from a partial CT scan (gray wireframe overlay). The green points depict the skull model vertices that (i) lie on the outside and (ii) overlap the extracted skull. For those vertices the FSTT radii can be computed and optimized.

4.4.2 Optimization of FSTT Radii

Given a CT scan, we compute the FSTT by first fitting the parametric skull model to the extracted CT skull and then determining an FSTT radius for each vertex of the outer skull surface. In Section 4.2.3, the radius r_i was computed as the minimum distance from the skull vertex $\mathbf{c}_i$ to the CT-extracted skin surface. However, noise in the CT data can lead to skin vertices perturbed into the interior, leading to erroneously too short distances and thus an underestimation of the radius. This manifests as dent-like artifacts shown in Figure 4.13(b). To overcome these problems, we optimize the FSTT radii such that the resulting sphere-mesh fits the skin surface in the least-squares sense.

In order to set up the optimization, we initialize the radii by our above-mentioned minimum distance heuristic. Since many CT scans are missing the calvaria part, we cannot estimate the FSTT for the skullcap (Figure 4.15). As the FSTT hardly varies in this region, we fill up the missing values by harmonic interpolation, i.e., we solve $\Delta r_i = 0$ for all missing radii, with the known valid radii as Dirichlet boundary constraints (see Section 3.3.3). This amounts to solving a sparse linear Laplace system in which the Laplacian Δr_i is discretized using the well-known cotangent weights and Voronoi areas [BKP+10]. We denote the resulting initial radii by $\bar{r}_i$.

We then optimize the FSTT radii such that the sphere-mesh closely fits the skin surface extracted from CT. To this end, we determine point-to-point correspondences between skin vertices $\mathbf{p}_c$ and their closest points $\mathbf{s}_c$ on the sphere-mesh and then minimize their squared distances. Finding the closest sphere-mesh point $\mathbf{s}_c$ for a given skin vertex $\mathbf{p}_c$ amounts

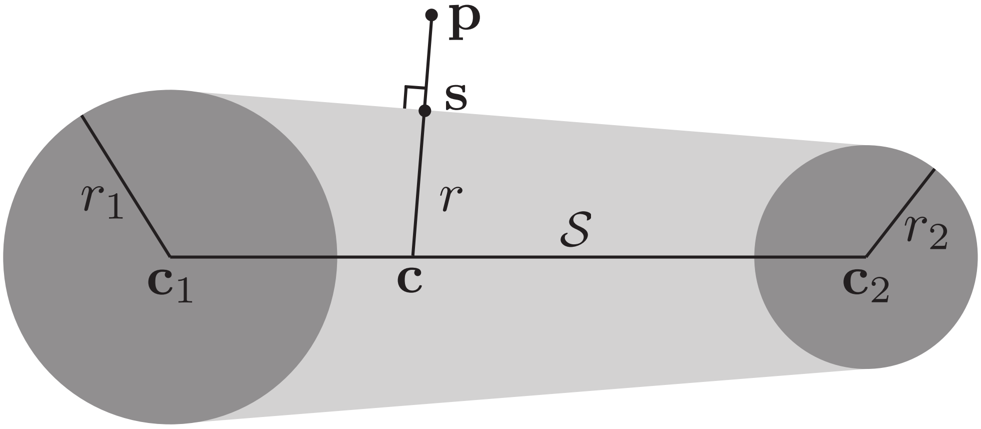


Figure 4.16: Computation of the nearest point s on a triangle wedge.

to first determining the triangle t and barycentric coordinates α, β, γ minimizing $\mathrm{dist}(\mathbf{p}_c)$ from (4.7) (using linear search for t and gradient descent for α, β, γ). From the interpolated values $\mathbf{c} = \alpha\mathbf{c}_1^t + \beta\mathbf{c}_2^t + \gamma\mathbf{c}_3^t$ and $r = \alpha r_1^t + \beta r_2^t + \gamma r_3^t$, we get the closest point on the sphere-mesh as $\mathbf{s}_c = \mathbf{c} + r\,(\mathbf{p}_c - \mathbf{c}) \,/\, \|\mathbf{p}_c - \mathbf{c}\|$ (Figure 4.16). As for Marching Cubes, the use of bounding spheres speeds up the computation of closest points considerably.

In order to remove unreliable correspondences, we prune correspondences $(\mathbf{p}_c, \mathbf{s}_c)$ if their distance is larger than $1\,\mathrm{mm}$, if the angle between their normal vectors $\mathbf{n}(\mathbf{p}_c)$ and $\mathbf{n}(\mathbf{s}_c)$ is larger than 20°, or if the angle between $\mathbf{n}(\mathbf{s}_c)$ and the normal vector $\mathbf{n}(t)$ of the wedge's skeleton triangle is larger than $45°$. Similarly to the symmetry heuristic of [ZPK16], if $\mathbf{p}_c'$ is the nearest point from $\mathbf{s}_c$ on the skin surface, then $\|\mathbf{p}_c - \mathbf{p}_c'\|$ should be at most $0.5\,\mathrm{mm}$. Finally, we prune correspondences that are located on the boundary of a sphere-mesh triangle and where the opposite wedge has no correspondences.

For each remaining correspondence $(\mathbf{p}_c, \mathbf{s}_c)$, we fix the barycentric coordinates $\alpha^c, \beta^c, \gamma^c$ and the triangle $(\mathbf{c}_1^c, \mathbf{c}_2^c, \mathbf{c}_3^c)$ such that the (squared) distance becomes a quadratic function of the radii r_1^c, r_2^c, r_3^c. If $\mathcal{C}$ denotes the set of correspondences and $\mathrm{B}(\mathbf{x}, \mathbf{c}, r)$ the sphere distance, the fitting term to be minimized becomes

$$E_{\text{fit}}(\mathcal{R}) \;=\; \frac{1}{|\mathcal{C}|}\sum_{c\in\mathcal{C}} \mathrm{B}(\mathbf{p}_c,\, \alpha^c\mathbf{c}_1^c + \beta^c\mathbf{c}_2^c + \gamma^c\mathbf{c}_3^c,\, \alpha^c r_1^c + \beta^c r_2^c + \gamma^c r_3^c)^2 \;.$$

The minimization of the fitting energy is regularized by two terms

$$E_{\text{init}}(\mathcal{R},\bar{\mathcal{R}}) \;=\; \frac{1}{m}\sum_{j=1}^{m}\|r_j - \bar{r}_j\|^2 \quad\text{and}\quad E_{\text{reg}}(\mathcal{R},\bar{\mathcal{R}}) \;=\; \frac{1}{m}\sum_{j=1}^{m}\|\Delta r_j - \Delta\bar{r}_j\|^2$$

penalizing the deviation of radii r_j and their Laplacians Δr_j from the initial state $\bar{r}_j$, where the Laplacian Δr_j is again discretized using the cotangent weights [BKP+10]. For given correspondences $\mathcal{C}$, we then minimize the combined objective

$$E_{\text{fstt}}(\mathcal{R}) \;=\; E_{\text{fit}}(\mathcal{R}) \;+\; 0.1 \cdot E_{\text{init}}\big(\mathcal{R}, \bar{\mathcal{R}}\big) \;+\; \lambda_{\text{reg}} E_{\text{reg}}\big(\mathcal{R}, \bar{\mathcal{R}}\big) \;, \tag{4.8}$$

which is quadratic in the radii $\mathcal{R}$ and hence amounts to solving a sparse linear system (see Section 2.2.2). Overall, we alternatingly compute correspondences $\mathcal{C}$ and optimize the radii $\mathcal{R}$ by minimizing (4.8). The process is iterated until convergence is reached. We start with $\lambda_{\text{reg}} = 1$ and decrease to $\lambda_{\text{reg}} = 0.1$ in an outer loop without any intermediate steps. When decreasing λ_{reg}, we also update $\Delta\bar{r}_j$ with Δr_j from our current guess. This process typically converges in 4 – 6 iterations and takes about $30\,\text{s}$ on average.

In comparison, Tkach et al. [TPT16, TTR+17] decompose wedges into triangles, spheres, and cones and solve a nonlinear optimization for fitting a sphere-mesh of 30 nodes to their hand model. In contrast, our approach is fully implicit and seamlessly handles even special cases when radii are larger than skeleton triangles. Our fitting requires simple linear least-squares systems only and efficiently and robustly optimizes our $m \approx 16.5\,\text{k}$ FSTT radii.

As shown in Figure 4.13(c), our optimization successfully removes the artifacts due to CT noise, leading to a smooth FSTT geometry. Based on the techniques presented in this section, we improve all FSTT distributions from our database. From the improved FSTT distributions, we are able to compute an improved parametric FSTT PCA model analogous to Section 4.3.4.

4.4.3 Fitting a Head Model

In order to reconstruct a 3D face from both a given skull and a given FSTT distribution, we fit our parametric head model to the FSTT-specified geometric offset from the outer skull surface similarly to Section 4.3.3. To this end, the template head model is first coarsely aligned through a similarity transform and through optimization of head-PCA parameters, followed by a fine-scale non-rigid deformation. To guarantee plausible reconstructions, the fitting process is regularized by penalizing large PCA weights as well as strong bending.

In Section 4.3.2, the FSTT-offset was represented as a union-of-spheres, leading to a discontinuous, non-smooth surface. Fitting the head model to this type of target geometry requires a rather strong bending regularization and even a dedicated weighting scheme (4.5) for up-weighting correspondences in the external part of the FSTT (see Figure 4.17, top row for an exemplary fitting result). In contrast, our proposed sphere-mesh representation with optimized FSTT radii provides a smoother and more accurate FSTT-offset. In Section 4.3.3, we fit our head model to an explicit representation of the FSTT-offset. For simplicity, we convert the implicit sphere-mesh to an explicit triangle mesh using Marching Cubes (grid spacing of $2\,\mathrm{mm}$) and then point-sample the triangle mesh to compute correspondences. The higher surface smoothness and FSTT accuracy allow for less regularization and therefore result in more precise fits (Figure 4.17, bottom row). A quantitative evaluation yields an average RMS fitting error of $0.51\,\mathrm{mm}$ for the optimized sphere-meshes, compared to an average RMS fitting error of $0.82\,\mathrm{mm}$ for the union-of-spheres, which is an improvement of $37\,\%$ in fitting accuracy.

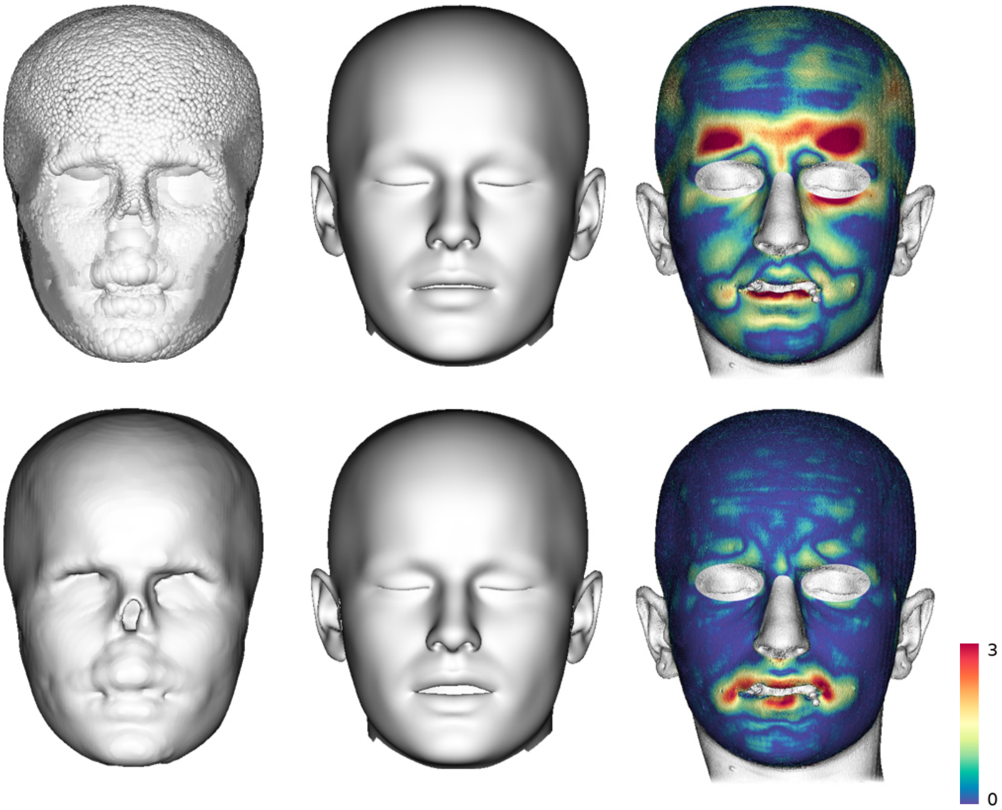


Figure 4.17: Fitting the head model to an FSTT-offset of a given skull. Representing the offset as union-of-spheres (top) leads to a larger geometric error as our proposed sphere-mesh representation (bottom). From left to right: FSTT-offset, fitted head model, color-coded distance (in mm) to true skin.

We presented an automated method based on a parametric skull model, a parametric head model, and FSTT statistics for reconstructing the face for a given skull. In the next chapter, we build on this work and generate an efficient computation model—unifying the relationship between facial skin, the underlying bony structures, and the facial soft tissue thickness—and demonstrate how it has several interesting and high-potential applications in the medical context.

5 A Multilinear Model for Bidirectional Craniofacial Reconstruction

The face constitutes a rather unique characteristic of our visual appearance and our identity. Its shape is mainly determined by the geometry of the underlying skull and the distribution of facial soft tissue on top of the bony structure. A better understanding—and an efficient computation model—of the relationship between facial skin (*head*), the underlying bony structures (*skull*), and the facial soft tissue thickness (*FSTT*) will bring forward a wide range of applications.

In Chapter 4, we proposed a method for the facial reconstruction from skeleton remains—an important topic in forensic medicine and archaeology. The other way around, i.e., deriving the skull from the face, also has high-potential applications. In a medical context, this technology can estimate the skull of a person based on a 3D face scan only—*without* the need for X-ray radiation or other expensive medical imaging methods. A reasonably accurate, radiation-free alternative would be beneficial, e.g., for patients with craniofacial malformations. Computed Tomography (CT) is currently the standard imaging procedure for such patients [CHP03]. Another application is radiation-free, bony cephalometric skull assessment in orthodontics. In such assessments, both the skull and face shape are often of interest, and a high radiation dose is prohibitive due to the typically young age of the patients [ECSS04].

While there are several approaches for facial reconstruction based on skull remains, we are not aware of any work that reconstructs accurate skull geometry from 3D face scans. Both problems are challenging and have to be regularized by statistical priors from medical imaging data. However, building a dense and accurate model of the correlation between skull, FSTT, and facial skin requires training data that sufficiently samples the *Cartesian product space* of skull shape times FSTT variation. Even with a large number of CT scans this is intractable since it would require measurements of the same individual at several tissue thickness states.

In this chapter, we present such a combined statistical model. We employ a multilinear model that maps from skull shape and FSTT—both represented in low-dimensional parameter spaces—to high-resolution triangle meshes of the skull and the head/facial skin. Varying just the skull parameters generates geometries of different individuals all sharing the same FSTT. Varying the FSTT parameters allows simulating weight changes of a particular individual. Thanks to its multilinear nature, our model can be evaluated as well as fitted in just a couple of seconds, allowing us to produce skull and skin variations from given skull shape parameters and FSTT parameters, or to determine these parameters by fitting the multilinear model to a given skull or skin measured, e.g., by medical imaging or a face scanner.

In order to train the multilinear model, we build on our previous work from Chapter 4: From a set of volumetric CT scans and 3D surface scans of heads/faces, we constructed three individual parametric models of skull shapes, head shapes, and FSTTs thereby decoupling these three models. This allows us to generate high-quality training data by computing face/head meshes from the Cartesian product of variations of skull geometries and variations of FSTTs; thereby, we effectively re-couple the previously decoupled parametric models (Section 5.2.1). The resulting dense sampling of the product space of skull and FSTT variations enables the construction of a multilinear model (Section 5.2.2). We can then fit this model to either given skull scans or face scans in a unified manner (Section 5.2.3). We show the versatility of our novel multilinear model and evaluate its reconstruction accuracy by estimating faces from given skulls as well as skulls from given faces (Section 5.3). Moreover, we made the model publicly available for research purposes.

My Contribution *The proposed multilinear model for bidirectional craniofacial reconstruction was developed in close cooperation with Thomas Gietzen, Robert Brylka, and Ulrich Schwanecke from RheinMain University of Applied Sciences in Wiesbaden. It was further developed in cooperation with Katja zum Hebel, Elmar Schömer, and Ralf Schulze from the Johannes Gutenberg University Mainz. The colleagues from Wiesbaden and Mainz prepared the CT data that were used for our method. I worked on the generation of training data for the multilinear model as well as on the generation of the multilinear model itself. Further, I implemented the approach for fitting the multilinear model. Based on an initial implementation from Thomas Gietzen and Robert Brylka, I also re-implemented the generation and fitting method of the linear model to make it as consistent as possible with the multilinear model. Moreover, I made the multilinear model publicly available for research purposes. Finally, the results about inferring skin surface from skull and vice versa were produced by Thomas Gietzen, Robert Brylka, and me. Furthermore, I worked on producing results for simulating weight changes for face scans.*
Corresponding publication:

[ABG^{+}18] A Multilinear Model for Bidirectional Craniofacial Reconstruction, VCBM, 2018

5.1 Related Work

Methods on skull-based facial reconstruction were already discussed in detail in the previous chapter where we proposed our method for forensic facial reconstruction using a parametric skull model, a parametric FSTT model based on dense FSTT measurements, and a parametric head model. This approach is fully automatic and allows the generation of different plausible head variants utilizing the parametric FSTT model. The current chapter improves upon this previous work by reconstructing not only faces from skulls but also skulls from faces and by being computationally much more efficient thanks to the proposed multilinear model of skull, FSTT, and head.

Reconstructing a skull from skin surface data has a wide range of applications, especially in medicine, but it is still relatively unexplored. The common techniques for reconstructing skulls with high precision are CT and MRI. To the best of our knowledge, there is currently no method that allows the skull structure to be accurately estimated from a face scan alone. A method for reconstructing a coarse approximation of the skull based on the correlation between skin surface, FSTT at few landmarks, and skull was presented in [BB14]. The authors estimated the rigid head transformation in a facial performance by fitting a simplified skull model to the animated face model. Later, Zoss et al. [ZBBB18] extended the skull model by a jaw and employed it for jaw animation. However, their skull model is too simplified to be utilized for medical purposes.

Ali-Hamadi et al. [AHLG$^+$13] presented a semi-automatic method for transferring a volumetric anatomical template model (consisting of bones, muscles, and viscera) to any target character. To map the internal anatomy into the target character, they manually estimate the fat distribution and warp the template by Laplacian deformation while satisfying additional constraints—e.g., that bones must stay straight and symmetric across the sagittal plane. Even if the reconstructed interior follows anatomical rules and gives visually pleasing results, the focus of this approach is to transfer the model to all kinds of targets, like animals or cartoon characters. It does not focus on precisely reconstructing the inner of a human body.

In [KIL$^+$16], a fully automated approach for reconstructing physics-based, anatomical models based on a tetrahedral template mesh representing an average male was presented. To fit the target as closely as possible, the template model was warped through a symmetric as-rigid-as-possible deformation. The work focuses on the reconstruction of large and medium anatomical details, leaving out parts like hands, toes, and the face. These smaller anatomical details are the main component of our current work. Another approach, presented by Ichim et al. [IKNDP16], builds a volumetric face rig based on thickness measurements from forensic studies and employs it for physics-based animation. In [IKKP17], this approach was extended to include a novel muscle activation model that separates active and passive soft tissue layers. Again, all the above approaches are based on skull models that are too simplified to be used for medical purposes.

In the next section, we present a method to generate training data for our multilinear model, generate the multilinear model itself, and fit it to scanner data.

5.2 Multilinear Model

Our goal is to develop a model that (i) maps from skull shape and FSTT distribution—both controlled by low-dimensional parameter vectors $\mathbf{w}_{\text{skull}}$ and $\mathbf{w}_{\text{fstt}}$—to a 3D head/skin surface and (ii) can also invert this map to infer skull and FSTT from a given face scan.

Our parametric PCA models for skull shape (Section 4.2.2) and FSTT (Section 4.4.2) can map skull and FSTT parameters to specific skull and FSTT instances. Adding the FSTT onto the skull through the sphere-mesh representation (Section 4.4.1) and fitting the head model to it (Section 4.4.3) eventually implements the forward mapping. However, this multi-step approach requires about $90\,\mathrm{s}$, which is prohibitive for interactive applications. Also, it cannot easily be inverted. Inspired by previous approaches that have successfully applied multilinear models in the context of faces using separate parameter sets for person identity and facial expressions [VBPP05, BW13, CWZ+14], we generate a multilinear model in the following. Our model can efficiently and robustly compute the head surface from skull and FSTT parameters and vice versa.

5.2.1 Generating Training Data

Multilinear models have to be trained on the full Cartesian product of their independent parameter sets. In our context, this means generating a set of skull shapes and a set of FSTT distributions and as training data, providing each skull shape equipped with each FSTT variation (input) and the respective head surface (output).

It is obviously not possible to collect such data from measurements alone, as it would require multiple CT scans of the same person under different, controlled body weight variations. Our CT scans include different skulls with different FSTT distributions, but the scans do not include their dense Cartesian product. In order to generate training data for our multilinear model, we use the parametric skull model, consisting of a tetrahedral mesh with $M \approx 69\,\mathrm{k}$ vertices, the parametric head model, and the parametric FSTT model from Chapter 4. In contrast to before, 5 pairs of skin surfaces and skulls from CT scans in our database (Section 4.2.1) are used for evaluation. They were therefore excluded from the parametric models. By building independent PCA models for skull shape and FSTT distributions, we effectively decouple these two components. This allows us to subsequently re-couple the components by generating synthetic head models for the Cartesian product of skull shape variation times FSTT variation as outlined above. We thus obtain statistically and anatomically plausible training data.

As a trade-off between computational effort and coverage of input data, we sample skull variations along six PCA-dimensions and FSTT variations along five PCA-dimensions. This covers more than $70\,\%$ of the variation included in our CT data. For each principal component, we sample two offsets at ± 2 standard deviations along that component. Furthermore, we include the mean skull with mean FSTT from our parametric models, which in total yields $2^{5+6} + 1 = 2049$ pairs of skulls and FSTT distributions. Following our approach presented in Chapter 4, we compute the sphere-mesh offset (Section 4.4.1) and fit the head model (Section 4.4.3) for each of these pairs. This takes about $90\,\mathrm{s}$ for each model on a desktop PC with $4 \times 3.6\,\mathrm{GHz}$ Intel Xeon (Figure 5.1). To achieve more accurate head fits, we gain geometric resolution by subdividing our head template, presented in Chapter 4, from about $6\,\mathrm{k}$ vertices to about $24\,\mathrm{k}$ vertices. We also experimented with sampling more PCA dimensions to cover more than $75\,\%$ of the variation in our CT data, but this did not lead to significant improvements in fitting accuracy and did not justify the increased computation effort for a more complex multilinear model.

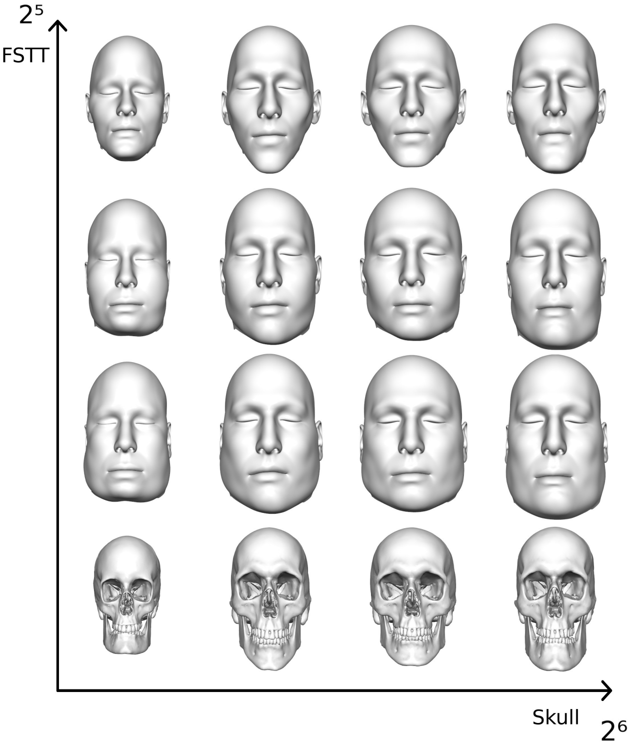


Figure 5.1: We sample skull variations along 6 PCA-dimensions and FSTT variations along 5 PCA-dimensions. For each of these pairs, we compute the sphere-mesh offset and fit the head model. This yields 2048 pairs of skull shapes and head surfaces that were used as training data.

