Data-driven Approaches for Personalized Head Reconstruction

Shu Liang

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2018

Reading Committee:

Linda G. Shapiro, Chair

Ira Kemelmacher-Shlizerman

Brian Curless

Program Authorized to Offer Degree:
Computer Science and Engineering

University of Washington

Abstract

Data-driven Approaches for Personalized Head Reconstruction

Shu Liang

Chair of the Supervisory Committee:
Professor Linda G. Shapiro
Paul G. Allen School of Computer Science & Engineering

Personalized 3D face reconstruction has produced exciting results over the past few years. However, traditional methods usually require complicated setups or controlled environments to get the detailed shape of a person's face. Most methods focus solely on the face area and mask out the hair due to the non-rigid nature and complicated layer structure of hairstyles. In this work, we explore data-driven approaches to reconstruct a person's 3D face or head including the hair from the devices that can be easily accessed by everyone.

The first part of our work introduces an algorithm that takes a single frame of a person's face from a commercial depth camera Kinect and produces a high-resolution 3D mesh of the input leveraging a large research dataset of 3D face meshes. We divide the input depth frame into semantically significant regions (eyes, nose, mouth, cheeks) and search the database for the best matching shape per region. We further combine the input depth frame with the matched database shapes into a single mesh that results in a high-resolution shape of the input person.

In order to free people from the capturing session, the larger portion of this thesis focuses on reconstructing not only the face, but also the rest of the head using in-the-wild image collections and videos. We first introduce a boundary-value growing algorithm to model a person's head from the person's large collection of photo data. We target reconstruction of the rough shape of the head. Our method is to gradually "grow" the head mesh starting from the frontal face and extending to the rest of the views using photometric stereo constraints.

Results on photos of celebrities downloaded from the Internet are given. However, in this algorithm, we have not reconstructed a complete head model and a specific model of the hair is lacked.

We further utilize a person's in-the-wild video to recover the full head model considering the multi-view information and hairstyle consistency across video frames. Given a video of a person's head, e.g., a TV interview, our method automatically reconstructs a 3D hair model leveraging a 3D hairstyle database. The resultant 3D hair model can be later deformed to change the hair shape, to make it brighter or darker. Our head reconstruction also includes facial modeling from the video, which is used to combine with the hair model. The method is completely automatic and requires as input only a single video taken "in the wild", found as is on the web or a selfie video taken by a smart phone. We demonstrate the capability of our method on a variety of celebrity videos and selfie videos, as well as comparing to the state of the art.

TABLE OF CONTENTS

Page

LIST OF FIGURES

LIST OF TABLES

GLOSSARY

CNN: a class of deep, feed-forward artificial neural networks to analysize vision imagery.

FCN: fully convonlutional network, a type of neural networks that is widely applied to semantic segmentation tasks.

VGG: a convolutional neural network for image classification tasks in computer vision.

RGB-D: a combination of RGB image and its corresponding depth image, in which each pixel is related to the distance between the camera and the object in the RGB image.

PS: photometric stereo, a technique in computer vision for estimating the surface normals of objects by obeserving that object under different lighting conditions.

GBR: generalized bas-relief, the ambiguity in determining the scale of the 3D structure recovered from unknown Lambertian surface.

PCA: principal component analysis, a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables for the purpose of feature extraction or data compression.

3DMM: 3D morphable model, an algorithm for 3D face reconstruction from a set of face shape or texture basis.

SIFT: scale-invariant feature transform, an algorithm in computer vision to detect and describe local features in images.

IOU: intersection over union, a metric for evaluating the accuracy of object detection or segmentation.

ACKNOWLEDGMENTS

The author wishes to express sincere appreciation to my advisors Linda G. Shapiro and Ira Kemelmacher-Shlizerman, the members in the GRAIL lab Ezgi, Yao, Supasorn and lots of other friends in the Computer Science Department of the University of Washington.

I am thankful to my collaborators Xiufeng Huang, Xianyu Meng, Kunyao Chen and Jason Lu from Owlii Inc. for their contribution to the hair reconstruction pipeline.

DEDICATION

to my parents

Chapter 1

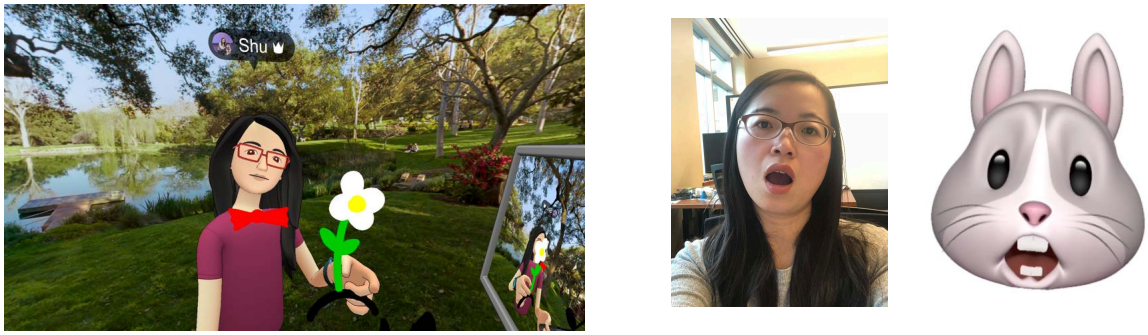
INTRODUCTION

Reconstructing a person's 3D face and hair is a very important topic with many applications in movies, video games and social VR/AR. Face-related products are popular among large companies and recent startups. For example, Facebook Story and Snapchat introduce lots of fun face masks; Oculus in Facebook creates Spaces application to allow people to socialize in VR; iPhone X comes with Animoji, which allows users to drive the emoji with their own faces in text messages as shown in Fig 1.1. However, most of the applications use cartoonish and generic 2D/3D avatars instead of personalized 3D heads.

To acquire a personalized 3D head of a subject is challenging due to the highly non-rigid nature of human faces and varying hairstyles. High-detailed face reconstruction methods currently require the subject to come to a lab equipped with a calibrated set of cameras and/or lights, e.g., multi-view stereo approaches [13, 14, 19], structured light [160], and light stages [4, 5, 56]. Traditional hair modeling algorithms are also developed in lab settings. Considering the complicated structures and specular reflections of hair, multi-view camera rigs, controlled lighting to reveal hair details and manual assistance to clean the hair volumes are usually necessary to get strand-level hair models with high fidelity [106, 66, 118, 117, 148]. However, we want to enable personalized 3D head reconstruction *anywhere* with easy inputs to allow more people to communicate in the digital world.

The development of commercial RGB-D cameras such as Kinect, extend the potiential of 3D shape capturing even in the comforts of one's home. KinectFusion [114] and DynamicFusion [115] works from Newcombe et al. allowed both the static and dynamic capture of a subject's face to be real-time. RGB-D cameras even made the capture of more complicated hairstyles such as braids possible [68].

With the development of movie, video games and healthcare industry, a lot of digital 3D shape databases become available to the public, providing high-quality shape priors for 3D



(a) Oculus Facebook Spaces VR Application

(b) Iphone X Animoji

Figure 1.1: Face-related application examples.

people. Paysan et al. [119] introduces the 3D Basel Face Model (BFM) including the high-resolution 3D face scans of 100 males and 100 females, which leads to the very traditional 3D face morphable model for face reconstruction from uncontrolled 2D photo inputs. The 3D Facial Norms Database [149] collects the 3D facial meshes of thousands of normal Caucasian individuals spanning a wide age range. Although hairstyles are hard to capture directly, a synthetic 3D hairstyle database of 343 hair models is collected from online game charactors for hair reconstruction, simulation and rendering purposes [67], which paves the way for serials of hair reconstruction work from in-the-wild photos [34, 161, 69].

In addition to the shape databases, large amounts of photo and video data are available on the Internet [104, 105]. For a celebrity, we can easily search for his/her photos in different poses and expressions on Google image search. YouTube videos of a celebrity's speechs or interviews are also available, containing the multi-view information for head reconstruction. For other people, photo collections and videos from Facebook albums or mobile storages could serve as a vast source of personalized information to produce a more realistic head model.

Our work focuses on freeing people from the complicated 3D head capturing session, better ultilizing the 3D shape databases and digital photos or videos available to for personalized 3D head reconstruction.

1.1 Thesis Outline

The rest of his thesis is organized as follows:

Chapter 2 summerizes the related work in face, hair and full head reconstruction and also reviews the work about hair segmentation and hair morphing for completeness. In Chapter 3, we utilize a RGB-D camera and demonstrates the algorithm to recover a high-quality face shape from a single depth view with the help of the 3D Facial Norms Database. We divide the face into 5 semantical regions and search the database for the best matching shape per region. The matched shapes are further combined with the input depth map to a high-resolution face shape of the input subject. In Chapter 4, we start to aim not only reconstructing the face, but also the rest of the head using in-the-wild 2D photo collections. We divide all the photos into 7 different pose clusters and gradually "grow" the head mesh from the frontal pose cluster to the side views to get a rough shape of the head. Chapter 5 tries to complete the unsolved problem in Chapter 4 to reconstruct a full head model with a personalized hairstyle. We use a person's in-the-wild video for the reconstruction, taking advantage of the multi-view information and hairstyle consistancy across video frames. We extract the hair directional information from all the views and retrieve for the best matching hairstyle from a synthetic hairstyle database. We further combine the hair model with the facial model fitted from the video. We provide qualitative, quantitative, and Mechanical Turk human studies that support the proposed system, and show results on a diverse variety of videos (8 different celebrity videos, 9 selfie mobile videos, spanning age, gender, hair length, type, and styling). Chapter 6 concludes the thesis.

Chapter 2

RELATED WORK

We describe related work in calibrated and uncalibrated settings for each of face, hair and head reconstruction. In addition, we also reviewed the relate work about hair segmentation and hair morphing for completeness.

2.1 Face Reconstruction

Calibrated face modeling has achieved great results over the last decade. Researchers were able to get a high-detailed head geometry with a stereo capturing system [13, 41, 4]. From their capture, they were able to create a photorealistic digital human character, which could be seen from any viewpoint, and could perform realistically. RGBD-based methods like DynamicFusion [115] and non-rigid reconstruction methods [138, 165] allowed capture to be real-time and much easier with an off-the-shelf device. However, all calibrated methods require a person to participate in a capturing session to achieve good results.

One way to solve the uncalibrated head reconstruction problem is to use the morphable model method introduced by Blanz and Vetter [17]. They proposed a principal component analysis basis to represent faces and used a linear combination of the bases to align and fit the input 2D image. Recently, CNN-based methods from Richardson et al. [123, 124] and Tran et al. [139] were also proposed to learn the face geometry directly from an input image, without relying on sparse facial features or accurate pose alignment.

Since a single 2D photo cannot fully reveal the true geometry of a person's face, Kemelmacher-Shlizerman et al. [84] raised the idea of using large photo collections of faces from the Internet by leveraging multi-image shading, but the shading-based method has the problem of scale ambiguity. Later, Roth et al. [126, 127] combined the shading information together with a prior 3D shape to achieve scale-correct and more robust results even with a smaller numbers of photos.

2.2 Hair Reconstruction

The hairstyle is an important characteristic and will dramatically change the appearance of a person. However, detailed hair modeling is highly complicated due to the high non-rigidity and variety of hairstyles. In the past decades, hair modeling also made great progress from reconstruction in a controlled lab setting to modeling from images in the wild.

With a well-controlled capturing environment, complicated hardware setups such as multi-view camera rigs and manual assistance, Paris et al. [118, 117], Luo et al. [106] and Hu et al. [66] were able to reconstruct strand-level hair models with high fidelity. More complicated hairstyles such as braids could even be reconstructed using a consumer RGB-D camera from Hu et al. [68]. Chai et al. [36, 35] explored detailed hair reconstruction from single-view images, but high-resolution inputs were required to recover the strands from per-pixel gradients. Later, Hu et al. [67] collected a database of synthetic hairstyles and proposed a data-driven approach to fit the hairstyle of a single photo to get a more natural-looking result. Although those methods produced high-quality results, human interactions such as hair segmentation and directional strokes are required. A fully automatic approach was proposed by Chai et al. [34] with CNN-based methods for hair segmentation, direction classification and a larger database for retrieving the best match for a single-view input. To utilize the information from more views, Vanakittistien et al. [142] used a hand-held phone camera to take photos from 8 views of the head to recover the hair strands. Zhang et al. [161] proposed a method to reconstruct the hair from four-view images starting from a rough shape retrieved from a database and synthesized hair textures to provide hair-growing directions to create detailed strands. However, their methods need human interactions for hair segmentation and pose alignment.

2.3 Full Head Reconstruction

Chai et al. [33] was the first to create a 2.5D portrait that combined face with hair using head shape priors and shading information from a frontal image. However, manual segmentation and directional strokes are required for portrait parsing. Utilizing deep learning models for portrait parsing, Hu et al. [69] later proposed an automatic framework to reconstruct a

full-head digital avatar ready for real-time animations with hair represented by polystrips from a single frontal image.

Cao et al. [30] extended the single-view portrait reconstruction work of Chai et al. [33] to reconstruct a full 3D head and showed that a complete head model with a rough but morphable hair model can be reconstructed from a set of captured images of one subject with hair depth estimated from each image and then fused together. With multi-view head images captured in an uncontrolled environment, Maninchedda et al. [109] used volumetric shape priors to reconstruct the geometry of a human head starting from structure-from-motion dense stereo matching. The shape volume was then semantically segmented into skin, hair, beard and eyebrow regions. However, both of these two previous methods created just a rough shape of the hair volume. Hair details were not recovered.

2.4 Hair Segmentation

Hair segmentation is an important part of face parsing and full head reconstruction. Yacoob et al. [157] detected hair based on the position relationship between face and hair and a simple color model. Wang et al. [146] proposed a coarse-to-fine hair segmentation method that starts from a coarse candidate region and performs graph-cuts to segment the hair. A CNN-based face parsing method from Luo et al. [107] hierarchically combined several detectors to detect face components. Liu et al. [100, 99] proposed multi-objective learning frameworks that could parse facial components as well as hair regions, but this model requires facial landmarks as prior inputs and can only handle simple hairstyles. To allow robust hair segmentation on various hairstyles, Chai et al. [34] trained a deep network specifically for the hair regions. However, their method requires pre-alignment of the face to detect the hair region. In recent years, fully convolutional networks (FCN) from Long et al. [102] have been widely used for pixel-level segmentation. We adopted the FCN model for robust hair segmentation and trained a network to segment hair regions across various poses.

2.5 Hair Morphing

While physical-based hair simulation, such as the mass-spring system [125], has been widely studied, hair morphing for achieving new looks of a subject was not the focus. Weng et al. [152] studied the problem of hair morphing to generate a set of intermediate hair models from one hairstyle to another from frontal hair reconstructions. In our work, we utilized our hair model for hairstyle morphing by interpolating by one-to-one strand correspondence.

Chapter 3

3D FACE HALLUCINATION FROM A SINGLE DEPTH MAP

3.1 Introduction

In this chapter, we demonstrate that a high quality face shape can be captured from a *single* depth view. The depth view is usually noisy and lacks details. So, we choose a single best database mesh per facial part, and then merge the individual parts, rather than assuming that the shape is spanned by a database. This enables high-detail shape reconstructions.

