Article:
Cao, C, Wu, H, Weng, Y et al. (2 more authors) (2016) Real-time Facial Animation with Image-based Dynamic Avatars. ACM Transactions on Graphics, 35 (4). ARTN 126. ISSN 0730-0301

https://doi.org/10.1145/2897824.2925873

Real-time Facial Animation with Image-based Dynamic Avatars

Chen Cao Hongzhi Wu Yanlin Weng Tianjia Shao Kun Zhou

State Key Lab of CAD&CG, Zhejiang University*

Figure 1: *From a set of sparsely captured images of a user, we construct our image-based dynamic avatar, consisting of a face blendshape and a hair morphable model, which represent the corresponding coarse geometry. The captured images are warped and blended under the guidance of the coarse geometry to generate real-time facial animation with fine-scale details. From left to right: input images, the reconstructed coarse geometry for face and hair, rendering results of our avatar with different poses and expressions.*

Abstract

We present a novel image-based representation for dynamic 3D avatars, which allows effective handling of various hairstyles and headwear, and can generate expressive facial animations with fine-scale details in real-time. We develop algorithms for creating an image-based avatar from a set of sparsely captured images of a user, using an off-the-shelf web camera at home. An optimization method is proposed to construct a topologically consistent morphable model that approximates the dynamic hair geometry in the captured images. We also design a real-time algorithm for synthesizing novel views of an image-based avatar, so that the avatar follows the facial motions of an arbitrary actor. Compelling results from our pipeline are demonstrated on a variety of cases.

Keywords: facial animation, face tracking, virtual avatar, image-based rendering, hair modeling

Concepts: •**Computing methodologies** → **Animation;** *Motion capture;*

1 Introduction

A personalized dynamic avatar is a custom face rig that matches the geometry, appearance and expression dynamics of a user at different poses. Combined with real-time face tracking techniques (e.g., [Weise et al. 2011; Cao et al. 2014a]), facial animation generated using a personalized avatar helps convey the realism of a person, in contrast to avatars of pre-defined virtual characters, which is useful and important in many real-world applications, including virtual reality, gaming or teleconferencing.

One major related challenge is how to easily create compelling user-specific avatars with fine-scale details for non-professionals with commodity hardware at home. Excellent recent work [Ichim et al. 2015; Garrido et al. 2016] introduces techniques that focus on modeling the face and head from image/video recordings of a user. The main idea is to use a geometry-based approach to adapt a coarse-scale generic blendshape model to the captured images/video, and then estimate fine-scale facial details with shape-from-shading. While very impressive results are demonstrated, only the face and head are explicitly modeled; hair, a non-negligible part of a user's appearance, is simply approximated as a diffuse texture map on the head model. A more expressive model is needed.

In this paper, we introduce a novel image-based representation for personalized, dynamic 3D avatars with hairs. Our representation is expressive, easy-to-construct and run-time efficient. We take a set of sparsely captured images of a user with predefined poses and expressions as input, and create a face blendshape model and a morphable hair model to represent the coarse dynamic geometry. Other components such as eyes and teeth are represented as billboards. During runtime, we generate novel views of the avatar by warping and blending pre-captured images under the guidance of the coarse dynamic geometry, based on the rigid head transformation and facial expressions estimated by a real-time face tracker. Different components of the avatar are then seamlessly integrated to render the final result. Our image-based avatars naturally exhibit all fine-scale facial details such as folds and wrinkles, since the captured images intrinsically record all related information.

To create an image-based avatar as described above, we need to construct coarse-scale face and hair geometries from a small set of images. The face model can be easily constructed by fitting a generic blendshape model to the captured images, in a way similar to the one used in existing work. The construction of the hair model, however, is challenging. There is no generic template models for hairs – different users may have dramatically different hairstyles. We have to build the model solely from the input images. Furthermore, hairs, especially long ones, may exhibit non-rigid deformations among different head poses, due to the effect of gravity or interaction between the body and hairs. Constructing the hair as a static geometry from the input images using structure-from-motion (SfM) techniques may result in a model that does not match the in-

*Corresponding authors: Hongzhi Wu (hongzhi.wu@gmail.com), Kun Zhou (kunzhou@acm.org)

SIGGRAPH '16 Technical Paper, July 24-28, 2016, Anaheim, CA,
ISBN: 978-1-4503-4279-7/16/07
DOI: http://dx.doi.org/10.1145/2897824.2925873

put well, leading to undesirable animation results. To tackle these challenges, we propose a method to build a morphable hair model that approximates the dynamic hair geometry well and is sufficient to guide the image warping and blending at runtime. We first infer depth information of the hair region in each image independently. All depth maps are then refined in a joint optimization, taking into account the inter-image consistency. Finally, we construct a topologically consistent morphable model from all depth maps. The constructed model is compact and allows high runtime efficiency.

During runtime animation, for each frame, we use a real-time face tracker to calculate the parameters of facial motion, which are then used to construct the coarse 3D geometry of the avatar. Next, the input images are warped and blended to generate a novel view of the avatar, guided by the coarse geometry. To this end, we introduce a real-time algorithm that makes use of the coarse geometry to determine the per-pixel weight for each warped image, ensuring smooth, seamless integration of different image regions. We demonstrate compelling results from our algorithm on a variety of cases with different hairstyles and headwear.

In summary, the main contributions of our work include:

- We introduce a novel image-based representation for dynamic 3D avatars, which allows effective handling of various hairstyles and headwear, and can generate expressive facial animations with fine-scale details.
- We develop algorithms for creating an image-based avatar from a set of sparsely captured images of a user. In particular, we propose an optimization method to construct a topologically consistent morphable model to approximate the dynamic hair geometry in the captured images.
- We design a real-time algorithm for synthesizing novel views of an image-based avatar so that the avatar follows the facial motions of an arbitrary actor.

2 Related Work

Dynamic Avatar Creation. In comparison with facial muscle simulation (e.g., [Venkataramana et al. 2005]) and parametric models (e.g., [Jimenez et al. 2011]), data-driven approaches are more widely used to create dynamic avatars in practice, as they produce realistic facial motions at modest computational costs. For example, Amberg et al. [2007] use a morphable model [Blanz and Vetter 1999] to reconstruct a static 3D face from multi-view images. The multi-linear models [Vlasic et al. 2005; Cao et al. 2014b] that capture a joint space of identity and expression are often used to create a user-specific blendshape model by optimizing the identity coefficients to fit a set of input images [Cao et al. 2013] or video frames [Cao et al. 2014a]. Dynamic geometry variations to the blendshape model can also be linearly modeled in real-time while tracking RGBD videos [Bouaziz et al. 2013; Li et al. 2013]. These linear models, while suitable for real-time face tracking and animation, are unable to capture facial details such as wrinkles.

In high-end production, special hardware setups (e.g., the Light Stage system) have been used to create photorealistic dynamic avatars with fine-scale skin details [Alexander et al. 2009; Alexander et al. 2013]. Jimenez et al. [2010] compute dynamic skin appearances by blending hemoglobin distributions captured with different expressions. In their subsequent work, expression-dependent normal maps are interpolated to add realistic wrinkles to an animated face [Jimenez et al. 2011]. Nagano et al. [2015] synthesize skin microstructures based on local geometric features derived from high-precision microgeometry, acquired with an LED sphere and a skin deformer. These custom devices, however, are unaffordable for average users.

Some techniques aim to create a dynamic avatar from a single image. Taking one image of an avatar and a user as input, Saragih et al. [2011] learn a mapping function between the expressions of the two. At runtime, the user's face is tracked by fitting a deformable face model, which is then used with the learned map to generate the corresponding shape for the avatar's face. Cao et al. [2014b] introduce a technique to animate a still face image with the facial performance of an arbitrary actor. It first uses a multi-linear face model to fit a blendshape model for the subject in the image, and then textures the model with the image. At runtime, the tracked head motion and expressions of the actor are transferred to animate the blendshape model, from which a novel image is rendered. To handle hair, it uses a single-view hair modeling technique [Chai et al. 2012] to reconstruct a strand-based 3D hair model, which is also transformed together with the face model and rendered into the final result. As only one input image is used, the resulting avatars are not very expressive and do not have fine-scale details such as dynamic wrinkles. Large head rotations and exaggerated expressions are also problematic to these techniques.

