# Rapid One-Shot Acquisition of Dynamic VR Avatars

Charles Malleson*[1], Maggie Kosek[1,4], Martin Klaudiny[1], Ivan Huerta[1], Jean-Charles Bazin[2], Alexander Sorkine-Hornung[2], Mark Mine[3], and Kenny Mitchell†[1,4]

[1] *Disney Research, UK* [2] *Disney Research, CH* [3] *Walt Disney Imagineering, USA* [4] *Edinburgh Napier University, UK*

**ABSTRACT**

We present a system for rapid acquisition of bespoke, animatable, full-body avatars including face texture and shape. A blendshape rig with a skeleton is used as a template for customization. Identity blendshapes are used to customize the body and face shape at the fitting stage, while animation blendshapes allow the face to be animated. The subject assumes a T-pose and a single snapshot is captured using a stereo RGB plus depth sensor rig. Our system automatically aligns a photo texture and fits the 3D shape of the face. The body shape is stylized according to body dimensions estimated from segmented depth. The face identity blendweights are optimised according to image-based facial landmarks, while a custom texture map for the face is generated by warping the input images to a reference texture according to the facial landmarks. The total capture and processing time is under 10 seconds and the output is a light-weight, game-engine-ready avatar which is recognizable as the subject. We demonstrate our system in a VR environment in which each user sees the other users' animated avatars through a VR headset with real-time audio-based facial animation and live body motion tracking, affording an enhanced level of presence and social engagement compared to generic avatars.

**Keywords:** Avatars, Capture, Virtual Reality

**Index Terms:** I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Virtual Reality

## 1 INTRODUCTION

Since high-quality consumer-level VR headsets such as the Oculus Rift and HTC Vive have become available there has been a proliferation of novel applications in VR both in terms of immersive media consumption and interactive applications. Although these headsets are high-resolution, high frame-rate and low-latency, a remaining obstacle is how to get the user (subject) to feel truly immersed in the experience. One approach to increasing the perception of immersion is for the participant to see a representation of themselves and the other participants in the virtual world. If this 'avatar' tracks their motion and can be recognized as the person it represents, the VR immersion is significantly enhanced as the experience becomes less isolating and more social.

In this paper, a system for rapid acquisition of bespoke avatars for each participant (subject) in a social VR environment is presented. For each subject, the system automatically customizes a parametric avatar model to match the captured subject by adjusting its overall height, body and face shape parameters and generating a custom face texture (in our application the body is in an astronaut character suit, thus we do not customize the texture of the body).

*email:cmalleson@gmail.com
†email:kenny.mitchell@disneyresearch.com

2017 IEEE Virtual Reality (VR)
March 18-22, 2017, Los Angeles, CA, USA
978-1-5090-6647-6/17/$31.00 © 2017 IEEE

A lightweight capture setup consisting of a stereo DSLR rig along with a depth sensor is used (Figure 1a). A single snapshot of each participant in a T-pose is captured and after only a few seconds of processing time on a standard desktop PC, the avatars are available in the VR environment (Figure 1b).

The parametric avatar model contains blendshapes for face and body identity, as well for animation of the face via audio-based lip animation. Blendweights for the rig are fitted from segmented depth data and 3D facial landmarks, while a gender classification on the face image is used to drive a subtle male/female stylization of the avatar. The custom texture map for the face is generated by warping input images to a reference map according to the facial landmarks.

(a) Capture setup

(b) VR avatar

Figure 1: Acquisition setup showing subject in the capture pose and the corresponding output avatar.

The contribution of this paper is a light-weight, fully automatic system for end-to-end capturing of avatars with minimal hardware requirements, a simple, instant capture procedure and fast processing time. The output avatars have custom shape and texture and are skeleton-rigged ready for input into a game engine, where they can be animated directly, making them suitable for VR applications in which recognition and interaction of participants is important.

The remainder of the paper is structured as follows. Section 2 covers related work in the field of VR avatar generation, Section 3 covers our capture hardware setup, Section 4 explains our customizable avatar rig structure and design choices and Section 5 details the method of fitting our template rig to the captured subject. Finally in Section 6, results of the approach are presented and are demonstrated in a representative social VR scenario. The overall pipeline of our system is shown in Figure 2.

## 2 Related Work

Our goal is to acquire an animatable, full-body avatar of the subjects that can then be used in a VR environment (Figure 1). We aim for a fully automatic end-to-end pipeline (from acquisition to final 3D model in the VR experience) running in less than 10 seconds. In the following, we review existing works on this topic.

**3D face reconstruction** Several methods have been proposed for performance-based facial animation, i.e., controlling a virtual character (e.g., a monster's face) via face tracking from RGBD sensors [12, 51] or monocular videos [15, 43], or for reenactment of another person's face in a video [48]. In contrast to tracking, we want to generate an animatable, textured 3D model of the actual subject's head and build his/her full-body 3D avatar, as illustrated in Figure 1b.

The 3D face models of a subject can be obtained in many different ways. One solution is to use sophisticated setups composed of many cameras and structured lights, for example like in the digital Emily project [2] and earlier blue-c [31], but they are not very practical to deploy with low cost. Moreover to create the facial blendshapes for later animation of Emily, the actress has to mimic 40 facial expressions, which is time consuming. In contrast, our method needs just a single snapshot of the subject acquired by a stereo DSLR camera.

Other solutions need several images from a moving camera (or moving subject) [16, 18, 27, 28, 34]. However, in addition to acquisition duration, they also require a long processing time, ranging from tens of seconds to tens of minutes. Some of these references [16, 18, 34] can also generate facial blendshapes/rigs, typically from different facial expressions. However they require additional acquisition that might be time consuming (e.g. around 10 minutes [16]) and have long processing time (several minutes). In contrast, our entire pipeline, including both acquisition and processing, runs in less than 10 seconds.

It is also possible to obtain impressive results from a single shot acquired by several synchronized cameras or stereo camera system [8], but the processing takes several minutes. Other approaches just need a single image obtained by a single camera: the depth ambiguity from a single view is constrained by a 3D template or priors on 3D face shapes learned from a collection of 3D scans. The seminal work of Blanz and Vetter [11] takes several minutes. Later methods can run in a few seconds [36], or even in real-time when the main goal is *not* accurate 3D reconstruction, for example in the context of gaze correction in video calls [29]. However none of these references provide a rigged 3D face model. Our method uses a stereo color camera and, in addition to short processing time, also provides an animatable face model that can be used for speech animation.

**Full-body reconstruction** An accurate and efficient approach to obtain the 3D full-body model of a subject is to use professional 3D body scanners [21]. However they are very expensive, need long acquisition time (up to minutes), require extensive computing resources and/or take several minutes of processing.

Some methods have been proposed to acquire dynamic humans from videos, and thus can be applied at one time instance to acquire a static 3D model. For example, the recent system of Collet et al. [19] obtains impressive results but uses 106 synchronized cameras and the processing time is in the order of several minutes per frame on a single machine. On the other end of the acquisition spectrum, Jain et al. [35] propose a method for monocular videos where the body shape can be manipulated in videos. However it requires few minutes of manual user interaction. In contrast, we aim for a fully automatic method.