The key idea of this work is that while a single depth frame of a person's face is extremely noisy and low resolution, it still encodes metric information about the person's underlying facial features. Our approach is to leverage a large dataset of 3D face scans (1204 meshes of distinct Caucasian individuals, with age ranging from 3 to 40) for *hallucination* of a new 3D shape. We were inspired by the texture synthesis approach from Hays et al. [64] that leverage a large number of photos to fill in missing parts in a new photo. However, instead of working with photos, we propose an approach that finds similarities between a depth image and high-resolution 3D scans. Our task is similar to image super-resolution problems [116, 44]. While the related work is in shape matching approaches such as [122, 89], our goal is different. Rather than searching for corresponding semantic parts, we search for best matches for a particular part. Specifically, we match small parts from the depth frame to parts of the dataset faces, copy the matched parts from the corresponding dataset meshes and finally combine them together. This approach works remarkably well and can even reconstruct shapes of people who fall outside of the dataset span, such as, for people of older age and Asian ethnicity.

Our complete approach takes as input a single RGBD frame of a person's face and outputs a high-resolution 3D mesh of the input face. We are given a large dataset of high-resolution 3D face meshes (just the mesh, without texture), captured in a neutral expression. Examples of high-resolution meshes are shown in Fig. 3.2. All the meshes in the dataset

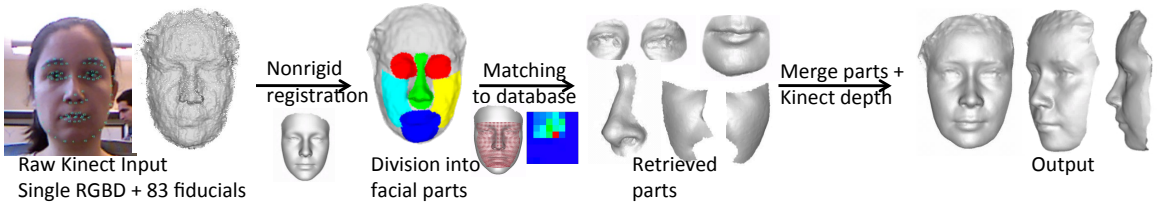


Figure 3.1: Our Pipeline.

have been put into dense correspondence using the deformable registration method [6]. Further, the aligned database meshes are averaged to produce the generic mesh G. Finally, we define five facial areas on G and, using the dense correspondence, propagate the areas to the database meshes.

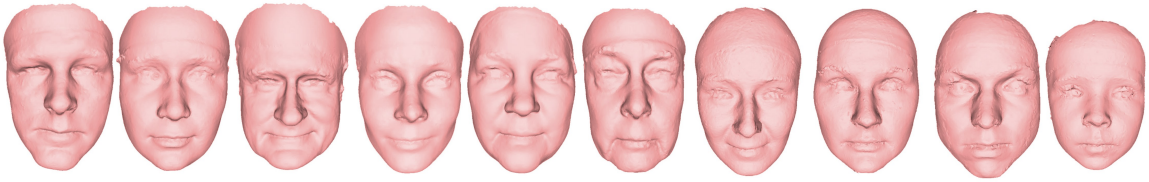


Figure 3.2: Example high-resolution face meshes. The database includes meshes (no texture) of 652 females and 552 males, ages 3 to 40, captured in a neutral expression.

Our approach is as follows. We first align the input RGBD frame to the generic mesh G. Then the input depth is divided into five facial parts via the alignment, and each facial part is matched independently to the dataset resulting in five high-resolution meshes. Finally, the matched meshes are combined with the input into a single mesh to produce the output. Fig. 3.1 illustrates all these steps. Below, we describe each of the steps in detail.

3.2 Aligning a single depth frame to the database

Given a single RGBD frame of a person's face in neutral facial expression, we first detect the face and 83 fiducial points. Any facial landmark detection method can be applied on

the RGB [48, 31] or depth image [49]. We use the software of Face++ [73]. Out of the 83 points, 19 are on the silhouette of the face, and the rest are on the internal part of the face. We use the internal facial points for rigid pose alignment via Procrustes analysis [58] and then all 83 points for dense alignment to the generic mesh G [6]. We obtain point-to-point correspondence between the depth frame and the generic shape, producing a deformed generic mesh G' which minimizes the difference to the depth frame. With the 83 points, all the faces in our data set are warped using [120] so that their global shapes are deformed to match the input depth image better. We define five facial parts on the input depth image based on the correspondence to the generic mesh. The five facial parts correspond to eyes, nose, mouth, left cheek, and right cheek as illustrated in Fig. 3.1.

3.3 Part-based matching to the database

The next step is to match each of the five facial parts in the input frame to the database. Prior to the matching process, we apply a curvature flow smoothing method [42] that preserves the low-frequency shape while smoothing out the noise.

Each of the five facial parts is then matched to the database using our distance function. The distance is a weighted combination of pseudo-landmarks and histograms of azimuth and elevation components of the surface normals, following Mercan et al. [110]and Atmosukarto et al. [8]. The distance function is described in detail in Sec. 3.6. The matching process results in five high-resolution meshes that are retrieved from the database. Each mesh matches to a different part of the input face.

3.4 Merging the matches

Once we get the five matches, the vertex normals are copied to replace the original normals of deformed generic shape G', part by part. Our query mesh can have hair while the high-resolution 3D head models do not. For each vertex V in the face region, using the nearest triangle $\triangle ABC$ in G', the normal vector of V can be interpolated as the weighted combination of the normal directions of $\triangle ABV$, $\triangle VBC$ and $\triangle VCA$. For the hair region, the original normals are kept. After we compute new normals for each vertex in the face region, we fuse the depth from the Kinect frame and the new normals together using the

method from Nehab et al.[113]. Then fine details on the facial part are transferred to the input face, but the hair style is kept.

3.5 Facial expressions

The above process produces a high-resolution mesh of the input face from a single noisy Kinect frame. While the focus of this work is on neutral faces, we further show that it is possible to produce high-resolution meshes of *facial expressions* using the same approach. It is challenging to acquire a database of high-resolution meshes of many distinct individuals making a large number of facial expressions. Instead, we show that given a single RGBD frame of a person in neutral expression and another frame that captures a facial expression, our approach can output a high-resolution expression mesh.

Specifically, we retrieve five matches from the database using the neutral input as described in 3.2 and 3.3, and then include the expression depth frame in the merging process. Each of the five database meshes are deformed towards the expression frame as in 3.2, and then we execute exactly the same merging process as in 3.4.

3.6 Similarity function

Our similarity function is used to match each of the five facial parts of the input depth frame to the corresponding parts of the database meshes. The similarity function is a weighted combination of *pseudo-landmarks* and *histograms of azimuth-elevation* components of the surface normals.

Pseudo-landmarks. To obtain pseudo-landmarks we sample the Kinect shape and each of the database meshes (which are at that stage in dense correspondence) following [110]. First, two anatomical landmarks (the sellion and chin tip), are computed and two base horizontal planes are computed through these points. Then, m parallel planes are computed between the two base planes, each sampled by n points. We chose $m = 33$ and $n = 35$ for a total of $1,225$ points, for mesh size of $19,033$ vertices. Additional details are described in the evaluation part below. Once pseudo-landmarks are estimated, the distance

per database mesh j is defined as

$$D^j_{\text{pts}} = \sum_{i=1}^{(m+2)n} ||P^j_i - P^{\text{input}}_i||^2 \tag{3.1}$$

where P^*_i is an xyz-coordinate of a pseudo-landmark.

Histograms of azimuth-elevation. We also compute distances between surface normals, as follows. Given the surface normal $\vec{n} = (n_x, n_y, n_z)$ at a point, the azimuth angle θ is defined as the angle between the positive x-axis and the projection of $\vec{n}$ to the xy plane. The elevation angle ϕ is the angle between the x-axis and $\vec{n}$:

$$\theta = \arctan(\frac{n_z}{n_x}), \phi = \arctan(\frac{n_y}{\sqrt{({n_x}^2 + {n_z}^2)}}) \tag{3.2}$$

with $\theta \in [-\pi, \pi]$, $\phi \in [-\frac{\pi}{2}, \frac{\pi}{2}]$. Histograms are useful to determine the "flatness" and the dominant orientation of a surface patch. We calculate a 32×32 histogram for each facial component, and define the distance as the χ^2-distance between the histograms

$$D_{\text{normals}} = \chi^2(H^j, H^{\text{input}}). \tag{3.3}$$

Combined distance. The combined distance for a single facial part is then defined as

$$D = D_{\text{pts}} + \alpha D_{\text{normals}} \tag{3.4}$$

The parameter α is chosen per facial part according to our evaluation experiment in Section 3.7.2. The cheek area typically has less variation in surface normals across points and thus has a small $\alpha = 1$; the mouth has higher normal variation and thus α will be larger ($\alpha = 10$). We chose $\alpha = 4$ for the eye area and $\alpha = 2$ for the nose area.

3.7 Experiments

Below we describe the details of our data, our implementation, and our results.

3.7.1 Implementation and data details

We used a Microsoft Kinect to capture the inputs in resolution 640×480; the face part of the frame was about 100×100. The database includes meshes of 1204 distinct Caucasian

individuals, ages 3-40 obtained by a 3dMD digital stereophotogrammetry system. The database does not include texture or color information due to privacy. Each mesh includes 15K-20K vertices. Subjects all face forward, have a neutral expression, and wear caps to remove hair occlusions. Meshes are cleaned by trained personnel and 15 anatomical facial landmarks were manually labeled by a single trained expert. Figure 3.2 shows examples of 3D meshes produced by the 3dMD system. The landmarks are used to register all the meshes to each other using deformable registration [6].

The experiments were run on an Intel Xeon 2.67GHz/2.66GHz CPU, 16GB RAM in Windows Server 2008 R2 64bit environment. For a typical result mesh of 15K vertices, the running time was 92.16s, with 1.2s for preprocessing (finding fiducial points, rigid alignment), 83.4s for non-rigid registration, 7.16s for retrieval (calculating features for the input, warping all the faces, finding the best matching parts), and 0.4s for merging. The non-rigid registration part (90% of the running time) could be replaced with a real-time registration method [165, 78].

3.7.2 Evaluation of similarity function

To evaluate our similarity measure we tested it with seven ground-truth meshes ($S1 - S7$). We included the ground-truth meshes in the database, and retrieved the best mesh per facial part. The inputs were Kinect depth images of the corresponding people. We compared pseudo-landmarks and azimuth-elevation histogram contributions at different resolutions as well as our final combined similarity distance. For each person, we obtained the ranking of the ground-truth in the retrieval results (lower is better). Note that the ground-truth meshes and Kinect inputs are not exactly the same, since the facial expression of the person may slightly change between the two captures. Tables 3.1, 3.2, 3.3, and 3.4 show the rankings for nose, cheeks, mouth and eyes areas respectively. Most of the cases show that increasing the resolution of pseudo-landmarks does not improve the retrieval result. As shown in Tables 3.1 and 3.2, the similarity function using the combined features worked extremely well on retrieving based on similarity of the nose and cheeks. For the nose, two individuals were returned as best matches, two others as second best, and another as third best (out of 1204

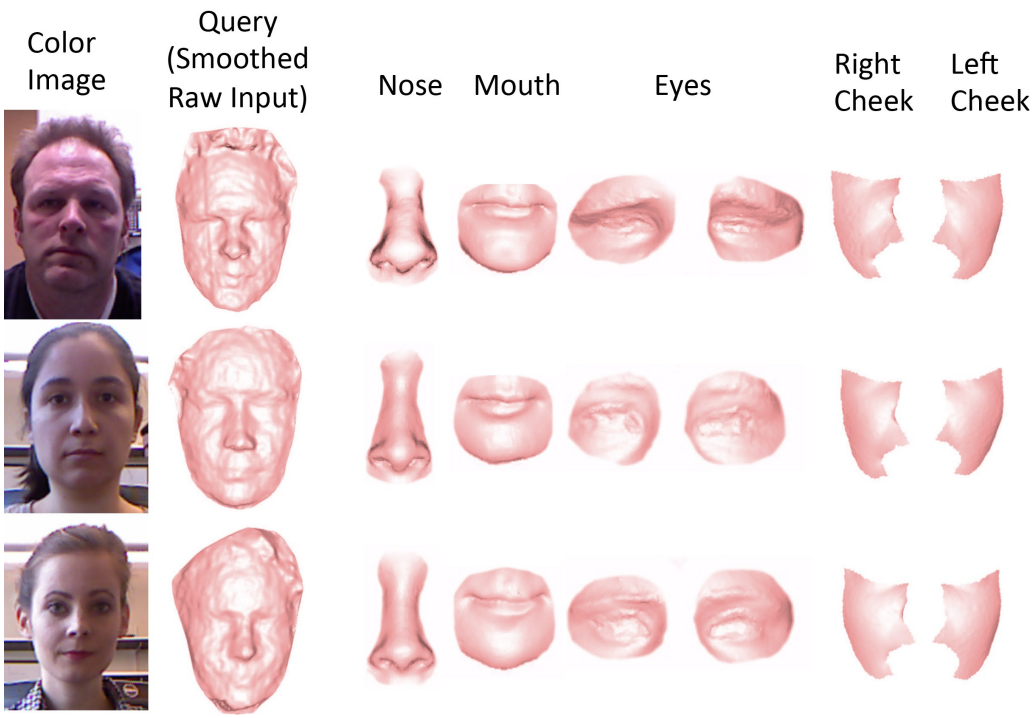


Figure 3.3: Similar parts that were retrieved using our approach. Photo shown only for reference.

+ 7 = 1211). For the cheeks, the similarity function with combined features returned the correct individuals with rankings of five through 68. The mouth region proved to be a little more difficult with the correct individuals achieving rankings from 1 to 229. The eyes were the most difficult with rankings from 12 to 482. We note that the eyes are the worst part of the Kinect depth frames, often not showing up well at all. Most of the obtained rankings were in the top 10% of the 1211 possible individuals in the expanded database. We show the five similar parts for input examples in Fig. 3.3. Note that while matching of 3D meshes is a widely studied research area [77, 15], there is no prior work on matching a noisy depth frame to high resolution meshes.

Table 3.1: Ranking from our distance function on the nose region.

Dist.	S1	S2	S3	S4	S5	S6	S7
Pts 35x35	157	2	809	1	14	1	58
Pts 65x65	157	2	813	1	14	1	38
A-E hist	24	7	1	33	99	238	9
Combined	**14**	**1**	**3**	**2**	**14**	**1**	**2**

3.8 Comparisons of reconstructions

We compared our reconstructions to reconstructions by KinectFusion [114](implementation by Kinect for Windows SDK v1.8 [111]) and to ground-truth shapes for people who were not part of the original database (since the people in the original database are unknown IRB-protected subjects). KinectFusion requires the subject to stay still and requires a few dozen Kinect frames, while our method requires a single frame. For each reconstruction we show the meshes and the error in surface normals (in angles). Fig. 3.4 shows the results on three meshes from our test set and includes the angle error for both KinectFusion and our result. In all tests, our result had a lower error than KinectFusion. We next implemented the face reconstruction technique by fitting the depth images to a 3D morphable model [164] and compared the results. Fig. 3.5 shows the morphable model results reconstructed from 200 principal components (more than 99% of the variances) of our face database. The fitted results are very dependent on the database and produce more generic results, while our results capture more individual details. We have also tested the contribution of using the database vs. just using the generic shape and non-rigid registration for the reconstruction and filling in the missing details in Kinect depth as shown in Fig. 3.6. Note that facial details are not captured with the generic model but appear once the database is used.