5.2.2 Generating the Multilinear Model

Our training data from the full Cartesian product (without mean) consist of 2048 pairs of skull mesh ($M \approx 69\,\mathrm{k}$ vertices) and skin mesh ($N \approx 24\,\mathrm{k}$ vertices). We stack each into a column vector $\mathbf{X}_i \in \mathbb{R}^{d_{\mathrm{vert}}}$ with $d_{\mathrm{vert}} = 3N + 3M$. These pairs are obtained as $d_{\mathrm{skull}} = 64$

skull variants, each containing $d_{\text{fstt}} = 32$ FSTT distributions. Following [BW13], we center each $\mathbf{X}_i$ by subtracting the model constructed from the mean skull with the mean FSTT from our parametric models, denoted by $\bar{\mathbf{X}}$. Alternatively, the mean out of the 2048 pairs can be subtracted from each $\mathbf{X}_i$.

To construct the multilinear model (MLM in the following), we arrange the 2048 mean-centered geometry vectors $\mathbf{X}_i$ into a three-dimensional array $\mathcal{D} \in \mathbb{R}^{d_{\text{vert}} \times d_{\text{skull}} \times d_{\text{fstt}}}$ which is formally called a third order (3-mode) *data tensor* [VBPP05]. This way, the three *mode spaces* of $\mathcal{D}$ are associated with skin/skull vertex geometry, skull variations, and FSTT variations. This data tensor $\mathcal{D}$ is then decomposed by *higher-order singular value decomposition* [DL97] as

$$\mathcal{D} \;=\; \mathcal{M} \times_{\text{skull}} \mathbf{U}_{\text{skull}} \times_{\text{fstt}} \mathbf{U}_{\text{fstt}}\,,$$

where

$$\mathcal{M} \;=\; \mathcal{D} \times_{\text{skull}} \mathbf{U}_{\text{skull}}^{\mathsf{T}} \times_{\text{fstt}} \mathbf{U}_{\text{fstt}}^{\mathsf{T}}$$

is a *multilinear model tensor* (or *core tensor*) $\mathcal{M} \in \mathbb{R}^{d_{\text{vert}} \times d_{\text{skull}} \times d_{\text{fstt}}}$. $\mathbf{U}_{\text{skull}} \in \mathbb{R}^{d_{\text{skull}} \times d_{\text{skull}}}$ and $\mathbf{U}_{\text{fstt}} \in \mathbb{R}^{d_{\text{fstt}} \times d_{\text{fstt}}}$ are orthogonal matrices containing the left singular vectors of the corresponding mode spaces. If we choose n to be either 'skull' or 'fstt', the matrix $\mathbf{U}_n$ is constructed as follows: We first unfold $\mathcal{D}$ along the *n*-th mode to a matrix $\mathbf{T}_n$ by stacking as columns all vectors of $\mathcal{D}$ aligned with the *n*-th mode. Then, the matrix $\mathbf{U}_n \in \mathbb{R}^{d_n \times d_n}$ can be computed via standard matrix SVD as $\mathbf{T}_n = \mathbf{U}_n \mathbf{S}_n \mathbf{V}_n^{\mathsf{T}}$. For instance, unfolding $\mathcal{D}$ along the skull-mode leads to a matrix $\mathbf{T}_{\text{skull}} \in \mathbb{R}^{d_{\text{skull}} \times (d_{\text{vert}} \cdot d_{\text{fstt}})}$. Given $\mathcal{D}$ and $\mathbf{U}_n$, the *n*-th *mode product* $\mathcal{D} \times_n \mathbf{U}_n^{\mathsf{T}}$ acts on each vector $\mathbf{v} \in \mathbb{R}^{d_n}$ in $\mathcal{D}$'s mode-n space via the linear transformation $\mathbf{v} \mapsto \mathbf{U}_n^{\mathsf{T}} \mathbf{v}$.

Finally, given skull parameters $\mathbf{w}_{\text{skull}} \in \mathbb{R}^{d_{\text{skull}}}$ and FSTT parameters $\mathbf{w}_{\text{fstt}} \in \mathbb{R}^{d_{\text{fstt}}}$, the MLM computes the corresponding combined skin/skull mesh $\mathbf{X} \in \mathbb{R}^{d_{\text{vert}}}$ by tensor contraction as

$$\mathbf{X}(\mathbf{w}_{\text{skull}}, \mathbf{w}_{\text{fstt}}) \;=\; \bar{\mathbf{X}} + \mathcal{M} \times_{\text{skull}} \mathbf{w}_{\text{skull}}^{\mathsf{T}} \times_{\text{fstt}} \mathbf{w}_{\text{fstt}}^{\mathsf{T}}\,. \tag{5.1}$$

This evaluation takes less than a second making the MLM well suited for interactive applications like exploring FSTT variations for a given skull in a forensic context.

5.2.3 Multilinear Model Fitting

The MLM maps skull parameters $\mathbf{w}_{\text{skull}}$ and FSTT parameters $\mathbf{w}_{\text{fstt}}$ to a geometry $\mathbf{X}(\mathbf{w}_{\text{skull}}, \mathbf{w}_{\text{fstt}})$ which includes both the N head vertices and the M skull vertices. Inverting this process means determining the parameters $\mathbf{w}_{\text{skull}}$ and $\mathbf{w}_{\text{fstt}}$ such that the corresponding model $\mathbf{X}(\mathbf{w}_{\text{skull}}, \mathbf{w}_{\text{fstt}})$ closely matches a given geometry observation—which could, for instance, be a face scan or a skull scan extracted from CT. The inverse process therefore amounts to nonrigid registration (or fitting) of the MLM to a given point cloud $\mathcal{P}$.

This fitting procedure requires a coarse initial alignment that can be performed manually (by selecting landmarks) or computed automatically depending on the type of scanner data available [GBA$^+$19, AZB15]. We initialize the MLM as the mean shape $\bar{\mathbf{X}} = \mathbf{X}(\bar{\mathbf{w}}_{\text{skull}}, \bar{\mathbf{w}}_{\text{fstt}})$. Note that setting $\bar{\mathbf{w}}_{\text{skull}} = \mathbf{0}$ is problematic since $\mathbf{X}(\bar{\mathbf{w}}_{\text{skull}}, \bar{\mathbf{w}}_{\text{fstt}}) = \bar{\mathbf{X}}$ holds irrespective of $\bar{\mathbf{w}}_{\text{fstt}}$. Instead, we follow [BW13] and compute $\bar{\mathbf{w}}_{\text{skull}}$ as the average of all rows of $\mathbf{U}_{\text{skull}}$ and compute $\bar{\mathbf{w}}_{\text{fstt}}$ analogously. To speed up the fitting process, we uniformly sub-sample the scanner data $\mathcal{P}$ to approximately $100\,\text{k}$ points without noticeably sacrificing geometric fidelity.

After this initialization, we alternatingly compute closest point correspondences $\mathcal{C}$ between the given point cloud $\mathcal{P}$ and the current state $\mathbf{X}(\mathbf{w}_{\text{skull}}, \mathbf{w}_{\text{fstt}})$ and optimize the model parameters. We again prune correspondences if their distance is too high or their normal vectors deviate too much. Furthermore, we prune correspondences for error-prone areas that we have pre-selected on the template like the teeth, the inner part of the skull, hair, ears, or eye regions.

Given a set of correspondences $(\mathbf{p}_c, \mathbf{x}_c) \in \mathcal{C}$, we minimize their squared distances by optimizing for similarity transform (scaling s, rotation $\mathbf{R}$, translation $\mathbf{t}$) and model parameters $\mathbf{w}_{\text{skull}}, \mathbf{w}_{\text{fstt}}$:

$$E_{\text{fit}}(\mathbf{w}_{\text{skull}}, \mathbf{w}_{\text{fstt}}, s, \mathbf{R}, \mathbf{t}) \;=\; \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \left\| s\mathbf{R}\mathbf{x}_c(\mathbf{w}_{\text{skull}}, \mathbf{w}_{\text{fstt}}) + \mathbf{t} - \mathbf{p}_c \right\|^2 .$$

Here $\mathbf{p}_c \in \mathcal{P}$ is a scanner point and $\mathbf{x}_c(\mathbf{w}_{\text{skull}}, \mathbf{w}_{\text{fstt}})$ its closest point on the current state $\mathbf{X}(\mathbf{w}_{\text{skull}}, \mathbf{w}_{\text{fstt}})$, which is typically located within a triangle and expressed through barycentric coordinates. To prevent over-fitting, we add a Tikhonov regularization term

$$E_{\text{reg}}(\mathbf{w}_{\text{skull}}, \mathbf{w}_{\text{fstt}}) \;=\; \frac{1}{d_{\text{skull}}} \sum_{k=1}^{d_{\text{skull}}} \left(\frac{w_{\text{skull},k} - \bar{w}_{\text{skull},k}}{\sigma_{\text{skull},k}} \right)^2 + \frac{1}{d_{\text{fstt}}} \sum_{l=1}^{d_{\text{fstt}}} \left(\frac{w_{\text{fstt},l} - \bar{w}_{\text{fstt},l}}{\sigma_{\text{fstt},l}} \right)^2 ,$$

with $\sigma^2_{\text{skull},k}$ and $\sigma^2_{\text{fstt},l}$ being the variance of the principal components computed from the covariance matrices after unfolding $\mathcal{D}$ along the respective modes [BW16]. Similarly to [VBPP05], we then minimize the combined objective function

$$E_{\text{mlm}}(\mathbf{w}_{\text{skull}}, \mathbf{w}_{\text{fstt}}, s, \mathbf{R}, \mathbf{t}) \;=\; E_{\text{fit}}(\mathbf{w}_{\text{skull}}, \mathbf{w}_{\text{fstt}}, s, \mathbf{R}, \mathbf{t}) \;+\; E_{\text{reg}}(\mathbf{w}_{\text{skull}}, \mathbf{w}_{\text{fstt}})$$

using block-coordinate descent, i.e., we alternatingly solve for either MLM parameters $\mathbf{w}_{\text{skull}}, \mathbf{w}_{\text{fstt}}$ or pose parameters $s, \mathbf{R}, \mathbf{t}$, while fixing the respective other parameters. This energy minimization is alternated with the computation of new correspondences and iterated until convergence. The process typically takes 3 – 5 iterations and requires about $30\,\text{s}$ on average.

The result of the fitting process are model parameters $\mathbf{w}_{\text{skull}}$ and $\mathbf{w}_{\text{fstt}}$. Through (5.1), $\mathbf{w}_{\text{skull}}$ and $\mathbf{w}_{\text{fstt}}$ can be evaluated to a skin mesh and a skull mesh that closely matches the scanner point cloud $\mathcal{P}$.

5.3 Results

We evaluate our method on 5 different pairs of skulls and corresponding skin surfaces extracted from CT scans that are not included in our training data introduced in Section 5.2.1. We present results for fitting our MLM to scanner data in order to either infer skin surface from skull or vice versa. We compare our MLM with two different approaches: (1) a linear model (LM) created through PCA of the 2049 combined skin/skull pairs $\mathbf{X}_i$ and (2) the forensic facial reconstruction approach [GBA+19] presented in Chapter 4. Note that due to privacy reasons the extracted or reconstructed skin surface can only be shown for one single subject (Figure 4.12, top right).

Generating and Fitting a Linear Model

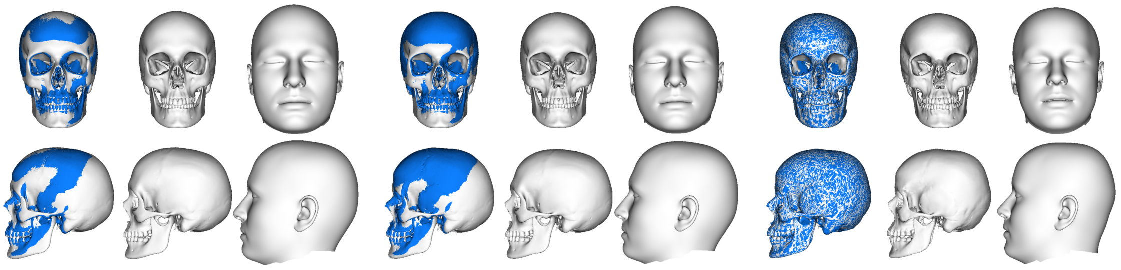
Analogous to the generation of the MLM, we use the 2049 pairs of skin/skull mesh from our synthetic training data to generate the LM. The vertices of each skin mesh and corresponding skull mesh are again stacked into a column vector $\mathbf{X}_i \in \mathbb{R}^{d_{\text{vert}}}$. After subtracting the mean $\bar{\mathbf{X}} \in \mathbb{R}^{d_{\text{vert}}}$ over all training data from each of the $\mathbf{X}_i$, we arrange the resulting mean-centered geometry vectors into a $d_{\text{vert}} \times 2049$-dimensional matrix. PCA of this matrix gives $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_d]$ consisting of the first d principal components. To obtain the same number of degrees of freedom as for the MLM, we chose $d = d_{\text{skull}} + d_{\text{fstt}} = 96$. Given a weight vector $\mathbf{w} \in \mathbb{R}^d$, the LM allows generation of a combined skin surface and skull mesh as

$$\mathbf{X}(\mathbf{w}) \;=\; \bar{\mathbf{X}} + \mathbf{U}\mathbf{w}\,.$$

Fitting the LM to a given face/skull geometry is very similar to fitting the MLM. Again, we distinguish between fitting to scanner data of skin/head and fitting to scanner data of skull. The fitting processes differ in the way their correspondences are computed. Given an initial alignment, we perform a non-rigid registration to estimate the weights $\mathbf{w}$ by minimizing a Tikhonov regularized linear least-squares problem (e.g., similar to Equation (4.4)).

5.3.1 Inferring Skin Surface from Skull

To analyze our skin reconstruction process, we fit both the LM and the MLM to the skulls extracted from our evaluation data set. Figure 5.2 shows skull fitting and skin surface reconstruction results for one specific subject based on the LM (Figure 5.2(a)) or the MLM (Figure 5.2(b)), respectively. The resulting skin reconstruction of the LM is an arbitrary skin surface related to the underlying PCA space and by no means a reconstruction based on the mean FSTT distribution. It is comparable to the MLM if $\mathbf{w}_{\text{fstt}}$ is not adjusted. Because both models are built on the same training data, both reconstructions are visually very similar. Moreover, while fitting the MLM takes about $28\,\text{s}$, fitting the LM takes $10\,\text{s}$ on average.



(a) LM *(b) MLM* *(c) Fine-scale non-rigid fitting [GBA+19]*

Figure 5.2: Comparison of LM (a) and MLM (b) fitting and reconstruction results. Automatic forensic facial reconstruction approach presented in [GBA+19] with fine-scale non-rigid fitting result and facial reconstruction based on mean FSTT for one subject of our evaluation data set (c). Each from left to right: Skull fit (white) with skull extracted from CT (blue) as overlay, skull fit, and skin reconstruction.

Figure 5.2(c) shows a fitted skull and a skin reconstruction based on our approach presented in Chapter 4 [GBA+19]. Since our skull fittings in Figures 5.2(a) and 5.2(b) are constrained by the LM and the MLM, respectively, the result is less accurate compared the fine-scale non-rigid registration of [GBA+19]. The RMS error based on the *skull* evaluation mask (Figure 5.4) results in $0.34\,\mathrm{mm}$ [GBA+19] vs. $1.13\,\mathrm{mm}$ for the MLM. However, the resulting *skin estimations* of both approaches are visually very similar.

Fitting based on the LM has the inherent drawback that there is no control over the FSTT distribution. This results in a single non-changeable skin surface reconstruction. The benefits of the MLM come into play when reconstructing skin surface variants for a specific skull because the MLM allows the generation of different head variants by varying $\mathbf{w}_{\mathrm{fstt}}$. Figure 5.3 shows different head surface variants generated by manipulating the FSTT for a given fixed skull. The presented MLM allows the generation of skin variants nearly in real-time, only at the cost of evaluating (5.1). In contrast, the skin reconstruction process [GBA+19] we presented in Chapter 4 is based on several time consuming steps resulting in a computing time of about $90\,\mathrm{s}$.

5.3.2 Inferring Skull Shape from Face Scan

To analyze the accuracy of our skull reconstruction process, we fitted the MLM and the LM to the extracted skin surfaces from the evaluation data sets. For privacy reasons, we can only show skull reconstructions and not the skin surface fittings. For the evaluation, we create a point mask which is limited to the facial area of the skull. Since our CT data

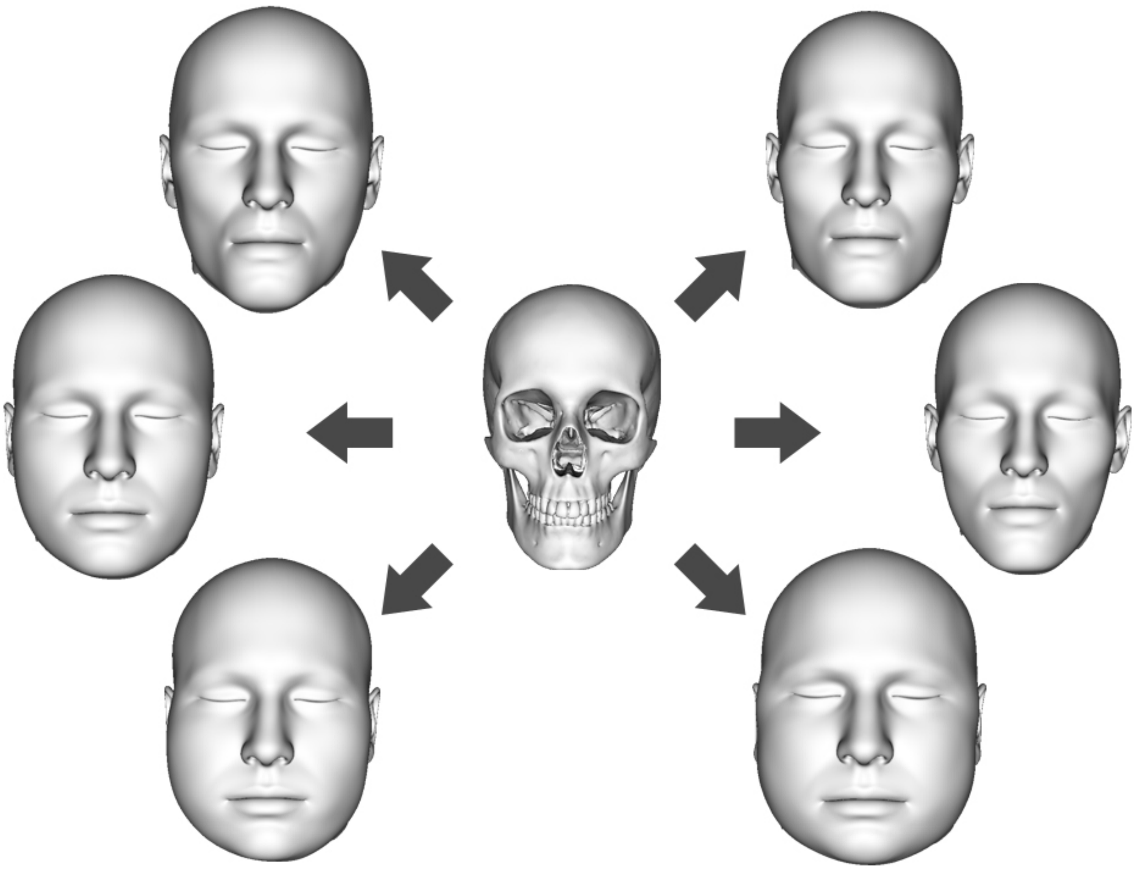


Figure 5.3: Multilinear model fitting: Skin surface variants given a skull. Skin variants can be simply generated by fixing skull parameters $\mathbf{w}_{\mathrm{skull}}$ and varying FSTT parameters $\mathbf{w}_{\mathrm{fstt}}$.

set for creating the FSTT statistics is partially incomplete for the upper part of the skull, we additionally restrict the evaluation mask to the smallest available calvaria part and also exclude teeth. The final evaluation mask used is shown in Figure 5.4. Points of interest are colored in green.

Distance is measured from each point of interest on a reconstructed skull to the surface of the corresponding extracted skull. The average RMS fitting error over all 5 reconstructed skulls is $1.72\,\mathrm{mm}$ using the MLM and $1.85\,\mathrm{mm}$ using the LM. As can be seen clearly in Figure 5.5, both models not only allow reconstruction of the correct size of the skull, but they also correctly reproduce the shape of the skull, in particular the emplacement of the mandibular. For both models, the RMS fitting error is below $2\,\mathrm{mm}$. While reconstructing skulls from given skin surfaces using the MLM gives slightly better results, it takes about $30\,\mathrm{s}$ compared to $8\,\mathrm{s}$ using the LM.

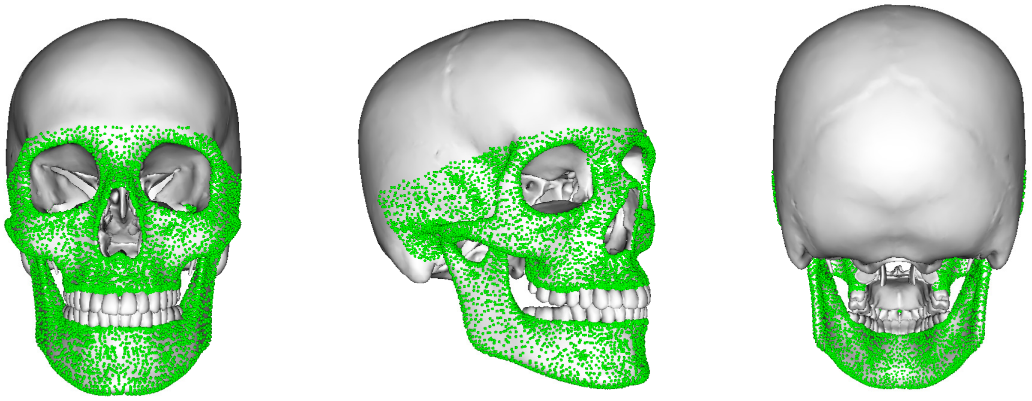


Figure 5.4: Evaluation mask for skull reconstructions (green).

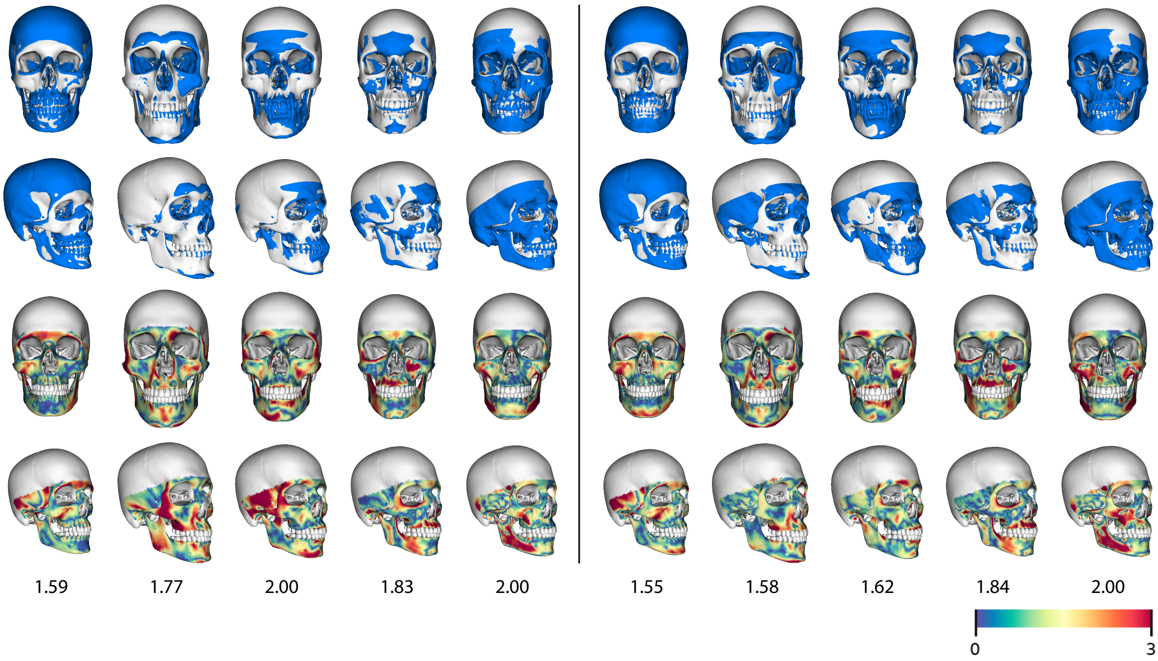


Figure 5.5: Skull reconstructions given skin surface scans from our 5 evaluation data sets. Reconstruction results for the linear model (left) and for the multilinear model (right). For each model from top to bottom: Skulls extracted from CT (blue) and our skull reconstructions (white) as overlay, the minimal distance to the actual skull, and the RMS errors in mm.

5.3.3 Simulating Weight Changes for Face Scans

Fitting the MLM to a face scan reveals not only the skull parameters $\mathbf{w}_{\text{skull}}$ but also the FSTT parameters $\mathbf{w}_{\text{fstt}}$ of the scanned individual. Given the skull shape and FSTT distribution of the person, we can simulate weight changes by varying $\mathbf{w}_{\text{fstt}}$.

Since the MLM does not reconstruct hairs or eyes, we start to obtain a realistic head reconstruction by fitting a head template to a photogrammetric face scan using our method from Chapter 3. This head template has the same triangulation as the template used in Section 4.4.3, but it additionally has open eyes, eyeballs, and teeth. Furthermore, the nonlinear fine-scale deformation allows a reasonable reconstruction of hair geometry (Figure 5.6, top row).