More related to our context, some methods have been designed to directly acquire 3D models of humans, for example using RGBD cameras such as with KinectFusion [40]. However this technique requires long acquisition time, and the subject must not move. This constraint can be relaxed using non-rigid alignment [20, 23, 24], where the subject typically rotates in front of one RGBD camera. However these techniques still have a relatively long scanning time (around 30 seconds to 1 minute), and require several minutes [20] or even hours [23, 24] of processing, e.g. due to computationally expensive non-rigid alignment and bundle adjustment. These references return a 3D surface mesh, not a rigged body model. Instead, our method is fast (less than 10 seconds) and provides a 3D body model that can then be animated, e.g. with pre-generated motions or body tracking systems (see results section).

To reduce the processing time, instead of processing a continuous image sequence, some methods use a limited number of RGBD snapshot views where the subject is observed in different orientations, typically between 4 and 10 selected views [39, 46, 50, 52, 53]. The acquisition still takes around 1-2 minutes and the processing time is in the order of minutes. Some of these methods provide a parametric body model but require longer processing time [46, 52]. Similarly, Feng el al. [26] produce good quality reshaping and rig customization from scans, again taking minutes of processing, whereas our requirement is to achieve greater throughput of subjects with acquisition completing in under ten seconds.

To reduce the acquisition time, instead of having the user rotating in front of one camera, several cameras can be used simultaneously, so that the acquisition is instantaneous (one shot). For example Plüss et al. [42] demonstrate a real-time system from 2 RGBD cameras in the context of 3D tele-presence but output a non-closed 3D surface. Tong et al. [49] use 3 synchronized RGBD cameras to generate a body mesh and then compute the skeleton and skin weights [7]. However the processing takes some minutes and the quality of the face is limited. De Aguiar et al. [22] use a single RGBD camera and provide a parametric body model. Song et al. [47] most recently use targeted feature descriptors with silhouettes and a regression scheme to constrain to the parametric body mode. The processing time of both these is in the order of a few seconds, but they target body model stylization. In contrast, our approach aims for a personalized avatar with realistic face of the subject in combination with body shape (see Figure 1b).

Taking advantage of parametric models [3–5, 33, 45], some methods have been proposed to estimate the 3D body from a single picture [32, 55]. While they can provide exciting results and a deformable body model, they require user interaction of some minutes. Instead, our approach runs in a fully automatic manner.

## 3 Capture Setup

Our capture setup was designed to be relatively inexpensive and easy to assemble and calibrate. The capture hardware comprises a

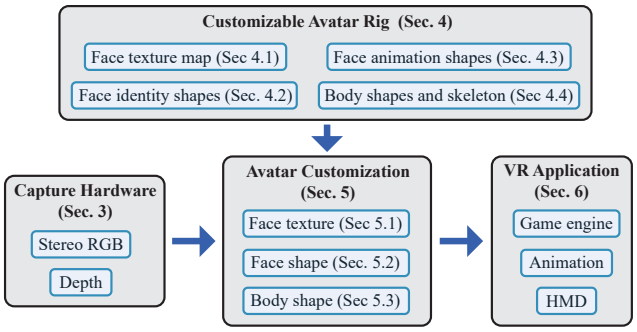



Figure 2: Avatar acquisition pipeline.

stereo RGB camera pair as well as a single RGBD sensor (Figure 3). Two Canon EOS 1200D DSLR cameras are used for the stereo pair, and the Microsoft Kinect v2 is used for depth sensing (RGB from the Kinect is not used). The capture setup is designed to handle subjects ranging in height from 150 to 200 cm (encompassing the 5th to 95th percentile of adult heights [14]) without requiring any mechanical adjustment. The requirements are to capture the face head-on, in stereo with high resolution RGB images for avatar face fitting , and to capture the whole body in the depth map for avatar body fitting. The Canon EDSDK[1] is used to trigger the camera captures and download the images over USB 2, while the Microsoft Kinect SDK[2] is used to capture depth maps from the Kinect over USB 3.

### 3.1 Layout

The DSLR cameras need to be positioned to capture a face with sufficient resolution across the assumed subject height range as shown in Figure 3. Also, the input image for face texture fitting should closely match the reference face view in Figure 4(a) to achieve a satisfying quality (specifically have no yaw and a small negative pitch to cover the bottom of the nose). The camera locations minimize deviation from this ideal viewpoint across the height range. The cameras are mounted 200 cm from the user in portrait orientation with a vertical baseline of 25 cm and toed in so that the full head of a user is visible across the height range (using lenses of focal length 55 mm to obtain a suitable field of view). Having the camera distance, height and baseline as described above means that the deviation from the ideal viewpoint is within 6° in at least one of the cameras (see Figure 8).

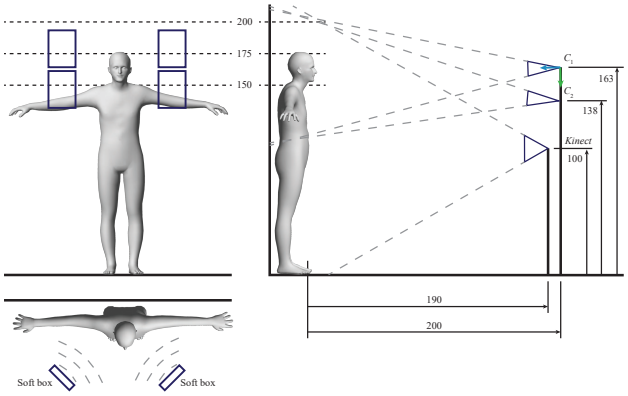



Figure 3: Avatar capture hardware configuration. The rectangles depict illumination soft boxes, $C_1$, $C_2$ and *Kinect* are capture devices.

[1] https://www.didp.canon-europa.com

[2] https://developer.microsoft.com/en-us/windows/kinect

The depth sensor (Kinect) is positioned at a distance of 190 cm from the subject so as to cover the whole of the tallest subjects within the field of view, while the height is chosen such that the device can be level (so that the subject is viewed head-on).

The face is lit using four Mosaic 'soft-box' LED lights positioned 50 cm from the subject, two on either side at 45° to the frontal view. They should illuminate the face without strong reflections or shadows and also not obstruct the camera or depth sensor views.

### 3.2 Calibration

Let $C_1$ refer to the top camera and $C_2$ to the bottom camera. They are set to the maximum resolution ($3456 \times 5184$), manual focus, fixed white balance, exposure, and aperture. The camera models are obtained using a checkerboard chart calibration [54]. An A3 sized chart was found to offer suitable image coverage at the subject range, allowing the intrinsics and extrinsics to be calibrated simultaneously, producing projection matrices $\mathbf{P}_1$ and $\mathbf{P}_2$, for $C_1$ and $C_2$, respectively. The 18-55 mm lenses used were found not to have significant distortion at the long end, thus distortion parameters were not included in the calibration. The standard factory calibration was used for focal length $f_d$ of the Kinect depth map. A background depth plate $D_{bg}$ is captured once to enable subsequent segmentation of each subject for body fitting.

## 4 Customizable Avatar Rig

Each output avatar is an instance of a template rig which is automatically customized according to the captured subject. This rig consists of a skeleton-rigged, skinned, textured mesh with blendshapes for body and face shape customization as well as blendshapes for eye and mouth animation. Using blendshape allows for efficient animation and shape adjustment of 3D meshes [37]. The overall size of the avatar, the body and face shape, and the face texture map are all estimated by the fitting process (refer to Figure 2).