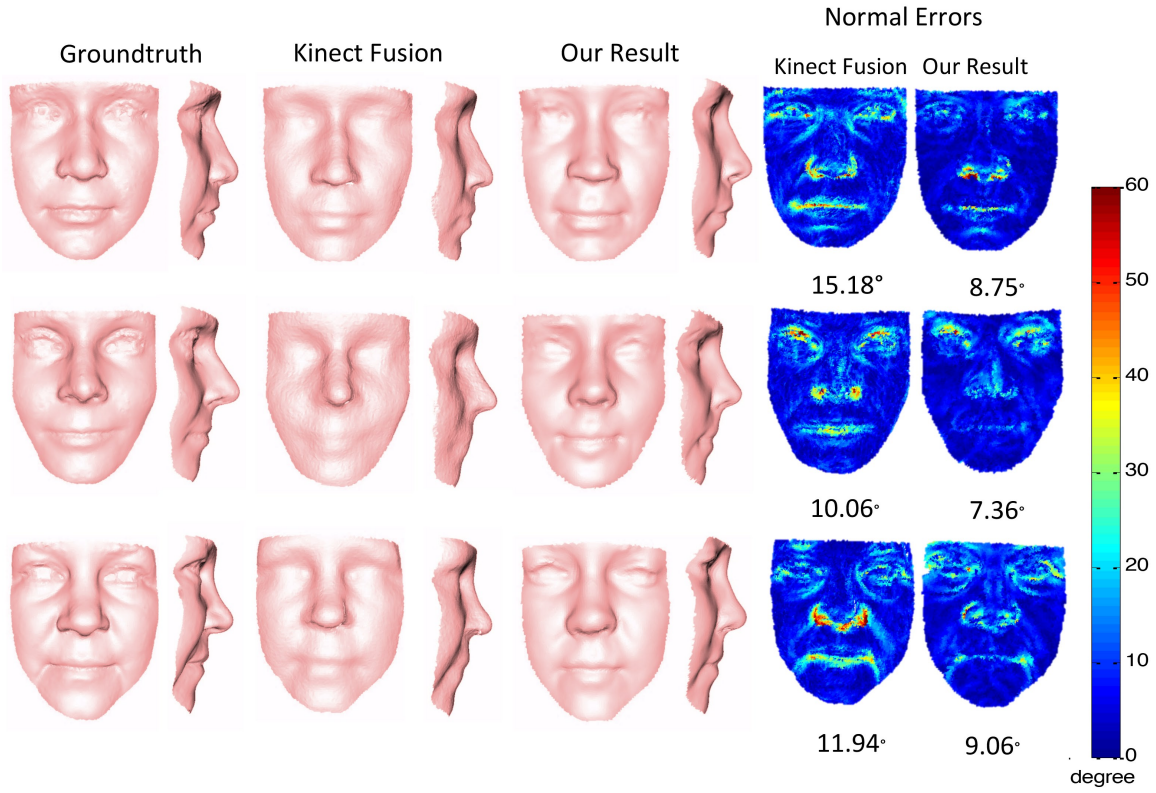


Figure 3.4: Comparison to ground-truth and KinectFusion [114].

Table 3.2: Ranking from our distance function on the cheek region.

Dist.	S1	S2	S3	S4	S5	S6	S7
Pts 35x35	17	64	88	**64**	49	3	89
Pts 65x65	17	76	83	70	47	3	83
A-E hist	229	98	47	314	334	11	38
Combined	**12**	**16**	**6**	68	**22**	**5**	**31**

Table 3.3: Ranking from our distance function on the mouth region.

Dist.	S1	S2	S3	S4	S5	S6	S7
Pts 35x35	229	408	441	73	22	619	342
Pts 65x65	227	382	478	90	22	581	276
A-E hist	27	108	1	119	17	95	262
Combined	**20**	**94**	**1**	**60**	**2**	**83**	**229**

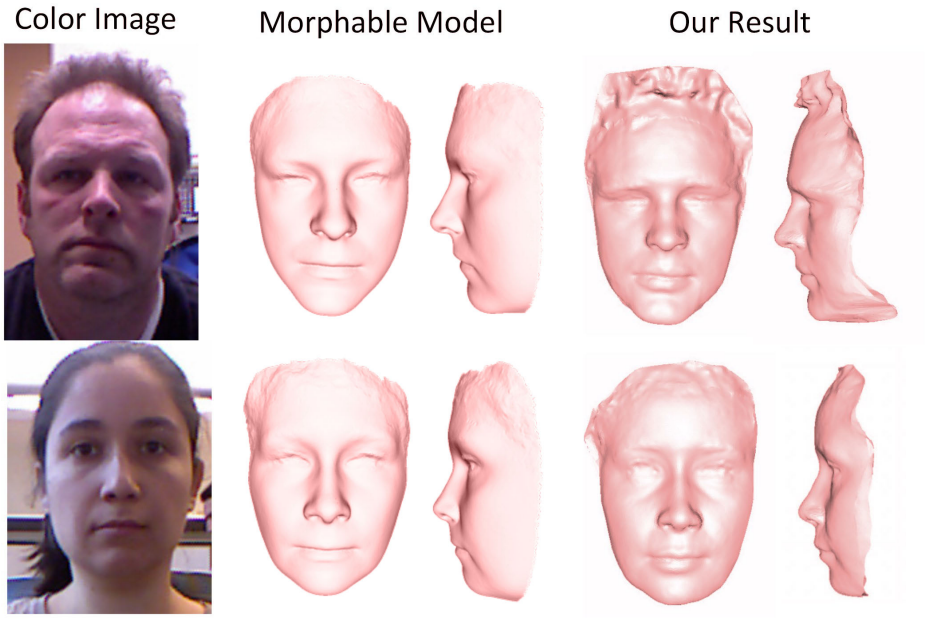


Figure 3.5: Comparison to reconstructions by fitting the detph to a morphable model [164].

Table 3.4: Ranking from our distance function on the eyes region.

Dist.	S1	S2	S3	S4	S5	S6	S7
Pts 35x35	92	57	543	43	102	351	475
Pts 65x65	90	67	544	56	103	395	429
A-E hist	184	617	484	713	334	11	231
Combined	**47**	**226**	**482**	**210**	**75**	**12**	**75**

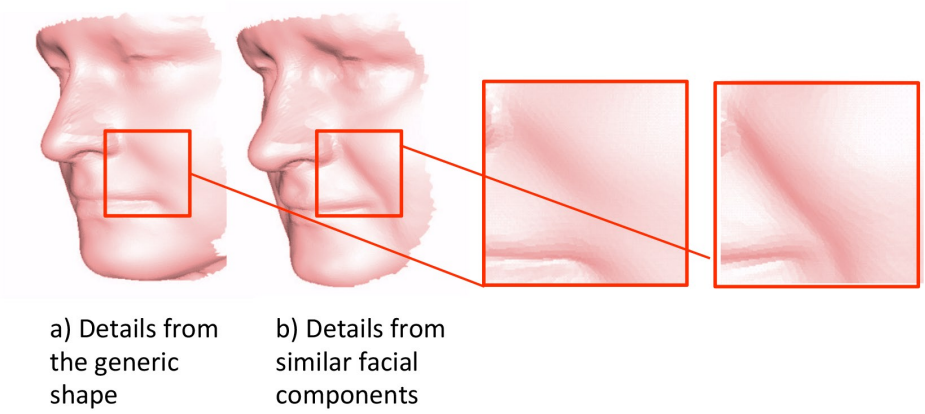


Figure 3.6: When a single generic shape (rather than the database) is used to fill in high-resolution details, individual details are not captured.

3.9 Additional results

Fig. 3.7 shows reconstructions of facial expressions from a single Kinect frame (given a neutral face frame). It is interesting to observe that the facial shape is reconstructed very well even though some of the people are not in the age span of the database or have a different ethnicity. The method is invariant to imaging conditions (light, pose) since the reconstruction is done based on depth-to-mesh matching and does not use the color channels.

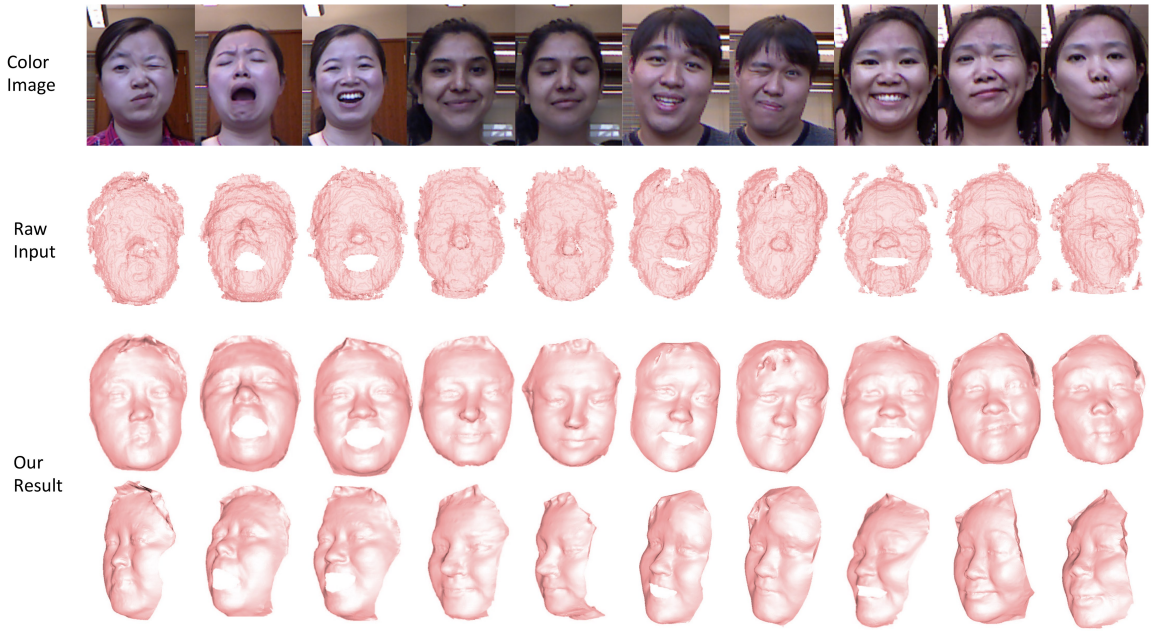


Figure 3.7: Reconstructions of facial expressions.

3.10 Conclusion

In this section, we described our approach for reconstruction of a high-quality 3D face mesh from a rough, noisy, low-resolution single Kinect depth frame. We leveraged a large dataset of high-resolution meshes of distinct individuals. Within that method, we have defined and tested a similarity measure that uses a linear combination of pseudo-landmark points and an azimuth-elevation angle histogram to retrieve parts of dataset faces that are most similar to the semantically equivalent parts of the query face. Our key contribution is to show that extremely simple part-based matching to a large set of faces enables the creation of remarkably accurate high-resolution meshes of novel people from noisy single-frame input. The resultant meshes can be further used for facial expression modeling, as we also demonstrated.

Chapter 4

HEAD RECONSTRUCTION FROM INTERNET PHOTOS

4.1 Introduction

In this chapter, we addresses the new direction of *head* reconstruction directly from Internet data. We propose an algorithm to create a rough head shape, and frame the problem as follows. Given a photo collection, obtained by searching for photos of a specific person on Google image search, we would like to reconstruct a 3D model of that person's head. Just like Kemelmacher-Shlizerman et al. [84] (that focused only on the face area), we aim to reconstruct an average rigid model of the person from the whole collection. This model can be then used as a template for dynamic reconstruction, e.g., [135] , and hair growing techniques, e.g., [67]. Availability of a template model is essential for those techniques.

Consider the top row photos in Fig. 4.1. The 3D shape of the head is clearly outlined in the different views (different 3D poses). However, if we are given only one or two photos per view, the problem is still very challenging due to lighting inconsistency across views, difficulty in segmenting the face profile from the background, and challenges in merging the images across views. Our key idea is that with many more (hundreds) of photos per 3D view, the challenges can be overcome. For celebrities, we can easily acquire such collections from the Internet; for others, we can extract such photos from Facebook or from mobile photos.

Our method works as follows: A person's photo collection is divided to clusters of approximately the same azimuth angle of the 3D pose. Given the clusters, a depth map of the frontal face is reconstructed, and the method gradually grows the reconstruction by estimating surface normals per view cluster and then constraining using boundary conditions coming from neighboring views. The final result is a head mesh of the person that combines all the views.

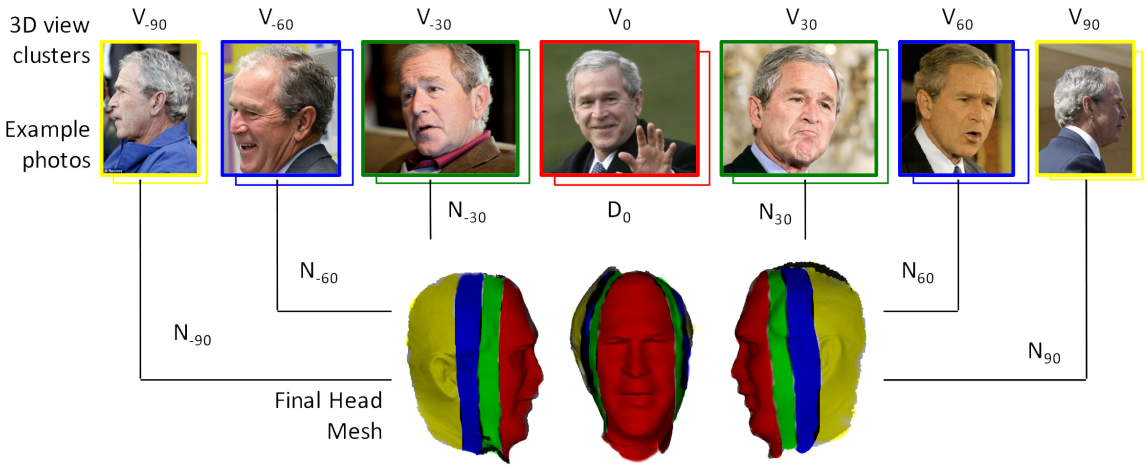


Figure 4.1: By looking at the top row photos we can imagine how Bush's head shape looks in 3D; however, existing methods fail to do so on Internet photos, due to such facts as inconsistency of lighting, challenging segmentation,and expression variation. Given many more photos per 3D view (hundreds), however, we show that a rough full head model can be reconstructed. The head mesh is divided into 7 parts, where each part is reconstructed from a different view cluster while being constrained by the neighboring view clusters.

4.2 Overview

We denote the set of photos in a view cluster as V_i. Photos in the same view cluster have approximately the same 3D pose and azimuth angle. Specifically, we divided the photos into 7 clusters with azimuths: $i = 0, -30, 30, -60, 60, -90, 90$. Figure 4.2 shows the averages of each cluster after rigid alignment using fiducial points (1st row) and after subsequent alignment using the Collection Flow method [85] (2nd row), which calculates optical flow for each cluster photo to the cluster average. A key observation is that each view cluster has one particularly well-reconstructed head area, e.g., the ears in views 90 and -90 are sharp while blurry in other views. Since our goal is to create a full head mesh, our algorithm will combine the optimal parts from each view into a single model. This is illustrated in Figure 4.1.