Since both models (the realistic face model and the MLM) were fitted to the same scanner data, they are well aligned to each other. When changing the FSTT of the scanned person from $\mathbf{w}_{\text{fstt}}$ to $\tilde{\mathbf{w}}_{\text{fstt}}$, we can therefore simply transfer the per-vertex *displacement*

$$\mathbf{X}(\mathbf{w}_{\text{skull}}, \tilde{\mathbf{w}}_{\text{fstt}}) \;-\; \mathbf{X}(\mathbf{w}_{\text{skull}}, \mathbf{w}_{\text{fstt}})$$

computed by the MLM onto the realistic face model. We thereby obtain the thinner or thicker models shown in Figure 5.6. In concept, this is similar to clothing transfer as proposed in Section 3.5.2. Finally, the positions of the eyeballs are adjusted to accommodate the slight displacements in the eyelids that result from the simulated weight changes.

In this chapter, we presented a multilinear model that maps a set of low-dimensional parameters for skull shape and FSTT distribution to an accurate and high-quality mesh of both the skin and the skull geometry. To foster further research in this direction, we made our multilinear model publicly available for research purposes at doi:10.4119/unibi/2930619. We demonstrated that our model has several interesting and high-potential applications in the medical context. This finalizes the second part of this thesis. We will continue with exploring virtual humans for domestic assistance.

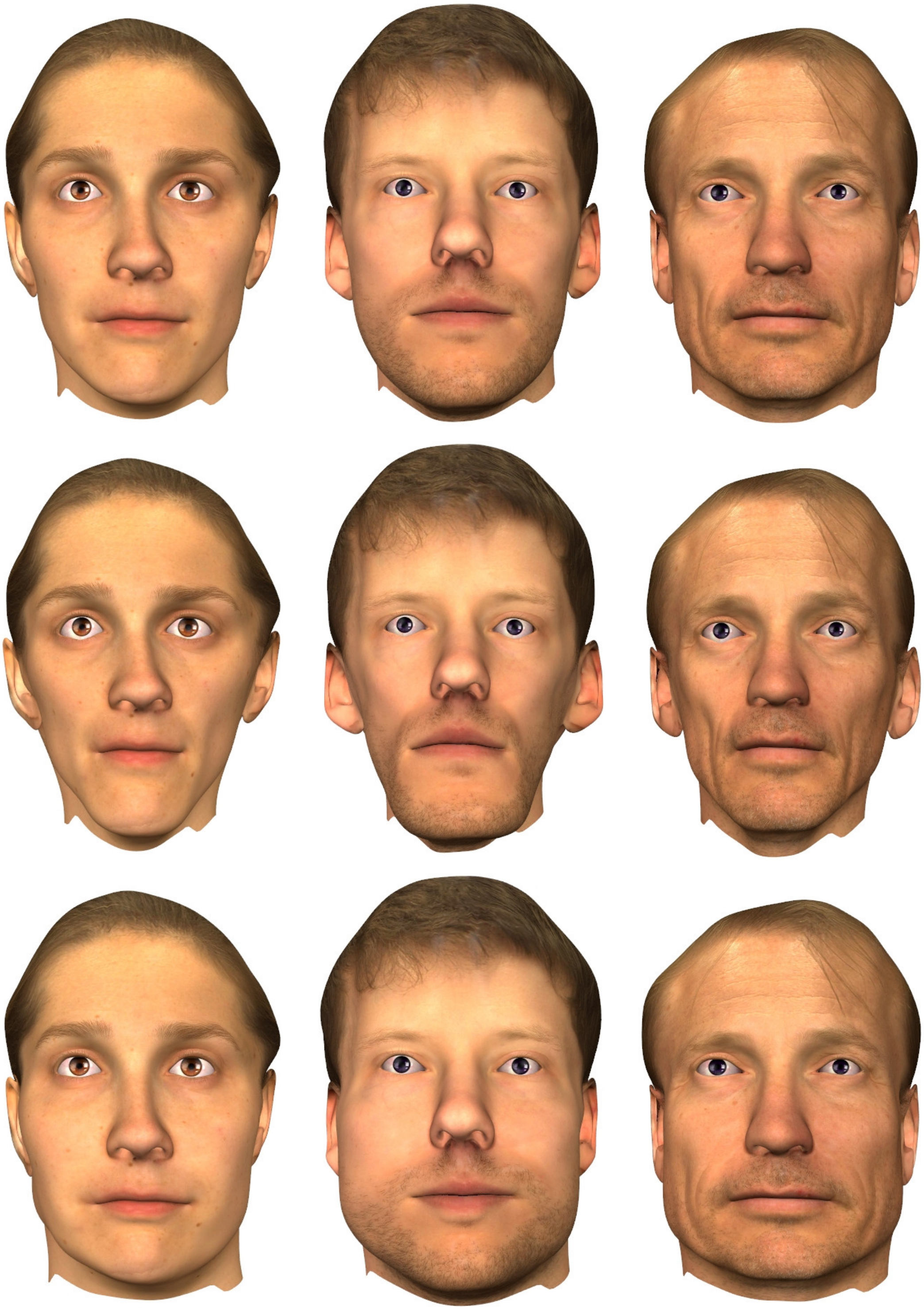


Figure 5.6: The multilinear model makes it possible to vary FSTT for a specific individual. Three scanned persons reconstructed by [AWLB17] (first row) with varied FSTT (second and third row).

Part III

PREFERENCES FOR VIRTUAL ASSISTANTS

6 Preferences for the Visual Attributes of Virtual Assistants

Conversational virtual agents that serve as assistive technologies have made their way into users' homes [Gmb18, ANA10, DvM00], facilitating their lives by providing information services, e.g., by obtaining information from the Internet [YKPK13]. Such virtual assistants will soon be used by demographically diverse target groups with personal needs and preferences for activities like cooking, planning leisure time, or physical rehabilitation. Despite the fact that virtual assistants will be widely available to provide services in a vast variety of domains, the fact remains that people report rather negative attitudes toward service robots and show little willingness to integrate them into their everyday lives [Gmb18, RE13, SBE$^+$16]. Thus, to increase users' acceptance of innovative technologies, user preferences have to be taken into account [SPC$^+$16, Nie94]. Previous research has indicated that virtual humans that assist in the smart home context should appear likeable, attractive, and competent [BM07, JB09, BEK12]. Moreover, the appearance of virtual assistants and even robots strongly influenced their evaluation [BM07, BEK12, HAAH02] and, consequently, had an impact on user motivation and performance [SSJ$^+$15]. For instance, human-like virtual assistants were considered more intelligent [KO96], more skilled [JVH08], were evaluated more positively [QB05], and elicited stronger social presence [CMB01] than virtual assistants that did not resemble humans. One aspect of human-likeness is hairstyle and hair color—features that have been varied frequently in research on the design of virtual assistants [RKBPD08, Gar00]. Moreover, hairstyle is a key facial feature indicating target gender—even in robots [BP93, EH12]. Thus, a robot's or a virtual agent's hairstyle may activate stereotypes and expectations regarding the agent's usefulness, credibility, and intelligence [Vel10, MMB94, HS96, Dev89]. Similarly, clothing may represent a virtual agent's social role, influencing judgments accordingly [Vel10]. Veletsianos has shown that students were better at learning to play punk music when the assistant's appearance was in line with the stereotypical appearance of a punk rock musician (i.e., featuring a Mohawk hairstyle) rather than being taught by a virtual assistant that was allegedly a scientist. That is, performance was better when the assistant's appearance and task type matched [GKP03]. Similarly, Rosenberg-Kima [RKBPD08] showed that perceived enjoyment, trust, and anxiety during a learning session with a virtual assistant depended on whether the assistant was portrayed as a peer or an expert. Because previous work lacks a differentiated analysis of preferences of demographically diverse users, we conducted a laboratory study to close this research gap. More specifically, we analyzed preferences of users who differed in terms of age, gender, and even hair color with regard to virtual assistants that were deployed in various smart home contexts.

Individual user preferences were determined with an adaptive choice-based conjoint analysis. Here, we systematically varied gender, age, hair color, hair length, and clothing of the virtual assistants that were used as stimuli. The virtual assistants were featured in a variety of contexts within a smart home. More specifically, they were depicted as assistants for planning leisure activities (*Leisure*), helping in the kitchen (*Kitchen*), coaching fitness exercises (*Fitness*), and providing support in the entrance hall (*Entrance*) to cover a wide array of contexts in which the assistants could potentially be used.

Furthermore, existing research has shown that individuals identify more strongly with virtual humans that they deem similar to themselves [BBG08, HB05, MK01]. For example, perceived similarity between users and virtual assistant's appearance influenced users' motivation to engage in a fitness program [MF06, Bay11]. Thus, the degree to which persons perceive a virtual assistant similar to themselves seems to be an interesting potential mediator [LFDK07]. According to the similarity-attraction hypothesis, a high degree of similarity predicts liking. In line with this, previous research found that female participants preferred interacting with other females [QB10, DWYW09, RG04]. Whereas research by Payne et al. [PSRJ13] confirmed this for female participants and female virtual agents, male users did not prefer same-sex virtual assistants. Another study investigated preferences for virtual agents' gender in senior participants [CKGIS11]. In that study, there was no specific preference for agent gender for the majority of senior participants. Among those who had a gender preference, more participants preferred male agents than female ones. However, as the authors stated, the main reason for that was the audibility of the voice. Based on users' preferences and a determined virtual assistant for each participant, we examined the evaluation of the virtual assistant with respect to perceived similarity between the users and the virtual assistants, with respect to the virtual assistant's appearance, and as a function of task domain. We analyzed whether the users chose a peer-like assistant, inspired by similarity-attraction hypothesis, or an expert-like assistant, who may be younger or older than themselves. That is, to a senior user, a relatively young virtual assistant may be seen as an expert in sports, whereas a younger user might regard a senior assistant as an expert in cooking. Complementing previous research, we considered diverse participants that differed in age groups, gender, and hair color. This way, we were able to comprehensively examine the effect of perceived similarity to the evaluation of virtual assistants.

Drawing on previous research on the perception of intelligent virtual agents, *warmth* and *competence* are the two core dimensions of social cognition [FCG07] and play a key role in impression formation about humans and non-human entities [BEK12]. While the dimension of warmth captures friendliness and positive intentions, the dimension of competence captures economical and educational success [FCG07]. In this study, we also examined how the individually determined preferred virtual assistants were perceived in terms of their warmth and competence. Moreover, we investigated the effect of openness toward technology on the evaluation of the preferred virtual assistants.

My Contribution *The present study was carried out in cooperation with Friederike Eyssel, Charlotte Diehl, Birte Schiffhauer, Ralf Wagner, and Stefan Kopp. We all conceived the experiment, and I implemented it. Specifically, I developed an approach to generate* 176 *different virtual assistants that were used as stimuli. Afterwards, the experiment was conducted by Charlotte Diehl, Birte Schiffhauer, student assistants, and me. Finally, Charlotte Diehl analyzed the results concerning our hypotheses, while I analyzed the user preferences for virtual assistants based on the adaptive choice-based conjoint analyses.*
Corresponding submission:

> *Preferences of different user groups for the visual attributes of virtual assistants, TAP, under submission*

6.1 Experiment

Preferences for Visual Attributes of Virtual Assistants

We explored user preferences for basic visual attributes of virtual assistants in smart home contexts. To this end, we grouped users of different age, gender, and hair color and examined their preferences for different smart home contexts. In this study, we focused on the following questions:

- How important are the selected customization categories, i.e., gender, age, hair color, hair length, and clothing?

- Which visual attributes of assisting virtual humans are preferred, and how are these attributes related to users of different age, gender, and hair color for different contexts? Do people prefer characteristics that are similar to themselves, i.e., do people choose a virtual assistant that features the same gender, age, and hair color as themselves?

Hypotheses

1. The more participants perceive the virtual assistant as being similar to themselves, the more positive the evaluation of the virtual assistant and its design is.

2. Female virtual assistants are perceived as warmer than male, whereas male virtual assistants are evaluated higher on competence and persuasive power compared to female ones.

3. Participants' openness toward technology moderates the acceptance of the virtual assistants.

6.1.1 Method

Sample and Design

$N = 131$ participants were recruited on campus, in schools, sport clubs, open youth clubs, nursing homes, and via advertisements in a local newspaper. 68 participants were female, 62 were male, and one participant did not indicate his/her gender. They ranged in age from 6 to 89 years ($M_{\text{age}} = 37.49; SD_{\text{age}} = 20.51$). In order to assure diversity of age groups, we distinguished three subgroups: young = under 21 years, middle-aged = between 30 and 45 years, and senior users, represented by persons older than 55 years. Considering hair color, 29 participants had blond hair, 67 had brown hair, and 24 had gray hair. Most of the participants ($88.5\,\%$) had little to no previous experience with the customization of a virtual assistant. The exact sample sizes grouped according to age category are displayed in Table 6.1.

age categories	female participants	male participants	total
young (< 21 yrs.)	n = 23	n = 20	n = 43
Mean (*SD*)	15.48 (3.50)	16.20 (4.48)	15.81 (3.95)
middle-aged (30 - 45 yrs.)	n = 21	n = 20	n = 41
Mean (*SD*)	32.62 (3.56)	33.50 (4.16)	33.05 (3.84)
senior (>55 yrs.)	n = 19	n = 21	n = 40
Mean (*SD*)	63.37 (7.71)	65.67 (5.61)	64.58 (6.70)

Table 6.1: Sample size and descriptive statistics concerning participant age and gender.

Seven participants who did not fit the predefined age categories or did not indicate his/her gender were excluded from analyses on age and gender, respectively, but their data were considered in the analyses on participants' hair color or in overall calculations.

Procedure

After giving written informed consent, participants were asked to imagine living in an intelligent apartment that would include a virtual assistant. In one out for four scenarios, this assistant would serve them as a virtual coach for fitness exercises. For example, participants learned that the virtual assistant would instruct them while learning new exercises, would remind them of training goals, and would provide information on their general health status. Similarly, participants were instructed that the virtual assistant would support them in the kitchen, help them plan leisure activities, and provide support upon leaving or entering the home in the entrance hall. With an adaptive choice-based conjoint analysis user preferences were determined. This procedure resulted in one particular virtual assistant for each context that was thereby tailored to fit the individual user preferences in terms of the virtual assistants' gender, age, hair color, hair length, and clothing. Participants were then presented with their preferred choice using a short video clip showing the virtual assistant performing an introducing animation with computer-generated gesture as well as with a gender- and age-specific synthetic voice. Subsequently, the participants had to evaluate the virtual assistant using a questionnaire. The selection process and evaluation of the virtual assistant took about 45 minutes. After completion of the study, participants were debriefed, reimbursed for their participation, and dismissed. Procedures were approved by the Bielefeld University Ethics Committee under the approval number No 2016-029 (date: March 4, 2016) and are in accordance with the guidelines and regulations of the German Society for Psychology (DGPs).

6.1.2 Design of the Virtual Assistants

Using the web-based service Autodesk Character Generator [Aut14], we semi-automatically generated 176 different virtual assistants which differed in gender, age, hair length, hair color, and clothing. To this end, eight "basis prototypes" were built manually, taking into account both gender (male, female) and age (child-like, adolescent, middle-aged, senior) of the virtual character (Figure 6.1). The further procedure was automated so that different clothing styles, different hair colors, and different hair lengths were added automatically to each of the basic prototypes at a time.

The virtual assistants featured three hair colors, i.e., blond, brown, and gray hair, with either short or long hairstyles. We omitted child-like virtual assistants with gray hair. The assistants were either dressed formally, casually, or using clothes based on the domain of use. That is, a virtual fitness coach or a virtual chef wore apparel that matched their particular profession. For instance, while the virtual cooking assistants featured a white top and a white chef's hat, the virtual fitness assistants featured a white sports jersey. The eight basic prototypes, presenting virtual assistants with different ages and gender, as well

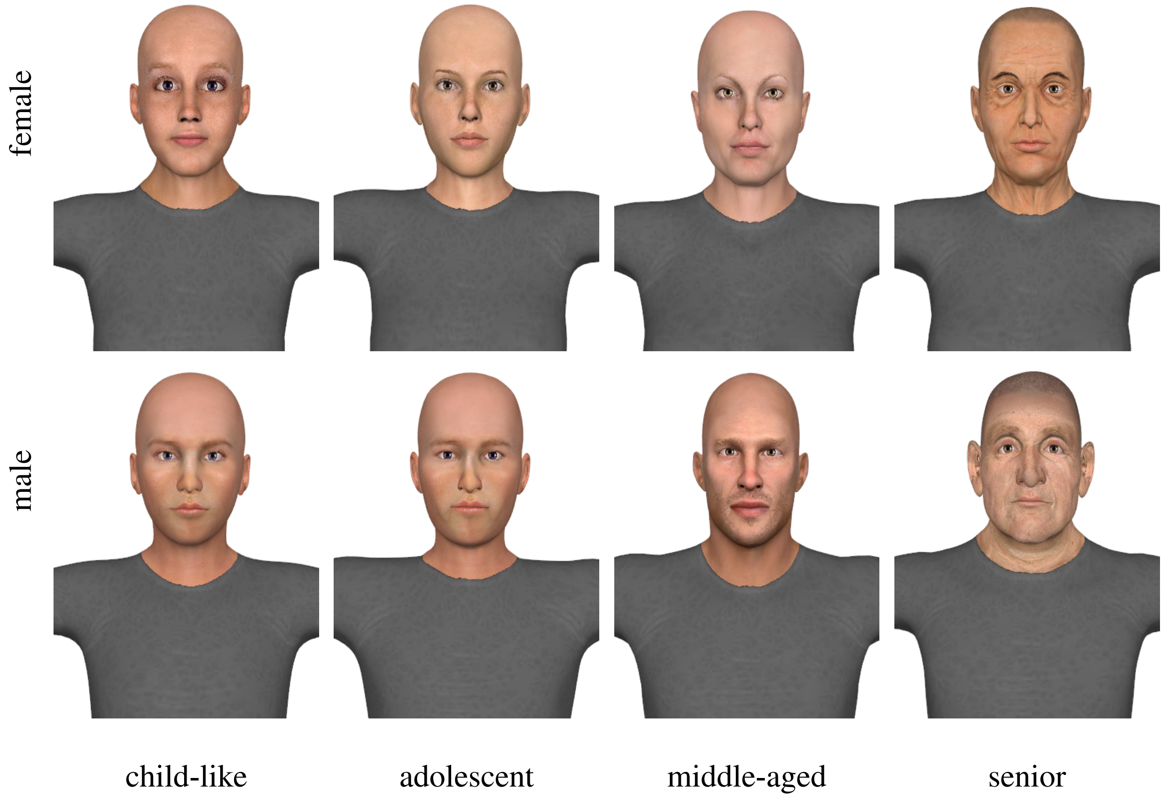
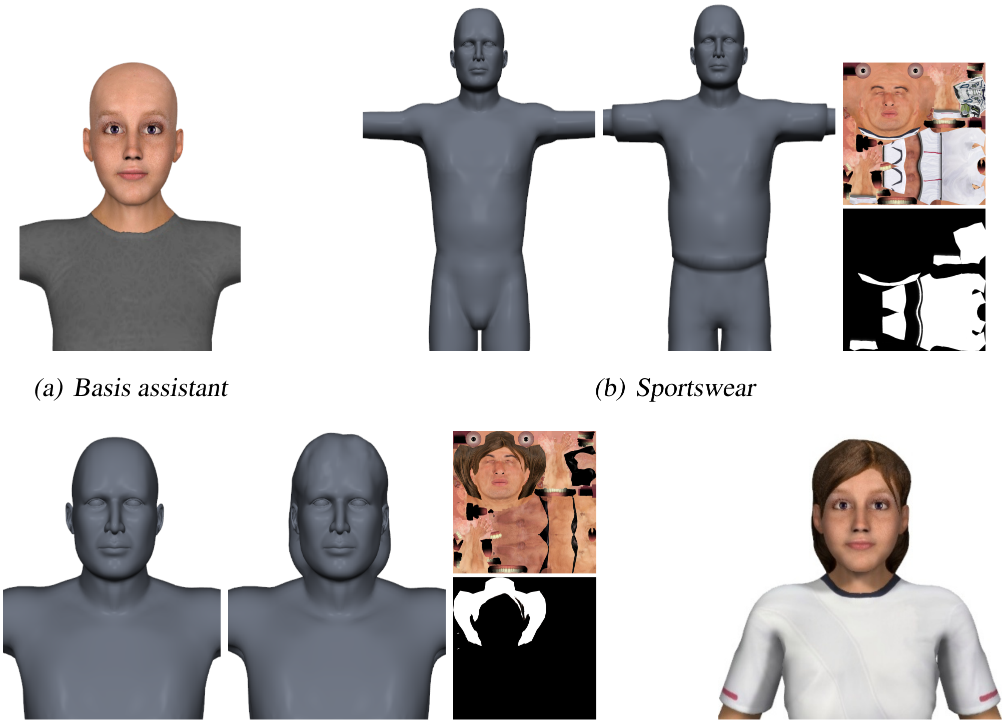


Figure 6.1: Eight virtual assistants of different gender and age were built manually and augmented to generate 176 variants.

as the clothing and hairstyles were chosen by eight people who reached a consensus on appropriateness of the visual attributes.

For one out of eight basis assistants (Figure 6.2(a)), we demonstrate how to add clothing and hairstyles. The underlying concept is very similar to clothing transfer from Section 3.5.2. When adding clothing, we distinguish between adding texture and adding geometry (Figure 6.2(b)). For the former, the regions of the textures (albedo-, normal-, specular maps) that represent the clothing-of-interest are just copied onto our character's textures. This requires a manually pre-created mask for each piece of clothing. For adding geometry, the difference between a model with clothing-of-interest and the same model without clothing is added to our character that is to be dressed. In particular, we encode the difference by deformation gradients and add it by deformation transfer. This process is similar to the generation of facial blendshapes as presented in Section 3.4.3.

For adding a hairstyle and setting a hair color, we discern whether the hair from our database is modeled as a separate mesh (wig) or not. In case of no wig, adding hair is analog to adding clothing (Figure 6.2(c)). This means that we add the difference between



(a) Basis assistant

(b) Sportswear

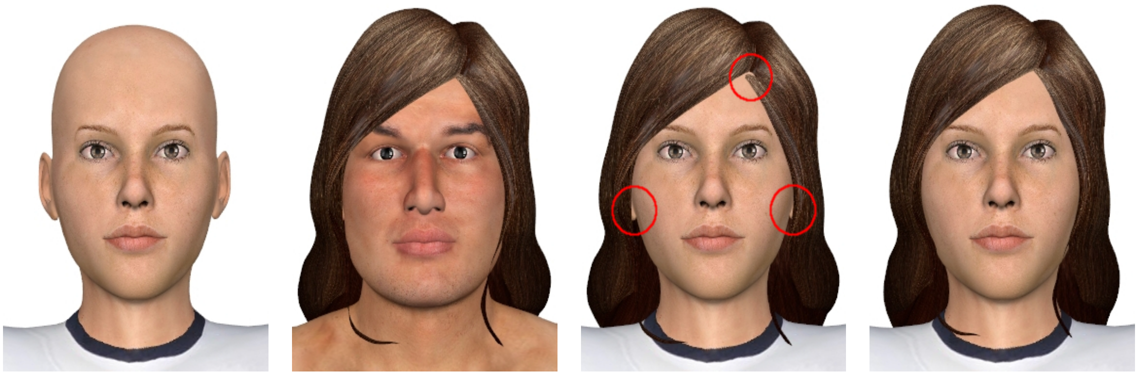
(c) Hairstyle model (long, brown, no wig)

(d) Final assistant

Figure 6.2: We dress a basis assistant (a) by adding the difference between a model with clothing-of-interest and the same model without clothing (b). The regions of the textures that represent the clothing are just copied. Similarly, we add hair (c) and set a default posture for our final assistant (d).

a model with hair of interest and the same model with a bald head to our character that we want to add a hairstyle to. As before, the regions of the textures that represent the hairstyle of interest are just copied to our character's textures based on a manually pre-created mask. Note that we distinguish albedo maps for different hair colors that can be added. In case of a wig, the idea is to add the separate wig mesh from the hairstyle model to our current character and adjust the geometry of the wig (Figure 6.3) by computing a *space deformation* field. This is based on radial basis functions [BKP$^+$10] and interpolates the deformation from the hairstyle model to our current character at vertices of the head. By this deformation field, we transform the wig to fit it to our current character. The texture of this separate wig mesh is simply taken from our database of different hair colors.

In case of generating a virtual chef, the chef's hat was added and adjusted automatically by optimizing position, orientation, and scaling based on pre-selected vertices. Subsequently, the hair was non-rigidly deformed to lay under the chef's hat and regularized by the Laplacian energy (3.3).



(a) Basis assistant wearing sportswear (b) Hairstyle model (long, brown, wig) (c) Wig not adjusted (d) Final assistant

Figure 6.3: We add a separate wig mesh from the hairstyle model (b) to our dressed character (a) and adjust the geometry of the wig (c–d).