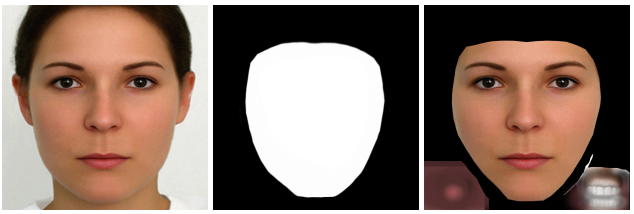
### 4.1 Face Texture Map

The face region of the texture map is customised for each subject with the texture of their face warped to align to the reference texture (Figure 4a) by the fitting process. The layout of the face texture map needs to be designed so as to produce a visually pleasing final result (limiting the appearance of misalignment or stretching on the rendered mesh). At the same time the texture map needs to closely resemble an image of the face, so as to limit distortion caused by the texture warping process. The resulting texture map (Figure 4) is based on an orthographic projection of an average reference face [13], but with modifications to improve resolution for oblique regions surrounding the nose. The aligned face image is composited into the texture map which includes the inside of the mouth, with teeth (Figure 4c) using the artist-defined mask, which defines the face mask of the avatar character (Figure 4b). A resolution of $1024 \times 1024$ is used for the person-specific face texture map. The meshes for the eyes have UVs in the corresponding area of the face texture map, but with a slight expansion of the texture so as to allow small amounts of eye animation without eyelids becoming visible on the edges of the eyeballs. The face texture fitting process is described in Section 5.1.

### 4.2 Face Identity Blendshapes

In addition to a personalised texture of the face, the identity of a person is also significantly influenced by facial shape. We customise the avatar face geometry using a blendshape rig. The personalised mesh $M$ is constructed by a linear combination of $n_F$ blendshape meshes $F_k$ as follows: $M(\mathbf{w}) = F_0 + \sum_{k=1}^{n_F} w_k(F_k - F_0)$, where $F_0$ is a base mesh and $\mathbf{w}$ is a vector of blendweights. In our rig, a compact set of $n_F = 15$ shapes was hand-selected based on those in MakeHuman[3] to cover the most prominent facial characteristics

[3] http://www.makehuman.org



(a) Reference face (b) Mask (c) Texture map

Figure 4: Face 2D texture mapping based on an artist-tuned orthographic projection of an average reference face.

while keeping complexity of the rig and fitting process down (see Figure 5).

Weights of individual blendshapes are typically restricted to the range $[0,1]$. This single range has to cover a full variation of a particular facial characteristic (e.g. eye separation). Therefore, the bounds 0 and 1 correspond to extreme cases (e.g. the smallest and largest possible eye separation). For this reason, the base shape of the rig does not correspond to a representative human face, but represents extreme facial proportions.

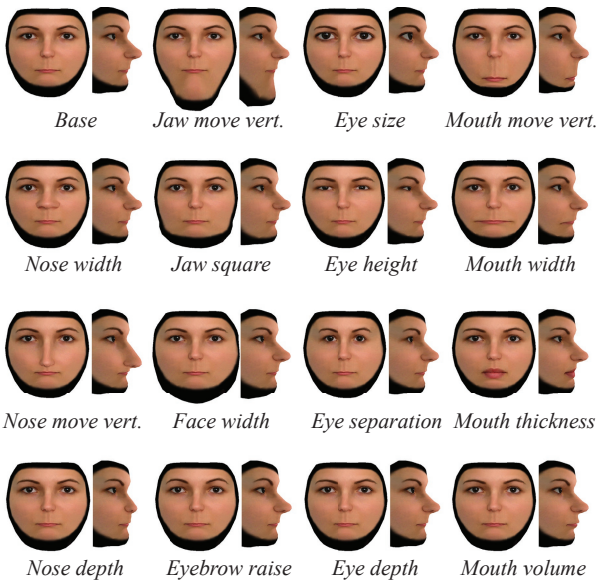



Figure 5: Face identity blendshapes.

The eyeballs, being separate meshes, require additional blendshapes to control their positioning. The blendweights for these are set equal to their corresponding face shape weights, for example *eye separation* changes the distance between the eyes both in the main mesh and the eyeballs. The face shape fitting process is described in Section 5.2.

### 4.3 Face Animation Blendshapes

The face animation blendshapes are geared towards speech animation of the lower part of the face, but also contain 2 shapes for animating the eye gaze (horizontal/vertical) and 1 shape for blinking. In the demo VR application presented in Section 6, an audio-based lip sync method (Oculus Lip Sync plugin[4]) is used. The plugin detects phonemes in the HMD microphone audio stream, which are mapped to viseme blendweights to produce the resulting animation.

[4] https://developer3.oculus.com/downloads/audio/1.0.1-beta/Oculus_OVRLipSync_for_Unity_5/

Thus, the mouth animation blendshapes are based on 15 visemes, as shown in Figure 6. Note that the viseme blendshapes include some motion of the upper part of the face in order to make the face dynamics appear more natural during speech.

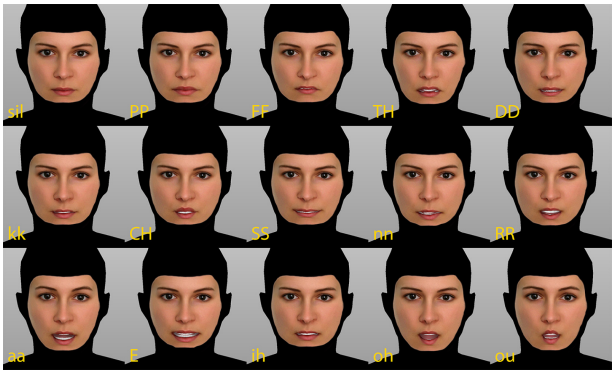



Figure 6: Face animation blendshapes (visemes) for audio-based speech animation.

### 4.4 Body Identity Blendshapes and Skeleton

The base body shape $B_0$ is a generic slim male of height 200 cm. The size of the avatar is matched to the size of the subject by scaling the entire avatar *down* according to the detected subject height. A relatively small set of blendshapes $B_m$ ($m \in [1,4]$) is combined according to a blendweight vector $\mathbf{v}$ to form the avatar body, making it more identifiable as the subject. Three of these are related to the width of upper, middle and lower torso, while the fourth is a stylization which is applied to subjects detected as female (see Figure 7). The body is rigged to a skeleton based on the CMU motion capture skeleton[5] allowing the body to be animated with pre-recorded or live motion capture data. A fixed $2048 \times 2048$ artist-created texture is used for all avatars. The body fitting process is described in Section 5.3.

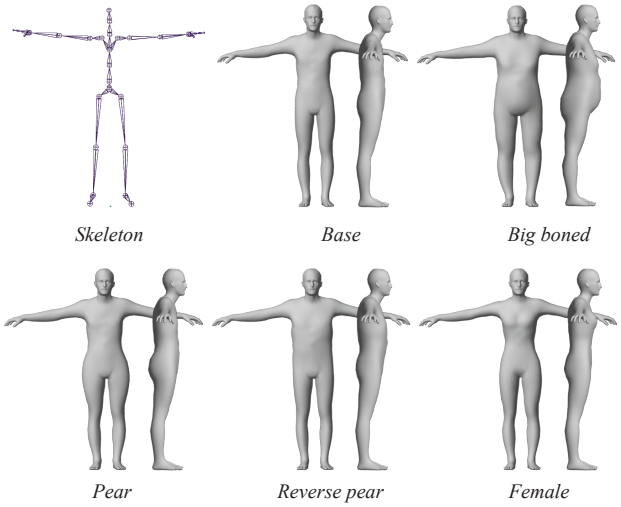



Figure 7: Body skeleton and identity blendshapes.