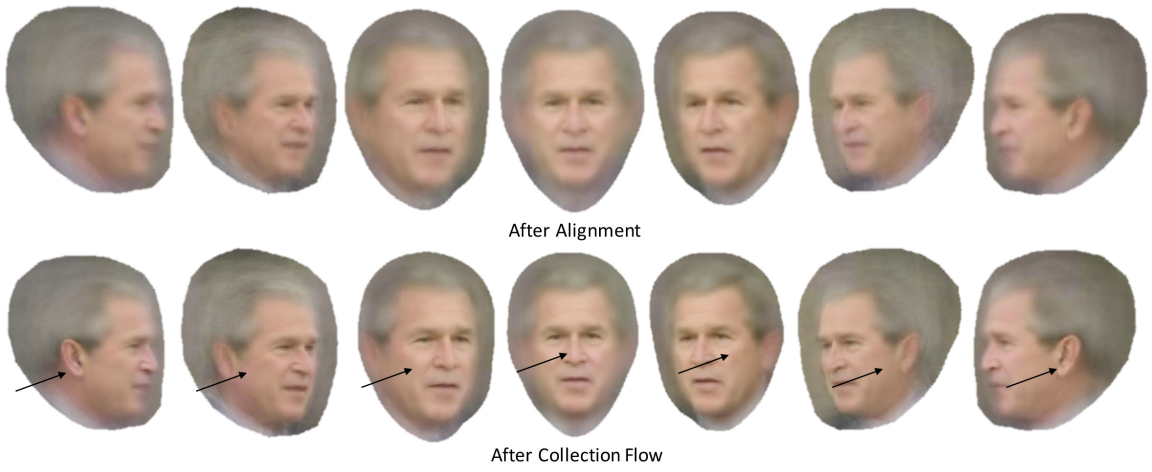


Figure 4.2: Averages of view clusters' photos after rigid alignment (1st row) and after collection flow (2nd row). The arrows visualize head parts that are sharper in each view, e.g., the ear is sharpest in 90 and -90 degrees (left and right). The key idea is to use the sharp (well-aligned) parts from the corresponding views to create an optimal mesh reconstruction.

It was shown in previous work that the face can be reconstructed from frontal photos using Photometric Stereo [84]. Thus, one way to implement our idea, of combining views into a single mesh, would be to reconstruct shape from each view cluster independently and then stitch them together. This turned out to be challenging as the individual shapes are reconstructed up to linear ambiguities. Although the photos are divided into pose clusters, the precise pose for each pose cluster is unknown. For example, V_{30} could have a variance from 25 to 35 in the azimuth rotation angle, depending on the dominant pose of the image cluster. This misalignment will also increase the difficulty of stitching all the views. We solve those challenges by growing the shape in stages works well. We begin by describing estimation of surface normals and a depth map for view cluster V_0 (frontal view) in section 4.3. This will be the initialization for our algorithm. In section 4.4, we describe how each view cluster uses its own photos and the depth of its neighbors to contribute to the creation of a full head mesh. Data acquisition and alignment details are given in the experiments section (Section 4.5).

4.3 Head Mesh Initialization

Our goal is to reconstruct the head mesh M. We begin by estimating a depth map and surface normals of the frontal cluster V_0, and assign each reconstructed pixel to a vertex of the mesh. The depth map is estimated by extending the method from Kemelmacher-Shlizerman et al. [84] to capture more of the head in the frontal face photos, i.e. , we extend the reconstruction mask to a bigger area to capture the chin, part of the neck and some of the hair. The algorithm is as follows:

1. **Dense 2D alignment:** Photos are first rigidly aligned using 2D fiducial points as the pipeline from Kemelmacher-Shlizerman [86]. The head region including neck and shoulder in each image is segmented using semantic segmentation by Zhang et al. [162]. Then Collection Flow [85] is run on all the photos in V_0 to densely align them to the average photo of that set. Note that the segementation works remarkably well on most photos. The challenging photos do not affect our method; given that the majority of the photos are segmented well, Collection Flow will correct for inconsistencies. Also, Collection Flow helps overcome differences in hair style by warping all the photos to the dominant style. See more details about alignment in Section 4.5.

2. **Surface normals estimation:** We used a template face mask to find the face region on all the photos. Photometric Stereo (PS) is then applied to the face region of the flow-aligned photos. The face region of the photos are arranged in an $n \times p_k$ matrix Q, where n is the number of images and p_k is the number of face pixels determined by the template facial mask. Rank-4 PCA is computed to factorize into lighting and normals: $Q = LN$. After we get the lighting estimation L for each photo, we can compute N for all p head pixels including ear, chin and hair regions.

   Two key components that made PS work on uncalibrated head photos are:

1) resolving the Generalized Bas-Relief (GBR) ambiguity using a template 3D face of a different individual, i.e., $\min_A ||N_{\text{template}} - AN_{face}||^2$,

2) using a per-pixel surface normal estimation, where each point uses a different subset of photos to estimate the normal. We follow the per-pixel surface estimation idea as in previous work, i.e., given the initial lighting estimate L, the normal is computed per point by selecting a subset of Q's rows that satisfy the re-projection constraint. In the full head case, we extend it to handle cases when the head is partially cropped out, by adding a constraint that a photo participates in normal estimation if it satisfies both the reprojection constraint and is inside the desired head area, i.e., part of the segmentation result from [162]. If the number of selected subset images is not enough (less than $n/3$), we will not use them in our depth map estimation step.

3. **Depth map estimation:** The surface normals are integrated to create a depth map D_0 by solving a linear system of equations that satisfy gradient constrains $dz/dx = -n_x/n_y$ and $dz/dy = -n_x/n_y$ where (n_x, n_y, n_z) are components of the surface normal of each point [10]. Combining these constraints, for the z-value on the depth map, we have:

$$n_z(z_{x+1,y} - z_{x,y}) = n_x \tag{4.1}$$

$$n_z(z_{x,y+1} - z_{x,y}) = n_y \tag{4.2}$$

In the case of $n_z \approx 0$, we use a different constraint,

$$n_y(z_{x,y} - z_{x+1,y}) = n_x(z_{x,y} - z_{x,y+1}) \tag{4.3}$$

This generate a sparse matrix of $2p \times 2p$ matrix M, and we can solve for:

$$\arg\min_z ||Mz - v||^2 \tag{4.4}$$

We do a least squares fit to solve for the z-value for each pixel.

Potentially, we could run the same algorithm for each view cluster. This, however, does not perform well, as we will see in the experiments section. Instead we are going to introduce two constraints, which we describe in the next section.

4.4 Boundary-Value Growing

In this section we describe our "growing" algorithm to complete the side views of the mesh. Starting from the frontal view mesh V_0, we gradually complete more regions of the head in the order of V_{30}, V_{60}, V_{90} and V_{-30}, V_{-60}, V_{-90}. For each view cluster we repeat the same algorithm as in Section 4.3 with two additional key constraints:

1. **Ambiguity recovery:** Rather than recovering the ambiguity A that arises from $Q = LA^{-1}AN$ using the template model, we use the already computed neighboring cluster, i.e., for $V_{\pm 30}$, N_0 is used, for $V_{\pm 60}$ we use $N_{\pm 30}$, and for $V_{\pm 90}$ we use $N_{\pm 60}$. Specifically, we estimate the out-of-plane pose from our 3D initial mesh V_0 to the average image of pose cluster V_{30} using the method proposed in [135]. We render the rotated mesh V_0' as a reference depth map D_0' to pose cluster V_{30}, accounting for visibility and occlusion using zbuffer. The normals on each projected pixels of D_0' will serve as the reference normals to solve for the GBR ambiguity of the overlapping head region as well as the newly grown head region.

2. **Depth constraint:** In addition to the gradient constraints that are specified in Sec. 4.3, we modify the boundary constraints from Neumann to Dirichlet. Let Ω_0 be the boundary of D_0'. Then we impose that the part of Ω_0 that intersects the mask of D_{30} will have the same depth values: $D_{30}(\Omega_0) = D_0'(\Omega_0)$. With both boundary constraints and gradient constraints, our optimization function can be written as:

$$\arg\min_z ||Mz - v||^2 + ||Wz - Wz_0||^2 \tag{4.5}$$

where z_0 is the depth constraint from D_0', and W is a blend mask with values decreasing from 1 to 0 on the boundary of D_0'. We will get the new vertex positions for grown

Table 4.1: Number of photos we used in each pose cluster

Pose	-90	-60	-30	0	30	60	90
Bush	185	62	118	371	113	80	191
Putin	131	58	151	413	121	61	151
Obama	65	51	126	284	177	55	75
Clinton	115	47	114	332	109	61	66

regions and we can also update vertices on the boundary of the already computed depth map, eliminating the distortion caused by lack of photos and inaccurate n_z. This process is repeated for every neighboring pair of depths.

After each depth stage reconstruction (0,30,60,.. degrees), the estimated depth is projected to the head mesh. By this process, the head is gradually filled in by gathering vertices from all the views.

4.5 Experiments

We describe the photo collection, alignment, evaluations and comparisons with other methods.

4.5.1 2D Photo Collections

We collected around $1,000$ photos per person (George Bush, Vladimir Putin, Barack Obama and Hillary Clinton) by searching for photos on Google image search. The numbers of images in each pose cluster are shown in Table 4.1. We noticed that the numbers of side view photos are usually much smaller than frontal view photos. In order to get more photos, we searched for "Bush shakes hands", "Bush shaking hand", "Bush portrait", "Bush meets" etc. to collect more non-frontal photos. The number of photos in each cluster will affect the final result; we will demonstrate the reconstruction quality vs. number of photos later in this section.

We ran face detection and fiducial detection using IntraFace [154]. For extreme side views, none of the state of the art fiducial detection algorithms was able to perform, and often times the face was not even detected. We therefore manually annotated each photo with 7 fiducials.

Once the photos are aligned, we run collection flow [85] on each view cluster. For completeness we review the method. The idea is to estimate a lighting subspace from all the photos in a particular cluster V_i via PCA. Then each photo in the cluster V_i^j is projected to the subspace to produce photo $\hat{V_i^j}$, which has a similar lighting as V_i^j but an average shape. Optical flow is then estimated between V_0^j and its relighted version $\hat{V_0^j}$. The process is iterated over the whole collection. In the end, all photos are warped to approximately average shape; however, they retain their lighting which makes them amenable for photometric stereo methods.

4.5.2 Results and Evaluation

Fig. 4.3 shows the reconstruction per view that was later combined to form a single mesh. For example, the ear in 90 and -90 views is reconstructed well, while the other views are not able to reconstruct the ear. In Figure 4.4, we shows how our two key constraints work well in the degree 90 view reconstruction result. Without the correct reference normals and depth constraint, the reconstructed shape is flat and the profile facial region is blurred, which increased the difficulty of aligning it back to the frontal view. Fig. 4.5 shows the reconstruction result for 4 subjects; each mesh is rotated to five different views. Note that the back and top part of the head are partly missing due to the lack of photos.

To evaluate how the number of photos affects the reconstruction quality, we took 600 photos for George Bush and estimated pose, lighting, texure for each image. We report the L2 intensity difference between the rendered photos and original photos. We tested our reconstruction method with 1/2, 1/4, 1/8 and 1/16 of the photos in each view cluster (see number of photos per cluster in Table 4.1.) The method did not work in 1/16 case because

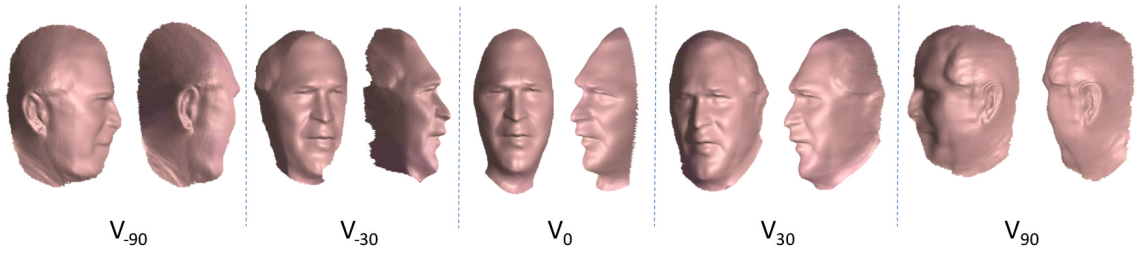


Figure 4.3: Individual reconstructions per view cluster, with depth and ambiguity constraints. We can see that the individual views provide different shape components. For each view we show the mesh in two poses.

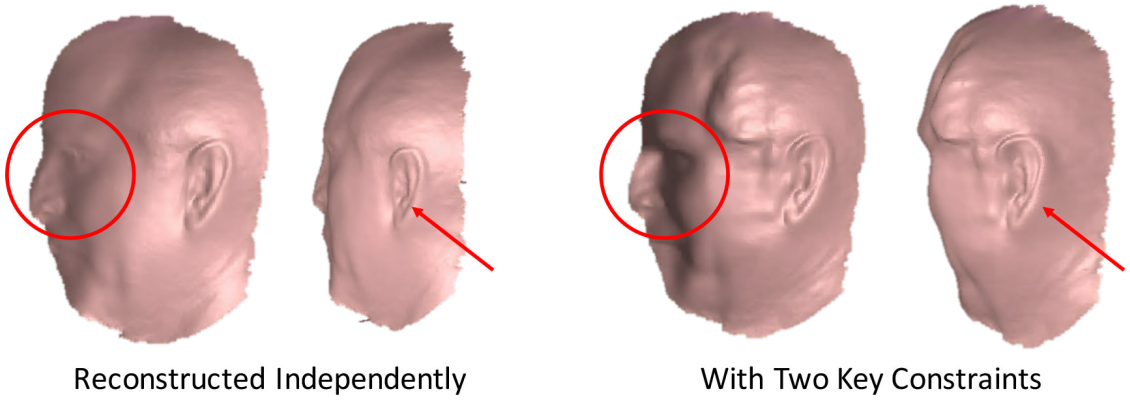


Figure 4.4: Comparison between without and with two key constraints. The left two shapes show the two views of 90 degree view shape reconstructed independently without two key constraints. The right two shapes show the two views of our result with two key constraints.

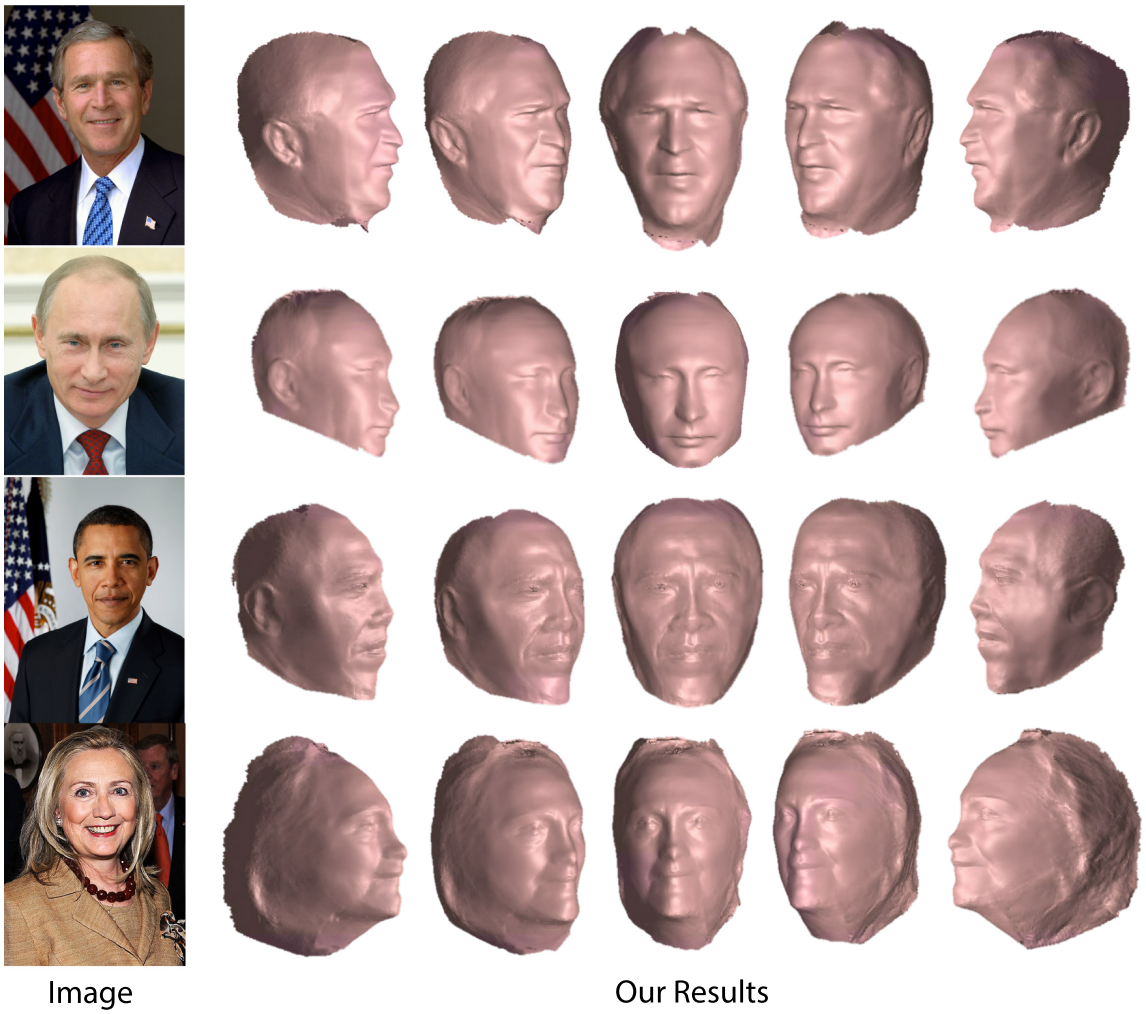


Figure 4.5: Final reconstructed mesh rotated to 5 views to show the reconstruction from all sides. Each color image is an example image among our around $1,000$ photo collection for each person.