Since all virtual assistants are equipped with a skeleton, we are able to automatically set a default posture (Figure 6.2(d)). Further, for each of these virtual assistants, both images and video clips were generated automatically. The videos featured a short animation including computer-generated gesture and speech. For each of these virtual assistants and for each context, a video clip was generated that showed the virtual assistant introducing itself as a personal assistant. The clip also briefly depicted several opportunities for assistance in that specific context. Note that the images and video clips only show the upper body. See Figure 6.4 for a representative subset. An overview of all virtual assistants that were used as stimuli can be found in the appendix.

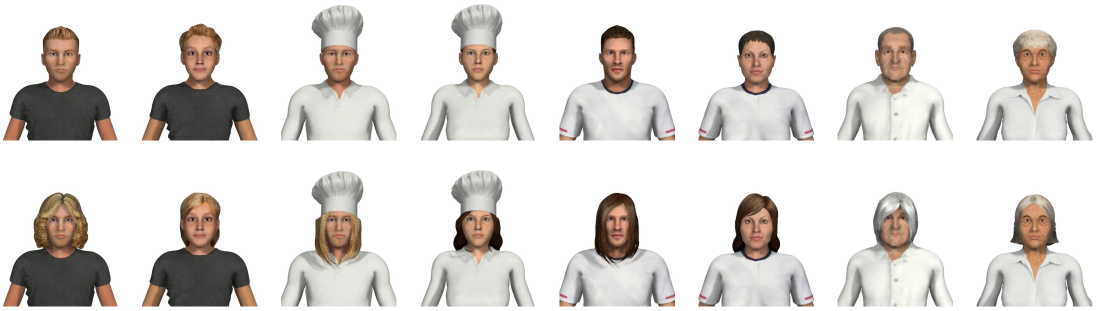
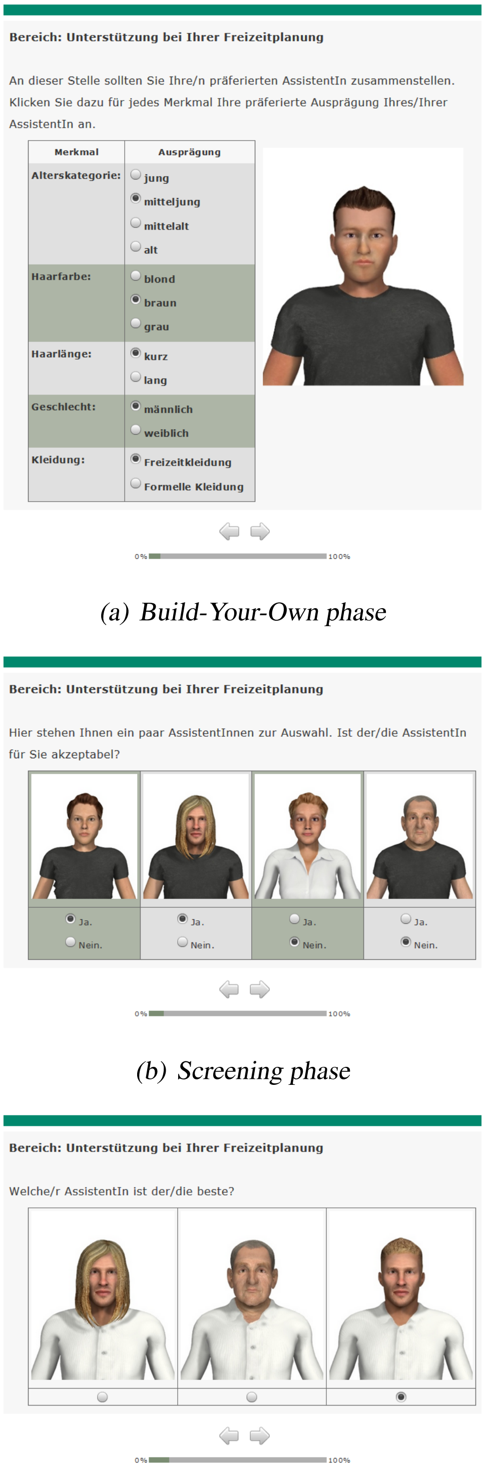


Figure 6.4: Design of the virtual assistants. A representative subset of 176 different virtual assistants that were designed and used as stimuli.

6.1.3 Adaptive Choice-Based Conjoint Analysis

To determine the preferred virtual assistant, we utilized adaptive choice-based conjoint analysis (ACBC) [CDC10]. Generally, conjoint analysis is a decompositional statistical approach that represents a de facto standard in market research to determine preferences of customers regarding particular features of products in a holistic way [GS78, GKW01]. ACBC is framed as a choice exercise where the respondent chooses the most preferred target from a set of competing alternatives. According to Cunningham et al. [CDC10], the ACBC approach is considered to be realistic, appealing to the participants, and seems to yield good predictions. In the current study, the adaptive procedure enables assessment of the large number of images of virtual assistants. We implemented the whole study with the proprietary software Lighthouse Studio [Sof17].



Bereich: Unterstützung bei Ihrer Freizeitplanung

An dieser Stelle sollten Sie Ihre/n präferierten AssistentIn zusammenstellen. Klicken Sie dazu für jedes Merkmal Ihre präferierte Ausprägung Ihres/Ihrer AssistentIn an.

Merkmal	Ausprägung
Alterskategorie:	○ jung ● mitteljung ○ mittelalt ○ alt
Haarfarbe:	○ blond ● braun ○ grau
Haarlänge:	● kurz ○ lang
Geschlecht:	● männlich ○ weiblich
Kleidung:	● Freizeitkleidung ○ Formelle Kleidung

(a) Build-Your-Own phase

Bereich: Unterstützung bei Ihrer Freizeitplanung

Hier stehen Ihnen ein paar AssistentInnen zur Auswahl. Ist der/die AssistentIn für Sie akzeptabel?

● Ja. ○ Nein.	● Ja. ○ Nein.	○ Ja. ● Nein.	○ Ja. ● Nein.

(b) Screening phase

Bereich: Unterstützung bei Ihrer Freizeitplanung

Welche/r AssistentIn ist der/die beste?

○	○	●

(c) Choice Task phase

Figure 6.5: Three phases of an adaptive choice-based conjoint analysis.

Work flow

The ACBC consists of three phases, called *Build-Your-Own*, *Screening*, and *Choice Task*. During the Build-Your-Own phase, the participants have to build their own presumably preferred virtual assistant for the current smart home context (Figure 6.5(a)). Participants could customize the virtual assistant by individually selecting gender, age, hair color, hair length, and clothing. Then, a picture of the upper body of the corresponding virtual assistant was displayed. At this point, participants were still able to change characteristics again and thus could choose for a different virtual assistant. Based on the specified answers, a pool of virtual assistants was created. To limit all possibilities to relevant ones, characteristics were relatively concentrated around the specified characteristics, although every characteristic was included.

During the Screening phase, four virtual assistants were shown at the same time (Figure 6.5(b)). Here, the participants had to decide if the proposed virtual assistants would come into question for acting as virtual assistants or not, respectively. We showed eight of such screens to ensure that enough virtual assistants were assessed so that we could derive strong individual utility estimates. During the Screening phase, a participant may have consistently avoided some levels of an attribute. In this case, we asked whether any of these avoided levels was a so-called *Unacceptable*. The participant could then mark the one characteristic that was most unacceptable or mark none. Later on, depending on the responses, another unacceptable screen may have been shown. All unacceptable levels that were specified by a participant were no longer displayed. Similarly, the virtual assistants that were marked as coming into question for being virtual assistants may have contained certain characteristics. In this case, we asked if that level is a *Must-Have*, so that the participant could mark the most important feature that is a Must-Have or mark none. Again, depending on the responses, another Must-Have screen may have been shown where the participant could decide for another Must-Have level. All Must-Have levels that were specified by the participant were shown in the following screens. If no virtual assistant was possible, the ACBC stopped for that specific smart home context.

During the Choice Task phase, the participants were shown a series of choice tasks in groups of three (Figure 6.5(c)). Here only the "surviving" virtual assistants that were considered a possibility during the Screening phase and that were conform to Unacceptable and Must-Have rules were shown. In subsequent rounds of the tournament, the winning virtual assistant from each triple competed until the overall preferred assistant was found.

6.1.4 Dependent Variables

Participants evaluated four different virtual assistants (i.e., one for each context) on a variety of dependent measures. Responses were provided using seven-point Likert scales (from 1 = not at all to 7 = very much). For the four contexts *Leisure*, *Kitchen*, *Fitness*, and *Entrance*, participants evaluated the individually determined preferred virtual assistant on attributed warmth, competence, and persuasiveness. Perceived similarity with the virtual assistant was assessed using the item "The assistant is similar to me." Participants further indicated their intention to use the virtual assistant and how much they would like to have it at home. They reported their satisfaction with the realization of the virtual assistant using three items (satisfaction with the appearance, the voice, and the animation of the assistant). These items read "How satisfied are you with the appearance of your virtual assistant?", "How do you feel about the animation of your virtual assistant?", and "How do you feel about the voice of your virtual assistant?". The items were rated using seven-point Likert scales (from 1 = robotic to 7 = human). At the end of the study, participants reported their individual openness toward technology using 13 items from Neyer, Felber, and Gebhardt [NFG12]. An example item read: "I would like to use new technology more often." Finally, participants indicated their gender, age, and German language skills.

6.2 Results

In the following, we will first present our results on preferences for visual attributes of virtual assistants. Thereafter, we give recommendations for the design of virtual assistants in smart home environments and continue with answering our hypotheses.

6.2.1 Preferences for Virtual Assistants

Importance of the Customization Categories

To test whether the customization categories gender, age, hair color, hair style, and clothing differ in their importance for the total utility of the final virtual assistant, we considered conjoint importance characterizing the relative importance of each category and conducted paired t-tests for each of the four contexts using Bonferroni adjustments. Descriptive statistics are summarized in Figure 6.6 (for details see Table 1 from the appendix). Note that conjoint importance is ratio data. Thus, e.g., the category age in the context *Leisure* with an importance of $46.82\,\%$ is more than twice as important as the category hair color with an importance of $22.36\,\%$. Age of the virtual assistant turned out to be the most important customization category for all contexts, $ts(130) \geq 13.50, ps < .001$.

Preference for Female vs. Male Virtual Assistants

When determining preferences of participants by ACBC, so-called individual *part-worth utilities* were estimated. Part-worth utilities are interval data that quantify the participants' preferences for each characteristic of each attribute and are to be preferred over the individually determined preferred virtual assistants. However, analyzing pure part-worth utilities might seem somewhat abstract. Fortunately, there are more powerful ways to analyze preferences by choice simulations that are based on part-worth utility estimates. We analyzed participants' preferences in different groups and for different smart home contexts by conducting choice simulations. In this chapter, all choice simulations were conducted by the *First Choice* method [LRDB06]. This method assumes that users will choose the product that has the highest overall utility. In our context, this method simulated the decision for the most preferred virtual assistant for many simulated users and is thus similar to the

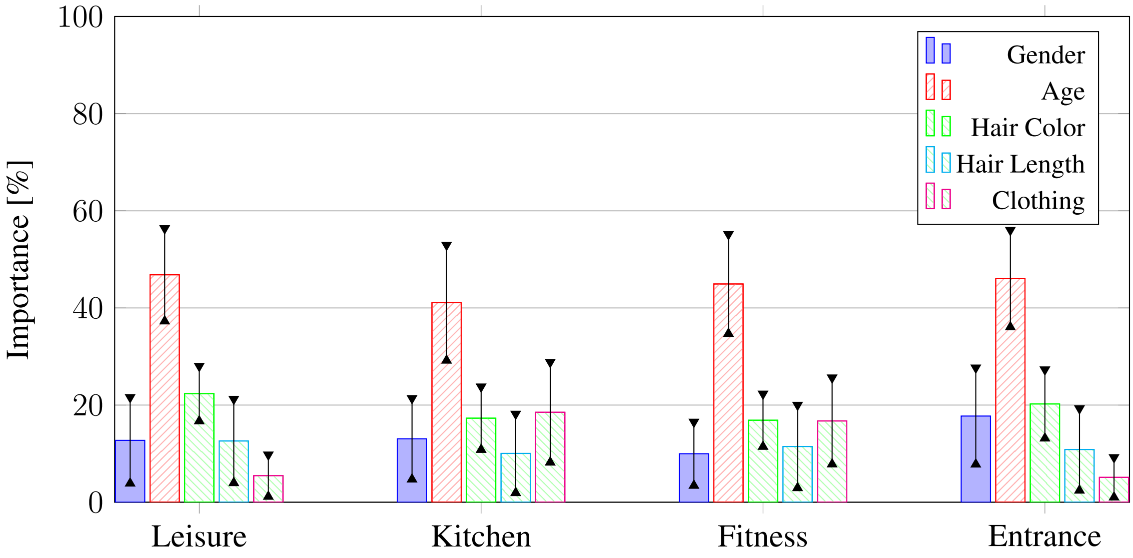


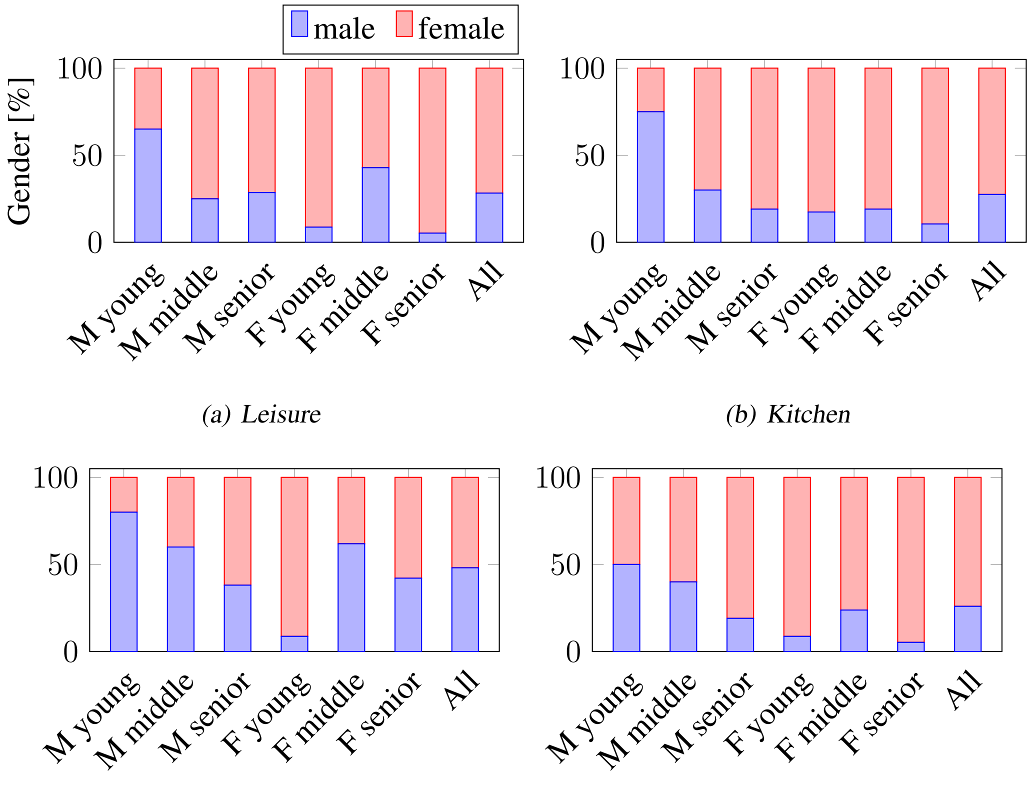
Figure 6.6: Importance of the customization categories. Means and standard deviations of the importance of the customization categories grouped according to context.

decision process of a real end-user for a preferred virtual assistant. Following the advice from Lighthouse Studio, we communicate the results of our conjoint analyses using choice simulators.

We distinguished male and female participants and coded participants' age as young, middle-aged, and senior (cf. Table 6.1). See Figure 6.7 for the choice simulations' output (for details see Table 2 from the appendix).

On a descriptive level, there was an overall preference for female virtual assistants. Further, we observed that young male participants preferred male virtual assistants equally or more than female virtual assistants in all contexts. However, middle-aged and senior male participants mostly preferred female virtual assistants. There is one exception for the context *Fitness*. Here, middle-aged male participants preferred male virtual assistants. We also observed the trend that, on average, male participants increasingly preferred female assistants with increasing age.

In contrast, the majority of female participants preferred a same gender assistant. An exception was the context *Fitness*. Here, middle-aged female participants prefer male virtual assistants, too (Figure 6.7).



(a) Leisure

(b) Kitchen

(c) Fitness

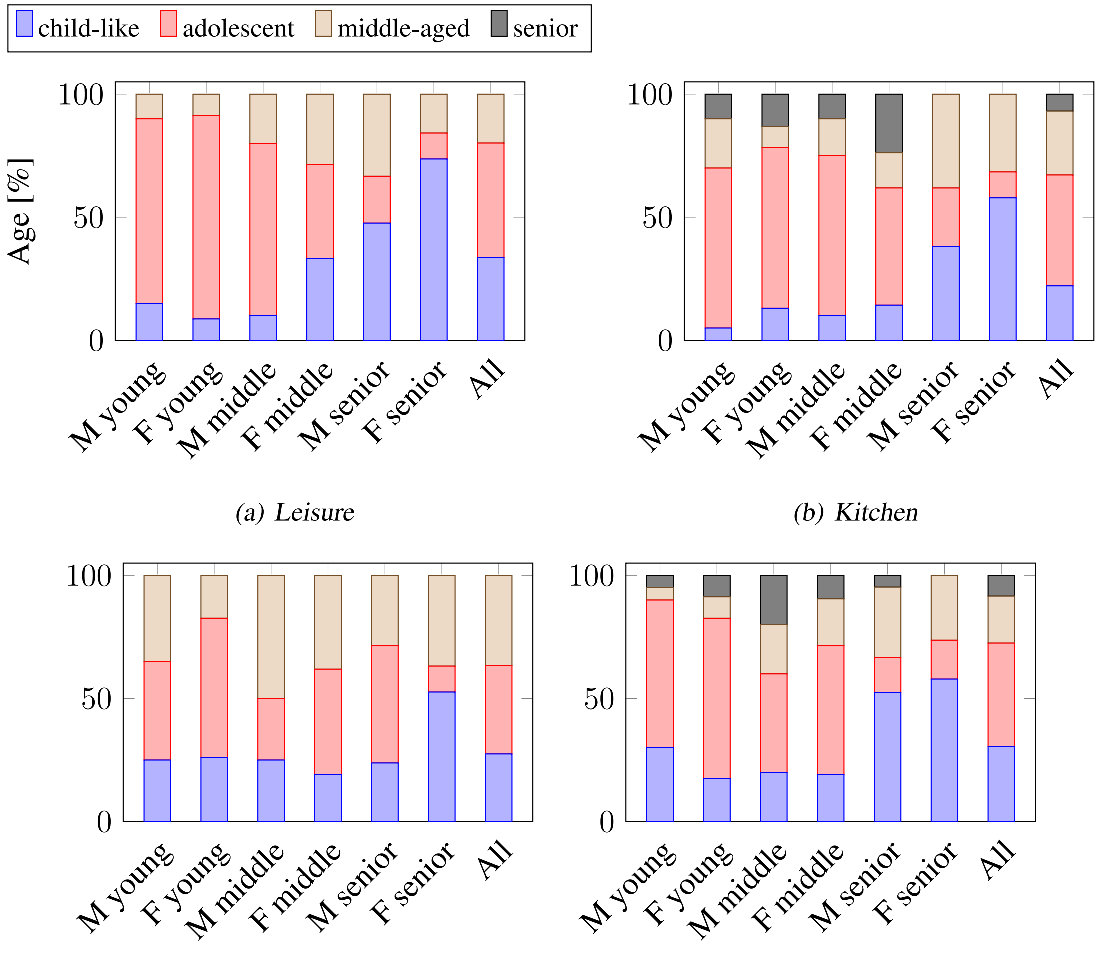
(d) Entrance

Figure 6.7: Preferences for female vs. male virtual assistants. Results of choice simulations for preferences for different genders grouped according to context and different groups of participants.

Preference for Child-like vs. Adolescent vs. Middle-aged vs. Senior Virtual Assistants

Paired samples t-tests were conducted to compare the preference for adolescent virtual assistants with the preference for the three other age categories for each of the four contexts using Bonferroni adjustments. All comparisons turned out significant, $ts(129) \geq 3.56, ps < .01$.

Considering the choice simulations, i.e., on a descriptive level, in the contexts *Leisure*, *Kitchen*, *Entrance*, there was a preference for adolescent virtual assistants. In comparison, middle-aged virtual assistants were mostly selected for the context fitness training, see Figure 6.8 (for details see Table 2 from the appendix). We observed that for young participants the adolescent virtual assistants were always preferred. For middle-aged participants, adolescent assistants were highly preferred, too. There is one exception for the context *Fitness*. In this context, middle-aged male participants preferred middle-aged vir-



(a) Leisure

(b) Kitchen

(c) Fitness

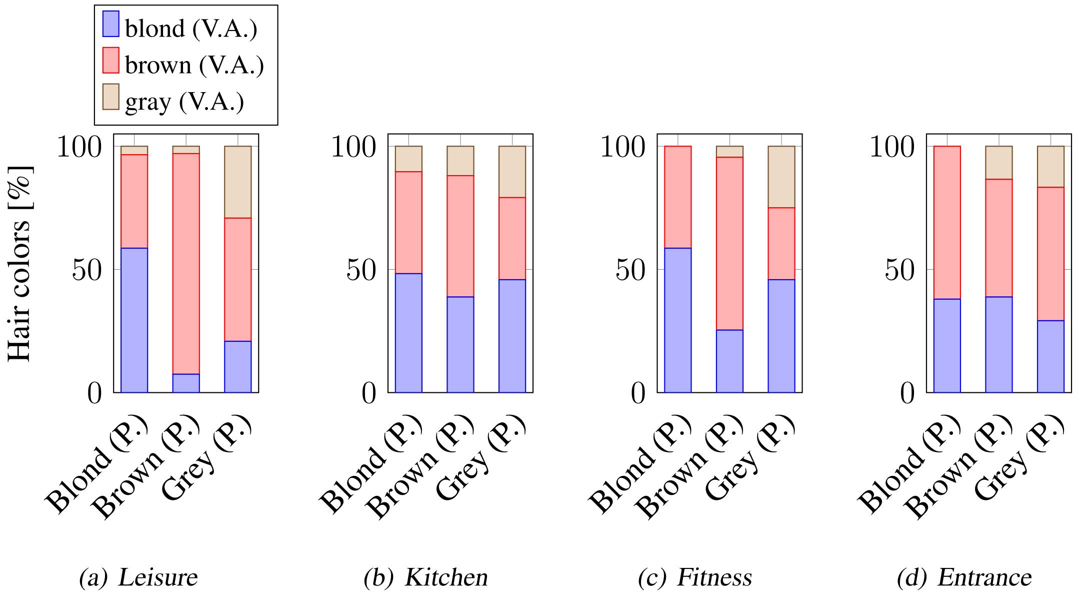
(d) Entrance

Figure 6.8: Preferences for child-like vs. adolescent vs. middle-aged vs. senior virtual assistants. Results of choice simulations for preferences for different ages grouped according to context and different groups of participants.

tual assistants more. Interestingly, for participants older than 55 years, child-like virtual assistants were preferred most. Again, there is one exception for the context *Fitness*. Here, senior male participants preferred adolescent virtual assistants.