[5] http://mocap.cs.cmu.edu/

## 5 Avatar Customisation

Given the customisable avatar rig template construction, we now follow to describe the process of rapid customisation from a single shot capture. The avatar customisation process begins with the capture of an input snapshot $S = \{I_1, I_2, D\}$, where $I_1$, $I_2$ and $D$ are the camera RGB images and Kinect depth map, respectively. Upon capture of $S$, a fully automatic process of generating the face texture and fitting the face and body shape is performed before the custom avatar is passed to the VR application.

### 5.1 Face Texture Generation

Using OpenFace [6], two sets of 2D facial landmarks $\{\mathbf{p}_{1,i}\}$ and $\{\mathbf{p}_{2,i}\}$ are detected in images $I_1$ and $I_2$ (where $i \in [1,l]; l = 68$). These comprise points on the main facial features(mouth, nose, eyes, eyebrows) as well as the contour of the face (see Figure 9). Using the camera projection matrices $\mathbf{P}_1$ and $\mathbf{P}_2$, the corresponding 2D points in $\{\mathbf{p}_{1,i}\}$ and $\{\mathbf{p}_{2,i}\}$ are triangulated to produce a set of 3D landmarks $\{\mathbf{q}_i\}$.

Next, one of the images $I_1$ and $I_2$ needs to be selected for texture generation. This is determined based on a head pose with respect to each camera viewpoint. The head pose is defined by three 3D landmarks selected from $\{\mathbf{q}_i\}$ which correspond to outer corners of the eyes and base of the nose. The ideal viewing direction for the texture capture is computed based on the head pose as shown in Figure 8. The best camera $C_b$ is selected according to how close the viewpoint aligns with the ideal viewing direction for the face.

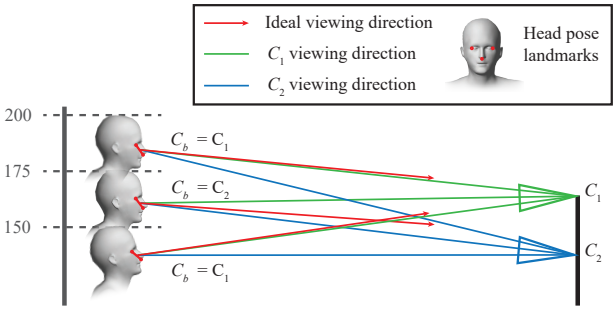



Figure 8: Selection of the best camera $C_b$ for face texturing. Note that it is the angle between the face direction and camera viewpoint which determines the image used, not the position of the face in the frame. Thus, it is possible to have a short subject textured from the top camera if they tilt their head upwards when captured.

The image $I_b$ from the camera $C_b$ is warped to the reference face texture $I_{ref}$ (see Figure 4). This is based on the image 2D landmarks $\{\mathbf{p}_{b,i}\}$ and the landmarks $\{\mathbf{p}_{ref,i}\}$ which are detected from $I_{ref}$ using the same process. Image warping using the original 68 landmarks alone was found to produce undesirable artefacts in some cases (e.g. distortions between eyebrow landmarks causing wavy looking eyebrows). To mitigate this, auxiliary points are introduced between the original detections to further constrain the warping. Two points are inserted along the line between consecutive landmarks on each contour (e.g. eyebrow, eye) resulting in 178 points altogether (see Figure 9).

Moving least squares (MLS) image warping [44] is used to warp $I_b$ to $I_{ref}$ based on the augmented sets of $\{\mathbf{p}_{b,j}\}$ and $\{\mathbf{p}_{ref,j}\}$. These landmarks operate as source and target control points, while image information is smoothly interpolated among them to produce a warped face image $I_{warped}$. The output texture map $I_{tex}$ is then generated by composing $I_{warped}$ into $I_{ref}$ using a pre-defined mask $I_{mask}$ that specifies the face region in the texture map:

$$I_{tex} = I_{mask} \cdot I_{warped} + (1 - I_{mask}) \cdot I_{ref} \quad (1)$$

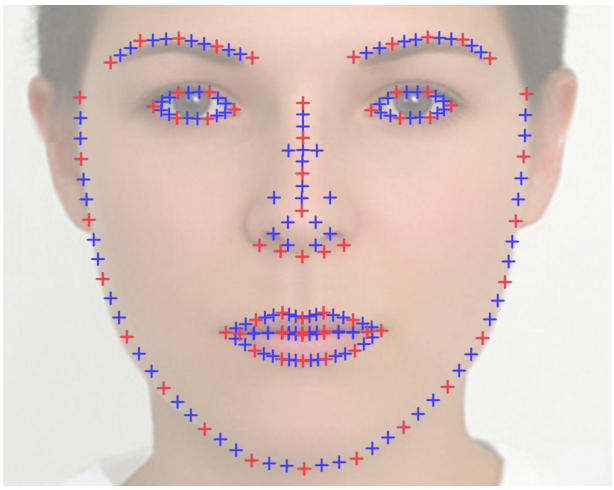


Figure 9: The set of original facial landmarks as detected by OpenFace (red) and the augmented set of points among them (blue).

Finally, specular highlights in eyes resulting from the lighting rig are removed from the texture. This is to avoid having baked-in highlights attached to the eyeballs as they are moving in the VR experience. Instead, reflections in the eyes are rendered at runtime in the game engine based on the virtual environment and eye motion. The highlight removal is performed using in-painting [10] inside a ROI of the texture $I_{tex}$ defined by the relevant eye landmarks in $\{\mathbf{p}_{ref,i}\}$. The specularity mask for in-painting is created by intensity thresholding and morphological dilation as shown in Figure 10.

(a) Raw eye texture (b) High-light mask in eye region of interest (c) In-painted eye texture

Figure 10: Specular highlight removal from eye texture by in-painting.

### 5.2 Face Shape Fitting

Face shape fitting is achieved through optimization of the blendweight vector $\mathbf{w}$ which controls face identity blendshapes. As a one-time pre-process, a correspondence is manually established between 68 detected landmarks and vertices on the face base mesh $F_0$. Let $M_i(\mathbf{w})$ denote the $i$-th landmark position on the mesh blended with blendweights $\mathbf{w}$. The face shape optimization aims to minimise the Euclidean distance between the 3D detected landmarks $\{\mathbf{q}_i\}$ and the corresponding blended vertices $\{M_i(\mathbf{w})\}$ aligned by a rigid transform $(\mathbf{R}, \mathbf{t})$. The optimisation is performed in the camera coordinate system established during the setup calibration. Note that the rigid transform is optimised to compensate for a head pose and a difference between camera and rig coordinate systems, but is not used as an output.