Recent research begins to investigate ways to create fully-rigged avatars with fine-scale details for average users at home. Ichim et al. [2015] propose to build a two-scale representation of a dynamic 3D face rig from a hand-held video. Using a set of images of a user in neutral expression, they first fit a morphable face model to the point cloud extracted with structure-from-motion, resulting in a user-specific neutral face model. Then, a user-specific blendshape model (i.e., the dynamic face rig) is reconstructed by fitting a generic blendshape template at medium resolution to the static neutral model and the user's expression in the captured video. Fine-scale facial details such as dynamic wrinkles are represented as normal and ambient occlusion maps, estimated with shape-from-shading. Garrido et al. [2016] propose an automatic approach to creating personalized 3D face rigs solely from monocular video data (e.g., vintage movies). Their rig is based on three-scale layers, ranging from the coarse geometry to static and transient fine details on the scale of folds and wrinkles. Casas et al. [2015] generate a user blendshape model with textures using an RGBD camera, which considers only the front face. It is not clear how to model a full head that includes ears and hair.

Unlike existing techniques that adopt multi-scale geometric representations of face rigs, we represent a rig as a set of images of the user's face, along with some coarse-scale geometric information needed for runtime animation synthesis. All fine-scale facial details such as folds and wrinkles are implicitly recorded in our image-based representation and do not need any special treatment. More importantly, this image-based representation enables us to effectively handle hair and headwear, which are crucial parts of a user's appearance but largely ignored in existing work.

Face Tracking and Facial Performance Capture. In film and game production, special equipments, such as facial markers, camera arrays and structured lighting, have long been used to capture high-fidelity facial performance [Zhang et al. 2004; Weise et al. 2009; Bradley et al. 2010; Beeler et al. 2011; Huang et al. 2011]. Such equipments are usually not available for consumer-level applications, where only commodity hardware such as video and RGBD cameras are accessible.

Video-based face tracking have been extensively studied in both computer vision and graphics [Essa et al. 1996; Pighin et al. 1999; DeCarlo and Metaxas 2000; Chai et al. 2003; Valgaerts et al. 2012; Garrido et al. 2013; Shi et al. 2014]. The latest techniques demonstrate robust real-time tracking and animation from an ordinary web

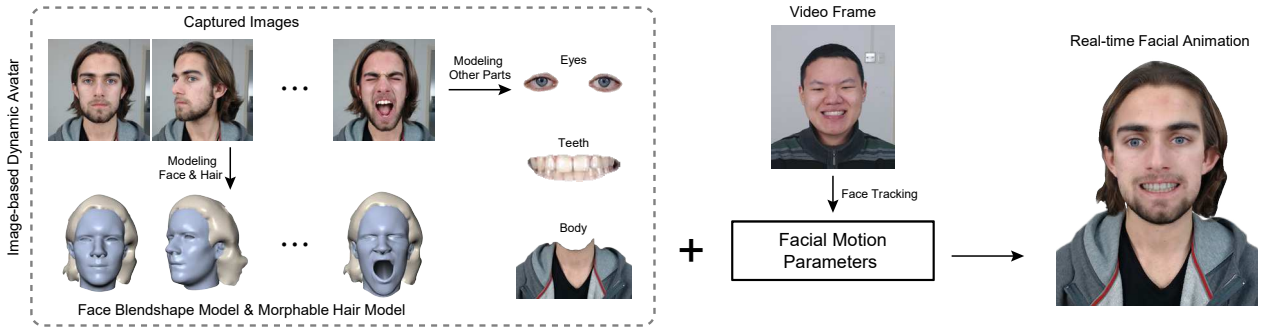

Figure 2: *Overview of our pipeline. From captured images of the user that sparsely sample over head poses and expressions, we build a head blendshape and a morphable hair model to represent the corresponding coarse geometry. Other components such as eyes and teeth are expressed as billboards. During runtime, an actor can drive our avatar using a single-camera-based face tracker. The captured images are warped and blended under the guidance of the coarse geometry to generate real-time facial animation with fine-scale details.*

camera [Cao et al. 2013; Cao et al. 2014a]. High-fidelity facial details such as wrinkles can also be reconstructed in real-time using shading cues [Cao et al. 2015]. Making use of commodity RGBD cameras, researchers have achieved highly impressive facial tracking and performance-based animation [Weise et al. 2011; Baltrušaitis et al. 2012; Bouaziz et al. 2013; Li et al. 2013; Li et al. 2015; Liu et al. 2015]. The head motion and facial expression coefficients tracked by these methods can be used to drive the facial animation of our image-based avatars in real-time.

Image-based Rendering. Our work is closely related to image-based rendering [Shum et al. 2007], in particular view morphing [Seitz and Dyer 1996] and view interpolation [McMillan 1997]. To generate a novel view of an object from images taken at different views, with or without the help of extra geometric information of the object, these techniques typically start with establishing correspondences among images; then images from nearby views are warped into the novel view based on the correspondences, and blended to obtain the final result. For example, Stich et al. [2008] synthesize novel views of general images captured at different views and times. Xu et al. [2011] generate novel motion/view of human performance, using a database of multi-view video sequences.

Face prior can be incorporated in novel-view image synthesis as extra source of information to improve the quality of the result. Zanella et al. [2007] use Active Shape Model (ASM) to identify facial feature points on images, and then linearly interpolate their locations for frontal view morphing. Yang et al. [2012] propose a method to morph two images with the help of a reconstructed 3D face model. Large rotation and expression changes between the two images can be handled. None of these techniques can create fully rigged dynamic avatars as achieved in this paper.

Image-based Hair Modeling. Our work is also related to image-based hair modeling algorithms [Wei et al. 2005; Paris et al. 2008; Luo et al. 2013; Hu et al. 2014a; Hu et al. 2014b]. Techniques in this category typically reconstruct a static, 3D geometric model of hair, based on multiple images taken from different view points or under different illuminations. Recently, hair modeling methods based on just a single image [Chai et al. 2012; Chai et al. 2013; Chai et al. 2015; Hu et al. 2015] are proposed to make hair modeling more accessible to non-professional users. Dynamic hair geometry can also be acquired from multi-view inputs using spacetime optimization [Xu et al. 2014]. One major difference between our work and existing image-based hair modeling work is that we do not model the hair geometry to the strand level. Instead, we take a hybrid approach that uses a 3D morphable hair model for the coarse geometry, and images to capture fine details.

3 Overview

Our approach constructs a personalized dynamic avatar from a set of sparsely captured images of the user's face. Unlike previous techniques that represent avatars as textured geometries [Ichim et al. 2015; Garrido et al. 2016], we represent avatars using the captured images, along with some coarse geometry information of the images. During real-time animation, the images are warped and blended under the guidance of the geometry information to make avatars exhibit the facial motions of an arbitrary actor. As the captured images contain all fine-scale facial details, we do not need to reconstruct multi-scale geometric details as in [Ichim et al. 2015; Garrido et al. 2016]. Furthermore, this image-based representation enables us to effectively handle the user's hair.

The pipeline for building and using an image-based dynamic avatar is shown in Fig. 2. It has three main steps: image acquisition, avatar construction (§4) and real-time animation (§5). The geometry information in our avatar representation consists of two main components: a face blendshape model and a morphable hair model. The construction of the face model is relatively easy: geometric priors from existing 3D facial expression databases can be utilized to fit a generic blendshape model to the captured images (§4.2). For the hair model, such priors are not available – different users may have drastically different hairstyles. We need to construct the hair model solely from the captured images (§4.3). For other components, such as eyes, teeth and body, we adopt a billboard-based approach to model them from the input images (§4.4). During runtime, we use the motion parameters computed by a state-of-the-art face tracker to drive the face blendshapes, as well as the morphable hair model (§5.1). We then warp and blend the captured images with the guidance from the coarse geometry (§5.2). Other facial components are seamlessly integrated to complete the final result (§5.3).