Preference for Blond- vs. Brown- vs. Gray-haired Virtual Assistants

We conducted choice simulations to analyze preferred hair colors (blond vs. brown vs. gray hair color) for participants that were divided into groups of blond-haired, brown-haired, and gray-haired individuals based on their own, self-reported hair color. Results of the choice simulations are provided in Figure 6.9 (for details see Table 3 from the appendix). Accordingly, we observed that brown-haired participants dominantly preferred brown-haired



(a) Leisure (b) Kitchen (c) Fitness (d) Entrance

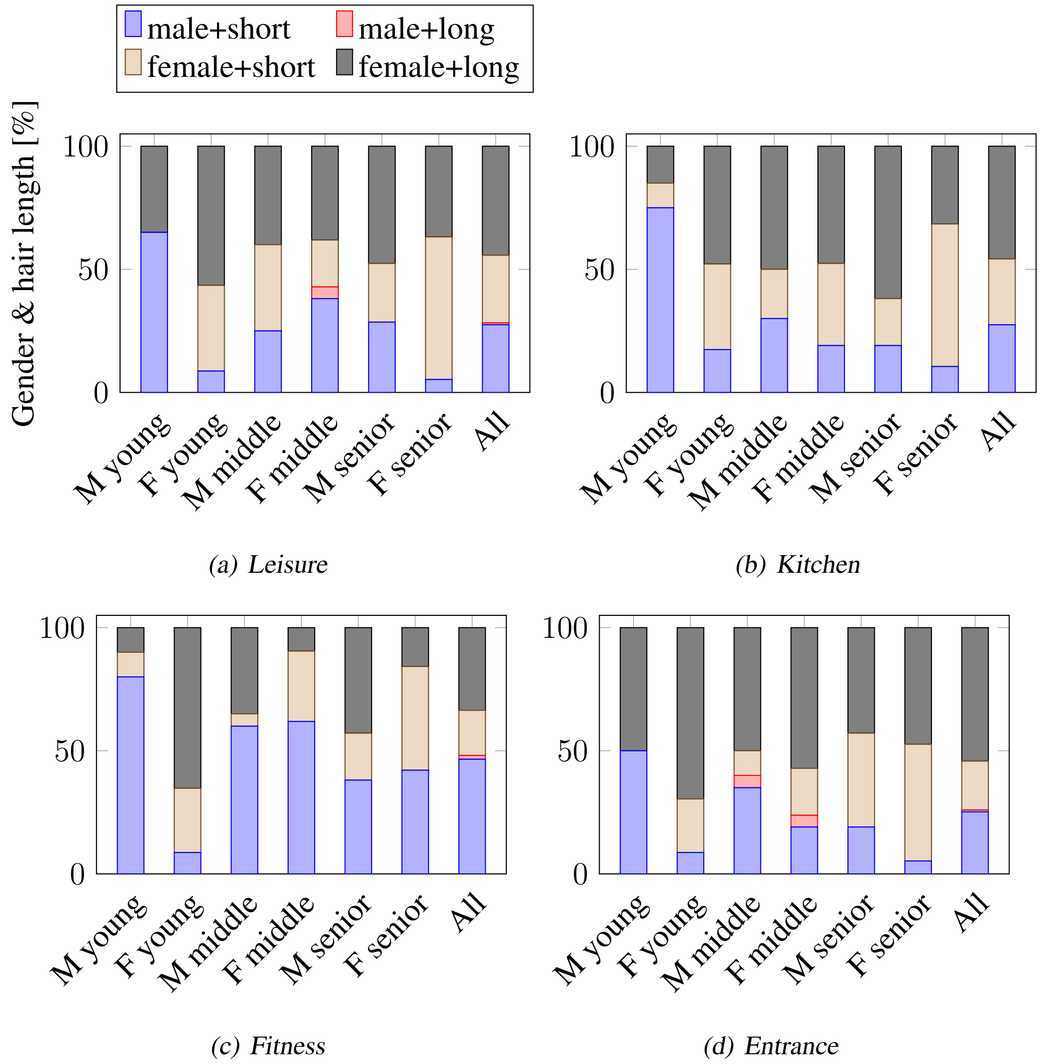
Figure 6.9: Preferences for blond- vs. brown- vs. gray-haired virtual assistants (V.A.). Results of choice simulations for different hair color preferences grouped according to context and different groups of participants (P.).

virtual assistants, while blond-haired and gray-haired participants prefer brown-haired or blond-haired virtual assistants.

Preference for Short- vs. Long-haired Virtual Assistants and the Correlation of Gender and Hair Length

Interaction effects between attributes were identified by interaction Chi-Square tests for all attributes taken two at a time. To test the strength and significance of the interaction effects, the 2 Log-Likelihood test was used respectively. For all groups, we observed interaction effects between hair length and gender at a 99 % level except for senior female participants older than 55 years in the context *Leisure*. However, in that case, hair length and gender still interact at a 95 % level.

To examine the actual preferences, we analyzed preferences for all participants together by conducting choice simulations. In addition, we analyzed preferences as a function of participants' age and gender. On average, the outputs of the choice simulations yielded higher rates for male assistants with short hair or female assistants with long hair compared to the other combinations (Figure 6.10, for details see Table 4 from the appendix).



(a) Leisure

(b) Kitchen

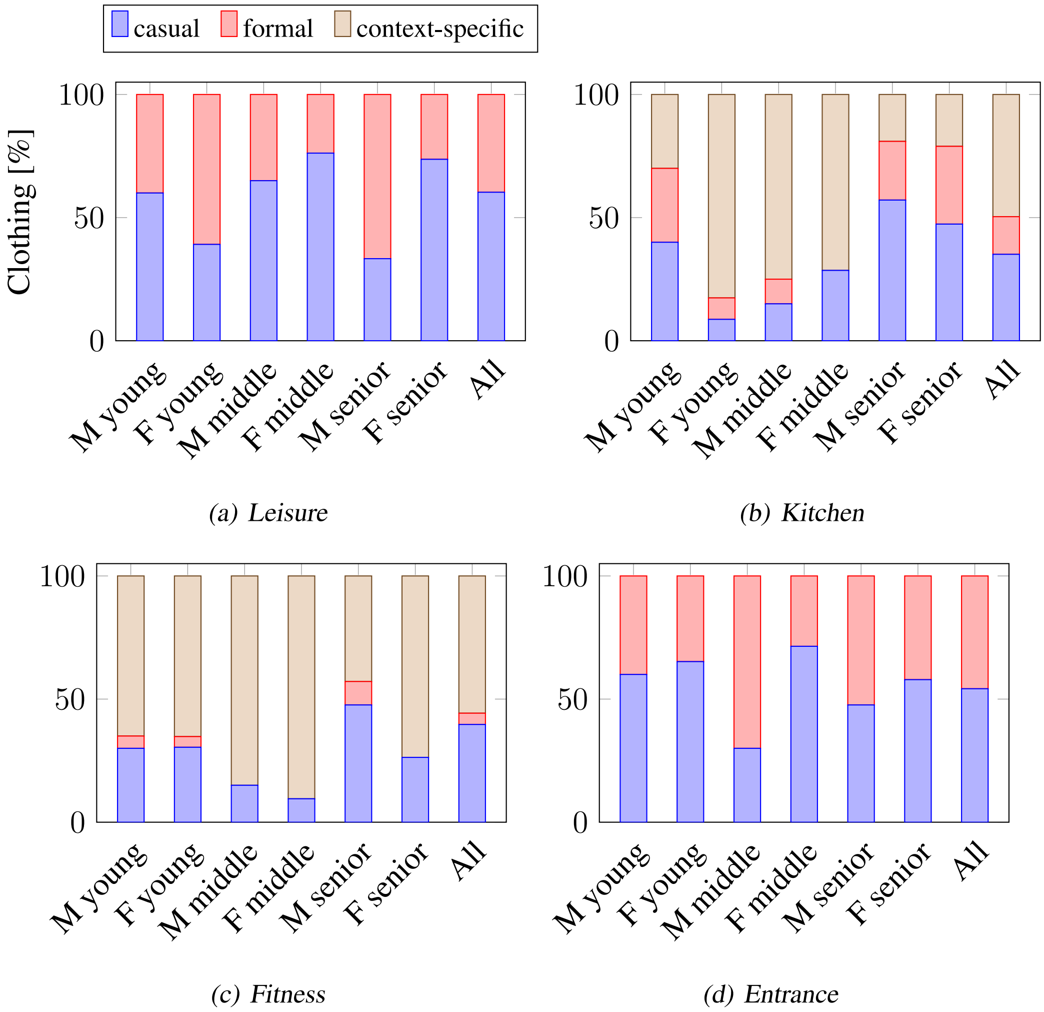
(c) Fitness

(d) Entrance

Figure 6.10: Preferences for hair lengths of the virtual assistants. Results of choice simulations for different hair length preferences in combination with gender grouped according to context and different groups of participants.

Preference for Context-Specific Clothing of the Virtual Assistants

Paired samples t-tests were conducted to compare the preference for context-specific clothing of the virtual assistants with the preference for alternative clothing for each of the four contexts using Bonferroni adjustments. Although there was no clear context-specific clothing for the *Leisure* and *Entrance* contexts, we nevertheless considered these contexts in our analysis for the sake of completeness. There was a significantly greater preference for casual clothing in the *Leisure* context compared to formal clothing ($M_{casual} = 4.52, SD_{casual} = 16.41; t(130) = 3.15, p = .002$), and a greater preference for sports



(a) Leisure

(b) Kitchen

(c) Fitness

(d) Entrance

Figure 6.11: Preferences for virtual assistants' context-specific clothing. Results of choice simulations for different clothing preferences grouped according to context and different groups of participants.

clothing in the context *Fitness* compared to casual and formal clothing ($M_{sports} = 36.95$, $SD_{sports} = 27.80$; $M_{casual} = 3.61, SD_{casual} = 16.85$; $M_{formal} = -40.56, SD_{formal} = 24.82$; $ts(130) \geq 9.86, ps < .001$). Further, there were no significant (ns) differences between the preference for kitchen clothing, casual, and formal clothing in the *Kitchen* context ($M_{kitchen} = 4.64, SD_{kitchen} = 62.41$; $M_{casual} = -4.63, SD_{casual} = 39.87$; $M_{formal} = -.02, SD_{formal} = 29.53$; $ts(129) \leq 1.13, ns$). There was also no difference between formal and casual clothing in the *Entrance* context ($M_{formal} = -.78, SD_{formal} = 16.03$; $t(129) = -.52, ns$).

Further, we analyzed preferences by performing choice simulations for all participants together. Here, it turned out that casual clothing was slightly more preferred for the contexts *Leisure* and *Entrance*. Moreover, kitchen-specific clothing was preferred most in the context *Kitchen* while sportswear was preferred most in the context *Fitness*.

We also analyzed participants' preferences according to age and gender by conducting choice simulations (Figure 6.11, for details see Table 4 from the appendix). For the contexts *Leisure* and *Entrance*, it turned out that casual clothing is preferred more often. In contrast, for *Fitness*, sportswear is mostly preferred. The results from the context *Kitchen* were not as clear. There, clothing that fit this particular context was only markedly preferred by middle-aged participants of both genders as well as by young female participants. Young male participants also preferred casual clothing but also accepted the context-specific clothing. Interestingly, senior participants preferred casual clothing ($52.25\,\%$), while kitchen-specific clothing was preferred least ($20.05\,\%$).

6.2.2 Recommendations for the Design of Virtual Assistants in the Smart Home Context

In this section, we derive guidelines for the design of virtual assistants in smart home contexts. The main finding of the present research is that the design of virtual assistants should be tailored to fit different target groups and different contexts. However, in order to derive more versatile and applicable virtual assistants, we also present positively evaluated characteristics of virtual assistants for situations in which the age, gender, hair color, and context are not known.

Our results suggest, that when context and user features remain unknown, an adolescent, female virtual assistant with long brown hair and casual clothing might be the character of choice, as these characteristics were preferred most over all participants (see the framed virtual assistant in Table 6.2). Furthermore, it became apparent across analyses that the virtual assistant's age turned out as the most important customization category across all contexts.

In case the context of the use case is known, we would recommend an adolescent female virtual assistant with long hair as well. Interestingly, for the context *Fitness*, middle-aged male virtual assistants with short hair turned out equally suitable. Further, depending on the context, we suggest opting for a casual clothing style in the contexts *Leisure* and *Entrance*, for kitchen-specific wear in the context *Kitchen*, and for sportswear in the context *Fitness*. Hence, the clothing should be context-specific and casual if no clear, context-specific clothing is available.

Going one step further, if the context as well as the gender and age of the user is known, we can give more tailored recommendations. See Table 6.2 for an overview of the most preferred virtual assistants based on choice simulations.

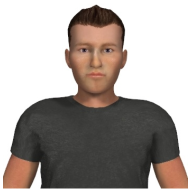
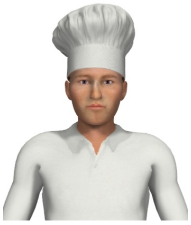
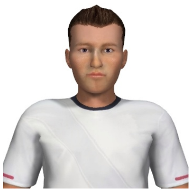
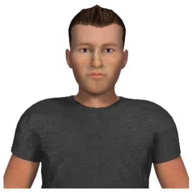
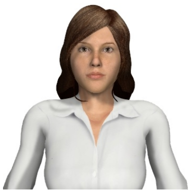
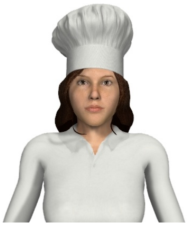
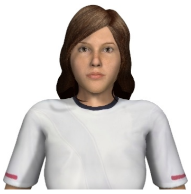
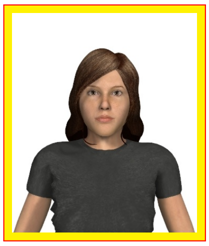
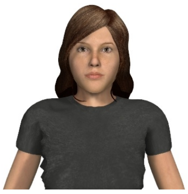
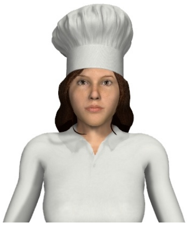
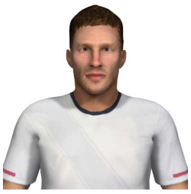
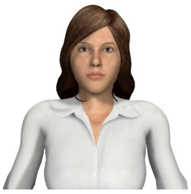
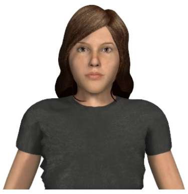
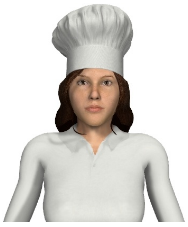
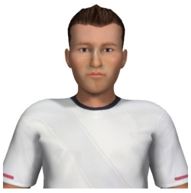
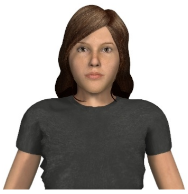
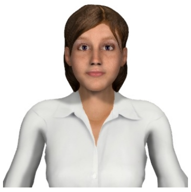
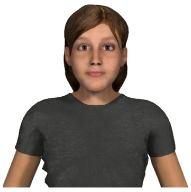
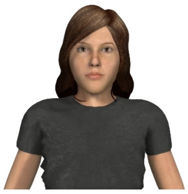
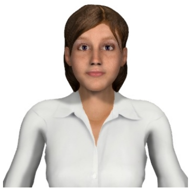
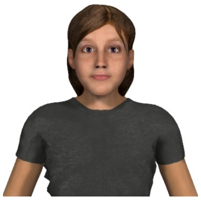
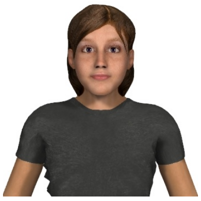
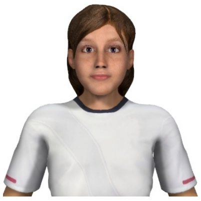
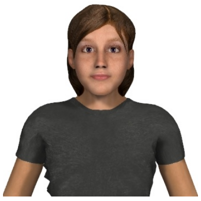
Participants	Leisure	Kitchen	Fitness	Entrance
M young				
F young				
M middle				
F middle				
M senior				
F senior				

Table 6.2: Results of choice simulations for the most preferred virtual assistants for different genders, age categories, and contexts.

With regard to gender of the virtual assistants, we observed that, on average, female virtual assistants were preferred most, particularly by female participants. Interestingly, on average, young male participants preferred male virtual assistants most in all contexts. Further, we observed the trend that, on average, male participants increasingly preferred female assistants by an increasing age. In contrast, for female participants, a female virtual assistant is always preferred by the majority. Interestingly, for the context *Fitness*, both male and female middle-aged participants preferred male virtual assistants.

Regarding preferences for age of the virtual assistants, we found that adolescent virtual assistants were strongly preferred. Interestingly, senior participants preferred child-like virtual assistants most, except in the context *Fitness*, where senior male participants preferred adolescent virtual assistants most. Furthermore, in the context *Fitness*, middle-aged male participants prefer middle-aged virtual assistants.

Considering the clothing, our results revealed an interesting fact when taking participant age into account. In the context *Kitchen*, there were no significant differences between the preference for kitchen clothing, casual clothing, and formal clothing. Further, senior participants preferred kitchen clothing least and instead preferred casual clothing. For more tailored clothing on virtual assistants with respect to age and gender, we refer to Figure 6.11.

In terms of hair color of the virtual assistant, we recommend a virtual assistant with brown hair in case the hair color of the user is not known because this color was chosen most often. Otherwise, we recommend blond- or brown-haired virtual assistants for blond- or gray-haired users. For brown-haired users we recommend brown-haired assistants because these users preferred brown-haired virtual assistants which resembled themselves. Thus, the aforementioned recommendations should be adjusted in case the hair color of the user is known.

Finally, we examined whether preferences regarding gender and hair length of the virtual assistants would interact. As predicted, we consistently observed interaction effects between gender and hair length of the virtual assistant. That is, congruent with gender stereotypical appearance, male virtual assistants should feature short hair, while female virtual assistants should have long hair.

6.2.3 Hypotheses

Hypothesis 1

Effects of Perceived Similarity

Hypothesis 1 assumed that the more people perceived their chosen virtual assistant as being similar to themselves, the better the evaluation would be. To analyze whether participants' evaluation of their final virtual assistant, their intention to use it, their wish to have it at home, and their satisfaction with the realization of the virtual assistant was positively correlated with perceived similarity of the assistant to themselves, a Pearson's product-moment correlation analysis was conducted separately for each use case. In every context, all four variables—evaluation of the assistant, intention to use, wish to have it at home, and satisfaction with the realization of the virtual assistant—turned out to be positively correlated with similarity. This result validates the hypothesis (cf. Table 6.3).

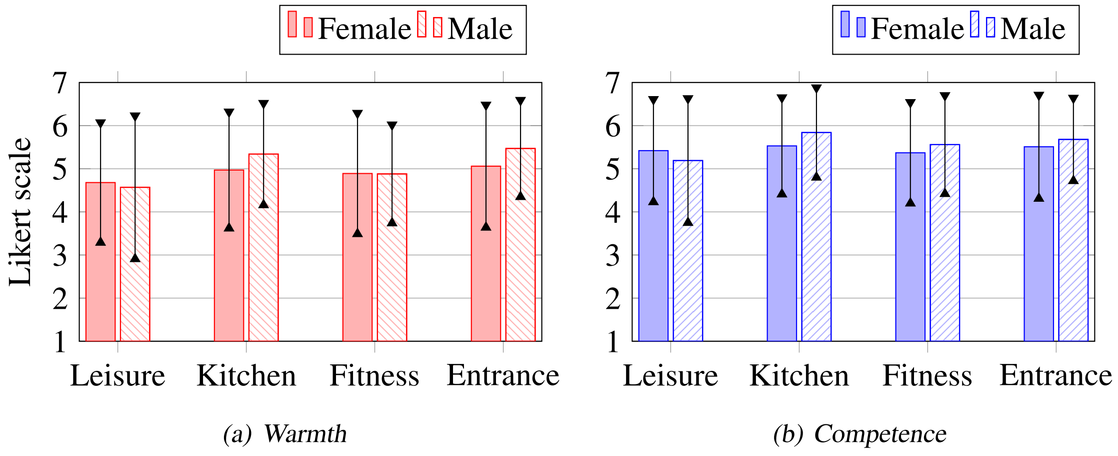
	evaluation of the assistant	intention to use	wish to have it at home	satisfaction with the realization
Leisure	$r(131) = .47$, $p < .001$	$r(131) = .32$, $p < .001$	$r(131) = .39$, $p < .001$	$r(131) = .47$, $p < .001$
Kitchen	$r(127) = .47$, $p < .001$	$r(127) = .21$, $p = .02$	$r(127) = .30$, $p = .001$	$r(127) = .47$, $p < .001$
Fitness	$r(126) = .32$, $p < .001$	$r(126) = .19$, $p = .03$	$r(126) = .32$, $p < .001$	$r(126) = .41$, $p < .001$
Entrance	$r(125) = .37$, $p < .001$	$r(125) = .20$, $p = .03$	$r(125) = .22$, $p = .01$	$r(125) = .43$, $p < .001$

Table 6.3: Effects of perceived similarity. In every context, all four variables—evaluation of the assistant, intention to use, wish to have it at home, and satisfaction with the realization of the virtual assistant—turned out to be positively correlated with perceived similarity.

Hypothesis 2

Perceived Warmth and Competence

Hypothesis 2 assumed that female virtual assistants are perceived as warm, whereas male virtual assistants are perceived as highly competent and high in persuasive power. A MANCOVA was conducted to analyze the effect of gender of both participants and the virtual assistants on perceived warmth and competence. Participant age was used as a covariate. Descriptive statistics are summarized in Figure 6.12 (for details see Table 5 from the appendix). Against expectations, there were no main effects of virtual assistant gender (*Leisure*: $Fs(1, 125) \leq .94, ns$; *Kitchen*: $Fs(1, 122) \leq 2.72, ns$; *Fitness*: $Fs(1, 120) \leq 1.50, ns$; *Entrance*: $Fs(1, 120) \leq 2.24, ns$) and no main effects of participants' gender (*Leisure*: $Fs(1, 125) \leq .86, ns$; *Kitchen*: $Fs(1, 122) \leq .74, ns$; *Fitness*:



(a) Warmth

(b) Competence

Figure 6.12: Perceived warmth and competence of the virtual assistants. Means and standard deviations of attributed warmth (a) and competence (b) to female and male virtual assistants grouped according to context.

$Fs(1,120) \leq .90, ns$; *Entrance*: $Fs(1,120) \leq .05, ns$). However, there was an effect of participants' age on warmth in the context *Leisure* ($F(1,125) = 4.86, p = .03$, effect size $\eta^2 = .04$): The older participants were, the more warmth they attributed to their self-chosen virtual assistant.

Thus, the hypothesis cannot be validated. Instead, one-sample t-tests against the neutral scale midpoint (scale value = 4 on a 7-point Likert scale) showed significant deviations from the mean value for warmth and competence ratings in the four contexts indicating that all virtual assistants that represented final choices were rated rather high in both warmth and competence (*Leisure*: $ts(130) \geq 5.07, ps < .001$; *Kitchen*: $ts(126) \geq 9.41$, $ps \leq .001$; *Fitness*: $ts(125) \geq 7.82, ps < .001$; *Entrance*: $ts(124) \geq 9.92, ps < .001$).

Hypothesis 3

Effects of Individual Openness Toward Technology

Hypothesis 3 assumed that peoples' individual openness toward technology would moderate the acceptance of the virtual assistant. To analyze the relationship between participants' openness toward technology and their evaluation of their preferred virtual assistant, we conducted regression analyses across the four use cases. Openness toward technology significantly predicted participants' intention to use the assistant (standardized coefficient $\beta = .22, t(130) = 2.57, p < .01$) and their wish to have the assistant at home ($\beta = .23, t(130) = 2.63, p = .01$). There was no effect of participants' openness toward technology on their satisfaction with the realization of the virtual assistant ($\beta = .06, t(130) = .65, p = .52$).

In this chapter, we described how we conducted a laboratory study to analyze preferences of users who differed in terms of age, gender, and even hair color with regard to virtual assistants that were deployed in various smart home contexts. This finalizes the third and last part of this thesis and brings us to the overall conclusion.

7 Conclusion

This thesis investigates how virtual humans can be generated and efficiently employed for *character reconstruction*, for *craniofacial reconstruction in medicine*, and for an interdisciplinary study. For the latter, we deploy virtual humans as stimuli to examine the *preferences for virtual assistants* in smart home contexts.

In the following, we conclude this thesis by summarizing our main results and describing limitations as well as possible directions of future work.

Character Reconstruction

We started with deriving a template fitting framework that provides accurate face reconstructions. Our method is based on a structured analysis of the different algorithmic building blocks for non-rigid registration. From the algorithmic building blocks, we combined 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 from multi-view stereo data in a couple of minutes.

An interesting direction for future work is the reconstruction of non-neutral facial expressions. 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 2.11) would make manual interaction less necessary and make our approach fully automatic.

In a next step, we presented a fast and reliable pipeline to digitally clone full real persons into realistic virtual humans. For 3D-scanning, we employ a custom-built camera rig with 40 cameras for the body and 8 cameras for the face and compute dense point clouds through multi-view stereo reconstruction. Similarly as before and in order to robustly deal with noise and missing data, we fit a generic human body model to the user's scanner data. By also transferring the skeleton, blendshapes, and eyes of the generic template to the model, our reconstructed virtual humans can be animated in standard game engines and VR frameworks. Furthermore, we demonstrated how to easily and seamlessly transfer clothing from one character to another while still being compatible to standard rendering engines.

Our character generation requires only a minimum amount of user interaction and takes less than ten minutes on a desktop PC. It is therefore fast enough to be performed at the beginning of each session in a VR experimental study.

While our pipeline produced convincing results with all tested subjects, some inherent limitations remain. Due to scanning subjects in A-pose, some areas are not visible from enough cameras and thus are not reconstructed well. While missing data can be compensated by template data during geometry reconstruction, these regions still suffer from texture artifacts.

In future work, one could use the proposed pipeline to generate characters for preference studies for personalized virtual agents. Another interesting direction for future work is the realistic modeling of clothing motion. Moreover, we will work on further speeding up the whole pipeline and making it fully automatic.

Craniofacial Reconstruction in Medicine

In the second part of this thesis, we presented an automated method based on a parametric skull model, a parametric head model, and FSTT statistics for reconstructing the face for a given skull. The models we are using were derived from head CT scans taken from an existing CT image repository and from 3D surface scans of real subjects. Our approach has three main outcomes: (i) a dense map of FSTT (i.e., a soft tissue layer), (ii) a visual presentation of a statistically probable head based on FSTT statistics and a parametric head model, and (iii) a method for generating plausible head or face variants, respectively.