In practice, the outer face contour points are relatively unconstrained compared to the inner landmarks such as eyes. Their detected locations slide along the chin depending on the viewpoint. For this reason, triangulated 3D positions of the outer landmarks drift in depth (along $z$-axis). This leads to large 3D distances to the fitted rig vertices, therefore we use only $x$-$y$ Euclidean distance for these points. This is achieved by a scaling matrix $\mathbf{S}_i = \text{diag}(1,1,1)$

for inner landmarks, and $\mathbf{S}_i = \text{diag}(0.5, 0.5, 0)$ for outer landmarks. The energy function for shape fitting is defined as follows,

$$E(\mathbf{w}, \mathbf{R}, \mathbf{t}) = \sum_{i=1}^{l} ||\mathbf{S}_i \cdot (\mathbf{q}_i - (\mathbf{R} \cdot M_i(\mathbf{w}) + \mathbf{t}))||_2^2. \quad (2)$$

This energy is minimised subject to $0 \leq w_k \leq 1, \forall k \in [1, n_F]$ using the Levenberg Marquardt algorithm in the Ceres library [1]. Automatic differentiation turns the problem to a sequence of linear solves until a defined tolerance of $1e^{-10}$ or maximum number of 50 iterations are reached.

The interpupillary distance (IPD) is also estimated directly from the 3D eye landmarks. This is used in the VR application to set the lens separation on the HMD to match the user.

### 5.3 Body Shape Fitting

The body shape customization is achieved using three types of input: the depth map $D$, face image $I_b$ and 3D facial landmarks $\{\mathbf{q}_i\}$. The depth map is used to obtain the height of the subject and approximate metric body proportions used to set the weights $\mathbf{v}$ of the body identity blendshapes. The face image is also utilised to detect the subject gender to stylize their body.

#### 5.3.1 Depth-Based Fitting

Upon capture of the subject in a T-pose, a segmented depth map $D_{sub}$ is produced by performing background subtraction on $D$ using the background plate $D_{bg}$ and $D$ (subject to depth noise tolerance of $5cm$). The average subject depth $d_{sub}$ is then computed. Next, a silhouette is extracted from $D_{sub}$ by thresholding non-zero depths and performing morphological operations to remove noise artefacts (see Figure 11).

Given the silhouette and subject depth $d_{sub}$, an in-plane metric body dimension $x_m$ is determined using the corresponding pixel distance $x_{pix}$ and the depth sensor focal length $f_d$:

$$x_m = d_{sub} x_{pix} / f_d \quad (3)$$

The subject height $h_{sub}$ is estimated by taking the distance between the top and the bottom of the silhouette in the depth map. The body blendweights $v_m$ for the blendshapes *Bigboned*, *Pear* and *Reversepear* of figure 7 (indexed $m = 1, 2, 3$) are determined according to the ratio of torso width in certain regions, and the overall height. The torso is split into height regions defined in proportion to the detected silhouette (see Figure 11). Each region is scanned row-by-row and in the case of more than one segment of silhouette in a row (e.g. due to blobs of noise or the arms), the largest segment of silhouette is assumed to be the torso. The torso widths for all rows are sorted in the region and a single body dimension $x_m$ is taken as a percentile of the detected widths (50th, 90th and 50th, for $m = 1, 2, 3$, respectively). Taking the percentiles makes the measurements more robust to outliers caused by silhouette noise or loose clothing. The ratio of $x_m$ to the body height $h_{sub}$ is used to determine the values of blendweight vector $\mathbf{v}$ as follows

$$v_m = \text{clamp}\left(\frac{x_m / h_{sub} - a_0}{a_m - a_0}\right), \quad (4)$$

where $a_m$ is a ratio of the body dimension $m$ to the avatar height for the corresponding blendshape $B_m$ and likewise $a_0$ for the base shape $B_0$. The function clamp() clamps the output to $[0, 1]$ to ensure that the resulting avatar body is within the range of the modelled blendshapes. Note that the range of the body identity blendshapes is somewhat limited - for instance subjects with a very large body mass index value will have avatars with a moderate value. This body shape fitting is thus more of an artistic stylization of the person. On the other hand the face blendshapes have a wide range allowing more geometrically accurate fitting across a wide range of face shapes and ensuring that the avatar is recognizable as the subject.

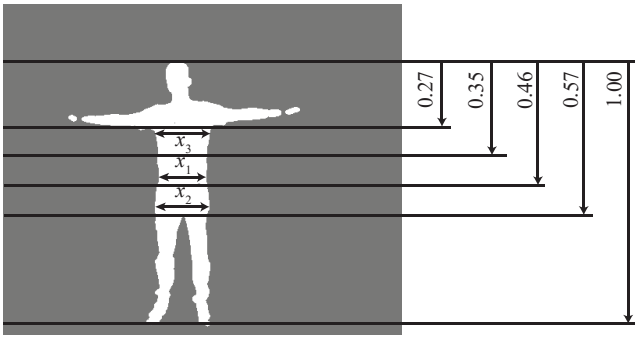



Figure 11: Body shape customization from a silhouette extracted from depth map $D_{sub}$.

The head size blendweight, initially set during the face fitting stage is updated according to the detected height, such that it maintains its metric size once the body is scaled down according to the detected height.

#### 5.3.2 Gender Stylization

Although the general shape of subject is well represented by the three blendshapes fitted from depth above, an additional stylization is applied to subtly modify the avatar to make it more gender specific. Because this gender blendshape is a stylization rather than representing the true geometry of the subject, the blendweight is based on the detected gender rather than the depth map. The *Female* blendweight of the rig is set continuously according to the output of a gender classification on the face image. A Fisherfaces classifier [9] is pre-trained using a subset of gender-labelled images from an existing database [25]. The full database - which includes all ages, occlusions, challenging lighting and viewpoints - was filtered to include only frontal images of adults which represent the conditions in our scenario. Note that the images in the training dataset as well as the images for runtime classification need to be pre scaled and aligned so that the eyes are in the same location in all images. This is achieved by a similarity transform of each image based on OpenFace landmark detections. The output of the classifier is a gender class $g \in \{0, 1\}$ (male/female) as well as a distance $c_g \in [0, \infty)$ indicating the classification confidence (larger distance means lower confidence). The *Female* body blendweight $v_4$ is set according to the confidence:

$$v_4 = g \cdot \max(c_{max} - c_g, 0) / c_{max}) \quad (5)$$

where $c_{max}$ is the empirically chosen as 70. Thus, female classifications made with a high confidence trigger the female blendshape fully. But as the confidence decreases, the blendshape is applied more conservatively. In our avatar stylisation, it is preferable to have a false negative (more neutral/male body shape for a female) than a false positive (feminine body shape for a male).

## 6 Results

### 6.1 Avatar Fitting

To evaluate the proposed approach, we performed the capture and fitting procedure with 12 subjects (9 male and 3 female). Input, intermediate and output data for a subset of these is presented in Figure 13 while the full set of avatars is shown in the supplementary video.

For numerical verification of the IPD and height estimation components, the height and IPD of 12 subjects were manually measured.

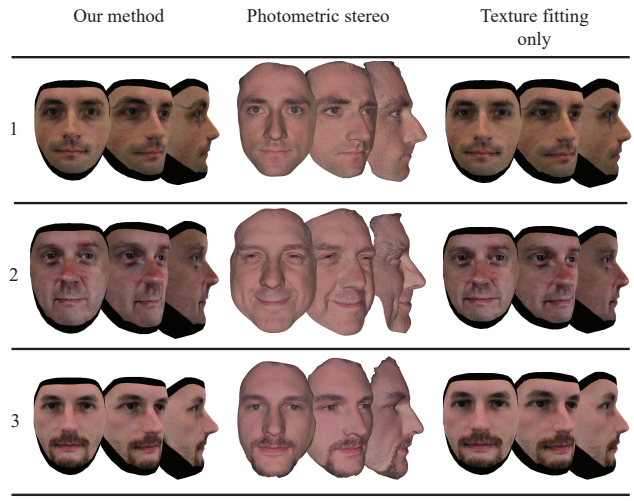



Figure 12: Comparison of the fitted avatar face shape without inclusion of a helmet, obtained using our method (left) and with dense static meshes obtained using photometric stereo [30] (centre). We also show the face texture with generic 50% values for the shape weights (right).