Table 4.2: Reconstruction Quality vs. Number of Photos

Number of photos	N	N/2	N/4	N/8	N/16
Reprojection Error(intensity)	18.29 ± 4.07	18.70 ± 4.07	18.71 ± 4.07	18.80 ± 4.04	N/A

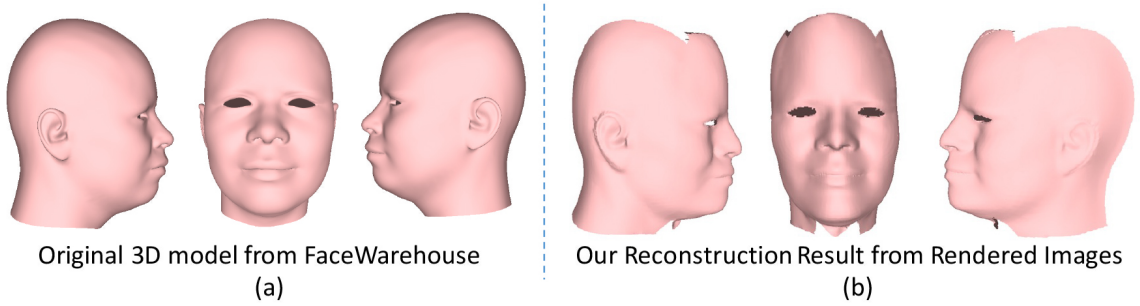


Figure 4.6: Reconstruction result from the synthetic photos rendered from a 3D model in FaceWarehouse. The left three shapes are the $-90, 0, 90$ views for the groundtruth shape, and the right three shapes are our reconstruction result.

some view clusters have less than 10 photos and there was not enough lighting variation within the collection for photometric stereo. Generally, we suggest using more than 100 photos for frontal view. The number of photos in side view clusters can be smaller (but larger than 30) because the side view of a human's head is more rigid than the frontal view.

We also rendered a 3D model from the FaceWareHouse dataset [29] with 100 lights and 7 poses. We applied our method on these synthetic photos and got a reconstruction result as shown in Fig 4.6. Since we use a template 3D model to correct GBR ambiguity, we cannot get the exact scale of the groundtruth. We do not claim that we have recovered the perfect shape, but the result looks reasonable with an average reprojection error of 11.1 ± 5.72.

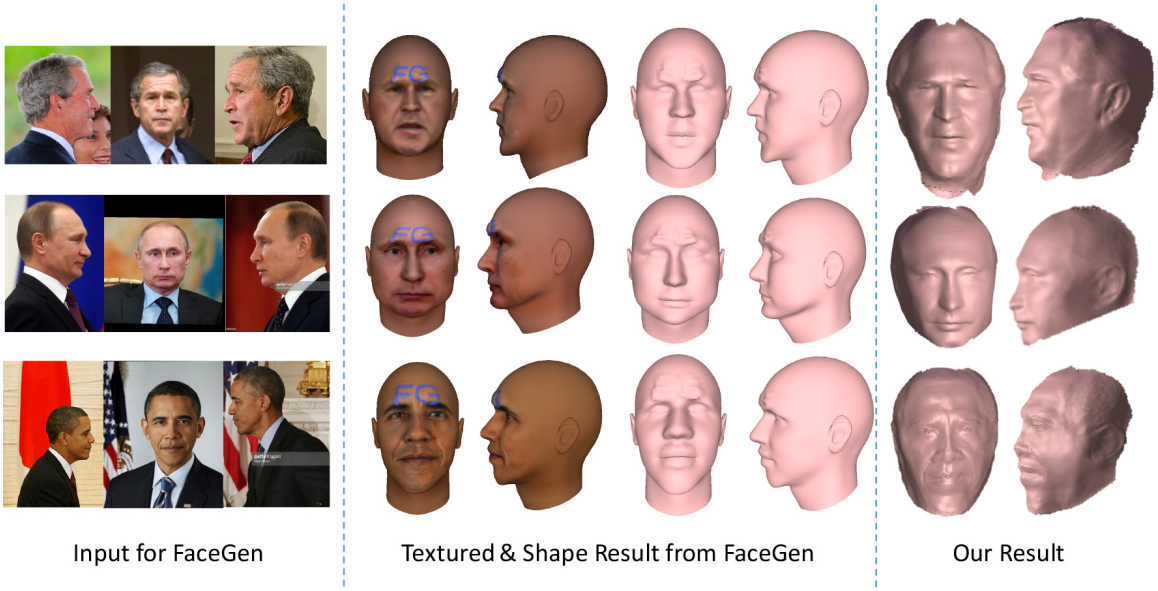


Figure 4.7: Comparison to FaceGen (morphable model). We show the textured results and shape results from FaceGen in the middle and our results are on the right as comparisons. Note that the head shape reconstructed by morphable models is average like and not personalized. Additionally, texture hides shape imperfections.

4.6 Comparison

In Figure 5.8 we show a comparison to the software FaceGen that implements a morphable model approach. For each person, we manually selected three photos (one frontal view and two side view photos) and used them as the input for FaceGen. The results of FaceGen are too averaged out and not personalized. Note that their ears look the same as one another.

We also tried the shape-from-silhouette method [50]. For each subject, we manually selected about 30 photos in different poses with a neutral expression. We used the segmentation result obtained from Section 4.3 as the silhouette. We assumed the camera focus length to be 100 and estimated the camera extrinisic parameters using a template 3D head model. We smoothed the visual-hull shapes using [42] and showed the reconstruction in Figure 4.8. The shape-from-silhouette method can produce a rough shape of the head.

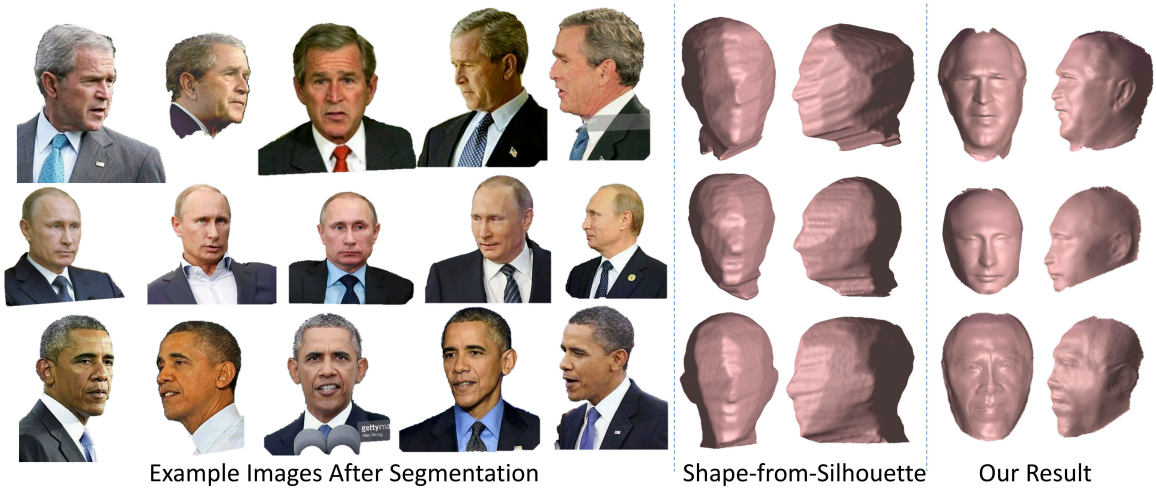


Figure 4.8: Comparison to shape-from-silhouette method. 5 example segmented images are shown on the left for each person. The segmentations were used as silhouettes. We used around 30 photos per person.

Increasing the number of photos to use does not improve the result.

We have also experimented with VisualSfM [153], but the software could not find enough feature points to run a structure from motion method. This is probably due to the lighting variation and expression change in the photo collection. Similarly, we have tried the software at http://www.123dapp.com/catch, and it was not able to reconstruct from such photos.

For a quantitative comparison, for each person, we calculated the reprojection error of the shapes from three methods (ours, Shape-from-Silhouette and FaceGen) to 600 photos in different poses and lighting variations. The 3D shape comes from each reconstruction method. The albedo all comes from average shapes of our clusters, since the Shape-from-Silhouette method and the FaceGen results do not include albedos. The average reprojection error is shown in Table 4.3. The error map of an example image is shown in Fig 4.9. We calculated the error for the overlapping pixels of the three rendered images. Notice that the shapes from FaceGen and Shape-from-Silhouette might look good from the frontal view, but they are not correct when rotating to the target view. See how different the ear part is

Table 4.3: Reprojection error from 3 reconstruction methods.

Reprojection error	FaceGen	Shape-from-Silhouette	Our method
Bush	20.6 ± 3.80	19.6 ± 3.55	18.3 ± 4.04
Putin	20.1 ± 4.84	17.2 ± 4.68	15.1 ± 5.06
Obama	21.5 ± 4.62	20.7 ± 4.58	19.7 ± 4.40

in the figure.

4.7 Conclusion

In this chapter, we have shown the first results of head reconstructions from Internet photos. Our method has a number of limitations. First, we have not reconstructed a complete model; the top of the head is missing. To solve this we would need to add photos with different elevation angles, rather than just focusing on the azimuth change. Second, we assume a Lambertian model for surface reflectance. While this works well for the skin region, this simple assumption does not work well for hair region. We need a specific model to reconstruct the hair. Third, fiducials for side views were labeled manually; we want to propose an automatic pipeline to reconstruct the full head model of a subject.

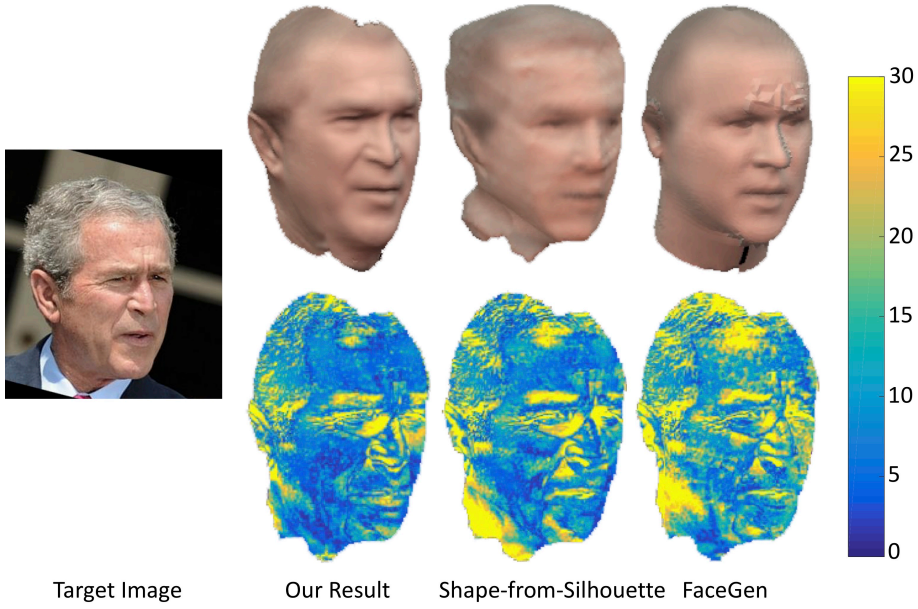


Figure 4.9: Visualization of the reprojection error for 3 methods.

Chapter 5

VIDEO TO FULLY AUTOMATIC 3D HEAD MODEL

5.1 Introduction

Any future virtual and augmented reality application that includes people must have a robust and fast system to capture a person's head (3D hair strands and face). The simplest capture scenario is taking a single selfie photo with a cell phone and create a digital avatar [69]. Autohair reconstruction from a single selfie [34], however, by definition, will not produce high fidelity results due to the ill-posedness of the problem–a single view does not show sides of the person. Using multiple frames, however, will create an accurate reconstruction. Indeed, a state-of-the-art method for hair modeling [161] needs four views, but it requires spanning the full 360° (front, back, and sides), as well as user interaction. Our method proposes to use a video as input and introduces solutions to three obstacles that prevent state-of-the-art work [161, 34, 69] from being applicable in simple automatic self capture:

1. Fixed views: Four-view reconstruction method [161] requires four views (front, back, and two side views), those are hard to acquire accurately with self capture. In this algorithm, we don't constrain the views, instead we use any available video frames in which the person talks or captures themselves; camera poses are estimated automatically with structure from motion. Results with various view ranges are demonstrated (as low as 90 degrees range). Autohair [34] and avatar digitalization method [69] assume a single frontal image.

2. Hair segmentation: Four-view method [161] relies on the user to label hair and face pixels. In this algorithm, we use automatic hair segmentation and don't require any user input. Using general video frames, rather than four fixed views, introduces motion blur, varying lighting, and resolution issues, all of which our system overcomes.

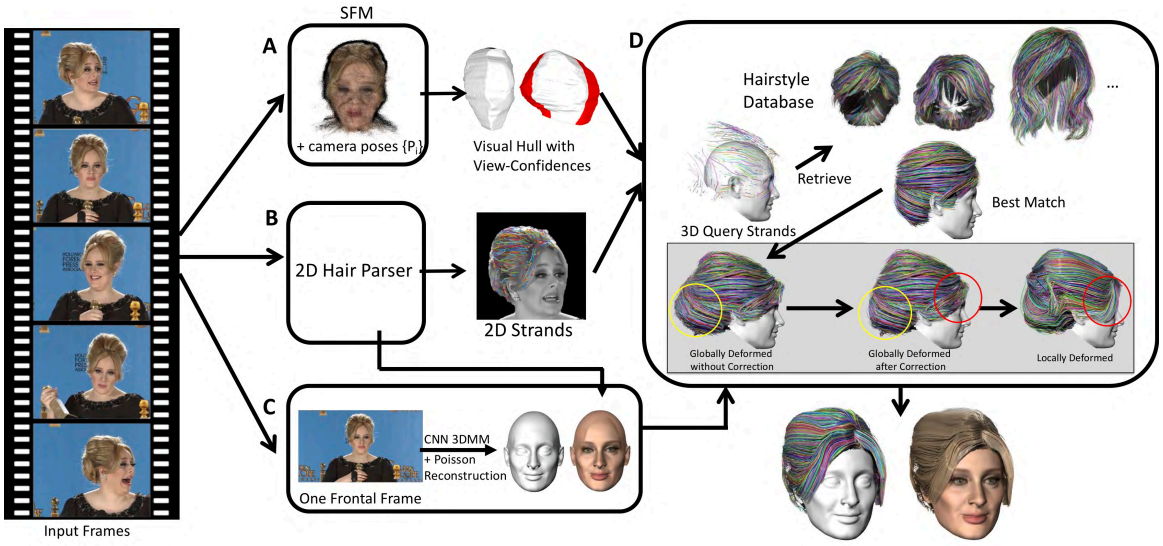


Figure 5.1: Overview of our method. The input to the algorithm is a video: (A) structure from motion is applied to the video to get camera poses, depth maps and a visual hull shape with view-confidence values, (B) hair segmentation and gradient direction networks are trained to apply on each frame and recover 2D strands, (C) the segmentations are used to recover the texture of the face area, and a 3D face morphable model is used to estimate face and bald head shapes. The core of the algorithm is (D) where the depth maps and 2D strands are used to create 3D strands, which are used to query a hair database; the strands of the best match are refined globally and locally to fit the input photos.