Image Acquisition. To construct an image-based avatar, we capture 32 images for a particular user with a web camera: 15 for different prescribed head poses, and 17 for different expressions.

The predefined head poses consist of head rotations over a set of angles with a neutral expression. Specifically, the set of rotations, expressed in Euler angles, are as follows: yaw from -60° to 60° at 20° intervals (with pitch and roll at 0°); pitch from -30° to 30° at 15° except for 0° (with yaw and roll at 0°); and roll is sampled similarly as pitch. Essentially, the user sweeps her head three times along different axes, to get to the prescribed angles. Note that head

rotations in exact angles are not required from the user; a rough approximation is sufficient for our algorithm.

To capture images of different expressions, the user is asked to keep the neutral head pose and perform the following 17 expressions: mouth stretch, smile, brow down, brow raise, disgust, squeeze left eye, squeeze right eye, roar, mouth left, mouth right, grin, mouth up, lip pucker, lip funnel, teeth, cheek blowing, eye close.

4 Image-based Avatar Construction

We build the image-based avatar representation from the captured images $\{I_1, I_2, ...I_{32}\}$ of a user, where I_1 corresponds to the image of neutral expression under the neutral head pose. Our representation is made up of preprocessed images and crude geometric proxies for various components of the avatar. We first apply user-assisted segmentation and facial landmark labeling to the captured images. Next, we build a face blendshape model and a morphable hair model from the images, as well as billboards for the rest parts, to represent the coarse geometry.

4.1 Image Preprocessing

We perform two steps in preprocessing: segmenting the captured images into different components, and landmark labelling that facilitates subsequent construction of geometric models.

Segmentation. The first step is to segment the images into several layers: head, hair (including headwear), eyes, teeth, body and background. We employ Lazy Snapping [Li et al. 2004] to perform the segmentation with minimal user assistance. In our experiments, a few strokes are sufficient to complete the decomposition. As the hair may overlap with other regions in a complex fashion, we further perform image matting [Levin et al. 2008] on the hair layer to refine the segmentation. This results in an additional alpha channel for the hair layer (see Fig. 3).

Landmark Labeling. Similar to [Cao et al. 2013], we semi-automatically label a few facial landmarks S_i for each image I_i. These landmarks indicate the 2D positions of a set of facial features, such as the contour of mouth and eyes, and the silhouette of face (see Fig. 3). Specifically, we use the face tracker [Cao et al. 2014a] to automatically locate these landmarks, and then manually adjust them with a drag-and-drop tool.

4.2 Face Blendshapes

We represent the coarse, dynamic geometry of the face (along with the head) with a blendshape model, using the labeled images $\{I_i, S_i\}$ as input. The blendshape model helps warp and blend segmented images to obtain the final appearance of face in our avatar. First, we compute an initial static 3D face model F_i^{init} for each image, based on the FaceWarehouse database [Cao et al. 2014b], which includes the 3D face geometry of 150 different individuals, each with 47 expressions. Then, F_i^{init} is refined by mesh deformation to obtain F_i. Finally, we compute the expression blendshapes $\{B_j\}$ from $\{F_i\}$.

Initial Modeling by Tensor Fitting. For each image I_i, we compute the initial face model by interpolating the 3-mode tensor C of FaceWarehouse, using the global identity coefficients $\mathbf{w}^{id}$ and per-image expression coefficients $\mathbf{w}_i^{exp}$:

$$F_i^{init} = C \times_2 \mathbf{w}^{id} \times_3 \mathbf{w}_i^{exp}, \tag{1}$$

where $\times_2$ and $\times_3$ are the tensor contraction operations of second and third mode. In order to generate F_i^{init}, we need to determine $\mathbf{w}^{id}$ and $\mathbf{w}_i^{exp}$.

Figure 3: *Image preprocessing. From left to right: a captured image, segmented layers, the hair layer after matting, and facial landmarks.*

Observe that F_i^{init} is related to its 2D image-space projection F_i^{im} as:

$$F_i^{im} = \Pi\left(\mathbf{R}_i \cdot F_i^{init} + \mathbf{T}_i\right). \tag{2}$$

Here $(\mathbf{R}_i, \mathbf{T}_i)$ is the object-to-camera-space transformation, and Π maps a point from the camera space to the image space, based on the intrinsic matrix of the camera. To compute the unknown $\mathbf{w}^{id}$ and $\{\mathbf{w}_i^{exp}, \mathbf{R}_i, \mathbf{T}_i\}$, we employ the method in [Cao et al. 2013], which minimizes the error between the labeled landmarks S_i and the corresponding mesh vertices of F_i^{im}:

$$E_{ld} = \sum_k \|\mathbf{v}_k - S_{i,k}\|^2. \tag{3}$$

Here $\mathbf{v}_k$ a vertex in F_i^{im}, $S_{i,k}$ is the landmark corresponding to $\mathbf{v}_k$ in S_i .

Geometry Refinement by Mesh Deformation. The initial face model interpolated from FaceWarehouse is only a rough approximation. To further refine the accuracy of the model, we optimize the positions of all mesh vertices by minimizing Eqn. (3) plus a Laplacian regularization term [Huang et al. 2006] defined as follows:

$$E_{lap} = \sum_k \left\| \Delta\mathbf{v}_k - \frac{\delta_k}{|\Delta\mathbf{v}_k|}\Delta\mathbf{v}_k \right\|^2. \tag{4}$$

Here Δ is the discrete mesh Laplacian operator based on the cotangent formula [Desbrun et al. 1999], δ_k is the magnitude of the Laplacian coordinates of vertex k in F_i^{init}. We denote the refined results as $\{F_i\}$.

Blendshape Generation. In the previous step, we have improved the quality of the 3D face model in each image. The blendshapes obtained from the FaceWarehouse database using $\mathbf{w}^{id}$ can be also refined to create more accurate, dynamic geometry. Thus, we employ an example-based face rigging method [Li et al. 2010] to compute the refined blendshapes $\{B_j\}$ from $\{F_i\}$.

4.3 Morphable Hair Model

The construction of the hair model is more difficult than that of the face model. Different users may have considerably different hairstyles. There are no generic templates available to model them. We have to construct the hair model solely from the captured images. Moreover, hairs, especially long ones, may exhibit non-rigid deformations when the head is rotating, due to the effect of gravity or interaction between the body and hairs. Representing the hair as a static geometry with rigid transformations may not match the captured images well and could result in undesirable animation results. Therefore, we build a morphable model to approximate the dynamic geometry of hair.

Due to the relatively low image quality of ordinary web cameras, hair strands are not distinguishable in the captured images. Thus it would be difficult to calculate accurate, strand-level correspondences among this sparse set of images. So unlike previous techniques that aim to construct an accurate geometry down to the

strand level for a static hair [Wei et al. 2005; Paris et al. 2008; Luo et al. 2013; Hu et al. 2014a; Chai et al. 2012; Hu et al. 2015], we only construct a coarse geometric proxy that is sufficient to help warp and blend related image segments during runtime animation. To build the model, we first infer depth information of the hair region in each image independently. All depth maps are then refined in a joint optimization, taking into account the inter-image consistency. Finally, we build a topologically consistent morphable model from all depth maps.