The heights ranged between 154 and 185 cm, while the IPDs ranged between 55 and 69 mm. The mean error in the automatically estimated height was 1.4 cm (std. 1.2 cm, max. 2.4 cm). The mean error in the estimated IPD was 1.6 mm (std. 1.2 mm, max. 4.2 mm) This level of accuracy results in the correct relative heights between avatars in VR. It is also useful to have a good height estimate as input for body tracking purposes. The accuracy of the estimated IPD allows to effectively set the lenses on the HMD and in the rendered VR world as described in Section 6.2.

A breakdown of the avatar acquisition timing on our hardware is shown in Table 1. The timings are for unoptimized single threaded C++ code running on an Intel Core i7 3.7 GHz CPU. The entire pipeline from snapshot to fitted avatar takes under 10 s. Apart from capture and file I/O, the most time intensive tasks are landmark detection and face shape fitting. It would be possible to run the texture warping/compositing along with the face shape fitting, which would bring the total time down by approx 1.5 seconds. However, the processing time is already short enough that it does not introduce a delay between capturing the subjects and starting the VR session, since it takes some time for them to put on the VR headset.

We further validate our face shape fitting approach by comparing our output faces with dense static meshes obtained using an offline active photometric stereo technique [30]. The results for three subjects are show in Figure 12. For completeness, we also show the result using our texturing method without face shape fitting - this demonstrates that customizing the face shape in addition to the texture results in more recognisable avatars.

### 6.2 Virtual Reality Demonstration

The Oculus Rift CV1[6] VR headset used in our demo application features movable lenses to accommodate a range of user IPDs. In order to provide the most comfortable viewing experience for the participants, the headset is adjusted according to the detected IPD, as determined by taking the distance between the average 3D landmarks in the left and right eyes, respectively.

A VR demonstration was created to verify the end-to-end pipeline from capture to VR application. In the demonstration four subjects are acquired and enrolled in the VR experience at a time. The subjects were sequentially captured and turned into avatars using the system. Upon completion of processing, the avatars were automatically loaded into the demonstration application in a popular video game engine framework, which is run over the Oculus VR headset.

The demonstration shows the avatars as futuristic astronauts in an alien planet environment, with pre-generated body animation, eye movement and blinks, with live audio-based speech animation for the face as well as body animation using the Perception Neuron[7] inertial body tracking system (see Figure 15). The supplementary video for this paper shows three sets of four representative captured avatars in the VR environment.

[6] https://www3.oculus.com/en-us/rift/

Table 1: Avatar acquisition timing

| Task | Time (s) | Proportion(%) |
|---|---|---|
| Image/depth capture and file I/O | 2.1 | 21.4 |
| Facial landmark detection | 3.54 | 36.1 |
| Face texture warping | 0.76 | 7.8 |
| Face texture composite | 0.78 | 8.0 |
| Face shape fitting | 2.38 | 24.3 |
| Depth based body fitting | 0.14 | 1.4 |
| Gender detection | 0.1 | 1.0 |
| Total | 9.8 | |

## 7 Discussion and Further Work

In this paper, we have presented a system allowing for fast acquisition of custom, animatable avatars using low-cost, easy to setup hardware. The avatars are of suitable quality for realtime display in VR applications within 9.8 seconds (Table 1). The careful selection of acquisition methods which operate on the scale of milliseconds has been key to achieve an overall fast end-to-end solution.

While the avatars created using the system are compelling, they lack the high level of detail available from systems which use more complex hardware setups, longer processing times or manual interaction. For example the system does not capture high resolution details of the face shape such as wrinkles and the detailed profile of the nose. Future work could consider improving the level of shape detail while keeping the capture and processing complexity low.

The appearance customization of our avatars is limited to the face. A generic character texture is used for the body and the ears and hair are covered by the character headgear (astronaut space suit). Future work could consider customizing ears, hair and clothing of the avatar as well expanding the range of applications to those where the avatar is in their own clothes rather than a character wearing an suit and headgear.

The audio-based lip-sync animation used in our demo application cannot detect eye motion or silent facial expression. Our avatars could however be used with existing full-face tracking systems, for instance using HMDs with built-in surface strain sensors for occluded face tracking [38].

The system can fail on subjects with very thick, dark beards, due to the OpenFace landmark detection failing around the mouth region. This causes error in the fitted face positions as well as the texture alignment (Figure 14). While the fitting works correctly with subject glasses, the glasses appear flush with the texture in the avatar - it may be possible to automatically remove glasses in the image [41] for such subjects.

As is common with body fitting methods, results are best when the clothing is fairly thin and tight-fitting, and body dimensions may be overestimated with thick or loose-fitting clothing. In practice over a large number of experimental subjects and a period of weeks, we found a very high percentage of successful acquisitions on the first take. By introducing a face capture landmarks preview in under