3. Accuracy: our method compares and deforms hair strands in 3D rather than 2D, and the availability of the back view is not required as in [161]. It achieves higher accuracy results as demonstrated with qualitative, quantitative, and human studies. Intersection of union rate of the hair region compared to ground truth photos is on average 80% for our method (compared to 60% by [161]). Amazon Turk raters prefer our results 72.4% over the four-view method [161] and 90.8% over the single-view method of [69].

In addition to [161] (and a previously multi-view based method by Vanakittistien et al. [142] that also required user interaction), there is a large body of work for modeling hair from photos. Earlier works assumed laboratory calibrated photos, e.g., Hu et al. [66]. More recently single-view hair reconstruction methods [35, 67, 69, 34] showed how to reconstruct hair from a single photo. Interesting hair-related applications inspire further research, e.g., depth-based portraits [34], effective avatars for games [69], photo-based hair morphing [152], and hair-try-outs [81]. Enabling reconstruction "in the wild" is an open problem where Internet photos have been explored in Chapter 4, as well as structure from motion [72] on a mobile video input. Both methods output a rough structure of the hair and head, without hair strands. This chapter proposes a system that can take in an in-the-wild video and automatically output a full head model with a 3D hair-strand model.

5.2 Overview

Figure 5.1 provides an overview of our method and the key components. The input to the algorithm is a video sequence of a person talking and moving naturally, as in a TV interview. There are four algorithmic components (correspond to the labeling of boxes in Figure 5.1):

(A) video frames are used to create a structure-from-motion model, estimate camera poses as well as per-frame depth, and compute a rough visual hull of the person with view-confidences, (B) two models are trained: one for hair segmentation, and another for hair direction; given those models, 2D hair strands are estimated and hair segmentation results are transferred to the visual hull to define the hair region, (C) the masks from the previous stage are used to separate the hair from the face and run the morphable model (3DMM) to

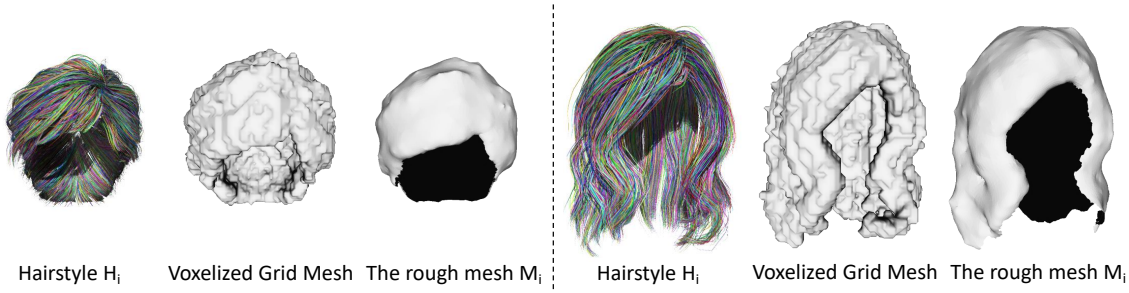


Figure 5.2: Example hair styles from the dataset. For each hairstyle H_i, we create its corresponding rough mesh M_i as described in the text.

estimate the face shape and later create the texture of the full head.

(D) is a key contribution in which first depth maps and 2D hair strands are combined to create 3D strands, and then 3D strands are used to query a database of hair styles. The match is deformed according to the visual hull, then corrected based on the region of confidence of the visual hull. Finally, it is deformed on the local strand level to fit the input strands. Texture is estimated from input frames to create the final hair model. The full head shape is a combination of the face model and the hair strands. In the next sections, we describe each of these components. We begin by explaining (D), since it is the main contribution, while the rest of the sections are improvements over previous work.

5.3 3D hair strand estimation

This section corresponds to part (D) in Figure 5.1, assuming parts (A-C) are done. We describe those parts later in the chapter. By utilizing a video, we deform the hairstyles in 3D instead of 2D because we are able to take advantage of the shape information of all the frames and its content continuity to estimate the per-frame pose.

2D to initial 3D strands: Each video frame i has an estimation of 2D strands; those are projected to depths D_i to estimate 3D strands. Large peaks of the 3D hair strands (distance to the neighboring vertex larger than 0.002 with a reference head width of 0.2) are removed. A merging procedure is performed to decrease future retrieval time (and reduce duplicate strands) as follows: for each pair of strands, if their directions are the same, the

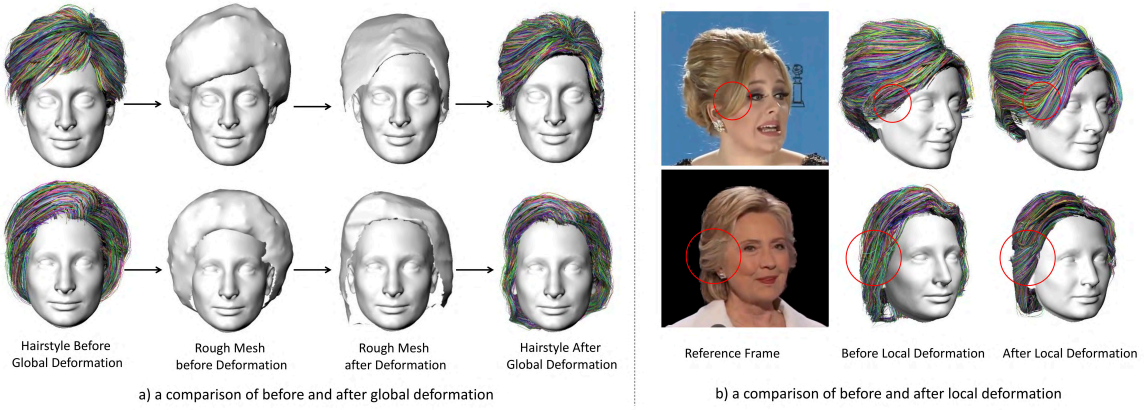


Figure 5.3: In (a), we show a comparison of before and after global deformation. The retrieved hairstyle is deformed under the control of its rough mesh to fit the visual hull shape. In (b), we show a comparison of before and after local deformation. A video frame is shown as a reference that after local deformation, we are able to recover more personalized hair details.

pairwise point-to-point distances of the vertices in these two 3D strands are checked, and the overlapping line segments are combined. If the directions are not the same, no merging occurs. This process iterates until around 100 3D strands are obtained.

3D Strands to Query a Hair Database: The recovered 3D strands in the previous stage are sparse and incomplete; thus we use them to query a database of hair models and adjust the retrieved matches with global and local deformations to create a full hair model. The sparseness is a result of resolution of the video frames, motion blur, quality, and coverage in views (in all of our input videos, the back of the head is not visible). While being sparse, the strands do capture the person's specific hairstyle. We describe the algorithm below.

We use the hair dataset created by Chai et al. [34], which contains $35,000$ different hairstyles, each hairstyle model consisting of more than $10,000$ hair strands. For each database hair model, we create a voxel grid around each hair strand vertex and combine all voxel grids into a voxelized mesh. The shape is further smoothed using Laplacian mesh

smoothing [133]. In order to remove the inner layer of the shape, a ray is shot from each vertex with the direction equivalent to the one from the center of the head to the current vertex. If the ray intersects any other part of the rough shape, the vertex is removed, because it is in the inner surface; otherwise it is kept. The resulting shape has $5,000$ to $7,000$ vertices. The final cleaned shape M (shown in Figure 5.2) will be used for retrieval and deformation.

For each 3D hair strand in our query hairstyle Q, the closest 3D hair strand from a hair style H is determined using the following distance:

$$Distance(Q,H) = \sum_{s_i \in Q} \sum_{p(s_i)} \min_{s_j \in H, \mathbf{n_{p(s_i)}} \cdot \mathbf{n_{p(s_j)}} > 0} |p(s_i) - p(s_j)|, \tag{5.1}$$

where s_i is a hair strand of Q and $p(s_i)$ is a vertex in strand s_i of H, $\mathbf{n_{p(s_i)}}$ is the tangent vector direction at $p(s_i)$.

This point-to-line distance comparison is very time-consuming. We performed experiments to accelerate the retrieval speed by pruning using a rough mesh X_h of the head obtained from step (A) (Section 5.4) and step (B) (Section 5.5) as follows:

1. **Hairstyle boundary:** Only the hairstyles with x-range in the range of $(0.8(maxX\{X_h\} - minX\{X_h\}), 1.2(maxX\{X_h\} - minX\{X_h\}))$ and y-range in the range of $(0.8(maxY\{X_h\} - minY\{X_h\}), 1.2(maxY\{X_h\} - minY\{X_h\}))$ are considered.

2. **Area of the hairstyle:** The surface area of the rough mesh X_h and of each hairstyle mesh M_i are computed. Only the hairstyles with surface area in the range of $(0.8S_{X_h}, 1.5S_{X_h})$ are considered.

Next, the retrieved matches are deformed in global and local fashion in 3D instead of 2D, taking advantage of the multi-view information in the video. Figure 5.3 illustrates the deformation process.

View Correction and Global Deformation After the top 20 best matching hairstyles are found, each retrieved hairstyle M_i is deformed towards the rough shape X_h (created by Step (A)(B) and shown in Figure 5.4(a)) using deformable registration [6] producing deformed hairstyle mesh M_i'. Furthermore, step (A) defines regions of *low-confidence* of

X_h (see Section 5.4), so further correction is needed on the corresponding regions of M_i'. The azimuth angle of each vertex is calculated on M_i', and the vertices that are outside the confident region are considered invalid as shown in Figure 5.4(c) marked as red. Naturally, we think of using the original shape of M_i to correct the invalid region. The correction is based on the idea of Laplacian Mesh Editing [133]. We denote the valid view range as $[R_1, R_2] \cup [R_3, R_4]$ for simplification. We assign a confidence value c_i to each vertex v_i' on M_i' and minimize the following energy function.

$$E(v_i') = \sum_{i=1}^{n}(1 - c_i)||\mathcal{L}(v_i') - \mathcal{L}(v_i)||^2 + \lambda \sum_{c_i=1} ||v_i' - x_i||^2 \tag{5.2}$$

where $\mathcal{L}$ is a Laplacian operator, v_i is the vertex position before deformation, x_i is the closest point of v_i' on X_h after direct deformable registration, λ is 10^{-5}. The confidence value c_i is 1 for the valid region and is defined for the invalid region as follows:

$$c_i = exp(-\frac{||\gamma(v_i') - R_j||^2}{2\sigma^2}) \begin{cases} j = 1, \gamma(v_i') \in (-\pi, R_1) \\ j = 2, \gamma(v_i') \in (R_2, 0) \\ j = 3, \gamma(v_i') \in [0, R_3) \\ j = 4, \gamma(v_i') \in (R_4, \pi] \end{cases}$$

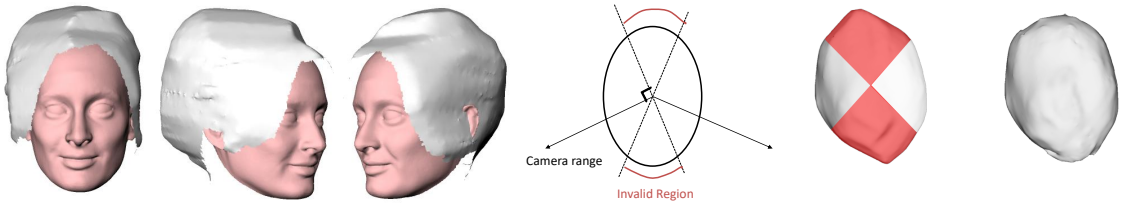
where $\sigma = \pi/18$. As shown in Figure 5.4(c), the stretched red region of M_i' is corrected to have a natural look in (d). After the correction, a transformation matrix T is obtained for each vertex on M_i.

We further deform the hair strands in H_i as shown in Figure 5.3(a). Each vertex v_i in M_i works as an anchor point for the hairstyle deformation. For each point p in H_i, its deformation will be decided by a set of neighboring anchor points as

$$T_p = \frac{\alpha I + \sum_{v_i \in N(p)} w_i T_i}{\alpha + \sum_{v_i \in N(p)} w_i}, \tag{5.3}$$

where $N(p)$ is the set of neighboring anchor points chosen to be the top 10 closest vertices of M_i. I is an identity matrix, and w_i is defined as a Gaussian function

$$w_i = exp(-\frac{||p - v_i||^2}{2\sigma^2}) \tag{5.4}$$



a) Hair Shape from Visual Hull on top of Head Mesh

b) Illustration of the Visual Hull with Incomplete Views

c) Before Correction

d) After Correction

Figure 5.4: In (a), we show the hair part shape X_h extracted from the visual hull (Figure 5.1(A)) plus the hair labels in Figure 5.1(B) on top of the head mesh from Figure 5.1(C). (b) shows an illustration of the top-down view of the visual hull with camera range and invalid regions. In (c)(d), we show a candidate hairstyle mesh M_i before and after correction.

In our experiments, we set α to 0.01 and σ to 0.015, while the width of our reference head is 0.2. We deform the top 20 best matching hairstyles, and use the same distance function as proposed in Equation 5.1 to find the final best match. We show a comparison in Figure 5.3 (a) before and after global deformation.

Local Deformation To add locally personalized hair details, we follow a method similar to Fu et al. [51] by converting the deformed hair strands into a 3D orientation field V. The orientation of the extracted hair strands from the video frames are also added to the 3D orientation field to bend the hair strands in the surface layer of the hairstyle. For each 3D query strand, we set an influence volume with a radius of 2 around it and diffuse it to fill in its surrounding voxels. The best matching hairstyle and the query 3D hair strands all contribute to the 3D orientation field by

$$E(\mathbf{v_i}) = \sum_{i \in V} ||\Delta(\mathbf{v_i})||^2 + w_1 \sum_{i \in C} ||\mathbf{v_i} - \mathbf{c_i}||^2 + w_2 \sum_{i \in Q} ||\mathbf{v_i} - \mathbf{q_i}||^2, \tag{5.5}$$

where Δ is the discrete Laplacian operator, C is the boundary constraints with the known directions $\mathbf{c_i}$ at certain voxel grids that contain 3D strands from the best-matching hairstyle, and Q is the boundary constraints from query hair strands.

In our experiment, we set the 3D orientation grid size to be $80 \times 80 \times 80$, w_1 to be 1 and w_2 to be 0.1. We show a comparison of results with and without the local deformation in

Figure 5.3(b). Notice the personalized hair strands in the red circles. We avoid the artifacts of hair going inside the head by growing new hair strands out of V from the scalp region of the complete head shape of Section 5.6 with pre-defined hair root points.