Single-Image Depth Estimation. We adapt the single-view hair modeling technique in [Chai et al. 2012], which minimizes a silhouette and a smoothness energy function. Initially, the hair silhouettes $\partial\Omega_h$ is identified in the segmented hair region Ω_h. The initial depth D^0 is set as follows: for a pixel on the interior silhouettes, we directly assign the depth of the corresponding point on the head model F_i; for a pixel on the exterior silhouettes, we assign the average depth of points on the exterior silhouettes of F_i. The silhouette energy term is defined as:

$$E_{sil} = \sum_{p \in \partial\Omega_h} \left(\left\| D_p - D_p^0 \right\|^2 + \left\| \mathbf{n}_p - \nabla\Omega_h \right\|^2 \right), \quad (5)$$

where D_p is the per-pixel depth we would like to solve, $\mathbf{n}_p$ is the normal and $\nabla\Omega_h$ denotes the image gradient along the hair silhouette. Next, the smoothness term, which encourages the smoothness of hair depths and normals, is defined as:

$$E_{sm} = \sum_{p \in \Omega_h} \sum_{q \in N(p)} \left(\omega_d \| D_p - D_q \|^2 + \omega_{\mathbf{n}} \| \mathbf{n}_p - \mathbf{n}_q \|^2 \right), \quad (6)$$

where $N(p)$ denotes the 4-connected neighbors of p, ω_d / $\omega_{\mathbf{n}}$ is the parameter that controls the depth / normal smoothness. By minimizing the energy of $E_{sil} + E_{sm}$, we obtain the depth map D_i for each image I_i.

Inter-image Depth Optimization. The above depth estimation is performed on each image independently, without taking into account the fact that all depths correspond to the same avatar component. This results in less accurate 3D geometry reconstructed from the depth maps. To address this issue, we propose a joint depth optimization to explicitly enforces inter-image consistency. The optimization is solved iteratively in an alternating fashion. In each iteration, we loop over all depth maps; we refine one depth map D_i at a time, while keeping others fixed. The idea is to constrain the current depth map with other ones, to enforce the consistency. Specifically, we transform all other depth maps $\{D_j\}_{j \neq i}$ to the camera space of D_i, denoted as $\{\hat{D}_j\}_{j \neq i}$; then the inter-image consistency term is expressed as the sum of pairwise differences between D_i and $\hat{D}_j$:

$$E_{con} = \sum_j \sum_p \left\| D_{i,p} - \hat{D}_{j,p} \right\|^2. \quad (7)$$

We solve the refined depth map, by minimizing this consistency term plus the silhouette term Eqn. (5) and the smoothness term Eqn. (6) defined in the previous subsection.

Now we describe in detail how to transform a depth map D_j to the camera space of D_i, which results in $\hat{D}_j$. For cases like short hairs that can be well approximated by a rigid proxy, we simply perform a rigid transformation between camera spaces using the parameters $\{\mathbf{R}_i, \mathbf{T}_i\}$ solved during face blendshape construction as:

$$P(\hat{D}_{j,p}) = (\mathbf{R}_i, \mathbf{T}_i) \cdot (\mathbf{R}_j, \mathbf{T}_j)^{-1} \cdot P(D_{j,p}), \quad (8)$$

where $P(\cdot)$ is a 3D point in the camera space, corresponding to a depth pixel.

Figure 4: *A visualization of hair region correspondences. We show the hair correspondences computed with our method in the right image, which is a color visualization based on the displacement vectors from each pixel in the hair region of the left image to a corresponding pixel in the center image.*

For cases like long hair, we need to further perform a non-rigid transformation to $\hat{D}_j$, as the geometry can no longer be well approximated as a rigid proxy. Specifically, we establish sparse correspondences C_{ij} between D_i and D_j, which will be detailed later. Then, based on the correspondences, we deform D_j to obtain $\hat{D}_j$, by minimizing the following energy:

$$E_j = \sum_{(c_i, c_j) \in C_{ij}} \left\| \hat{D}_{j,c_j} - D_{i,c_i} \right\|^2 + \omega_l \sum_k \left\| \Delta\mathbf{v}_k - \frac{\delta_k}{|\Delta\mathbf{v}_k|} \Delta\mathbf{v}_k \right\|^2, \quad (9)$$

where (c_i, c_j) is a pair of correspondences, vertex $\mathbf{v}_k$ is the k-th vertex of the mesh corresponding to the depth map $\hat{D}_j$, and the Laplacian term is the same as in Eqn. (4). ω_l is the parameter to control the regularization term, which is set to 10 in our experiments.

Correspondence Computation. As aforementioned, hairs may be non-rigidly deformed when the head moves. Thus, before solving for D_i in the inter-image optimization, we need to determine correspondences between D_i and another depth map D_j, in order to compute the deformation that produces $\hat{D}_j$. Our algorithm consists of three steps: image-space correspondence computation, low-quality match pruning, and guided correspondence refinement.

In the first step, we compute dense correspondences between the hair regions in I_i and I_j, using the PatchMatch algorithm [Barnes et al. 2009]. The generated correspondences may not be accurate for all pixels. So in the second step, we prune low-quality matches and use a deformation algorithm to compute other potential correspondences. To do this, we first construct a regular mesh C_i of the hair region of I_i, where each pixel is regarded as a vertex whose depth value is assigned from D_i. Next, for each vertex of C_i, if the PatchMatch errors of all pixels in a 3x3 neighborhood of the vertex are lower than a given threshold 0.05 and the PatchMatch offsets of these pixels are similar, we average the offsets of neighboring pixels to obtain that of the current vertex; otherwise, the correspondence of the vertex is set as invalid. For the vertices of C_i with valid correspondences, we set their corresponding points in I_j as positional constraints, and deform C_i using the Laplacian deformation algorithm [Huang et al. 2011]. The deformed mesh C_i' is rendered to the image plane of I_j, and each vertex of C_i' passing the depth test can find its corresponding point in I_j. In the final step, we refine the correspondences at vertices of C_i, by searching for the best matching patch in a local 9x9 neighborhood in I_j. The center of the neighborhood in patch search is determined by the initial correspondence. If the PatchMatch error at a vertex is still greater than the threshold, we simply set the correspondence at this vertex as

invalid and leave it out in the optimization of Eqn. (9). An example of the computed correspondences is visualized in Fig. 4.

Note that we use PatchMatch instead of more traditional approaches such as optical flow to compute the dense correspondences. The reason is that our input is images with different head poses and expressions, which has large inter-image variations; PatchMatch is more robust in this case, while optical flow is better suited for processing data like a video sequence, where the neighboring frames have small variations.

Morphable Model Generation. We build the morphable hair model from all hair depth maps. Specifically, we first transform and deform all depth maps into a common coordinate system, the camera space of I_1. The is achieved by constructing a regular mesh for each hair region and deforming the mesh using Laplacian deformation in the same way we compute the correspondences. Each pixel in a depth map is then converted into a 3D point. We remove the outlier points with the normal's z direction smaller than a given threshold 0.5 (i.e., grazing view angles). Next, Poisson surface reconstruction [Kazhdan et al. 2006] is applied to the point cloud to generate the coarse hair geometry H_1 for I_1. As the camera space transformations and deformations between depth maps are all known, we transform and deform H_1 to the other 14 images $\{I_i\}$ of different head poses, resulting in $\{H_i\}_{i=1,2,...,15}$. We regard the set of geometries $\{H_i\}$ as the morphable hair model, which spans the space of a user's coarse geometry of hair with different head poses. Note that we assume that the facial expression has no effect over the hair shape.

4.4 Handling Eyes, Teeth and Body

We have described the representations for face and hair in previous subsections. To complete our avatar, we create simple billboards for eyes, teeth and body. Unlike face and hair, these components vary less with different expressions. Therefore, we build the eye and body billboards based on the image with neutral expression under frontal view (i.e., I_1), and the tooth billboards from the image with the teeth expression.