[7] https://neuronmocap.com/

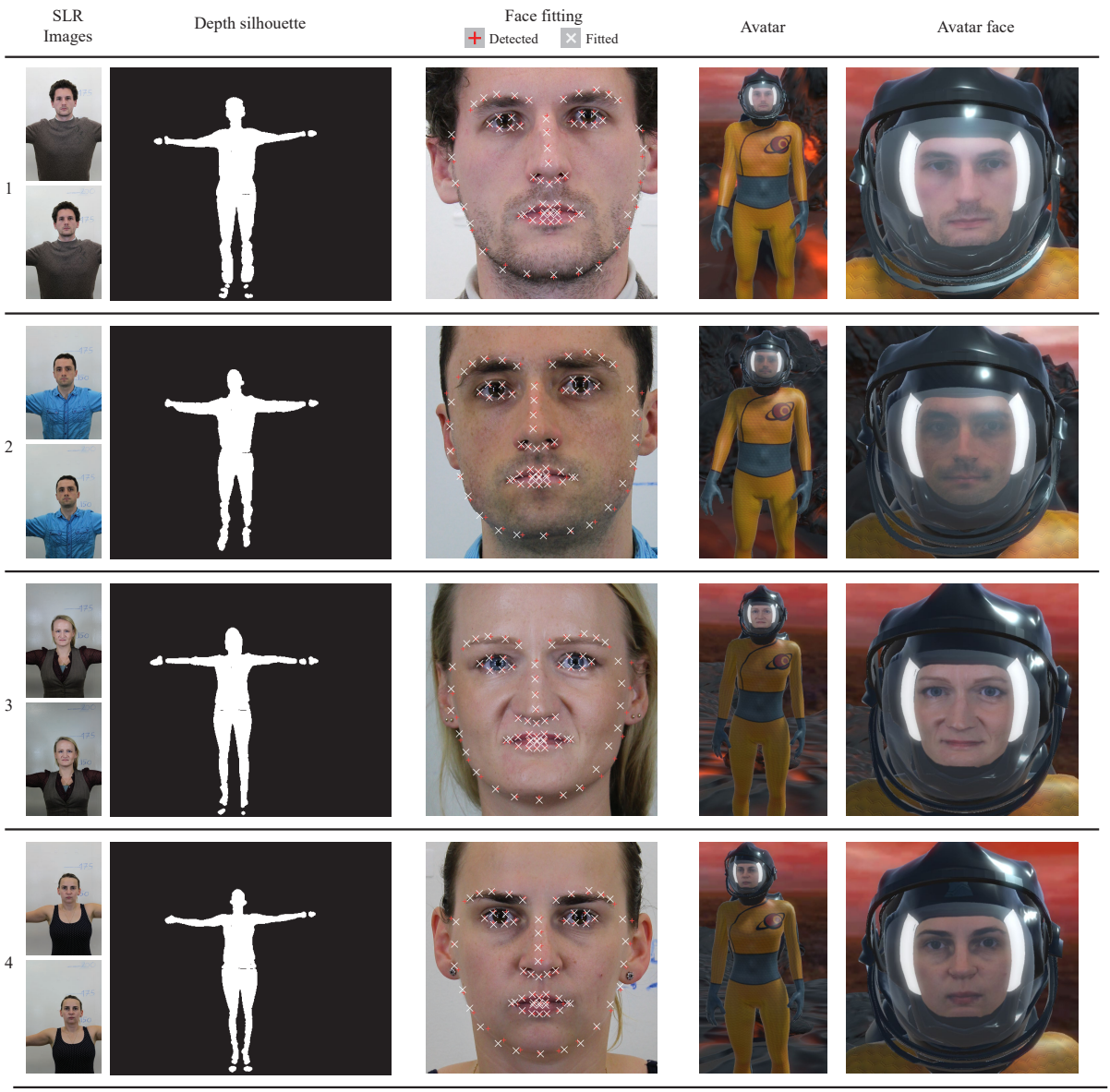



Figure 13: Sample input and captured avatars.

6 seconds, the advice to the subject for quick re-takes in failure cases was also accelerated to comfortable operational times. This addressed further problems with blinking at the wrong time, face tilted or non-neutral expression, body moving away from T-pose shape and distance, etc. Multiple participants were pipelined at up to double rates with parallel processing operations after initial preview check indicated success.

For the infrequent occasion of an insurmountable case, the space suit (and related scenarios such as biker, diver, etc.) allowed us to mist the visor and optionally select from a set of default body shapes through manual overrides.

For avatars without helmets, we consider in further work to select from a range of hair styles and colors manually, or through optimization where possible of an image-based hair model [17] for rapid integration into a VR environment. With the same acquired input data from Figure 13 subjects 1 and 3, we provide a range of views captured from the runtime without headgear to illustrate the extents of the face capture data unobstructed (Figure 16).

## Acknowledgements

The authors wish to thank the reviewers, Douglas Fidaleo, Pearce Bergh, Marco Grubert, Sara Thacher, Christopher Gabriel, Joanna Jamrozy, Babis Koniaris, David Sinclair, Isadora Sanna, Jose Guitain, and Steven McDonagh.

## References

[1] S. Agarwal, K. Mierle, and Others. Ceres solver. `http://ceres-solver.org`.

[2] O. Alexander, M. Rogers, W. Lambeth, J. Chiang, W. Ma, C. Wang, and P. E. Debevec. The digital Emily project: Achieving a photorealistic digital actor. *IEEE Computer Graphics and Applications*, 30(4):20–31, 2010.

[3] B. Allen, B. Curless, and Z. Popovic. The space of human body shapes: reconstruction and parameterization from range scans. *TOG (SIGGRAPH)*, 22(3):587–594, 2003.

Figure 14: Face fitting failure case - poorly detected mouth landmarks causing the mouth to be misaligned avatar.

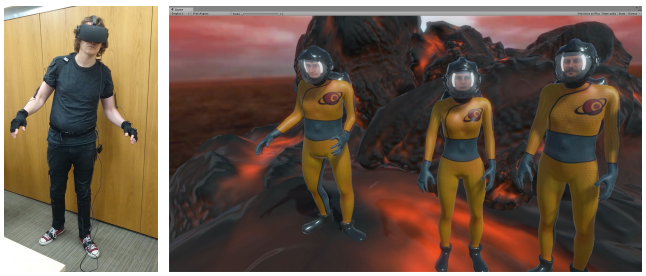


Figure 15: Avatars in an video game engine VR environment with live body motion capture using a Perception Neuron IMU based motion capture system.


[4] B. Allen, B. Curless, Z. Popovic, and A. Hertzmann. Learning a correlated model of identity and pose-dependent body shape variation for real-time synthesis. In *SCA*, pages 147–156, 2006.

[5] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis. SCAPE: shape completion and animation of people. *TOG (SIGGRAPH)*, 24(3):408–416, 2005.

[6] T. Baltrušaitis, P. Robinson, and L.-P. Morency. OpenFace: an open source facial behavior analysis toolkit. In *IEEE Winter Conference on Applications of Computer Vision*, 2016.

[7] I. Baran and J. Popovic. Automatic rigging and animation of 3D characters. *TOG (SIGGRAPH)*, 26(3):72, 2007.

[8] T. Beeler, B. Bickel, P. A. Beardsley, B. Sumner, and M. H. Gross. High-quality single-shot capture of facial geometry. *TOG (SIGGRAPH)*, 29(4):40:1–40:9, 2010.

[9] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *TPAMI*, 19(7):711–720, 1997.

[10] M. Bertalmío, A. L. Bertozzi, and G. Sapiro. Navier-stokes, fluid dynamics, and image and video inpainting. In *CVPR*, pages 355–362, 2001.

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

[12] S. Bouaziz, Y. Wang, and M. Pauly. Online modeling for realtime facial animation. *TOG (SIGGRAPH)*, 32(4):40:1–40:10, 2013.

[13] C. Braun, M. Gründl, C. Marberger, and C. Scherber. Beautycheck-ursachen und folgen von attraktivitaet. 2003.

[14] M. Brubaker, L. Sigal, and D. J. Fleet. Physics-based human motion modeling for people tracking: A short tutorial. *Image (Rochester, NY)*, pages 1–48, 2009.

[15] C. Cao, Q. Hou, and K. Zhou. Displaced dynamic expression regression for real-time facial tracking and animation. *TOG (SIGGRAPH)*, 33(4):43:1–43:10, 2014.

[16] C. Cao, H. Wu, Y. Weng, T. Shao, and K. Zhou. Real-time facial animation with image-based dynamic avatars. *TOG (SIGGRAPH)*, 35(4):126, 2016.

[17] C. Cao, H. Wu, Y. Weng, T. Shao, and K. Zhou. Real-time facial animation with image-based dynamic avatars. *ACM Trans. Graph.*, 35(4):126:1–126:12, July 2016.

[18] D. Casas, O. Alexander, A. W. Feng, G. Fyffe, R. Ichikari, P. E. Debevec, R. Wang, E. A. Suma, and A. Shapiro. Rapid photorealistic blendshapes from commodity RGB-D sensors. In *Symposium on Interactive 3D Graphics and Games (I3D)*, page 134, 2015.

[19] A. Collet, M. Chuang, P. Sweeney, D. Gillett, D. Evseev, D. Calabrese, H. Hoppe, A. G. Kirk, and S. Sullivan. High-quality streamable free-viewpoint video. *TOG (SIGGRAPH)*, 34(4):69, 2015.