The main advantage of our approach over landmark-based FSTT measurements (see references in [Ste17]) is the density of the FSTT map without the need for error-prone normal information. A FSTT value can be derived from the FSTT statistics for any vertex of the parametric skull model. It is important to note that the statistical evaluation of the FSTT is fully automatic without any manual interaction. This is different from other CT-based FSTT assessments which often still rely on error-prone manual measurements (see, e.g., [Cha13]). Our fully automated method can help to generate a more accurate database in the future because it largely overcomes the accuracy issues well-known for manual, landmark-based FSTT assessments [SS08]. However, as our method is based on CT scans, it is still prone to typical artifacts and gravity effects due to supine patient position. Although our statistic sample of FSTT so far is generated from only 43 CT scans, the data we derived (Figure 4.5) clearly indicate good agreement with data just recently published in a meta-analysis [Ste17]. If enough appropriate CT scans are available, rapid processing by means of an automated pipeline can aid the creation of a large statistical database. It seems most likely that methods such as ours constitute the future for the generation of statistical models from 3D medical imagery. Therefore, enlarging the database will be part of future work to produce more precise statistics.

FSTT statistics play a significant role in facial approximation [SS08] and are also an integral part of modern orthodontic treatment planning [APS99, Cha13]. For forensic reconstruction, FSTT statistics form the basis for further steps in the reconstruction process. The advantage of our approach in comparison to other automated methods [TBK+05, TBL+07, RME+14, SZD+16, SZM+17] is that our facial reconstruction process is fully automated. The only manual steps necessary in our approach are done during the model generation processes. As mentioned before, our FSTT statistics are independent of the measurement direction; thus, we utilize union-of-spheres in the reconstruction process. Therefore, error-prone strategies such as averaging over normal vectors to define a measurement direction are completely avoided. By representing the FSTT-offset by a sphere-mesh, we could improve the FSTT and thus further improve the quality of the resulting facial reconstructions. Moreover, our parametric FSTT model allows us to create plausible head variants in a statistical sense without having any prior knowledge about the head characteristics.

Indeed, our method already provides a good approximation of the facial skin surface in a statistical sense. Nevertheless, the quality of the reconstruction depends on the size of the statistic sample. In order to use additional descriptive factors (e.g., age, sex, ancestry, weight, or skeletal classes [HTKK14]), a larger sample size representing the variance of each of the factors is required. We thus aim to enlarge our skull and head database to further elaborate on the method introduced here. Part of future work is the evaluation of accuracy and recognition of a reconstruction based on our method. Inspired by Miranda's approach [MWR+18], we are planning to collect existing CT datasets and frontal standardized photographs that are voluntarily donated by subjects for publication and also for the assessment of accuracy and recognition.

Building upon this work, we presented a multilinear model that maps a set of low-dimensional parameters for skull shape and FSTT distribution to an accurate and high-quality mesh of both the skin and the skull geometry.

The required training data, a dense Cartesian product of skull and FSTT variations, cannot be obtained by measurements alone. Based on individual parametric models of skull shape, FSTT distributions, and head shapes, we generate synthetic—but statistically plausible—training data by computing head models for given skull shapes and FSTT distributions.

We demonstrated that our model has several interesting and high-potential applications in the medical context. First, it allows simulation of plausible head shapes for given skull and FSTT variations at interactive rates. This is an important component in explorative, computer-aided forensics. Second, by fitting the multilinear model to face scans, we can infer both the skull shape and the FSTT of the scanned individual and successively simulate weight changes. More importantly, the estimation of a reasonably accurate skull shape from a photogrammetric face scan helps to reduce or even avoid X-ray radiation for bony cephalometric skull assessments in orthodontics or for surgical planning for patients with craniofacial malformations. To the best of our knowledge, there is no approach to infer the skull shape from a skin surface scan with the high precision we obtained.

One limitation of our model is that there is no guarantee that eyes and mouth stay at their anatomically correct positions relative to the skull when varying the FSTT parameters. The slight movements of eyes and mouth can be avoided in the future by incorporating corresponding constraints into the generation of training data. Furthermore, since real faces and FSTTs are asymmetric to a certain extent, our derived FSTT component inherits this asymmetry. This may result in asymmetric head shapes when varying FSTT parameters.

Besides improving the theoretical properties of our model, collecting more CT scans from a larger variation of real people is required in order to increase the variability and expressiveness of our model. Access to more training data would also allow exploration of different learning algorithms or investigation of other approaches for filling up the sparsely scanned data with the goal of achieving a dense set of training data. Moreover, our model could be used for realistic animations in movies.

Preferences for Virtual Assistants

The main strength of the study presented in the last part of this thesis is the fact that the participants that were involved differed in age, gender, and hair color. This way, we were able to comprehensively examine the effect of perceived similarity on the evaluation of virtual assistants. Our results showed that, for participants, the evaluation of their final virtual assistant, their intention to use it, their wish to have it at home, and their satisfaction with the realization of the virtual assistant is positively correlated with the self-perceived similarity to the assistant.

Contrary to our expectations, we found that the individually determined preferred virtual assistants were perceived as both warm and competent regardless of the assistant's gender. This might be due to the fact that the individually determined preferred virtual assistants were positively evaluated. Thus, such assistants seem to be perceived as both warm and competent.

Considering the individual openness toward technology of the participants, openness toward technology significantly predicted participants' intention to use the assistant and their wish to have the assistant at home. However, there was no effect of participants' openness toward technology on their satisfaction with the realization of the virtual assistant.

These findings emphasize the importance of differentially investigating user preferences for virtual assistants. The key finding of the current research highlights that the design of virtual assistants should be tailored to different target groups and different contexts, i.e., participant features such as age, gender, and even their hair color, as well as the use contexts for virtual assistants do matter. To this end, we give recommendations for the design of virtual assistants in smart home environments. We can conclude that people do not necessarily prefer characteristics that are similar to themselves, i.e., people did not always choose a virtual assistant of their gender, age, and hair color.

Our research explored judgments about a broad variety of virtual characters to be used in a variety of domains in the smart home context. These judgments were based on a demographically diverse sample encompassing various age groups, were counterbalanced in terms of gender, and even took into account participant hair color as a cue for similarity with the virtual assistants that had to be rated. While our work provides important insights into user preferences, it nevertheless has shortcomings that need to be discussed and improved in follow-up studies. For example, the virtual assistants used in our study differed in terms of five core features. This already provided a large array of stimuli. However, we could only rely on these particular prototypes. There might be a vast number of alternative ways to instantiate our operationalization. For instance, manipulating body weight has not been considered at all in the present research; however, it would be interesting to manipulate in future research. Clearly, the graphical models used in our research were far from perfect in terms of realistic appearance. On the other hand, more human-like appearance could give rise to more human-like behavior that could not be fulfilled [Rui15]. Overall, there might be many more equally relevant visual features that might influence the evaluation of virtual assistants; our results surely cannot generalize across all of them.

Thus, future studies could shed more light on this by investigating settings of other virtual assistants at other levels of realism and even more attributes. Besides considering visual aspects, it turned out that the animation and voice of the virtual assistants were evaluated as being quite robot-like. Future studies should investigate the effect of different synthetic voices and animation complexities on the evaluation of virtual assistants in smart home contexts. Motivated by our results on self-perceived similarity to the assistant, an interesting direction for future work is the evaluation of virtual assistants that are generated by our approach from Chapter 3 and, thus, are digital clones of the participants. Furthermore, virtual assistants that are morphed versions [ZB13] of digital clones and assistants as used in this study could be investigated. Additionally, the technique presented in Section 3.5.2 could be used to transfer clothing, e.g., to add one's own clothing to assistants used in our study or to change the clothing of a digital clone to reduce potential confounds caused by different clothing. Finally, it could be examined how simulated weight changes of the digitally cloned virtual assistants effect participants' evaluation of the virtual assistants, e.g., by our approach presented in (Section 5.3.3).

Supplemental Material for Preferences Study

Supplementary Figures

Figures 1, 2, 3, and 4 show all assistants that were designed and used as stimuli.

Supplementary Tables

The tables in this document provide the exact numbers for the diagrams that are shown in the main article.

context	gender [%]	age [%]	hair color [%]	hair length [%]	clothing [%]
Leisure	12.74	46.82	22.36	12.61	5.47
Mean (SD)	(8.59)	(9.28)	(5.39)	(8.36)	(4.03)
Kitchen	13.05	41.07	17.30	10.05	18.52
Mean (SD)	(8.07)	(11.61)	(6.21)	(7.85)	(10.05)
Fitness	9.98	44.93	16.88	11.48	16.73
Mean (SD)	(6.28)	(9.95)	(5.16)	(8.26)	(8.64)
Entrance	17.74	46.04	20.23	10.86	5.12
Mean (SD)	(9.67)	(9.71)	(6.80)	(8.16)	(3.84)

Table 1: Means and standard deviations of the importance of the customization categories grouped according to context.

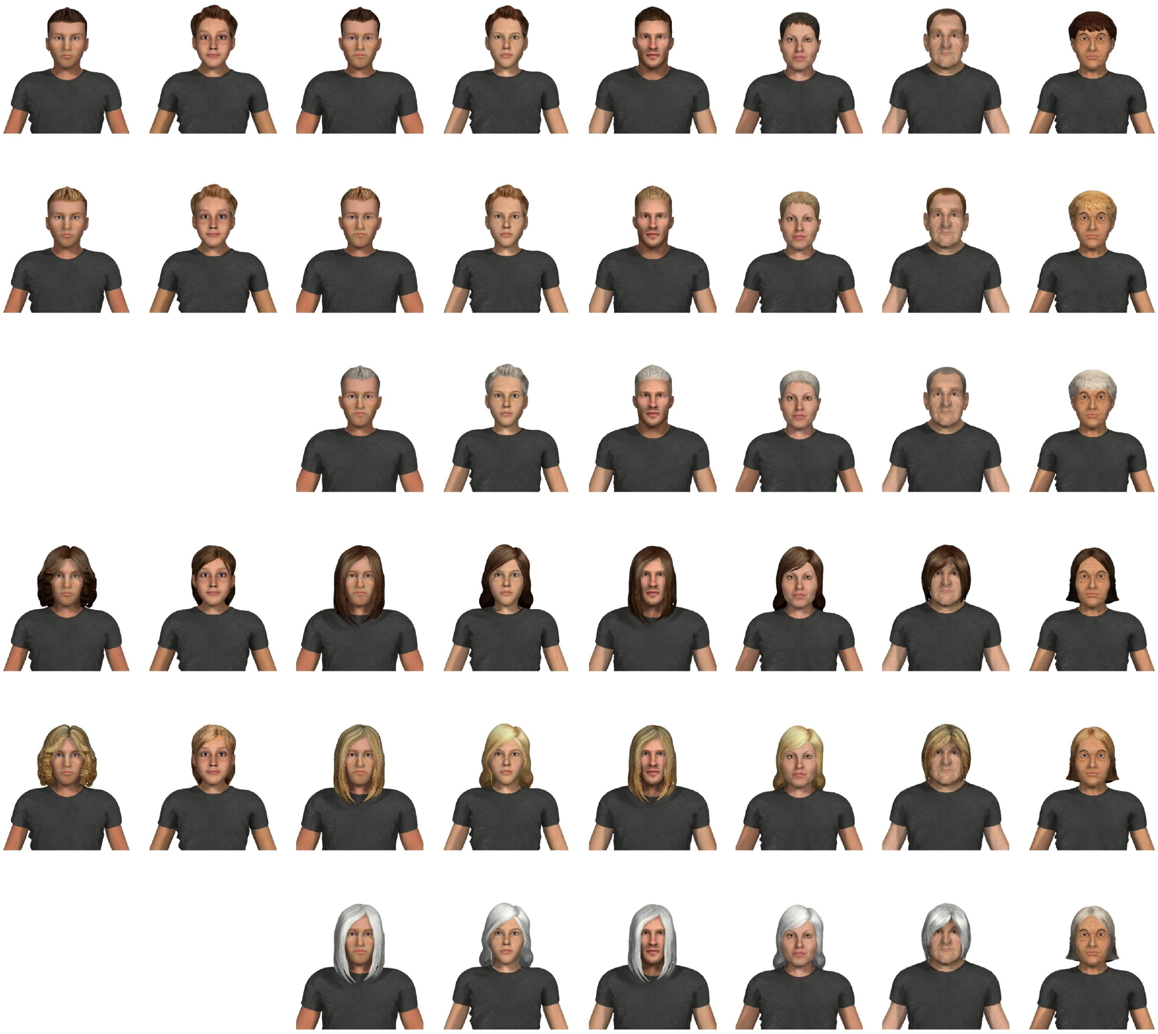


Figure 1: All virtual assistants with casual clothing that were designed and used as stimuli.

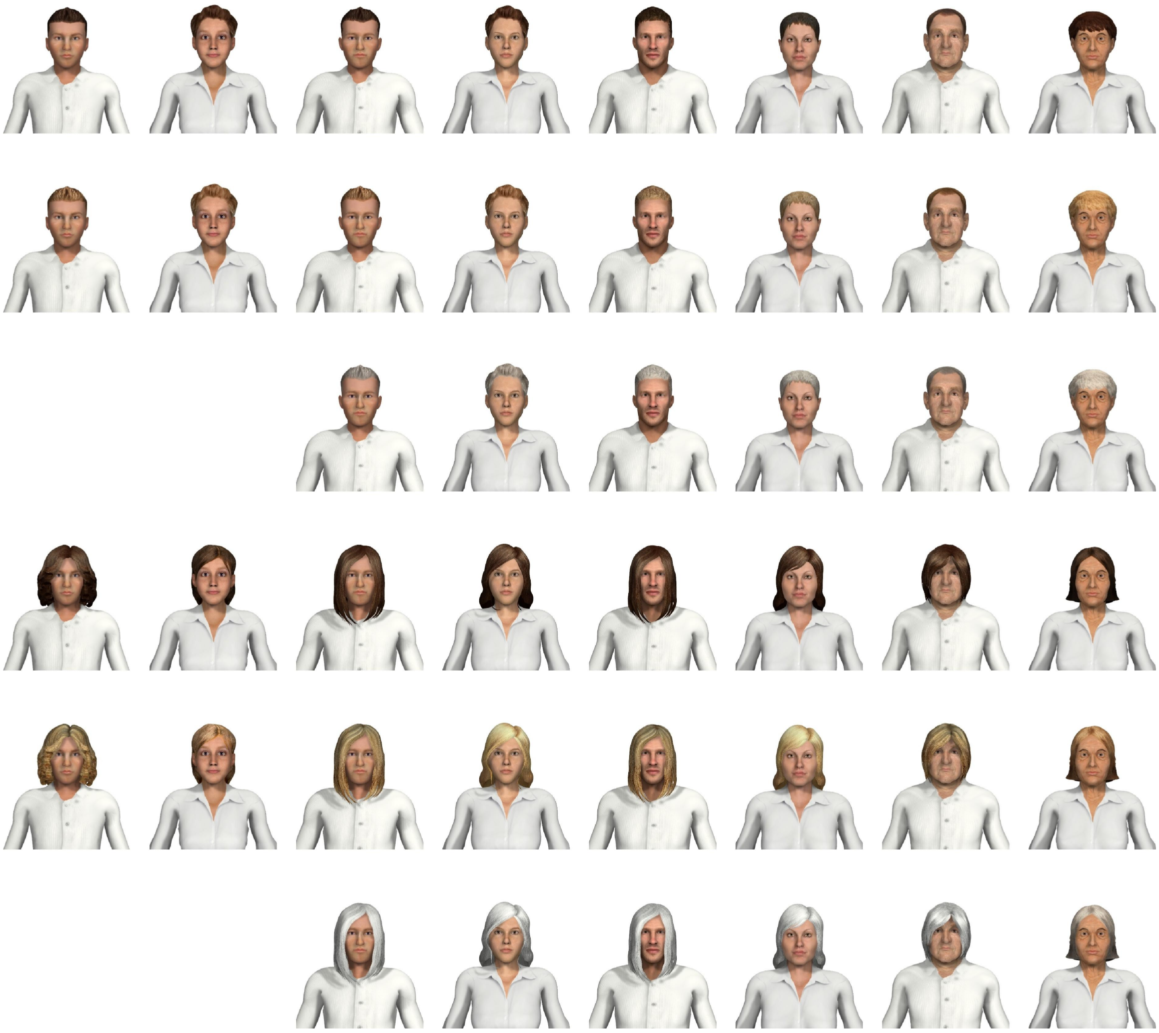


Figure 2: All virtual assistants with formal clothing that were designed and used as stimuli.

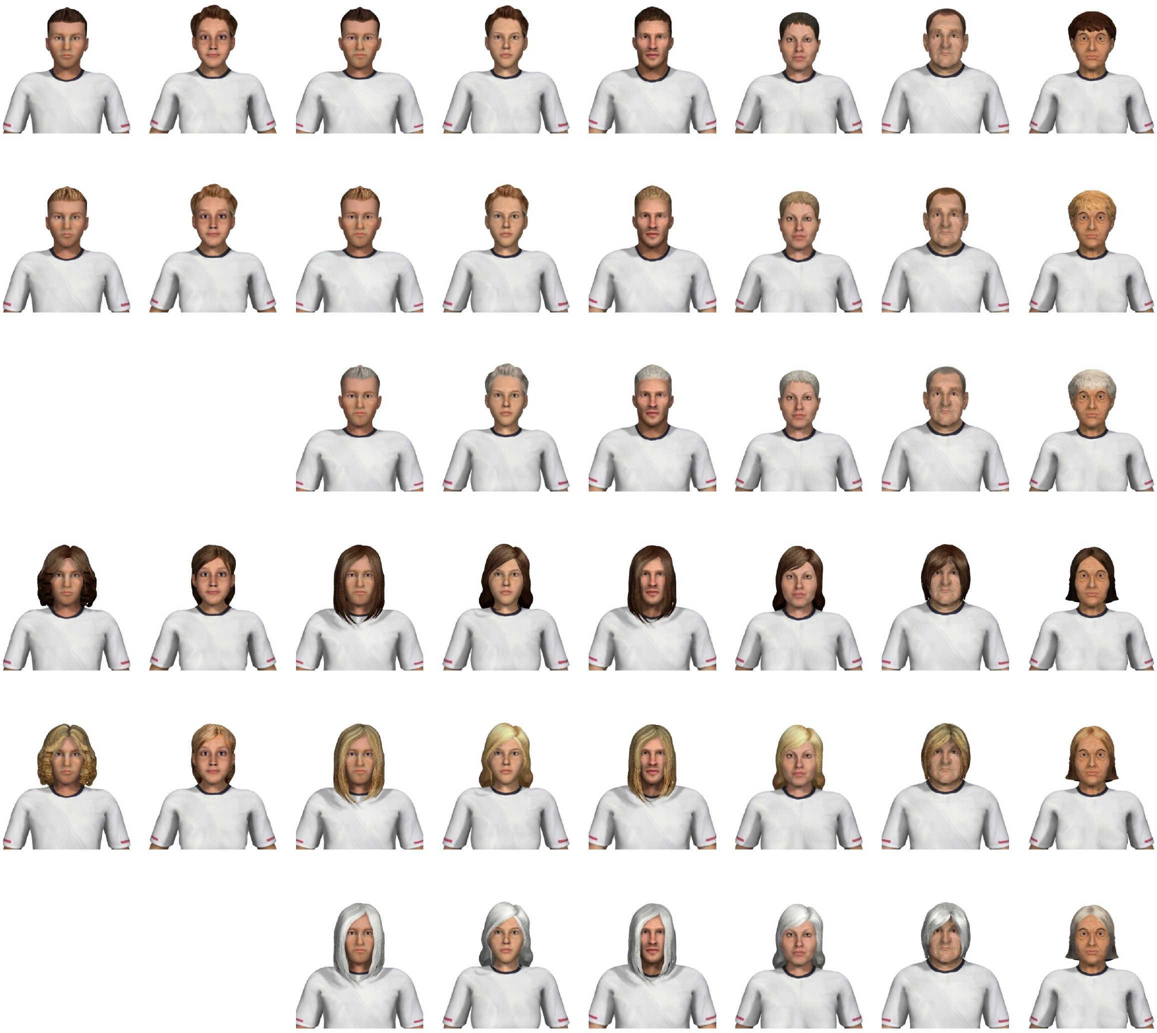


Figure 3: All virtual assistants with clothing for fitness training that were designed and used as stimuli.

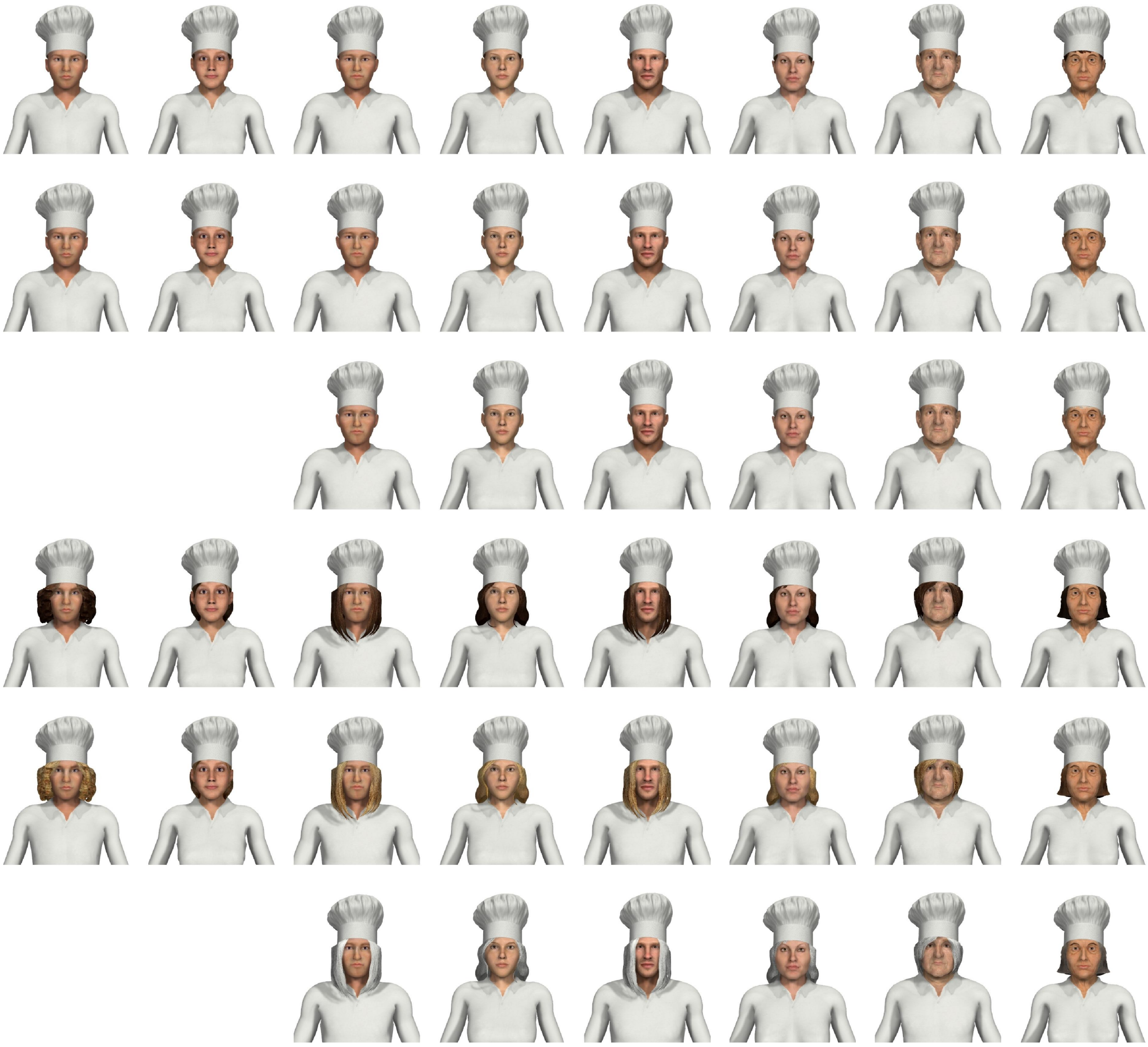


Figure 4: All virtual assistants with clothing for the kitchen domain that were designed and used as stimuli.

	Participants	child-like [%]	adolescent [%]	middle-aged [%]	senior [%]	male [%]	female [%]
Leisure	M young	15	75	10	0	65	35
	F young	8.7	82.6	8.7	0	8.7	91.3
	M middle	10	70	20	0	25	75
	F middle	33.32	38.12	28.56	0	42.86	57.14
	M senior	47.62	19.05	33.33	0	28.57	71.43
	F senior	73.68	10.53	15.79	0	5.26	94.74
	All	33.59	46.56	19.85	0	28.24	71.76
Kitchen	M young	5	65	20	10	75	25
	F young	13.04	65.22	8.7	13.04	17.39	82.61
	M middle	10	65	15	10	30	70
	F middle	14.29	47.62	14.29	23.8	19.05	80.95
	M senior	38.1	23.8	38.1	0	19.05	80.95
	F senior	57.89	10.53	31.58	0	10.53	89.47
	All	22.14	45.04	25.95	6.87	27.48	72.52
Fitness	M young	25	40	35	0	80	20
	F young	26.09	56.52	17.39	0	8.7	91.3
	M middle	25	25	50	0	60	40
	F middle	19.05	42.86	38.09	0	61.9	38.1
	M senior	23.81	47.62	28.57	0	38.1	61.9
	F senior	52.63	10.53	36.84	0	42.11	57.89
	All	27.48	35.88	36.64	0	48.09	51.91
Entrance	M young	30	60	5	5	50	50
	F young	17.4	65.2	8.7	8.7	8.7	91.3
	M middle	20	40	20	20	40	60
	F middle	19.05	52.38	19.05	9.52	23.81	76.19
	M senior	52.38	14.29	28.57	4.76	19.05	80.95
	F senior	57.89	15.79	26.32	0	5.26	94.74
	All	30.53	41.98	19.08	8.41	25.95	74.05

Table 2: Results of choice simulations for preferences for different ages and genders grouped according to context and different groups of participants.