Eyes. For each eye, we represent it with two billboards, one for the iris, and the other for the sclera, following the work of [Saragih et al. 2011]. We first detect a rectangle of interest using the image-space bounding box of selected vertices on the head mesh. The position and size of the iris is automatically determined by the largest ellipse found inside the rectangle. We then copy the detected iris in the image to its billboard. For the sclera billboard, we copy from the eye region of the image, and remove the part that belongs to the iris. To fill in the missing pixels in the billboard, we apply the PatchMatch algorithm [Barnes et al. 2009] to synthesize their colors, using the sclera region as the source.

Teeth. We build two billboards for upper and lower jaw teeth, respectively. In the image with the teeth expression, we also identify a rectangle of interest using the image-space bounding box of selected vertices on the face mesh. The sizes of teeth billboards are automatically determined by a general teeth model [Thies et al. 2015] and the face mesh. We also provide a drag-and-drop tool, for manually refining the selection of billboard contents from the image.

Body. We approximate the upper body as a single billboard, which is directly filled with pixels from the body layer of I_1. The depth of body billboard is assigned as the average depth of points on the exterior silhouettes of the head model.

Algorithm 1 Rendering algorithm for an image-based avatar

Input: rigid head transformation $(\mathbf{R}, \mathbf{T})$ and facial expression coefficients $\mathbf{e}$, estimated from the face tracker
Output: a rendered image of the avatar
Begin
  Render the body billboard;
  Compute the head mesh F based on $(\mathbf{R}, \mathbf{T})$ and $\mathbf{e}$;
  Compute the new head mesh $\hat{F}$ after neck stablization;
  Compute the new head transformation $(\hat{\mathbf{R}}, \hat{\mathbf{T}})$ from $\hat{F}$;
  Compute the hair model H based on $(\hat{\mathbf{R}}, \hat{\mathbf{T}})$ and $\mathbf{e}$;
  for captured image I_i **do**
    Warp it to the current view with the guidance of $\hat{F}/H$;
    Compute per-vertex weights;
    Generate a weight map w_i using RBF-based interpolation;
  end for
  Blend all warped images using $\{w_i\}$ and render the result;
  Render the billboards for eyes and teeth with computed masks;
End

5 Real-time Animation

We employ the single-camera-based face tracker [Cao et al. 2014a] to capture facial motions and drive our image-based dynamic avatar. At runtime, for an input frame, the face tracker automatically computes the parameters of facial motion, including the rigid head transformation $(\mathbf{R}, \mathbf{T})$ and facial expression coefficients $\mathbf{e}$. Based on these parameters, we construct the coarse 3D geometry of the avatar for the current frame, and then warp and blend pre-captured images to get the final result, guided by the coarse geometry.

5.1 Geometry Construction

We describe how to construct the coarse geometry for face and hair, from the rigid head transformation and the expression coefficients.

Face. With the precomputed face blendshape model $\{B_j\}$, we build the face geometry for an input frame as:

$$F = \mathbf{R} \cdot (B_0 + \sum_{j=1}^{47} \mathbf{e}_j B_j) + \mathbf{T}, \tag{10}$$

where B_0 is the neutral expression blendshape, $\mathbf{e}_j$ is the j-th coefficient of $\mathbf{e}$.

To make the head seamlessly connected with the neck, we need to constrain the positions of the vertices near the neck to conform with the body billboard, which only moves with the translation $\mathbf{T}$. Then we adjust the positions of the rest vertices, using Laplacian deformation [Huang et al. 2006]. The new head geometry is denoted as $\hat{F}$. We then update the rigid transformation of the new geometry as $\left(\hat{\mathbf{R}}, \hat{\mathbf{T}}\right)$, via 3D registration between the two meshes F and $\hat{F}$.

Hair. We construct the hair geometry H based on the face mesh $\hat{F}$, to which it is attached. Similar to generating the head geometry, we obtain the hair geometry by interpolating precomputed hair models $\{H_i\}$ as:

$$H = \hat{\mathbf{R}} \cdot (\sum_{i=1}^{15} r_i H_i) + \hat{\mathbf{T}}. \tag{11}$$

Here r_i is the weight for H_i, calculated based on the head rotation transformations of the current frame $\hat{\mathbf{R}}$ and captured images $\{\mathbf{R}_i\}$

as:

$$r_i = \frac{e^{-\omega_r \|\hat{\mathbf{R}} - \mathbf{R}_i\|^2}}{\sum_{j=1}^{15} e^{-\omega_r \|\hat{\mathbf{R}} - \mathbf{R}_j\|^2}}, \tag{12}$$

where ω_r is an interpolation parameter (set to 10 in our experiments). We express $\mathbf{R}$ and $\hat{\mathbf{R}}$ as quarternions. Note that we assume that the shape of the hair depends only on the rotation of the head model, $\hat{\mathbf{R}}$.

5.2 Image Warping & Blending

We warp the captured images of face and hair regions to the current frame, guided by the corresponding coarse geometry. The results are denoted as $\{I_i^{warp}\}$.

Next, each pixel in the final image of the avatar is computed by blending the warped images, using a weighted average. To determine the per-pixel weight for each warped image, we compute corresponding weights on vertices of the coarse geometry F/H in the current frame, then interpolate the weights to pixels via radial basis functions (RBFs).

Specifically, for a warped image I_i^{warp}, we compute a per-vertex weight $w(v_k)$ as the product of an orientation term, a normal similarity term and an expression similarity term. The idea is to assign a large weight in cases of similar normals/expressions between the geometry of the current frame and that of the captured image, and when the vertex is oriented close to the view direction:

$$w(v_{i,k}) = e^{-\omega_z (1-\mathbf{n}_{i,k}^z)^2} \cdot e^{-\omega_n (1-\mathbf{n}_{i,k} \cdot \mathbf{n}_k)^2} \cdot \alpha_{i,k} e^{-\omega_e (1-\psi(\mathbf{e}_i, \mathbf{e}))^2}.$$

Here v_k is a vertex of F/H, $v_{i,k}$ is its corresponding vertex on the precomputed geometry F_i/H_i for image I_i. $\mathbf{n}_{i,k}/\mathbf{n}_k$ is the normal of vertex $v_{i,k}/v_k$, $\mathbf{n}_{i,k}^z$ is the z component of $\mathbf{n}_{i,k}$. ω_z, ω_n and ω_e are the parameters to control the relative importance of each term (we use 5, 10 and 30 respectively in experiments). $\alpha_{i,k}$ is a manually specified binary mask for a particular expression, which is 1 for regions related to the semantic information of the expression, and 0 otherwise. Note that we set α in a simple painting interface, as shown in the inset figure. This is only a one-time process performed on the template blendshapes, independent of avatars. $\mathbf{e}_i$ and $\mathbf{e}$ are the expression coefficients for I_i and I, and $\psi(\mathbf{e}_i, \mathbf{e})$ measures the similarity between them as follows:

$$\psi(\mathbf{e}_i, \mathbf{e}) = \frac{(\mathbf{e}_i \cdot \mathbf{e})}{\|\mathbf{e}_i\| \|\mathbf{e}\|}. \tag{13}$$

Here $(\cdot)$ is the dot product of two vectors.

Once we obtain *per-vertex* weights for each warped image, we compute the *per-pixel* weights $\{w_{i,p}\}$ via RBF-based interpolation as:

$$w_{i,p} = \sum_k e^{-\omega_u \|u_p - u_{i,k}\|^2} \beta_k w(v_{i,k}). \tag{14}$$

Here u_p is the 2D coordinates of pixel p, and $u_{i,k}$ is the 2D image-space coordinates of $v_{i,k}$. ω_u is the parameter to control the region of influence of $v_{i,k}$. β_k is a visibility term, which is 1 if v_k is visible in the current frame and 0 otherwise. $w_{i,p}$ is further normalized to satisfy that $\sum_i w_{i,p} = 1$.

Note that in practice, we compute the per-vertex weights on a subset of original vertices ($1/10$ of the number of original vertices in our experiments), obtained with uniform sampling. We find that using a smaller number of vertices helps smooth the boundaries when blending different images, resulting in visually satisfactory results.