[20] Y. Cui, W. Chang, T. Nöll, and D. Stricker. KinectAvatar: fully automatic body capture using a single Kinect. In *Workshop at Asian Conference on Computer Vision (ACCV)*, pages 133–147, 2012.

[21] H. A. M. Daanen and F. B. T. Haar. 3D whole body scanners revisited. *Displays*, 34(4):270–275, 2013.

[22] E. de Aguiar, L. L. Costalonga, L. O. R. Junior, and R. da Silva Villaça. Capture and stylization of human models. In *Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 282–287, 2013.

[23] M. Dou, H. Fuchs, and J. Frahm. Scanning and tracking dynamic objects with commodity depth cameras. In *ISMAR*, pages 99–106, 2013.

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

[25] E. Eidinger, R. Enbar, and T. Hassner. Age and gender estimation of unfiltered faces. *IEEE Transactions on Information Forensics and Security*, 9(12):2170–2179, Dec 2014.

[26] A. Feng, D. Casas, and A. Shapiro. Avatar reshaping and automatic rigging using a deformable model. In *Proceedings of the 8th ACM SIGGRAPH Conference on Motion in Games*, MIG '15, pages 57–64, New York, NY, USA, 2015. ACM.

[27] D. Fidaleo and G. Medioni. Model-assisted 3D face reconstruction from video. In *International Workshop on Analysis and Modeling of Faces and Gestures (AMFG)*, pages 124–138, 2007.

[28] P. Garrido, M. Zollhöfer, D. Casas, L. Valgaerts, K. Varanasi, P. Pérez, and C. Theobalt. Reconstruction of personalized 3D face rigs from monocular video. *TOG*, 35(3):28, 2016.

[29] D. Giger, J.-C. Bazin, C. Kuster, T. Popa, and M. Gross. Gaze correction with a single webcam. In *ICME*, pages 1–6, 2014.

[30] P. F. U. Gotardo, T. Simon, Y. Sheikh, and I. Matthews. Photogeometric scene flow for high-detail dynamic 3d reconstruction. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[31] M. Gross, S. Wuermlin, M. Naef, E. Lamboray, C. Spagno, A. Kunz, E. Koller-Meier, T. Svoboda, L. V. Gool, S. Lang, K. Strehlke, A. V. Moere, and O. Staadt. blue-c: A spatially immersive display and 3d video portal for telepresence. In *ACM SIGGRAPH*, 2003.

[32] P. Guan, A. Weiss, A. O. Balan, and M. J. Black. Estimating human shape and pose from a single image. In *ICCV*, pages 1381–1388, 2009.

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

[34] A. E. Ichim, S. Bouaziz, and M. Pauly. Dynamic 3D avatar creation from hand-held video input. *TOG (SIGGRAPH)*, 34(4):45, 2015.

[35] A. Jain, T. Thormählen, H. Seidel, and C. Theobalt. MovieReshape: tracking and reshaping of humans in videos. *TOG (SIGGRAPH Asia)*, 29(6):148, 2010.

[36] I. Kemelmacher-Shlizerman and R. Basri. 3D face reconstruction from a single image using a single reference face shape. *TPAMI*, 33(2):394–405, 2011.

[37] J. P. Lewis, K. Anjyo, T. Rhee, M. Zhang, F. H. Pighin, and Z. Deng. Practice and theory of blendshape facial models. In *Eurographics State of the Art Repors (STAR)*, pages 199–218, 2014.

[38] H. Li, L. C. Trutoiu, K. Olszewski, L. Wei, T. Trutna, P. Hsieh, A. Nicholls, and C. Ma. Facial performance sensing head-mounted display. *TOG (SIGGRAPH)*, 34(4):47, 2015.

[39] H. Li, E. Vouga, A. Gudym, L. Luo, J. T. Barron, and G. Gusev. 3D self-portraits. *TOG (SIGGRAPH Asia)*, 32(6):187:1–187:9, 2013.

[40] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. W. Fitzgibbon. KinectFusion: real-time dense surface mapping and tracking. In *ISMAR*, pages 127–136, 2011.

Figure 16: Avatars resulting without headgear obstructions. In this example we extend the masked face texture map area and blended with the average skin tone sampled from photographic face capture. A little texel mapping distortion is visible under the chin before the blended area. Neither hair nor ear acquisitions are addressed, instead we attach the face rig to a generic head shape as part of the customised body rig.


[41] M. H. Nguyen, J.-F. Lalonde, A. A. Efros, and F. de la Torre. Image-based shaving. *Computer Graphics Forum Journal (Eurographics 2008)*, 27(2):627–635, 2008.

[42] C. Plüss, N. Ranieri, J.-C. Bazin, T. Martin, P. Laffont, T. Popa, and M. Gross. An immersive bidirectional system for life-size 3D communication. In *CASA*, pages 89–96, 2016.

[43] J. M. Saragih, S. Lucey, and J. F. Cohn. Real-time avatar animation from a single image. *International Conference on Automatic Face and Gesture Recognition (FG)*, pages 117–124, 2011.

[44] S. Schaefer, T. McPhail, and J. Warren. Image deformation using moving least squares. *TOG (SIGGRAPH)*, 25(3):533–540, 2006.

[45] H. Seo and N. Magnenat-Thalmann. An example-based approach to human body manipulation. *Graphical Models*, 66(1):1–23, 2004.

[46] A. Shapiro, A. W. Feng, R. Wang, H. Li, M. T. Bolas, G. G. Medioni, and E. A. Suma. Rapid avatar capture and simulation using commodity depth sensors. *Journal of Visualization and Computer Animation*, 25(3-4):201–211, 2014.

[47] D. Song, R. Tong, J. Chang, X. Yang, M. Tang, and J. J. Zhang. 3D Body Shapes Estimation from Dressed-Human Silhouettes. *Computer Graphics Forum*, 2016.

[48] J. Thies, M. Zollhöfer, M. Stamminger, C. Theobalt, and M. Nießner. Face2Face: real-time face capture and reenactment of RGB videos. *CVPR*, 2016.

[49] J. Tong, J. Zhou, L. Liu, Z. Pan, and H. Yan. Scanning 3D full human bodies using Kinects. *TVCG*, 18(4):643–650, 2012.

[50] R. Wang, J. Choi, and G. G. Medioni. Accurate full body scanning from a single fixed 3D camera. In *International Conference on 3D Imaging, Modeling, Processing, Visualization & Transmission (3DIMPVT)*, pages 432–439, 2012.

[51] T. Weise, S. Bouaziz, H. Li, and M. Pauly. Realtime performance-based facial animation. *TOG (SIGGRAPH)*, 30(4):77, 2011.

[52] A. Weiss, D. A. Hirshberg, and M. J. Black. Home 3D body scans from noisy image and range data. In *ICCV*, pages 1951–1958, 2011.

[53] M. Zeng, J. Zheng, X. Cheng, and X. Liu. Templateless quasi-rigid shape modeling with implicit loop-closure. In *CVPR*, pages 145–152, 2013.

[54] Z. Zhang. A flexible new technique for camera calibration. *TPAMI*, 22(11):1330–1334, 2000.

[55] S. Zhou, H. Fu, L. Liu, D. Cohen-Or, and X. Han. Parametric reshaping of human bodies in images. *TOG (SIGGRAPH)*, 29(4):126:1–126:10, 2010.