Hair Texture The color of the rough hair shape X_h is averaged from all the frames, and the unseen regions are assigned by an average color of the visible regions. The color of each hair strand vertex is determined by its closest anchor point on X_h.

5.4 Input Frames to Rough Head Shape

This section describes part (A) in Figure 5.1.

We begin by preprocessing each frame i using semantic segmentation [162] to roughly separate the person from the background resulting in masks S_i. Our goal is to estimate camera pose per frame and to create a rough initial structure from all the frames. Since the background is masked out, having the head moving while the camera is fixed is roughly equivalent to the head being fixed while the camera is moving; thus we use structure from motion [153] to estimate camera pose P_i per frame and per-frame depth D_i using multiview stereo method [57].

Given S_i and P_i per frame, we estimate an initial visual hull of the head using shape-from-silhouette [91]. The method takes a list of pairs P_i and S_i as input and carves a 3D voxel space to obtain a rough shape of the head. Meanwhile, each segmented video frame is processed using the IntraFace software [155], which provides 49 inner facial landmarks per frame. The 2D facial landmarks are transferred to 3D using D_i and averaged.

The hair segmentation classifier trained in step(B) (Section 6) is run on all of our video frames. Each pixel is assigned a probability of being in the hair region. We drop the video frames with large motion blurs by calculating the surface area S_i of the detected hair region on each frame. Assuming the head size is relatively fixed across frames, a valid frame should have a hair region size of at least $0.33\overline{S_i}$. The corresponding probabilities of the valid frames are transferred to the visual-hull shape. A vertex with a mean probability larger than 0.5 is considered hair. Thus, we extract the hair part X_h out of the visual-hull as shown in Figure 5.4(a), and the remaining is the skin part.

The resultant visual-hull shape is relatively rough due to the non-rigid nature of the

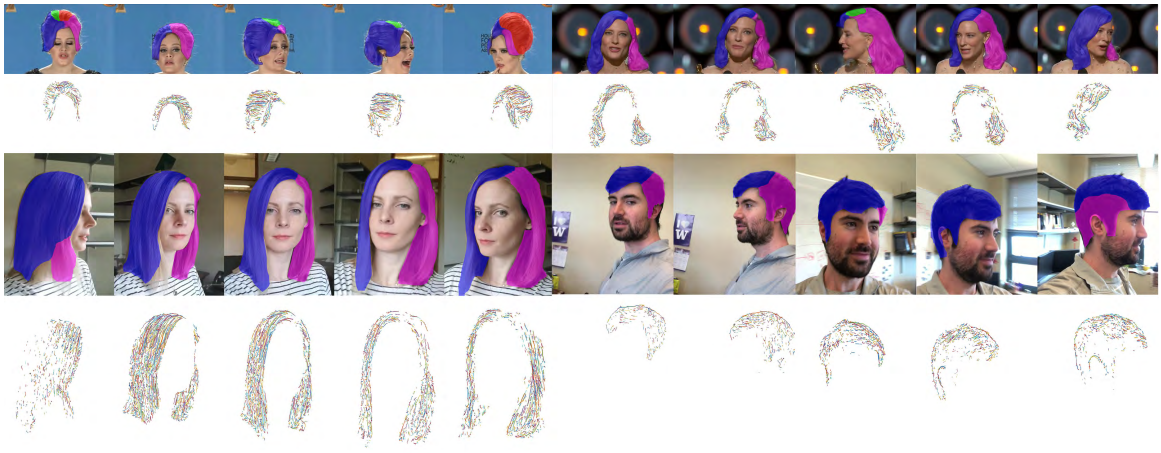


Figure 5.5: Examples of Figure 5.1(B). Hair segmentation, directional labels and 2D hair strands of example video frames. For the color of the directional subregions, red stands for $[0, 0.5\pi)$, pink stands for $[0.5\pi, \pi)$, blue stands for $[\pi, 1.5\pi)$ and green stands for $[1.5\pi, 2\pi)$.

subject's head and might be stretched due to the incomplete views. Ideally, assuming the camera distance is always larger than the size of the head, we will get a complete visual hull if our video covers a full azimuth range of -90 to 90 degree. However, for in-the-wild videos, we usually cannot guarantee full coverage. We rigidly align the rough visual hull to a generic head model using 3D facial landmarks, and each camera pose P_i is also transformed to a corresponding P_i' based on this alignment. We connect each P_i' to the center of the generic head (the origin point in our case) and calculate the azimuth angle γ_i of each camera. The vertices on the visual hull with an azimuth angle in $[min(\gamma_i) - \pi/2, max(\gamma_i) - \pi/2] \cup [min(\gamma_i) + \pi/2, max(\gamma_i) + \pi/2]$ as illustrated in Figure 5.4(a) are denoted *high-confidence* vertices.

5.5 Images to 2D Strands

This section describes part (B) in Figure 5.1. Inspired by the strand direction estimation method by Chai et al. [34], we trained our own hair segmentation and hair directional classifiers to label and predict the hair direction in hair pixels of each video frame. Unlike [34] that estimates a direction label per pixel, we chose to estimate a direction label per

hair region. This is since videos in the wild turn out to have lower resolution heads, as well as motion blur, thus labeling per pixel produces many ambiguous pixel labels with low-confidence scores. Additionally, it allows easier creation of continuous 2D and then 3D strands, which is needed to compare to a hair database in 3D space. More details on our hair segmentation method can be found in Appendix A. Results of the classifier are shown on examples in Figure 5.5 (1st and 3rd row).

To estimate 2D strands, we select one video frame every $\pi/8$ degrees according to its camera azimuth angle γ_i spanning the camera view range $[min(A_i), max(A_i)]$. Similar to previous hair orientation estimation methods [75, 36], we filter each image with a bank of oriented filters that are uniformly sampled in $[0, \pi)$. We choose the orientation θ_p with the maximum response for each pixel to get the 2D non-directional orientation map for each image. We further trace the hair strands on each non-directional orientation map following the method proposed by Chai et al. [36] as shown in Figure 5.5 (2nd and 4th rows). The hair-direction labels are used to resolve the ambiguity of each traced 2D hair strand. If half of the points in a single strand have opposite directions to their directional labels, we flip the direction of the strand.

5.6 Face Model

This section corresponds to part (C) in Figure 5.1. Each segmented video frame from the previous stage is processed using the IntraFace software [155], which provides head pose, and 49 inner facial landmarks. From all the frames, the frame that is closest to frontal face is picked first (where yaw and pitch are approximately 0), and fed to a morphable-model-based face model estimator 3DMM CNN [139]. This method generates a linear combination of the Basel 3D face dataset [17] with both identity shape, expression weights and texture. Here, we only use the identity weights to generate a neutral face shape. In the future, it will be easy to add facial expressions to our method. The result of the estimation is a masked face model.

We complete the head shape using a generic 3D head model from the Facewarehouse dataset [28]. We choose to use the Basel dataset instead of using the Facewarehouse dataset to fit directly, because the Facewarehouse dataset contains only about $11k$ vertices for the

whole head, while the Basel dataset contains about $53k$ for just the face region in which more facial shape details are provided. We pre-define 66 3D facial landmarks (49 landmarks on the inner face and 17 landmarks on the face contour and the ears) on both the 3D face dataset used by 3DMM CNN [139] and the generic head shape. Since all the face shapes in the 3D face dataset are in dense correspondence, we transfer these 66 landmarks to all the output 3D faces. We then deform the generic shape towards the 3D face shape using the landmarks, following [95]. We fuse the deformed generic head shape and the face shape using Poisson surface reconstruction [79] and get a complete head shape.

For the texture of the head, we project the full head to the selected frontal image and extract per-vertex colors from the non-hair region of the frontal image. We complete the side-view textures by projecting the head to all the frames. For the remaining invisible region, we assign an average skin color.

5.7 Experiments

In this section we describe the parameters used and the data collection process; we show results as well as comparisons to state-of-the-art methods.

5.7.1 Data Collection and Processing of Video Sequences

We collected 8 video clips of celebrities by searching for key words like "Hillary speech", "Adele award" on YouTube with an HD filter. The typical resolution of our videos is 720p (1280×720) with video duration around 40 seconds sampled at 10 fps (380 frames for Adele, 340 and 240 frames for Cate Blanchett, 250 and 350 frames for Hillary Clinton, 500 frames for Justin Trudeau, 390 frames for Theresa May, 310 frames for Angela Merkel). The camera view point is relatively fixed across all the frames, while the subject is making a speech with his/her head turning. We processed our frames at 10fps. We ran the face detection method [155] on all the frames to determine a bounding box around the head (box height varies from 200 to 600). Our online video sequences typically cover the frontal, left and right view of the person. The minimum view range we have is for Angela Merkel: only -15 to 75 degrees. There are no back views of any person's head in the videos.

For mobile selfie videos, 9 subjects were asked to take a selfie video of themselves from left to right and switch hands in the front using their own smart phones (video resolution varies from 720p to 1080p). The subjects were not required to stay rigid and could take the video at their ease. The videos were taken in arbitrary environments and the lightings were not controlled. Note that the quality of mobile selfie videos are usually worse than the online videos due to large motion blurs caused by hand moving, auto focus, and auto exposure from phone cameras, although a higher frame resolution is accessible. The selfie video is approximately 15 seconds sampled at 20fps (369 frames, 277 frames, 229 frames, 300 frames, 267 frames, 376 frames, 383 frames, 235 frames and 256 frames for each individual from top to bottom of Figure 5.6).

Later, the semantic segmentation method [162] was run on video frames to remove the background and foreground occlusions such as microphones. We ran VisualSFM [153] on the pre-processed frames. In VisualSFM[153], the non-rigid face expression change might cause large distortions in the reconstructed views; thus we set radial distortion to zero.

Runtime We ran our algorithm on a single PC with a 12 core i7 CPU, 16GB of memory and four NVIDIA GTX 1080 Ti graphics cards. For a typical online video, the preprocessing and structure from motion plus visual hull in Figure 5.1(A) takes 40 minutes. Extracting 2D query strands in Figure 5.1(B) takes 3 minutes. The head shape reconstruction and texture extraction takes 3 minutes to run. Hair database retrieval and deformation in Figure 5.1(D) is 40 minutes with 500 candidates. The 3D orientation local deformation and final hair strand generation from the reconstructed head takes 2 min.

5.7.2 Results and Comparisons

Figure 5.6 and Figure 5.7 show the results together with the reference frames from the videos. We can see that the reconstructions are good for a variety of lighting conditions, diverse people and hairstyles.

Next, we compared our results to the state-of-the-art hair in-the-wild reconstruction methods [34, 161, 69]. [1] We performed qualitative, quantitative and user study comparisons

[1] We thank the authors of those papers for helping creating comparison results.

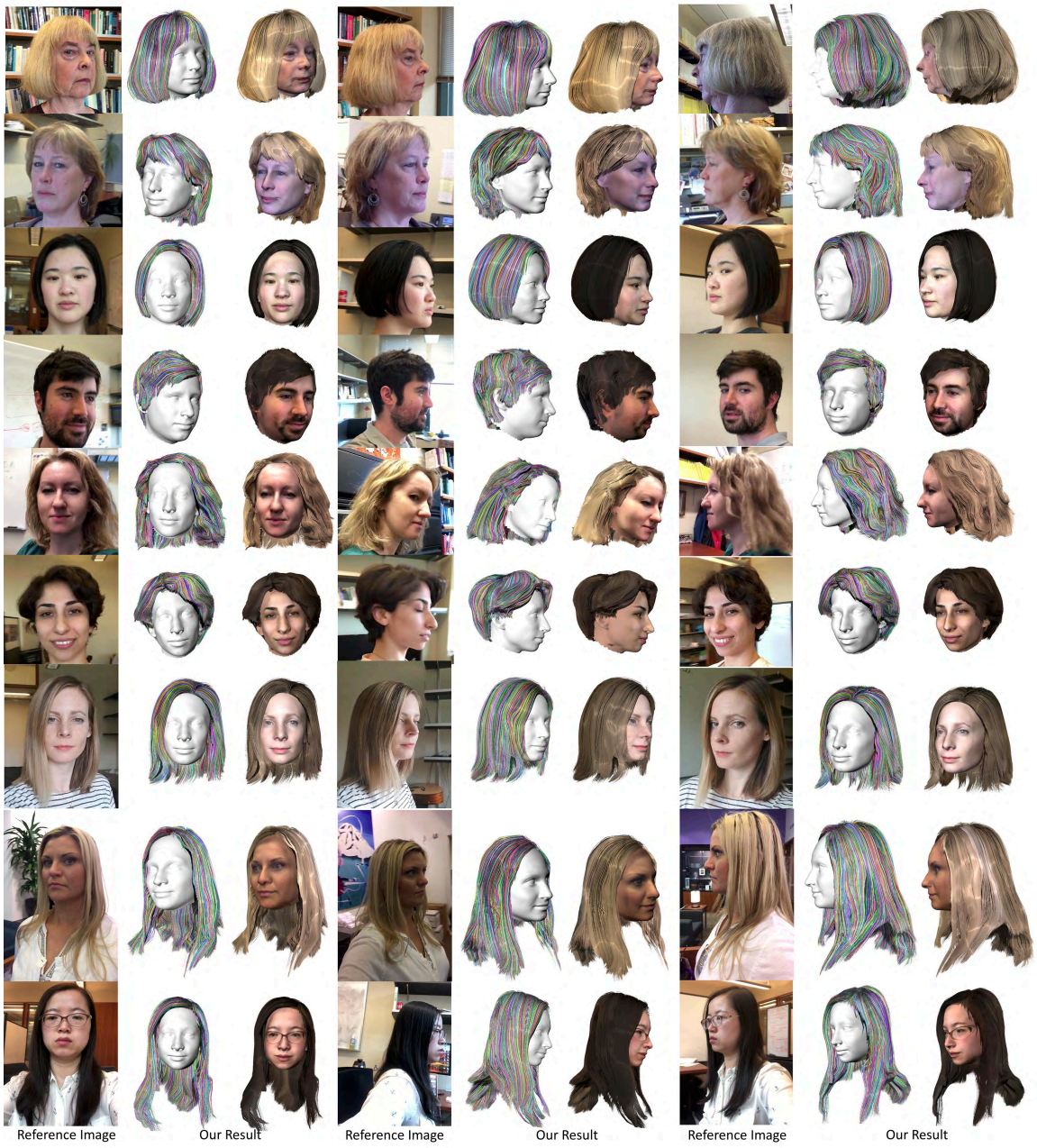


Figure 5.6: Reconstruction results from mobile selfie videos of different people in different environments.

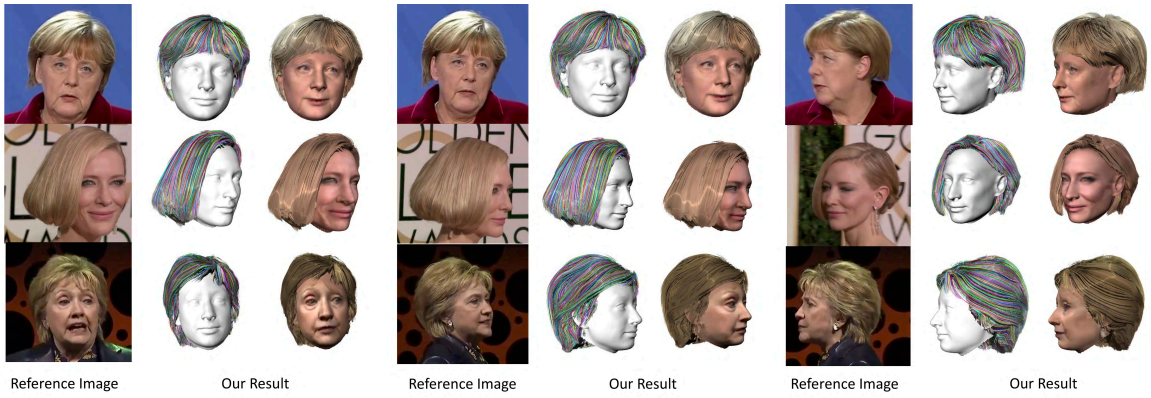


Figure 5.7: Example results of our method. From top to bottom, the view coverage for Angela Merkel's video is 15 degree to −75 degree, 67 to −75 degree for Cate Blanchett and 60 to −74 degree for Hillary Clinton. Note that we can even create a natural looking result for Angela Merkel with a small view coverage.

below.