Figure 5: *Seamless integration of eyes and teeth to the avatar. We composite the billboards of eyes and teeth into the final rendering result using computed masks.*

5.3 Other Components

We describe how to generate the final appearance of eyes, teeth and body of our avatar in this subsection.

For eyes, we first add two landmarks for pupils as in [Cao et al. 2014a] to accurately track and reproduce the iris at runtime. For the sclera billboards, we directly apply the rigid transformation $(\hat{\mathbf{R}}, \hat{\mathbf{T}})$ to render them. For the iris billboards, we perform a translation in addition to $(\hat{\mathbf{R}}, \hat{\mathbf{T}})$, estimated from the tracked pupil locations.

For teeth, the billboard of upper jaw teeth are connected to and move with the head, by applying the rigid transformation $(\hat{\mathbf{R}}, \hat{\mathbf{T}})$, similar to [Cao et al. 2014b]; the lower jaw teeth are connected to and moved with the tip of the chin.

For the body, we simply translate the corresponding billboard with $\hat{\mathbf{T}}$ and render it as part of the background, before processing any other components of the avatar.

To seamless integrate the appearance of eyes and teeth with the rest of the avatar, we employ masks computed on-the-fly for the image composition. Specifically, we add new triangles to the head model by zippering the lids of eyes (or the lips), which are open holes in the template model. The rendered region of these triangles then serve as masks for compositing the eye and tooth billboards. Please see Fig. 5 for an illustration.

The whole rendering algorithm for our image-based avatar is summarized in Algorithm 1.

6 Experimental Results

We have implemented the described algorithms on a workstation with a quad-core Intel i7 CPU running at 3.6GHz, 32 GB of memory and an NVIDIA GTX 760 graphics card. All input images are captured with an off-the-shelf web camera at a resolution of 1280x720. To construct one avatar, it typically takes 10 minutes for image acquisition, 40 minutes for image preprocessing, and 15 minutes for computing the head blendshape model and the morphable hair model. A completed image-based avatar requires on average 50MB of memory storage. During runtime, it takes about 30ms to animate and render the avatar for an input video frame. Combined with the face tracker [Cao et al. 2014a], our CPU-based system generates real-time facial animation of the avatar at 25 frames per second. Please refer to the accompanying video for live demos of our system.

We show in Fig. 13 the main results of our approach on a wide variety of users, with different hairstyles and headwear. Note that the

Figure 6: *Validation results. The captured video frames of a user are shown on the bottom, and the rendering results of our avatars of the same user are on the top. Our results well match the corresponding video frames.*

face blendshape model and the morphable hair model in our algorithm are only used as geometry proxies to warp images. Fine details of the avatar are encoded in images. In Fig. 13 we demonstrate fine-scale details such as folds and wrinkles on our avatars. No special care is needed to infer a fine-scale detail layer as in previous work, thanks to our image-based representation. Challenging cases like large rotations and hair deformations are also well handled, as our face blendshape model and morphable hair model efficiently capture such dynamics.

Moreover, we validate our technique by comparing a user's recorded video with the animation of the avatar constructed for this specific user. As shown in Fig. 6, our animation results well match the recorded video, indicating that our image-based avatars can help convey the realistic appearance of the individuals being modeled.

Evaluations. We evaluate the key components of our pipeline through several experiments. First, we demonstrate the effectiveness of the inter-image depth optimization in Fig. 7. As more iterations of the optimization are applied, we can see that the estimated hair geometry better approximates the hair in the images. A plot of the converging energy function is also visualized in the figure. In Fig. 8, we demonstrate the necessity of inter-image depth optimization, by comparing our results with those generated from the independent depth estimation, where the coarse hair geometry is generated by applying Poisson surface reconstruction to the point cloud produced by transforming the depth maps estimated independently to a common coordinate system. As shown, the inconsistent geometry from different images downgrades the quality of the hair model, leading to rendering artifacts.

In Fig. 9, we compare our weighting strategy for image blending, against using uniform weights across all images. Our weighting scheme produces more realistic results, compared with uniform weighting, which does not take into account similarity between the current pose/expression and those in the captured images.

Comparisons. We compare our technique with related methods. In Fig. 10, we show our results along with those computed with one structure-from-motion technique [Jancosek and Pajdla 2011]. As SfM only handles rigid motions, in this example, the user always keeps neutral expression and his hair can be well approximated by rigid geometry. Using the same set of captured images as our algorithm, the reconstructed result using SfM has small holes, particularly on the hair. Our result generated from 15 images looks smoother and more visually pleasing than that from SfM even with 64 images. While capturing more images can further improve the quality of SfM results, it also increases the capturing effort. More

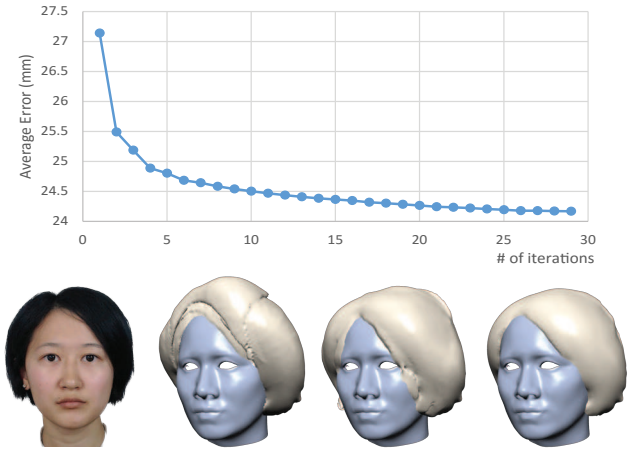

Figure 7: *Progress of the inter-image depth optimization. A plot of the energy function is shown on top. The bottom row illustrates the estimated hair geometry, whose quality gets improved as the iteration of the optimization increases.*

Figure 8: *Comparisons of inter-image depth optimization and independent depth estimation. For each image pair, we show our result on the left, and the result using independent depth estimation on the right.*

importantly, SfM techniques are not suitable for handling non-rigid face/hair deformations as shown in this paper.

Next, we compare our method with the image-based view morphing technique [Seitz and Dyer 1996], on a same set of input images. It is clear that using coarse geometric proxies to guide the warping and blending of images, as in our method, produces higher quality results than view morphing, a pure image-based approach. In fact, the results from view morphing exhibit ghosting artifacts, mainly due to the sparsity in inter-image correspondences.

In Fig. 12, we compare our method with a single-image-based avatar technique [Cao et al. 2014b]. Taking multiple images of the user allows us to properly sample the appearance variation of the user with different poses and expressions, so that challenging cases like large rotations or folds and wrinkles can be handled. Such effects are missing from the result of [Cao et al. 2014b], using only a single image.

7 Conclusions

We have introduced an image-based representation for dynamic 3D avatars. It allows effective handling of various hairstyles and headwear, and can generate expressive facial animations with fine-scale details. We also presents algorithms for constructing such an image-based avatar from a set of sparsely captured images of a user, and rendering it for real-time facial animation, driven by the facial motion of an arbitrary actor.

Our work is subject to a few limitations, which may lead to interesting future research. The avatars constructed in this paper do not support 360° rotations around the neck, where the geometry-based avatars of [Ichim et al. 2015] are more appropriate. It is possible to

Figure 9: *Comparisons of different weighting schemes for image blending. For each image pair, we show our result on the left, and the result using uniform weights on the right. Our results look more realistic as we take into account the similarity between the current pose/expression and those in the captured images.*

Figure 10: *Comparisons with one Structure-from-Motion technique. From left to right: a captured image, our result, SfM results using 15 and 64 images, respectively. Our results look smoother and more visually pleasing than SfM results.*

address the problem in our framework with more images captured from the back view. To reconstruct the geometry of the back of the head, one could estimate the rigid transformation of the head, by labeling additional landmarks, such as ears. In addition, our method does not handle well complex deformations of the hair, especially those causing significant occlusion changes. This is the limitation of our morphable hair model. To create a truly realistic hair animation, a strand-based hair representation is inevitable [Chai et al. 2014]. Finally, our current pipeline directly warps and blends captured images for novel view synthesis, without considering relighting effects. It is possible to apply intrinsic decomposition jointly on all images, to factor out the lighting and perform subsequent relighting.