	Participants	blond-haired virtual assistants [%]	brown-haired virtual assistants [%]	gray-haired virtual assistants [%]
Leisure	Blond-haired	58.62	37.93	3.45
	Brown-haired	7.46	89.55	2.99
	Gray-haired	20.83	50	29.17
Kitchen	Blond-haired	48.28	41.38	10.34
	Brown-haired	38.81	49.25	11.94
	Gray-haired	45.83	33.33	20.84
Fitness	Blond-haired	58.62	41.38	0
	Brown-haired	25.37	70.15	4.48
	Gray-haired	45.83	29.17	25
Entrance	Blond-haired	37.93	62.07	0
	Brown-haired	38.81	47.76	13.43
	Gray-haired	29.18	54.15	16.67

Table 3: Results of choice simulations for different hair color preferences grouped according to context and different groups of participants.

	Participants	M+s [%]	M+l [%]	F+s [%]	F+l [%]	casual [%]	formal [%]	context [%]
Leisure	M young	65	0	0	35	60	40	–
	F young	8.7	0	34.78	56.52	39.13	60.87	–
	M middle	25	0	35	40	65	35	–
	F middle	38.1	4.75	19.05	38.1	76.19	23.81	–
	M senior	28.57	0	23.81	47.62	33.33	66.67	–
	F senior	5.26	0	57.9	36.84	73.68	26.32	–
	All	27.48	0.76	27.48	44.28	60.31	39.69	–
Kitchen	M young	75	0	10	15	40	30	30
	F young	17.39	0	34.78	47.83	8.7	8.7	82.6
	M middle	30	0	20	50	15	10	75
	F middle	19.05	0	33.33	47.62	28.57	0	71.43
	M senior	19.05	0	19.05	61.9	57.13	23.83	19.04
	F senior	10.53	0	57.89	31.58	47.37	31.58	21.05
	All	27.48	0	26.72	45.8	35.11	15.27	49.62
Fitness	M young	80	0	10	10	30	5	65
	F young	8.7	0	26.08	65.22	30.43	4.35	65.22
	M middle	60	0	5	35	15	0	85
	F middle	61.9	0	28.58	9.52	9.52	0	90.48
	M senior	38.1	0	19.05	42.85	47.62	9.52	42.86
	F senior	42.11	0	42.11	15.78	26.32	0.0	73.68
	All	46.56	1.53	18.32	33.59	39.69	4.58	55.73
Entrance	M young	50	0	0	50	60	40	–
	F young	8.7	0	21.74	69.56	65.22	34.78	–
	M middle	35	5	10	50	30	70	–
	F middle	19.05	4.76	19.05	57.14	71.43	28.57	–
	M senior	19.05	0	38.1	42.85	47.62	52.38	–
	F senior	5.26	0	47.37	47.37	57.89	42.11	–
	All	25.19	0.76	19.85	54.2	54.2	45.8	–

Table 4: Results of choice simulations for different hair length preferences and clothing preferences grouped according to context and different groups of participants.

		female assistant	male assistant
Leisure	competence Mean (SD)	5.42 (1.16)	5.19 (1.41)
	warmth Mean (SD)	4.68 (1.36)	4.57 (1.63)
Kitchen	competence Mean (SD)	5.53 (1.09)	5.84 (1.01)
	warmth Mean (SD)	4.97 (1.32)	5.34 (1.15)
Fitness	competence Mean (SD)	5.37 (1.14)	5.56 (1.11)
	warmth Mean (SD)	4.89 (1.37)	4.88 (1.11)
Entrance	competence Mean (SD)	5.51 (1.17)	5.68 (.93)
	warmth Mean (SD)	5.06 (1.39)	5.47 (1.09)

Table 5: Means and standard deviations of attributed competence and warmth to female and male virtual assistants grouped according to context.

Bibliography

[ABG$^+$18] Jascha Achenbach, Robert Brylka, Thomas Gietzen, Katja zum Hebel, Elmar Schömer, Ralf Schulze, Mario Botsch, and Ulrich Schwanecke. A multilinear model for bidirectional craniofacial reconstruction. In *Proc. of Eurographics Workshop on Visual Computing for Biology and Medicine*, pages 67–76, 2018.

[ACP03] Brett Allen, Brian Curless, and Zoran Popović. The space of human body shapes: Reconstruction and parameterization from range scans. *ACM Transactions on Graphics*, 22(3):587–594, 2003.

[ACPH06] Brett Allen, Brian Curless, Zoran Popović, and Aaron Hertzmann. Learning a correlated model of identity and pose-dependent body shape variation for real-time synthesis. In *Proc. of Eurographics Symposium on Computer Animation*, pages 147–156, 2006.

[Agi17] Agisoft. Photoscan pro. `http://www.agisoft.com/`, 2017.

[AHLG$^+$13] Dicko Ali-Hamadi, Tiantian Liu, Benjamin Gilles, Ladislav Kavan, François Faure, Olivier Palombi, and Marie-Paule Cani. Anatomy transfer. *ACM Transactions on Graphics*, 32(6):188:1–188:8, 2013.

[AMX$^+$18a] Thiemo Alldieck, Marcus Magnor, Weipeng Xu, Christian Theobalt, and Gerard Pons-Moll. Detailed human avatars from monocular video. In *Proc. of International Conference on 3D Vision*, pages 98–109, 2018.

[AMX$^+$18b] Thiemo Alldieck, Marcus Magnor, Weipeng Xu, Christian Theobalt, and Gerard Pons-Moll. Video based reconstruction of 3d people models. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pages 8387–8397, 2018.

[ANA10] Juan Carlos Augusto, Hideyuki Nakashima, and Hamid Aghajan. *Ambient intelligence and smart environments: A state of the art*, pages 3–31. Springer, 2010.

[APS99] James L. Ackerman, William R. Proffit, and David M. Sarver. The emerging soft tissue paradigm in orthodontic diagnosis and treatment planning. *Clinical Orthodontics and Research*, 2(2):49–52, 1999.

[ARL$^+$09] Oleg Alexander, Mike Rogers, William Lambeth, Matt Chiang, and Paul Debevec. The digital emily project: Photoreal facial modeling and animation. In *ACM SIGGRAPH 2009 Courses*, pages 1–15, 2009.

[ARV07] Brian Amberg, Sami Romdhani, and Thomas Vetter. Optimal step nonrigid ICP algorithms for surface registration. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[ASCE02] Nicolas Aspert, Diego Santa-Cruz, and Touradj Ebrahimi. Mesh: Measuring errors between surfaces using the hausdorff distance. In *Proc. of IEEE International Conference on Multimedia and Expo*, pages 705–708, 2002.

[ASK$^+$05] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. Scape: Shape completion and animation of people. *ACM Transactions on Graphics*, 24(3):408–416, 2005.

[Aut14] Autodesk. Character generator. `https://charactergenerator.autodesk.com/`, 2014.

[AWLB17] Jascha Achenbach, Thomas Waltemate, Marc Erich Latoschik, and Mario Botsch. Fast generation of realistic virtual humans. In *Proc. of ACM Symposium on Virtual Reality Software and Technology*, pages 1–10, 2017.

[AZB15] Jascha Achenbach, Eduard Zell, and Mario Botsch. Accurate face reconstruction through anisotropic fitting and eye correction. In *Proc. of Vision, Modeling & Visualization*, pages 1–8, 2015.

[AZCP13] Akshay Asthana, Stefanos Zafeiriou, Shiyang Cheng, and Maja Pantic. Robust discriminative response map fitting with constrained local models. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pages 3444–3451, 2013.

[Bay11] Amy L. Baylor. The design of motivational agents and avatars. *Educational Technology Research and Development*, 59(2):291–300, 2011.

[BB14] Thabo Beeler and Derek Bradley. Rigid stabilization of facial expressions. *ACM Transactions on Graphics*, 33(4):44:1–44:9, 2014.

[BBB$^+$10] Thabo Beeler, Bernd Bickel, Paul Beardsley, Bob Sumner, and Markus Gross. High-quality single-shot capture of facial geometry. *ACM Transactions on Graphics*, 29(4):40:1–40:9, 2010.

[BBG08] Jeremy N. Bailenson, Jim Blascovich, and Rosanna E. Guadagno. Self-representations in immersive virtual environments. *Journal of Applied Social Psychology*, 38(11):2673–2690, 2008.

[BBGB16] Pascal Bérard, Derek Bradley, Markus Gross, and Thabo Beeler. Lightweight eye capture using a parametric model. *ACM Transactions on Graphics*, 35(4):117:1–117:12, 2016.

[BBK$^+$15] Amit Bermano, Thabo Beeler, Yeara Kozlov, Derek Bradley, Bernd Bickel, and Markus Gross. Detailed spatio-temporal reconstruction of eyelids. *ACM*

Transactions on Graphics, 34(4):44:1–44:11, 2015.

[BBLR15] Federica Bogo, Michael J. Black, Matthew Loper, and Javier Romero. Detailed full-body reconstructions of moving people from monocular rgb-d sequences. In *Proc. of IEEE International Conference on Computer Vision*, pages 2300–2308, 2015.

[BBN+12] Thabo Beeler, Bernd Bickel, Gioacchino Noris, Steve Marschner, Paul Beardsley, Robert W. Sumner, and Markus Gross. Coupled 3D reconstruction of sparse facial hair and skin. *ACM Transactions on Graphics*, 31(4):117:1–117:10, 2012.

[BBN+14] Pascal Bérard, Derek Bradley, Maurizio Nitti, Thabo Beeler, and Markus Gross. High-quality capture of eyes. *ACM Transactions on Graphics*, 33(6):223:1–223:12, 2014.

[BDS+12] Sofien Bouaziz, Mario Deuss, Yuliy Schwartzburg, Thibaut Weise, and Mark Pauly. Shape-up: Shaping discrete geometry with projections. *Computer Graphics Forum*, 31(5):1657–1667, 2012.

[BEK12] Kirsten Bergmann, Friederike Eyssel, and Stefan Kopp. A second chance to make a first impression? how appearance and nonverbal behavior affect perceived warmth and competence of virtual agents over time. In *Proc. of International Conference on Intelligent Virtual Agents*, pages 126–138, 2012.

[BHB+11] Thabo Beeler, Fabian Hahn, Derek Bradley, Bernd Bickel, Paul Beardsley, Craig Gotsman, Robert W. Sumner, and Markus Gross. High-quality passive facial performance capture using anchor frames. *ACM Transactions on Graphics*, 30(4):75:1–75:10, 2011.

[Bjo96] Ake Bjorck. *Numerical methods for least squares problems*, volume 51. Siam, 1996.

[BKP+10] Mario Botsch, Leif Kobbelt, Mark Pauly, Pierre Alliez, and Bruno Levy. *Polygon Mesh Processing*. AK Peters, 2010.

[BM92] Paul J. Besl and Neil D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.

[BM07] Stéphanie Buisine and Jean-Claude Martin. The effects of speech–gesture cooperation in animated agents' behavior in multimedia presentations. *Interacting with Computers*, 19(4):484–493, 2007.

[BML+14] Sofien Bouaziz, Sebastian Martin, Tiantian Liu, Ladislav Kavan, and Mark Pauly. Projective dynamics: fusing constraint projections for fast simulation. *ACM Transactions on Graphics*, 33(4):154:1–154:11, 2014.

[BP93] Elizabeth Brown and David I. Perrett. What gives a face its gender? *Perception*, 22(7):829–840, 1993.

[BP07] Ilya Baran and Jovan Popović. Automatic rigging and animation of 3d characters. *ACM Transactions on Graphics*, 26(3), 2007.

[BR07] Benedict J. Brown and Szymon Rusinkiewicz. Global non-rigid alignment of 3-D scans. *ACM Transactions on Graphics*, 26(3), 2007.

[BRLB14] Federica Bogo, Javier Romero, Matthew Loper, and Michael J. Black. Faust: Dataset and evaluation for 3d mesh registration. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pages 3794–3801, 2014.

[BS91] Jules Bloomenthal and Ken Shoemake. Convolution surfaces. In *Proc. of the ACM on Computer Graphics and Interactive Techniques*, pages 251–256, 1991.

[BS08] Mario Botsch and Olga Sorkine. On linear variational surface deformation methods. *IEEE Transactions on Visualization and Computer Graphics*, 14(1):213–230, 2008.

[BS14] Domna Banakou and Mel Slater. Body ownership causes illusory self-attribution of speaking and influences subsequent real speaking. *Proc. of the National Academy of Sciences*, 111(49):17678–17683, 2014.

[BSB14] Francesco Bonarrigo, Alberto Signoroni, and Mario Botsch. Deformable registration using patch-wise shape matching. *Graphical Models*, 76(5):554–565, 2014.

[BSS07] Volker Blanz, Kristina Scherbaum, and Hans-Peter Seidel. Fitting a morphable model to 3D scans of faces. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[BTP14] Sofien Bouaziz, Andrea Tagliasacchi, and Mark Pauly. Dynamic 2D/3D registration. In *Eurographics Tutorials*, 2014.

[BTS$^+$14] Matthew Berger, Andrea Tagliasacchi, Lee Seversky, Pierre Alliez, Joshua Levine, Andrei Sharf, and Claudio Silva. State of the art in surface reconstruction from point clouds. In *Eurographics State of the Art Reports*, pages 161–185, 2014.

[Bus04] Samuel R. Buss. Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods. *IEEE Journal of Robotics and Automation*, 17, 2004.

[BV99] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *Proc. of ACM SIGGRAPH*, pages 187–194, 1999.

[BW13] Timo Bolkart and Stefanie Wuhrer. Statistical analysis of 3d faces in motion. In *Proc. of International Conference on 3D Vision*, pages 103–110, 2013.

[BW16] Timo Bolkart and Stefanie Wuhrer. A robust multilinear model learning framework for 3d faces. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pages 4911–4919, 2016.

[BWP13] Sofien Bouaziz, Yangang Wang, and Mark Pauly. Online modeling for realtime facial animation. *ACM Transactions on Graphics*, 32(4):40:1–40:10, 2013.

[CBZB15] Chen Cao, Derek Bradley, Kun Zhou, and Thabo Beeler. Real-time high-fidelity facial performance capture. *ACM Transactions on Graphics*, 34(4):46:1–46:9, 2015.

[CDC10] Charles E. Cunningham, Ken Deal, and Yvonne Chen. Adaptive choice-based conjoint analysis. *The Patient: Patient-Centered Outcomes Research*, 3(4):257–273, 2010.

[Cha13] Kyung-Suk Cha. Soft-tissue thickness of south korean adults with normal facial profiles. *Korean Journal of Orthodontics*, 43(4):178–185, 2013.

[CHP03] Paul A. Caruso, Gordon J. Harris, and Bonnie L. Padwa. Ct imaging of craniofacial malformations. *Neuroimaging Clinics of North America*, 13(3):541–572, 2003.

[CHZ14] Chen Cao, Qiming Hou, and Kun Zhou. Displaced dynamic expression regression for real-time facial tracking and animation. *ACM Transactions on Graphics*, 33(4):42:1–42:10, 2014.

[CKGIS11] Veena Chattaraman, Wi-Suk Kwon, Juan E. Gilbert, and Soo In Shim. Virtual agents in e-commerce: representational characteristics for seniors. *Journal of Research in Interactive Marketing*, 5(4):276–297, 2011.

[CLM$^+$10] Will Chang, Hao Li, Niloy Mitra, Mark Pauly, and Michael Wand. Geometric registration for deformable shapes. In *Eurographics Tutorials*, 2010.

[CM92] Yang Chen and Gérard Medioni. Object modelling by registration of multiple range images. *Image and Vision Computing*, 10(3):145–155, 1992.

[CMB01] Yung Kyun Choi, Gordon E. Miracle, and Frank Biocca. The effects of anthropomorphic agents on advertising effectiveness and the mediating role of presence. *Journal of Interactive Advertising*, 2(1):19–32, 2001.

[CS16] Jodi Caple and Carl N. Stephan. A standardized nomenclature for craniofacial and facial anthropometry. *International Journal of Legal Medicine*, 130(3):863–879, 2016.

[CVDG+06] Peter Claes, Dirk Vandermeulen, Sven De Greef, Guy Willems, and Paul Suetens. Statistically deformable face models for cranio-facial reconstruction. *Journal of Computing and Information Technology*, 14(1):21–30, 2006.

[CWZ+14] Chen Cao, Yanlin Weng, Shun Zhou, Yiying Tong, and Kun Zhou. Face-Warehouse: A 3D facial expression database for visual computing. *IEEE Transactions on Visualization and Computer Graphics*, 20(3):413–425, 2014.

[DDB+15] Mario Deuss, Anders Holden Deleuran, Sofien Bouaziz, Bailin Deng, Daniel Piker, and Mark Pauly. Shapeop—a robust and extensible geometric modelling paradigm. In *Modelling Behaviour*, pages 505–515. Springer, 2015.

[Dev89] Patricia G. Devine. Stereotypes and prejudice: Their automatic and controlled components. *Journal of personality and social psychology*, 56(1):5–18, 1989.

[DL97] Lieven De Lathauwer. *Signal Processing Based on Multilinear Algebra*. PhD thesis, Katholike Universiteit Leuven, 1997.

[DvM00] Doris M. Dehn and Susanne van Mulken. The impact of animated interface agents: A review of empirical research. *International journal of human-computer studies*, 52(1):1–22, 2000.

[DWYW09] Nicolas Ducheneaut, Ming-Hui Wen, Nicholas Yee, and Greg Wadley. Body and mind: a study of avatar personalization in three virtual worlds. In *Proc. of SIGCHI Conference on Human Factors in Computing Systems*, pages 1151–1160, 2009.

[ECSS04] Directorate-General for Energy European Commission, Transport ; Directorate H Nuclear Safety, and Safeguards. Radiation protection 136 : European guidelines on radiation protection in dental radiology; the safe use of radiographs in dental practice. `https://ec.europa.eu/energy/sites/ener/files/documents/136_0.pdf`, 2004. Accessed 2018-08-21.

[EF78] Paul Ekman and Wallace Friesen. *Facial Action Coding System: A Technique for the Measurement of Facial Movement*. Consulting Psychologists Press, 1978.

[EH12] Friederike Eyssel and Frank Hegel. (S) he's got the look: gender stereotyping of robots. *Journal of Applied Social Psychology*, 42(9):2213–2230, 2012.

[FCG07] Susan T. Fiske, Amy J. C. Cuddy, and Peter Glick. Universal dimensions of social cognition: Warmth and competence. *Trends in cognitive sciences*, 11(2):77–83, 2007.

[FCS15] Andrew Feng, Dan Casas, and Ari Shapiro. Avatar reshaping and automatic rigging using a deformable model. In *Proc. of ACM Motion in Games*, pages 57–64, 2015.

[FGT$^+$16] Graham Fyffe, Paul Graham, Borom Tunwattanapong, Abhijeet Ghosh, and Paul Debevec. Near-instant capture of high-resolution facial geometry and reflectance. 35(2):353–363, 2016.

[FRS17] Andrew Feng, Evan Suma Rosenberg, and Ari Shapiro. Just-in-time, viable, 3d avatars from scans. In *Proc. of ACM SIGGRAPH*, pages 19:1–19:2, 2017.

[FSR$^+$14] Andrew Feng, Ari Shapiro, Wang Ruizhe, Mark Bolas, Gerard Medioni, and Evan Suma. Rapid avatar capture and simulation using commodity depth sensors. In *Proc. of ACM SIGGRAPH*, pages 16:1–16:1, 2014.

[Gar00] Toby Gard. Building character. *Game Developer*, pages 28–37, 2000.

[GBA$^+$19] Thomas Gietzen, Robert Brylka, Jascha Achenbach, Katja zum Hebel, Elmar Schömer, Mario Botsch, Ulrich Schwanecke, and Ralf Schulze. A method for automatic forensic facial reconstruction based on dense statistics of soft tissue thickness. *PLOS ONE*, 14(1):1–19, 2019.

[GBSS17] Thomas Gietzen, Robert Brylka, Ulrich Schwanecke, and Elmar Schömer. A dense statistical model of facial soft tissue thickness. In *INFORMATIK*, pages 891–898, 2017.

[GFPMSS10] Mar González-Franco, Daniel Perez-Marcos, Bernhard Spanlang, and Mel Slater. The contribution of real-time mirror reflections of motor actions on virtual body ownership in an immersive virtual environment. In *Proc. of IEEE Virtual Reality Conference*, pages 111–114, 2010.

[GFT$^+$11] Abhijeet Ghosh, Graham Fyffe, Borom Tunwattanapong, Jay Busch, Xueming Yu, and Paul Debevec. Multiview face capture using polarized spherical gradient illumination. *ACM Transactions on Graphics*, 30(6):129:1–129:10, 2011.

[GH97] Michael Garland and Paul S. Heckbert. Surface simplification using quadric error metrics. In *Proc. of the ACM on Computer Graphics and Interactive Techniques*, SIGGRAPH '97, pages 209–216, 1997.

[GKP03] Jennifer Goetz, Sara Kiesler, and Aaron Powers. Matching robot appearance and behavior to tasks to improve human-robot cooperation. In *Proc. of IEEE International Workshop on robot and Human Interactive Communication*, pages 55–60, 2003.

[GKW01] Paul E. Green, Abba M. Krieger, and Yoram Wind. Thirty years of conjoint analysis: Reflections and prospects. *Interfaces*, 31(3 - b):56–73, 2001.

[Gmb18] Fittkau & Maaß Consulting GmbH. Digitaler Assistent als Ersatz für die Musikanlage. `https://http://www.fittkaumaass.de/news/digitaler-assistent-als-ersatz-fuer-die-musikanlage`, 2018. Accessed 2016-07-03.

[GS78] Paul E. Green and Venkatachary Srinivasan. Conjoint analysis in consumer research: issues and outlook. *Journal of consumer research*, 5(2):103–123, 1978.

[GWBB09] Peng Guan, Alexander Weiss, Alexandru Balan, and Michael J. Black. Estimating human shape and pose from a single image. In *Proc. of IEEE International Conference on Computer Vision*, pages 1381–1388, 2009.

[GZC$^+$16] Pablo Garrido, Michael Zollhöfer, Dan Casas, Levi Valgaerts, Kiran Varanasi, Patrick Pérez, and Christian Theobalt. Reconstruction of personalized 3d face rigs from monocular video. *ACM Transactions on Graphics*, 35(3):28:1–28:15, 2016.

[HAAH02] Morten Hertzum, Hans H. K. Andersen, Verner Andersen, and Camilla B. Hansen. Trust in information sources: seeking information from people, documents, and virtual agents. *Interacting with computers*, 14(5):575–599, 2002.

[HAWG08] Qi-Xing Huang, Bart Adams, Martin Wicke, and Leonidas J. Guibas. Non-rigid registration under isometric deformations. *Computer Graphics Forum*, 27(5):1449–1457, 2008.

[HB05] Cynthia Hoffner and Martha Buchanan. Young adults' wishful identification with television characters: The role of perceived similarity and character attributes. *Media psychology*, 7(4):325–351, 2005.

[HCH$^+$15] Hyeon-Shik Hwang, Seon-Yeong Choe, Ji-Sup Hwang, Da-Nal Moon, Yanan Hou, Won-Joon Lee, and Caroline Wilkinson. Reproducibility of facial soft tissue thickness measurements using cone-beam ct images according to the measurement methods. *Journal of Forensic Sciences*, 60(4):957–965, 2015.

[HLRB12] David A. Hirshberg, Matthew Loper, Eric Rachlin, and Michael J. Black. Coregistration: Simultaneous alignment and modeling of articulated 3d shape. In *Proc. of European Conference on Computer Vision*, pages 242–255, 2012.

[HMYL15] Pei-Lun Hsieh, Chongyang Ma, Jihun Yu, and Hao Li. Unconstrained realtime facial performance capture. In *Proc. of IEEE Computer Vision and*

Pattern Recognition, pages 1675–1683, 2015.

[Hor87] Berthold K. P. Horn. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America A*, 4(4):629–642, 1987.

[HP04] Klaus Hildebrandt and Konrad Polthier. Anisotropic filtering of non-linear surface features. *Computer Graphics Forum*, 23(3):391–400, 2004.

[HS96] David L. Hamilton and Steven J. Sherman. Perceiving persons and groups. *Psychological review*, 103(2):336–355, 1996.

[HS13] Lei He and Scott Schaefer. Mesh denoising via L0 minimization. *ACM Transactions on Graphics*, 32(4):64:1–64:8, 2013.

[HSS$^+$09] Nils Hasler, Carsten Stoll, Martin Sunkel, Bodo Rosenhahn, and Hans-Peter Seidel. A statistical model of human pose and body shape. *Computer Graphics Forum*, 28(2):337–346, 2009.

[HSW$^+$17] Liwen Hu, Shunsuke Saito, Lingyu Wei, Koki Nagano, Jaewoo Seo, Jens Fursund, Iman Sadeghi, Carrie Sun, Yen-Chun Chen, and Hao Li. Avatar digitization from a single image for real-time rendering. *ACM Transactions on Graphics*, 36(6):195:1–195:14, 2017.