Acknowledgments

We thank Jesse Caron, Ruitao Cai, Keara Cole, Emily Hadfield, Jorg Hoefer, Huiwen Jiang, Veit Lehmann, Zachary Smith, Riley Temple, Xichen Yang, Xiaoyao Yu for being our performers, Yiying Tong for proofreading the paper and the SIGGRAPH reviewers for their helpful comments. This work is partially supported by the NSF of China (No. 61272305, No. 61303135 and No. 61572429), the National High-tech R&D Program of China (No. 2012AA010903), the National Program for Special Support of Eminent Professionals of China, and Lenovo's Program for Young Scientists.

References

ALEXANDER, O., ROGERS, M., LAMBETH, W., CHIANG, M., AND DEBEVEC, P. 2009. The digital Emily project: photo-real facial modeling and animation. In *ACM SIGGRAPH 2009 Courses*, 12:1–12:15.

ALEXANDER, O., FYFFE, G., BUSCH, J., YU, X., ICHIKARI, R., JONES, A., DEBEVEC, P., JIMENEZ, J., DANVOYE, E., ANTIONAZZI, B., EHELER, M., KYSELA, Z., AND VON DER PAHLEN, J. 2013. Digital Ira: creating a real-time photoreal digital actor. In *ACM SIGGRAPH 2013 Posters*.

AMBERG, B., BLAKE, A., FITZGIBBON, A., ROMDHANI, S., AND VETTER, T. 2007. Reconstructing high quality face-surfaces using model based stereo. In *Proceedings of ICCV*, 1–8.

BALTRUŠAITIS, T., ROBINSON, P., AND MORENCY, L.-P. 2012. 3D constrained local model for rigid and non-rigid facial tracking. In *Proceedings of IEEE CVPR*, 2610–2617.

BARNES, C., SHECHTMAN, E., FINKELSTEIN, A., AND GOLDMAN, D. B. 2009. Patchmatch: A randomized correspondence algorithm for structural image editing. *ACM Trans. Graph. 28*, 3 (July), 24:1–24:11.

BEELER, T., HAHN, F., BRADLEY, D., BICKEL, B., BEARDSLEY, P., GOTSMAN, C., SUMNER, R. W., AND GROSS, M. 2011. High-quality passive facial performance capture using anchor frames. *ACM Trans. Graph. 30*, 4, 75:1–75:10.

BLANZ, V., AND VETTER, T. 1999. A morphable model for the synthesis of 3d faces. In *Proceedings of SIGGRAPH*, 187–194.

Figure 11: *Comparisons with the view morphing technique [Seitz and Dyer 1996]. For each image pair, we show our result on the left, and the view-morphing result on the right. Results from view-morphing suffer from ghosting artifacts, due to the lack of 3D geometric information.*

Figure 12: *Comparisons with a single-image-based avatar technique [Cao et al. 2014b]. Rendering results of our avatar are shown on the top, while those of the single-image-based avatar are on the bottom. Unlike the single-image-based method, our method can handle large rotations and generate fine details such as folds and wrinkles.*

BOUAZIZ, S., WANG, Y., AND PAULY, M. 2013. Online modeling for realtime facial animation. *ACM Trans. Graph. 32*, 4 (July), 40:1–40:10.

BRADLEY, D., HEIDRICH, W., POPA, T., AND SHEFFER, A. 2010. High resolution passive facial performance capture. *ACM Trans. Graph. 29*, 4, 41:1–41:10.

CAO, C., WENG, Y., LIN, S., AND ZHOU, K. 2013. 3d shape regression for real-time facial animation. *ACM Trans. Graph. 32*, 4 (July), 41:1–41:10.

CAO, C., HOU, Q., AND ZHOU, K. 2014. Displaced dynamic expression regression for real-time facial tracking and animation. *ACM Trans. Graph. 33*, 4 (July), 43:1–43:10.

CAO, C., WENG, Y., ZHOU, S., TONG, Y., AND ZHOU, K. 2014. Facewarehouse: A 3d facial expression database for visual computing. *IEEE Transactions on Visualization and Computer Graphics 20*, 3 (Mar.), 413–425.

CAO, C., BRADLEY, D., ZHOU, K., AND BEELER, T. 2015. Real-time high-fidelity facial performance capture. *ACM Trans. Graph. 34*, 4 (July), 46:1–46:9.

CASAS, D., ALEXANDER, O., FENG, A. W., FYFFE, G., ICHIKARI, R., DEBEVEC, P., WANG, R., SUMA, E., AND SHAPIRO, A. 2015. Rapid photorealistic blendshapes from commodity rgb-d sensors. In *Proceedings of the 19th Symposium on Interactive 3D Graphics and Games*, i3D '15, 134–134.

CHAI, J.-X., XIAO, J., AND HODGINS, J. 2003. Vision-based control of 3d facial animation. In *Symp. Comp. Anim.*, 193–206.

CHAI, M., WANG, L., WENG, Y., YU, Y., GUO, B., AND ZHOU, K. 2012. Single-view hair modeling for portrait manipulation. *ACM Trans. Graph. 31*, 4 (July), 116:1–116:8.

CHAI, M., WANG, L., WENG, Y., JIN, X., AND ZHOU, K. 2013. Dynamic hair manipulation in images and videos. *ACM Trans. Graph. 32*, 4 (July), 75:1–75:8.

CHAI, M., ZHENG, C., AND ZHOU, K. 2014. A reduced model for interactive hairs. *ACM Trans. Graph. 33*, 4 (July), 1–11.

CHAI, M., LUO, L., SUNKAVALLI, K., CARR, N., HADAP, S., AND ZHOU, K. 2015. High-quality hair modeling from a single portrait photo. *ACM Trans. Graph. 34*, 6, 204.

DECARLO, D., AND METAXAS, D. 2000. Optical flow constraints on deformable models with applications to face tracking. *Int. Journal of Computer Vision 38*, 2, 99–127.

DESBRUN, M., MEYER, M., SCHRODER, P., AND BARR, A. H. 1999. Implicit fairing of irregular meshes using diffusion and curvature flow. In *Proceedings of ACM SIGGRAPH*, 317–324.

ESSA, I., BASU, S., DARRELL, T., AND PENTLAND, A. 1996. Modeling, tracking and interactive animation of faces and heads: using input from video. In *Computer Animation*, 68–79.

GARRIDO, P., VALGAERTS, L., WU, C., AND THEOBALT, C. 2013. Reconstructing detailed dynamic face geometry from monocular video. *ACM Trans. Graph. 32*, 6, 158.

GARRIDO, P., ZOLLHOFER, M., CASAS, D., VALGAERTS, L., , VARANASI, K., PEREZ, P., AND THEOBALT, C. 2016. Reconstruction of personalized 3d face rigs from monocular video. *ACM Trans. Graph. to appear*.

HU, L., MA, C., LUO, L., AND LI, H. 2014. Robust hair capture using simulated examples. *ACM Trans. Graph. 33*, 4 (July), 126:1–126:10.

HU, L., MA, C., LUO, L., WEI, L.-Y., AND LI, H. 2014. Capturing braided hairstyles. *ACM Trans. Graph. 33*, 6 (Nov.), 225:1–225:9.

HU, L., MA, C., LUO, L., AND LI, H. 2015. Single-view hair modeling using a hairstyle database. *ACM Trans. Graph. 34*, 4 (July), 125:1–125:9.

HUANG, J., SHI, X., LIU, X., ZHOU, K., WEI, L.-Y., TENG, S.-H., BAO, H., GUO, B., AND SHUM, H.-Y. 2006. Subspace gradient domain mesh deformation. *ACM Trans. Graph. 25*, 3 (July), 1126–1134.

HUANG, H., CHAI, J., TONG, X., AND WU, H.-T. 2011. Leveraging motion capture and 3d scanning for high-fidelity facial performance acquisition. *ACM Trans. Graph. 30*, 4, 74:1–74:10.

ICHIM, A. E., BOUAZIZ, S., AND PAULY, M. 2015. Dynamic 3d avatar creation from hand-held video input. *ACM Trans. Graph. 34*, 4 (July), 45:1–45:14.

JANCOSEK, M., AND PAJDLA, T. 2011. Multi-view reconstruction preserving weakly-supported surfaces. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, 3121–3128.

JIMENEZ, J., SCULLY, T., BARBOSA, N., DONNER, C., ALVAREZ, X., VIEIRA, T., MATTS, P., ORVALHO, V., GUTIERREZ, D., AND WEYRICH, T. 2010. A practical appearance model for dynamic facial color. *ACM Trans. Graph. 29*, 6 (Dec.), 141:1–141:10.

JIMENEZ, J., ECHEVARRIA, J. I., OAT, C., AND GUTIERREZ, D. 2011. *GPU Pro 2*. AK Peters Ltd., ch. Practical and Realistic Facial Wrinkles Animation.

KAZHDAN, M., BOLITHO, M., AND HOPPE, H. 2006. Poisson surface reconstruction. In *Proceedings of the Fourth Eurographics Symposium on Geometry Processing*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, SGP '06, 61–70.

LEVIN, A., LISCHINSKI, D., AND WEISS, Y. 2008. A closed-form solution to natural image matting. *IEEE Transactions on Pattern Analysis and Machine Intelligence 30*, 2 (Feb), 228–242.

LI, Y., SUN, J., TANG, C.-K., AND SHUM, H.-Y. 2004. Lazy snapping. *ACM Trans. Graph. 23*, 3 (Aug.), 303–308.

LI, H., WEISE, T., AND PAULY, M. 2010. Example-based facial rigging. *ACM Trans. Graph. 29*, 4 (July), 32:1–32:6.

LI, H., YU, J., YE, Y., AND BREGLER, C. 2013. Realtime facial animation with on-the-fly correctives. *ACM Trans. Graph. 32*, 4 (July), 42:1–42:10.

LI, H., TRUTOIU, L., OLSZEWSKI, K., WEI, L., TRUTNA, T., HSIEH, P.-L., NICHOLLS, A., AND MA, C. 2015. Facial performance sensing head-mounted display. *ACM Trans. Graph. 34*, 4, 47.

LIU, Y., XU, F., CHAI, J., TONG, X., WANG, L., AND HUO, Q. 2015. Video-audio driven real-time facial animationy. *ACM Trans. Graph. 34*, 6, 182.

LUO, L., LI, H., AND RUSINKIEWICZ, S. 2013. Structure-aware hair capture. *ACM Trans. Graph. 32*, 4, 76.

MCMILLAN, L. 1997. *An Image-Based Approach to Three-Dimensional Computer Graphics*. PhD thesis, University of North Carolina at Chapel Hill.

NAGANO, K., FYFFE, G., ALEXANDER, O., BARBIÇ, J., LI, H., GHOSH, A., AND DEBEVEC, P. 2015. Skin microstructure

deformation with displacement map convolution. *ACM Trans. Graph. 34*, 4 (July), 109:1–109:10.

PARIS, S., CHANG, W., KOZHUSHNYAN, O. I., JAROSZ, W., MATUSIK, W., ZWICKER, M., AND DURAND, F. 2008. Hair photobooth: Geometric and photometric acquisition of real hairstyles. *ACM Trans. Graph. 27*, 3 (Aug.), 30:1–30:9.

PIGHIN, F., SZELISKI, R., AND SALESIN, D. 1999. Resynthesizing facial animation through 3d model-based tracking. In *Int. Conf. Computer Vision*, 143–150.

SARAGIH, J., LUCEY, S., AND COHN, J. 2011. Real-time avatar animation from a single image. In *AFGR*, 213–220.

SEITZ, S. M., AND DYER, C. R. 1996. View morphing. In *Proceedings of ACM SIGGRAPH*, ACM, New York, NY, USA, SIGGRAPH '96, 21–30.

SHI, F., WU, H.-T., TONG, X., AND CHAI, J. 2014. Automatic acquisition of high-fidelity facial performances using monocular videos. *ACM Trans. Graph. 33*, 6, 222.

SHUM, H.-Y., CHAN, S.-C., AND KANG, S. B. 2007. *Image-Based Rendering*. Springer.

STICH, T., LINZ, C., ALBUQUERQUE, G., AND MAGNOR, M. 2008. View and time interpolation in image space. In *Computer Graphics Forum*, vol. 27, Wiley Online Library, 1781–1787.

THIES, J., ZOLLHÖFER, M., NIESSNER, M., VALGAERTS, L., STAMMINGER, M., AND THEOBALT, C. 2015. Real-time expression transfer for facial reenactment. *ACM Trans. Graph. 34*, 6, 183.

VALGAERTS, L., WU, C., BRUHN, A., SEIDEL, H.-P., AND THEOBALT, C. 2012. Lightweight binocular facial performance capture under uncontrolled lighting. *ACM Trans. Graph. 31*, 6, 187.

VENKATARAMANA, K., LODHAA, S., AND RAGHAVAN, R. 2005. A kinematic-variational model for animating skin with wrinkles. *Computer & Graphics 29*, 5 (Oct), 756–770.

VLASIC, D., BRAND, M., PFISTER, H., AND POPOVIĆ, J. 2005. Face transfer with multilinear models. *ACM Trans. Graph. 24*, 3 (July), 426–433.

WEI, Y., OFEK, E., QUAN, L., AND SHUM, H.-Y. 2005. Modeling hair from multiple views. *ACM Trans. Graph. 24*, 3 (July), 816–820.

WEISE, T., LI, H., GOOL, L. V., AND PAULY, M. 2009. Face/off: Live facial puppetry. In *Symp. Computer Animation*, 7–16.

WEISE, T., BOUAZIZ, S., LI, H., AND PAULY, M. 2011. Realtime performance-based facial animation. *ACM Trans. Graph. 30*, 4 (July), 77:1–77:10.

XU, F., LIU, Y., STOLL, C., TOMPKIN, J., BHARAJ, G., DAI, Q., SEIDEL, H.-P., KAUTZ, J., AND THEOBALT, C. 2011. Video-based characters: Creating new human performances from a multi-view video database. In *ACM SIGGRAPH 2011 Papers*, SIGGRAPH '11, 32:1–32:10.

XU, Z., WU, H.-T., WANG, L., ZHENG, C., TONG, X., AND QI, Y. 2014. Dynamic hair capture using spacetime optimization. *ACM Trans. Graph. 33*, 6 (Nov.), 224:1–224:11.

YANG, F., SHECHTMAN, E., WANG, J., BOURDEV, L., AND METAXAS, D. 2012. Face morphing using 3d-aware appearance optimization. In *Proceedings of Graphics Interface 2012*, GI '12, 93–99.

ZANELLA, V., VARGAS, H., AND ROSAS, L. V. 2007. Active shape models and evolution strategies to automatic face morphing. In *Proceedings of the 8th International Conference on Adaptive and Natural Computing Algorithms, Part II*, Springer-Verlag, Berlin, Heidelberg, ICANNGA '07, 564–571.

ZHANG, L., SNAVELY, N., CURLESS, B., AND SEITZ, S. M. 2004. Spacetime faces: high resolution capture for modeling and animation. *ACM Trans. Graph. 23*, 3, 548–558.

Figure 13: *Our image-based avatars on a wide variety of users, with different hairstyles and headwear. From left to right: one captured image, reconstructed coarse geometry, and rendering results of the avatar with different poses and expressions.*