[HTKK14] Utsuno Hajime, Kageyama Toru, Uchida Keiichi, and Kibayashi Kazuhiko. Facial soft tissue thickness differences among three skeletal classes in japanese population. *Forensic Science International*, 236:175–180, 2014.

[IBP15] Alexandru Eugen Ichim, Sofien Bouaziz, and Mark Pauly. Dynamic 3d avatar creation from hand-held video input. *ACM Transactions on Graphics*, 34(4):45:1–45:14, 2015.

[IKKP17] Alexandru-Eugen Ichim, Petr Kadleček, Ladislav Kavan, and Mark Pauly. Phace: Physics-based face modeling and animation. *ACM Transactions on Graphics*, 36(4):153:1–153:14, 2017.

[IKNDP16] Alexandru-Eugen Ichim, Ladislav Kavan, Merlin Nimier-David, and Mark Pauly. Building and animating user-specific volumetric face rigs. In *Proc. of ACM SIGGRAPH / Eurographics Symposium on Computer Animation*, pages 107–117, 2016.

[JB09] Seung-A. Annie Jin and Justin Bolebruch. Avatar-based advertising in second life: The role of presence and attractiveness of virtual spokespersons. *Journal of Interactive Advertising*, 10(1):51–60, 2009.

[JSW05] Tao Ju, Scott Schaefer, and Joe Warren. Mean value coordinates for closed triangular meshes. *ACM Transactions on Graphics*, 24(3):561–566, 2005.

[JVH08] Richard D. Johnson, Natasha F. Veltri, and Steven Hornik. Attributions of responsibility toward computing technology: The role of interface social cues and user gender. *International Journal of Human–Computer Interaction*, 24(6):595–612, 2008.

[KBJM18] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pages 7122–7131, 2018.

[KHS03] Kolja Kähler, Jörg Haber, and Hans-Peter Seidel. Reanimating the dead: Reconstruction of expressive faces from skull data. *ACM Transactions on Graphics*, 22(3):554–561, 2003.

[KIL$^+$16] Petr Kadleček, Alexandru-Eugen Ichim, Tiantian Liu, Jaroslav Křivánek, and Ladislav Kavan. Reconstructing personalized anatomical models for physics-based body animation. *ACM Transactions on Graphics*, 35(6):213:1–213:13, 2016.

[KO96] William Joseph King and Jun Ohya. The representation of agents: Anthropomorphism, agency, and intelligence. In *Proc. of ACM Conference Companion on Human Factors in Computing Systems*, pages 289–290, 1996.

[KPS03] Jure Kovac, Peter Peer, and Franc Solina. Human skin color clustering for face detection. In *Proc. of IEEE Region 8 EUROCON 2003. Computer as a Tool.*, volume 2, pages 144–148, 2003.

[LAGP09] Hao Li, Bart Adams, Leonidas J. Guibas, and Mark Pauly. Robust single-view geometry and motion reconstruction. *ACM Transactions on Graphics*, 28(5):175:1–175:10, 2009.

[LAR$^+$14] J. P. Lewis, Ken Anjyo, Taehyun Rhee, Mengjie Zhang, Fred Pighin, and Zhigang Deng. Practice and theory of blendshape facial models. In *Eurographics State of the Art Reports*, 2014.

[LBB$^+$17] Tianye Li, Timo Bolkart, Michael J. Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4d scans. *ACM Transactions on Graphics*, 36(6):194:1–194:17, 2017.

[LC87] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM SIGGRAPH Computer Graphics*, 21(4):163–169, 1987.

[LFDK07] Ian Li, Jodi Forlizzi, Anind Dey, and Sara Kiesler. My agent as myself or another: effects on credibility and listening to advice. In *Proc. of ACM Conference on Designing Pleasurable Products and Interfaces*, pages 194–208, 2007.

[LKS14] Shu Liang, Ira Kemelmacher-Shlizerman, and Linda G. Shapiro. 3D face hallucination from a single depth frame. In *Proc. of International Conference on 3D Vision*, pages 31–38, 2014.

[LLL15] Jean-Luc Lugrin, Johanna Latt, and Marc Erich Latoschik. Anthropomorphism and illusion of virtual body ownership. In *Proc. of International Conference on Artificial Reality and Telexistence and Eurographics Symposium on Virtual Environments*, pages 1–8, 2015.

[LLR16] Marc Erich Latoschik, Jean-Luc Lugrin, and Daniel Roth. Fakemi: a fake mirror system for avatar embodiment studies. In *Proc. of ACM Symposium on Virtual Reality Software and Technology*, pages 73–76, 2016.

[LMB14] Matthew Loper, Naureen Mahmood, and Michael J. Black. Mosh: Motion and shape capture from sparse markers. *ACM Transactions on Graphics*, 33(6):220:1–220:13, 2014.

[LMR$^+$15] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. Smpl: A skinned multi-person linear model. *ACM Transactions on Graphics*, 34(6):248:1–248:16, 2015.

[LRDB06] Gary L. Lilien, Arvind Rangaswamy, and Arnaud De Bruyn. *Conjoint Analysis: Marketing Engineering technical note*. Principles of Marketing Engineering, 2006.

[LRG$^+$17] Marc Erich Latoschik, Daniel Roth, Dominik Gall, Jascha Achenbach, Thomas Waltemate, and Mario Botsch. The effect of avatar realism in immersive social virtual realities. In *Proc. of ACM Symposium on Virtual Reality Software and Technology*, pages 39:1–39:10, 2017.

[LSP08] Hao Li, Robert W. Sumner, and Mark Pauly. Global correspondence optimization for non-rigid registration of depth scans. *Computer Graphics Forum*, 27(5):1421–1430, 2008.

[LVG$^+$13] Hao Li, Etienne Vouga, Anton Gudym, Linjie Luo, Jonathan T. Barron, and Gleb Gusev. 3d self-portraits. *ACM Transactions on Graphics*, 32(6):187:1–187:9, 2013.

[LWP10] Hao Li, Thibaut Weise, and Mark Pauly. Example-based facial rigging. *ACM Transactions on Graphics*, 29(4):32:1–32:6, 2010.

[MB93] Muhammad J. Mirza and Kim L. Boyer. Performance evaluation of a class of M-estimators for surface parameter estimation in noisy range data. *IEEE Transactions on Robotics and Automation*, 9(1):75–85, 1993.

[MF06] Roxana Moreno and Terri Flowerday. Students' choice of animated pedagogical agents in science learning: A test of the similarity-attraction hypothesis on gender and ethnicity. *Contemporary educational psychology*, 31(2):186–207, 2006.

[MK01] Daniel G. McDonald and Hyeok Kim. When i die, i feel small: Electronic game characters and the social self. *Journal of Broadcasting & Electronic Media*, 45(2):241–258, 2001.

[MKK$^+$17] Charles Malleson, Maggie Kosek, Martin Klaudiny, Ivan Huerta, Jean-Charles Bazin, Alexander Sorkine-Hornung, Mark Mine, and Kenny Mitchell. Rapid one-shot acquisition of dynamic vr avatars. In *Proc. of IEEE Virtual Reality*, pages 131–140, 2017.

[MMB94] C. Neil Macrae, Alan B. Milne, and Galen V. Bodenhausen. Stereotypes as energy-saving devices: A peek inside the cognitive toolbox. *Journal of personality and Social Psychology*, 66(1):37–47, 1994.

[MWR$^+$18] Geraldo Elias Miranda, Caroline Wilkinson, Mark Roughley, Thiago Leite Beaini, and Rodolfo Francisco Haltenhoff Melani. Assessment of accuracy and recognition of three-dimensional computerized forensic craniofacial reconstruction. *PLOS ONE*, 13(5):1–13, 2018.

[NFG12] Franz J. Neyer, Juliane Felber, and Claudia Gebhardt. Entwicklung und Validierung einer Kurzskala zur Erfassung von Technikbereitschaft. *Diagnostica*, 58(2):87–99, 2012.

[Nie94] Jakob Nielsen. *Usability Engineering*. Morgan Kaufmann, 1994.

[OLPM$^+$18] Mohamed Omran, Christoph Lassner, Gerard Pons-Moll, Peter Gehler, and Bernt Schiele. Neural body fitting: Unifying deep learning and model based human pose and shape estimation. In *Proc. of International Conference on 3D Vision*, pages 484–494, 2018.

[OLY$^+$17] Kyle Olszewski, Zimo Li, Chao Yang, Yi Zhou, Ronald Yu, Zeng Huang, Sitao Xiang, Shunsuke Saito, Pushmeet Kohli, and Hao Li. Realistic dynamic facial textures from a single image using gans. In *Proc. of IEEE International Conference on Computer Vision*, pages 5429–5438, 2017.

[PB11] Chavdar Papazov and Darius Burschka. Deformable 3D shape registration based on local similarity transforms. *Computer Graphics Forum*, 30(5):1493–1502, 2011.

[PGB03] Patrick Pérez, Michel Gangnet, and Andrew Blake. Poisson image editing. *ACM Transactions on Graphics*, 22(3):313–318, 2003.

[PMPHB17] Gerard Pons-Moll, Sergi Pujades, Sonny Hu, and Michael J. Black. Clothcap: Seamless 4d clothing capture and retargeting. *ACM Transactions on Graphics*, 36(4):73:1–73:15, 2017.

[Pol02] Konrad Polthier. Polyhedral surfaces of constant mean curvature. Habilitation Thesis, Berlin Technical University, 2002.

[PSAS13] Tabitha C. Peck, Sofia Seinfeld, Salvatore M. Aglioti, and Mel Slater. Putting yourself in the skin of a black avatar reduces implicit racial bias. *Consciousness and cognition*, 22(3):779–787, 2013.

[PSRJ13] Jeunese Payne, Andrea Szymkowiak, Paul Robertson, and Graham Johnson. Gendering the machine: Preferred virtual assistant gender and realism in self-service. In *Proc. of International Workshop on Intelligent Virtual Agents*, pages 106–115, 2013.

[PWH$^+$17] Leonid Pishchulin, Stefanie Wuhrer, Thomas Helten, Christian Theobalt, and Bernt Schiele. Building statistical shape spaces for 3d human modeling. *Pattern Recognition*, 67(C):276–286, 2017.

[QB05] Lingyun Qiu and Izak Benbasat. Online consumer trust and live help interfaces: The effects of text-to-speech voice and three-dimensional avatars. *International journal of human-computer interaction*, 19(1):75–94, 2005.

[QB10] Lingyun Qiu and Izak Benbasat. A study of demographic embodiments of product recommendation agents in electronic commerce. *International Journal of Human-Computer Studies*, 68(10):669–688, 2010.

[RBSB18] Anurag Ranjan, Timo Bolkart, Soubhik Sanyal, and Michael J. Black. Generating 3D faces using convolutional mesh autoencoders. In *Proc. of European Conference on Computer Vision*, volume 11207, pages 725–741, 2018.

[RC11] Radu Bogdan Rusu and Steve Cousins. 3D is here: Point cloud library (PCL). In *Proc. of IEEE International Conference on Robotics and Automation*, pages 1–4, 2011.

[RE13] Natalia Reich and Friederike Eyssel. Attitudes towards service robots in domestic environments: The role of personality characteristics, individual interests, and demographic variables. *Paladyn, Journal of Behavioral Robotics*, 4(2):123–130, 2013.

[RG04] Laurie A. Rudman and Stephanie A. Goodwin. Gender differences in automatic in-group bias: Why do women like women more than men like men? *Journal of personality and social psychology*, 87(4):494–509, 2004.

[RKBPD08] Rinat B. Rosenberg-Kima, Amy L. Baylor, Ashby Plant, and Celeste E. Doerr. Interface agents as social models for female students: The effects of agent visual presence and appearance on female students' attitudes and beliefs. *Computers in Human Behavior*, 24(6):2741–2756, 2008.

[RL01] Szymon Rusinkiewicz and Marc Levoy. Efficient variants of the ICP algorithm. In *Proc. of International Conference on 3D Digital Imaging and*

Modeling, pages 145–152, 2001.

[RME+14] Rafael Romeiro, Ricardo Marroquim, Claudio Esperança, Andreia Breda, and Carlos Marcelo Figueredo. Forensic facial reconstruction using mesh template deformation with detail transfer over hrbf. In *Proc. of Conference on Graphics, Patterns and Images*, pages 266–273, 2014.

[RTB17] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics*, 36(6):245:1–245:17, 2017.

[Rui15] Peter A. M. Ruijten. *Responses to human-like artificial agents: effects of user and agent characteristics*. PhD thesis, Eidnhoven University of Technology, 2015.

[RWP10] Christopher Rynn, Caroline Wilkinson, and Heather L. Peters. Prediction of nasal morphology from the skull. *Forensic Science, Medicine, and Pathology*, 6(1):20–34, 2010.

[RWS+16] Daniel Roth, Kristoffer Waldow, Felix Stetter, Gary Bente, Marc Erich Latoschik, and Arnulph Fuhrmann. Siamc: a socially immersive avatar mediated communication platform. In *Proc. of ACM Symposium on Virtual Reality Software and Technology*, pages 357–358, 2016.

[SA07] Olga Sorkine and Marc Alexa. As-rigid-as-possible surface modeling. In *Proc. of Symposium on Geometry Processing*, volume 4, pages 109–116, 2007.

[SBB07] Leonid Sigal, Alexandru O. Balan, and Michael J. Black. Combined discriminative and generative articulated pose and non-rigid shape estimation. In *Proc. of International Conference on Neural Information Processing Systems*, pages 1337–1344, 2007.

[SBE+16] Birte Schiffhauer, Jasmin Bernotat, Friederike Eyssel, Rebecca Bröhl, and Jule Adriaans. Let the user decide! user preferences regarding functions, apps, and control modalities of a smart apartment and a service robot. *Lecture Notes in Artificial Intelligence*, 9979:971–981, 2016.

[SBKC13] Jürgen Sturm, Erik Bylow, Fredrik Kahl, and Daniel Cremers. Copyme3d: Scanning and printing persons in 3d. In *Proc. of German Conference on Pattern Recognition*, pages 405–414, 2013.

[SHRB12] Matthias Straka, Stefan Hauswiesner, Matthias Ruther, and Horst Bischof. Rapid skin: Estimating the 3d human pose and shape in real-time. In *Proc. of International Conference on 3D Imaging, Modeling, Processing, Visualization Transmission*, pages 41–48, 2012.

[Si15] Hang Si. Tetgen, a delaunay-based quality tetrahedral mesh generator. *ACM Transactions on Mathematical Software*, 41(2):11:1–11:36, 2015.

[SKR$^+$06] Carsten Stoll, Zachi Karni, Christian Rössl, Hitoshi Yamauchi, and Hans-Peter Seidel. Template deformation for point cloud fitting. In *Proc. of Symposium on Point-Based Graphics*, pages 27–35, 2006.

[Sof17] Sawtooth Software. Lighthouse studio. `http://www.sawtoothsoftware.com/`, 2017.

[SP04] Robert W. Sumner and Jovan Popović. Deformation transfer for triangle meshes. *ACM Transactions on Graphics*, 23(3):399–405, 2004.

[SPC$^+$16] Ben Shneiderman, Catherine Plaisant, Maxine S. Cohen, Steven Jacobs, Niklas Elmqvist, and Nicholas Diakopoulos. *Designing the user interface: strategies for effective human-computer interaction*. Pearson, 2016.

[SS08] Carl N. Stephan and Ellie K. Simpson. Facial soft tissue depths in craniofacial identification (part i): An analytical review of the published adult data. *Journal of Forensic Sciences*, 53(6):1257–1272, 2008.

[SSJ$^+$15] Youssef Shiban, Iris Schelhorn, Verena Jobst, Alexander Hörnlein, Frank Puppe, Paul Pauli, and Andreas Mühlberger. The appearance effect: Influences of virtual agent features on performance and motivation. *Computers in Human Behavior*, 49:5–11, 2015.

[SSP07] Robert W. Sumner, Johannes Schmid, and Mark Pauly. Embedded deformation for shape manipulation. *ACM Transactions on Graphics*, 26(3):80:1–80:7, 2007.

[SSSVB10] Mel Slater, Bernhard Spanlang, Maria V. Sanchez-Vives, and Olaf Blanke. First person experience of body transfer in virtual reality. *PLOS ONE*, 5(5):1–9, 2010.

[Ste17] Carl N. Stephan. 2018 tallied facial soft tissue thicknesses for adults and sub-adults. *Forensic Science International*, 280:113–123, 2017.

[SWTC14] Fuhao Shi, Hsiang-Tao Wu, Xin Tong, and Jinxiang Chai. Automatic acquisition of high-fidelity facial performances using monocular videos. *ACM Transactions on Graphics*, 33(6):222:1–222:13, 2014.

[SZD$^+$16] Wuyang Shui, Mingquan Zhou, Qingqiong Deng, Zhongke Wu, Yuan Ji, Kang Li, Taiping He, and Haiyan Jiang. Densely calculated facial soft tissue thickness for craniofacial reconstruction in chinese adults. *Forensic Science International*, 266:573:1–573:12, 2016.

[SZM$^+$17] Wuyang Shui, Mingquan Zhou, Steve Maddock, Taiping He, Xingce Wang, and Qingqiong Deng. A pca-based method for determining craniofacial relationship and sexual dimorphism of facial shapes. *Computers in Biology and Medicine*, 90:33–49, 2017.

[TBK$^+$05] W. D. Turner, R. E. Brown, T. P. Kelliher, P. H. Tu, M. A. Taister, and K. W. Miller. A novel method of automated skull registration for forensic facial approximation. *Forensic Science International*, 154(2-3):149–158, 2005.

[TBL$^+$07] Peter Tu, Rebecca Book, Xiaoming Liu, Nils Krahnstoever, Carl Adrian, and Phil Williams. Automatic face recognition from skeletal remains. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–7, 2007.

[TCL$^+$13] Gary K. L. Tam, Zhi-quan Cheng, Yu-kun Lai, Frank C. Langbein, Yonghuai Liu, David Marshall, Ralph R. Martin, Xian-fang Sun, and Paul L. Rosin. Registration of 3D point clouds and meshes: A survey from rigid to non-rigid. *IEEE Transactions on Visualization and Computer Graphics*, 19(7):1199–1217, 2013.

[TGB13] Jean-Marc Thiery, Émilie Guy, and Tamy Boubekeur. Sphere-meshes: Shape approximation using spherical quadric error metrics. *ACM Transactions on Graphics*, 32(6):178:1–178:12, 2013.

[THHI06] Jose Rafael Tena, Miroslav Hamouz, Adrian Hilton, and John Illingworth. A validated method for dense non-rigid 3D face registration. In *Proc. of IEEE International Conference on Video and Signal Based Surveillance*, 2006.

[TMB14] Aggeliki Tsoli, Naureen Mahmood, and Michael J. Black. Breathing life into shape: Capturing, modeling and animating 3d human breathing. *ACM Transactions on Graphics*, 33(4):52:1–52:11, 2014.

[TPT16] Anastasia Tkach, Mark Pauly, and Andrea Tagliasacchi. Sphere-meshes for real-time hand modeling and tracking. *ACM Transactions on Graphics*, 35(6):222:1–222:11, 2016.

[TTR$^+$17] Anastasia Tkach, Andrea Tagliasacchi, Edoardo Remelli, Mark Pauly, and Andrew Fitzgibbon. Online generative model personalization for hand tracking. *ACM Transactions on Graphics*, 36(6):243:1–243:11, 2017.

[TZG$^+$18] Ayush Tewari, Michael Zollhöfer, Pablo Garrido, Florian Bernard, Hyeongwoo Kim, Patrick Pérez, and Christian Theobalt. Self-supervised multi-level face model learning for monocular reconstruction at over 250 hz. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pages 2549–2559, 2018.

[TZK$^+$17] Ayush Tewari, Michael Zollöfer, Hyeongwoo Kim, Pablo Garrido, Florian Bernard, Patrick Perez, and Theobalt Christian. Mofa: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction.

In *Proc. of IEEE International Conference on Computer Vision*, volume 2, pages 3735–3744, 2017.

[TZL$^+$12] Jing Tong, Jin Zhou, Ligang Liu, Zhigeng Pan, and Hao Yan. Scanning 3d full human bodies using kinects. *IEEE Transactions on Visualization and Computer Graphics*, 18(4):643–650, 2012.

[TZN$^+$15] Justus Thies, Michael Zollhöfer, Matthias Nießner, Levi Valgaerts, Marc Stamminger, and Christian Theobalt. Real-time expression transfer for facial reenactment. *ACM Transactions on Graphics*, 34(6):183:1–183:14, 2015.

[TZS$^+$16] Justus Thies, Michael Zollhofer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Face2face: Real-time face capture and reenactment of rgb videos. In *Proc. of IEEE Computer Vision and Pattern Recognition*, pages 2387–2395, 2016.

[VBPP05] Daniel Vlasic, Matthew Brand, Hanspeter Pfister, and Jovan Popović. Face transfer with multilinear models. *ACM Transactions on Graphics*, 24(3):426–433, 2005.

[Vel10] George Veletsianos. Contextually relevant pedagogical agents: Visual appearance, stereotypes, and first impressions and their impact on learning. *Computers & Education*, 55(2):576–585, 2010.

[WBGB16] Chenglei Wu, Derek Bradley, Markus Gross, and Thabo Beeler. An anatomically-constrained local deformation model for monocular face capture. *ACM Transactions on Graphics*, 35(4):115:1–115:12, 2016.

[WBH$^+$07] Max Wardetzky, Miklós Bergou, David Harmon, Denis Zorin, and Eitan Grinspun. Discrete quadratic curvature energies. *Computer Aided Geometric Design*, 24(8–9):499–518, 2007.

[WBLP11] Thibaut Weise, Sofien Bouaziz, Hao Li, and Mark Pauly. Realtime performance-based facial animation. *ACM Transactions on Graphics*, 30(4):77:1–77:10, 2011.

[WGR$^+$18] Thomas Waltemate, Dominik Gall, Daniel Roth, Mario Botsch, and Marc Erich Latoschik. The impact of avatar personalization and immersion on virtual body ownership, presence, and emotional response. *IEEE Transactions on Visualization and Computer Graphics*, 24(4):1643–1652, 2018.

[WHB11] Alexander Weiss, David Hirshberg, and Michael J. Black. Home 3d body scans from noisy image and range data. In *Proc. of IEEE International Conference on Computer Vision*, pages 1951–1958, 2011.

[Wil10] Caroline Wilkinson. Facial reconstruction–anatomical art or artistic anatomy? *Journal of Anatomy*, 216(2):235–250, 2010.

[WJH+07] Michael Wand, Philipp Jenke, Qixing Huang, Martin Bokeloh, Leonidas Guibas, and Andreas Schilling. Reconstruction of deforming geometry from time-varying point clouds. In *Proc. of Symposium on Geometric Processing*, pages 49–58, 2007.

[WPB+14] Stefanie Wuhrer, Leonid Pishchulin, Alan Brunton, Chang Shu, and Jochen Lang. Estimation of human body shape and posture under clothing. *Computer Vision and Image Understanding*, 127:31–42, 2014.

[YKPK13] Ramin Yaghoubzadeh, Marcel Kramer, Karola Pitsch, and Stefan Kopp. Virtual agents as daily assistants for elderly or cognitively impaired people. In *Proc. of International Workshop on Intelligent Virtual Agents*, pages 79–91, 2013.

[YMYK14] Yusuke Yoshiyasu, Wan-Chun Ma, Eiichi Yoshida, and Fumio Kanehiro. As-conformal-as-possible surface registration. *Computer Graphics Forum*, 33(5):257–267, 2014.

[YSN+18] Shuco Yamaguchi, Shunsuke Saito, Koki Nagano, Yajie Zhao, Weikai Chen, Kyle Olszewski, Shigeo Morishima, and Hao Li. High-fidelity facial reflectance and geometry inference from an unconstrained image. *ACM Transactions on Graphics*, 37(4):162:1–162:14, 2018.

[ZB13] Eduard Zell and Mario Botsch. ElastiFace: Matching and blending textured faces. In *Proc. of International Symposium on Non-Photorealistic Animation and Rendering*, pages 15–24, 2013.

[ZBBB18] Gaspard Zoss, Derek Bradley, Pascal Bérard, and Thabo Beeler. An empirical rig for jaw animation. *ACM Transactions on Graphics*, 37(4):59:1–59:12, 2018.

[ZNI+14] Michael Zollhöfer, Matthias Nießner, Shahram Izadi, Christoph Rehmann, Christopher Zach, Matthew Fisher, Chenglei Wu, Andrew Fitzgibbon, Charles Loop, Christian Theobalt, and Marc Stamminger. Real-time non-rigid reconstruction using an RGB-D camera. *ACM Transactions on Graphics*, 33(4):156:1–156:12, 2014.

[ZPK16] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Fast global registration. In *Proc. of European Conference on Computer Vision*, pages 766–782, 2016.

[ZTG+18] Michael Zollhöfer, Justus Thies, Pablo Garrido, Derek Bradley, Thabo Beeler, Patrick Pérez, Marc Stamminger, Matthias Nießner, and Christian Theobalt. State of the art on monocular 3d face reconstruction, tracking, and applications. 37(2):523–550, 2018.