Figure 5.8 shows comparisons for single-view methods. We picked a frontal frame from each of the 5 video clips of celebrities as input. We compared our untextured results with autohair [34] and textured results with avatar digitalization work [69]. Note that our 3D models captured more personalized hairstyles; for example in Adele's case (the 1st row), autohair [34] produced a short hairstyle, while Adele has a long hairstyle. Compared to digital avatars [69], where each hairstyle has a flat back, our results show more variety.

In four-view method [161], frontal, left, and right views are manually chosen from the same video clip. Since we do not have the back view in our video frames and the back view is necessary for the four-view reconstruction method, the authors were allowed to use any back view image they could find to reconstruct (the authors did not reconstruct Adele). In our algorithm, we did not use the back view photo of the person. Our results are similar to [161]; however ours are closer to the input; this can be seen by looking at the result of Justin (the 4th row) produced by four-view method [161] which has a larger volume.

We did a **quantitative comparison** by projecting the reconstructed hair as lines onto the images, computing the intersection-over-union rate to the ground truth hair mask (man-

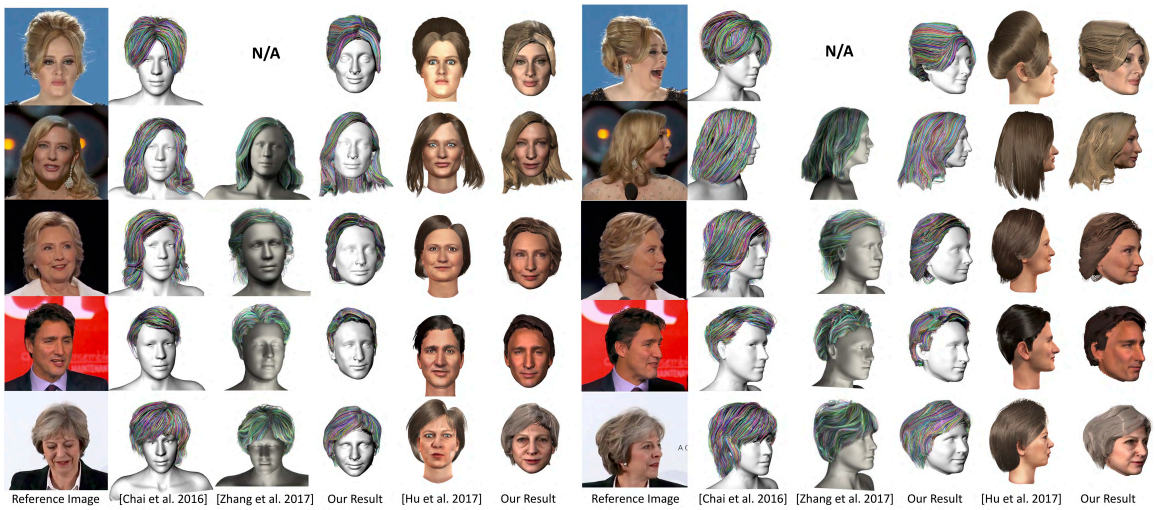


Figure 5.8: This figure shows our results compared to the state-of-the-art methods. For each subject, we show the results in frontal and side views. For each view, the first column shows a reference frame from the video, then we show in the order of the untextured results from autohair [34], four-view method [161], our method and the textured results from avatar digitalization method [69], our method. Note how our result captures more personalized hair details, as also indicated by human studies and quantitative comparisons.

Table 5.1: IOU accuracy between the projected reconstructed hair and the hair segmentation (manually labeled ground truth).

Subject	Frames	four-view [161]	Ours
Hillary	266	0.6295 ± 0.0411	$\mathbf{0.8294 \pm 0.0446}$
Theresa	252	0.6598 ± 0.0258	$\mathbf{0.8111 \pm 0.0216}$
Cate	255	0.5991 ± 0.0474	$\mathbf{0.7749 \pm 0.0669}$
Justin	307	0.4787 ± 0.0882	$\mathbf{0.8028 \pm 0.0187}$

ually labeled, but not used in our training or testing of the hair classifiers) per frame. We show the average IOUs over all the frames of each subject in Table 5.1. A larger IOU means that the reconstructed hair approximates the input better. In total, our reconstruction results get an average IOU rate of around 0.8, while the four-view reconstruction method gets an average IOU of around 0.6. We showed the projection and ground truth hair mask of some example frames in Figure 5.9. Our results resemble the hairstyles better in all the frames.

User Study We performed Amazon Mechanical Turk studies to compare our results to other methods. We showed two results side by side with the ground truth image in different views and asked which shape was more similar to the input, ignoring the face, shoulder and rendering qualities. For each subject, we did 3 groups of studies comparing the frontal, left, and right views. To remove bias, we switched the order of the two results and did 3 more groups of studies. Each view was rated by 40 Turkers, giving a total of 120 different Turkers for each subject. We reported the rate of preference to our results over total in Table 5.2. Our results achieved an average preference rate of 72.7% in all the study groups. Similarly, we did a comparison of the textured results to the avatar digitalization work [69] showing the 0°, 15° and 90° views. The ratio of preferences is reported in Table 5.3. In total, our results were considered better by 90.8% of the Turkers. All the comparison results can be found in Appendix B.

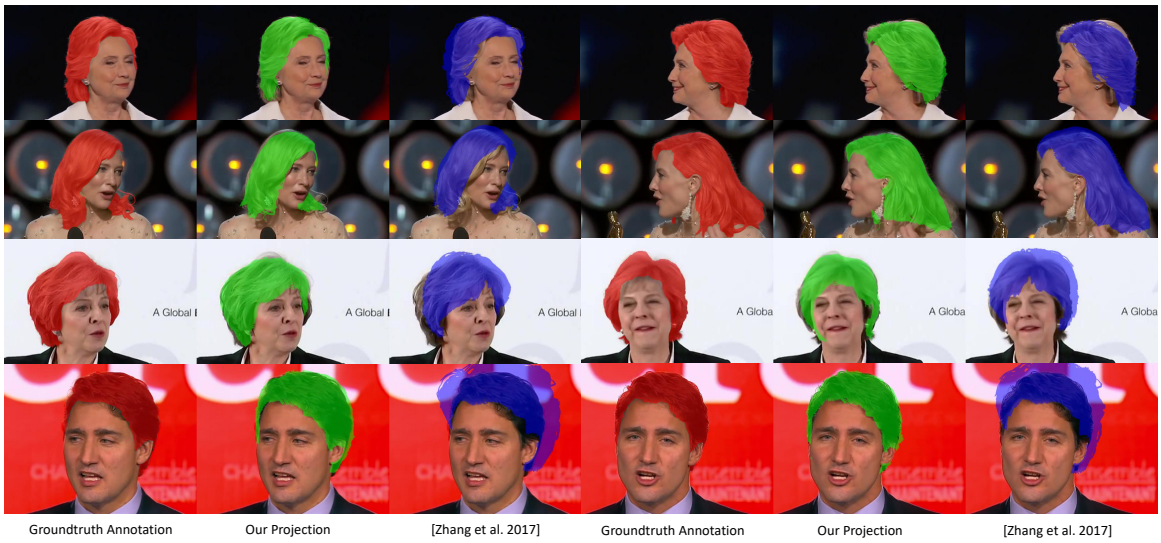


Figure 5.9: This figure shows four example frames comparing the silhouettes of the reconstructed hairstyles to the hair segmentation results. The red mask is the annotated groundtruth hair mask over the image frame. The green mask shows the projected silhouettes from our method over the image and the blue mask shows the projected silhouettes from four-view method [161].

Table 5.2: The ratio of preference to our results over total compared to four-view method [161] based on Amazon Mechanical Turk tests.

Subject	frontal	left	right	total
Hillary	39/40	27/40	27/40	93/120
Theresa	13/40	26/40	27/40	66/120
Cate	30/40	26/40	32/40	88/120
Justin	27/40	37/40	38/40	102/120

Table 5.3: The ratio of preference to our results over total compared to avatar digitalization method [69] based on Amazon Mechanical Turk test.

Subject	0°	15°	90°	total
Adele	29/40	32/40	38/40	99/120
Hillary	35/40	38/40	36/40	109/120
Theresa	35/40	37/40	39/40	111/120
Cate	40/40	40/40	40/40	120/120
Justin	39/40	35/40	32/40	106/120

5.8 Limitation and Applications

5.8.1 Limitations

Our method cannot work on highly dynamic hairstyles due to the high non-rigidity of the hair volumes across the in-the-wild videos. See the example video frames of Olivia Culpo in Figure 5.10(a). The human hand interaction and body occlusion make the segmentation difficult. Our method also fails on videos where the background is too complicated as shown in Figure 5.10(b). The other people or crowds in the background make it hard to estimate the head silhouette of the person and will lead to incorrect correspondences when running structure-from-motion.

For the low-confidence view corrections, we require an input video covering a view range of at least 90 degrees. Fewer views will cause the visual hull to be extremely distorted as shown in Figure 5.10(c), where our deformable registration will fail with a large fitting error. Note that as the view coverage decreases, this problem will be reduced to a single-view reconstruction problem.

(a) Highly Non-rigid Hair (b) Complicated Background c) Fewer view coverages

Figure 5.10: Limitations of our algorithm. In (a) we show example video frames of highly non-rigid hairstyle. In (b) we show an example video frame with a complicated background. In (c) we show the back of a deformed hair mesh towards a visual hull from a small view coverage input.

5.8.2 Applications

Our reconstructed models can be now used for a variety of applications as shown in Figure 5.11(a)(b); we can also change the overall color of the hairstyle, making it darker or lighter, or morphing it to another hairstyle.

Since the hair roots of all our hair models are transfered from the generic shape used to compute the head model, we can assume that hair root points and hair strands are in correspondence for the same person. We resampled all the hair strands with the same number of vertices (50 in our implementations), which can be used for applications such as personalized strand-level hairstyle morphing.

In Figure 5.11(c), we show the hair morphing result of Cate Blanchett in two hairstyles from two different videos. The intermediate results are created by a one-to-one strand interpolation of the source and target hair strands. We can also morph the reconstructed hairstyle to a given hairstyle from the dataset as shown in Figure 5.11(d). The given hairstyle was re-grown from its 3D orientation field using the same set of scalp points to

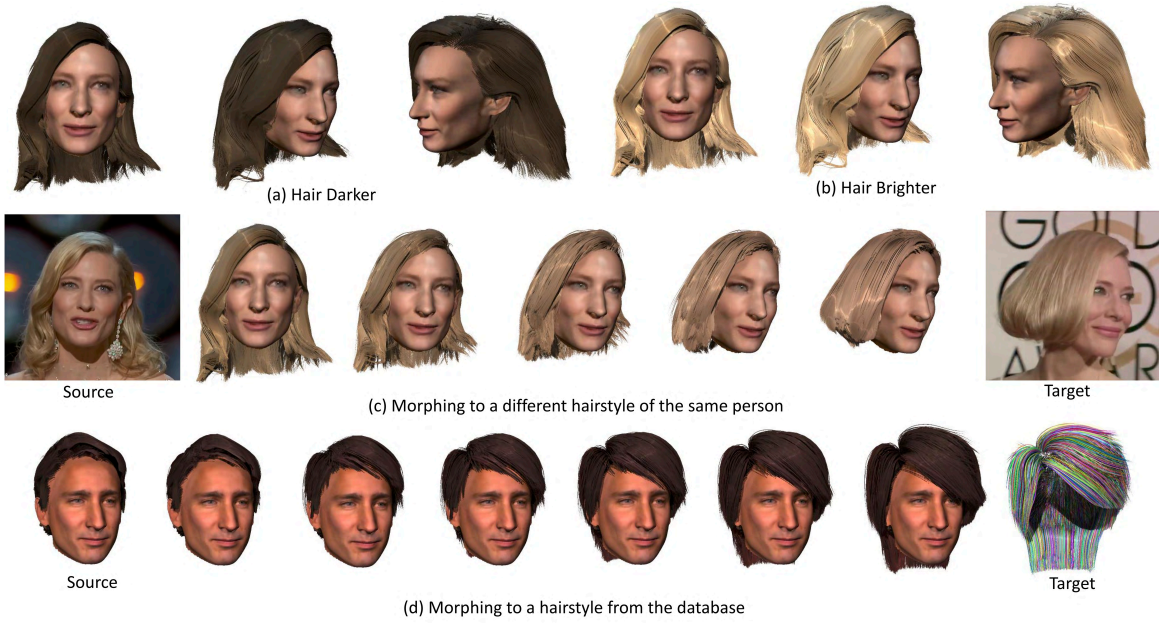


Figure 5.11: Hairstyle change examples. We show a darker and lighter version of Cate Blanchett's hairstyle in (a)(b). (c) shows the hair morphing intermediate results from two different hairstyles of the same person. (d) shows the hair morphing from the person's reconstructed hairstyle to a given hairstyle from the dataset.

create correspondences to the original hairstyle. We trimmed the hair strands that intersect with the face during interpolation.

5.9 Discussion and Future Work

We have described a method that takes as input a video of a person's head in the wild and outputs a detailed 3D hair strand model combined with a reconstructed 3D head to produce a full head model. This method is fully automatic and shows that a head model with higher fidelity can be recovered by combining information from video frames. Our method is not restricted to specific views and head poses, making the full head reconstruction from in-the-wild videos possible. We showed our results on several celebrities as well as mobile selfie videos and compared our work to the most recent state of the art.

However, there are still a number of limitations and possible extensions to explore. One direction is to refine the rough hair mesh estimation for non-rigid hairstyles. Currently in our input, the person's head moves gently and our rough hair mesh is generated from the visual hull, which is only an approximation of the real hair volume, since the hair is non-rigid. We might explore using the ARkit of a smart phone to provide extra depth information to align the hair volume densely across frames. Also, we currently rely on the rigid camera poses estimated using structure from motion with SIFT features to connect all the views. In the future, we want to use facial features and hair specific features to increase the robustness of the frame alignment.

We created face textures from video frames, which caused artifacts due to hair occlusions, decoration occlusions and low resolution of the faces. In the future, we can use generate photo-realistic textures as proposed by Saito et al. [128]. Finally, a future extension to incorporate a facial blend shape model or estimate per frame facial dynamics as proposed by Suwajanakorn et al. [135] to create a fully animatable model would be desirable.

Chapter 6

CONCLUSION AND FUTURE WORK

The purpose of this work was to reconstructing a personalized 3D head shape of a subject from easy inputs using data-driven approaches. We developed algorithms and pipelines to reconstruct a high-detailed face from a single depth image taken from a commercial RGB-D camera and move on to reconstruct more parts of the head from in-the-wild photos and videos.

We first reconstructed a high-quality 3D face mesh from a rough, noisy single Kinect depth image as presented in Chapter 3. We defined a similarity measurement that uses a combination of pseudo-landmarks and azimuth-elevation angle histogram to retrieve from a large 3D face database for the most similar parts of the input face. We did both quantitative comparison and visual comparison to other face reconstruction methods.

Then, in Chapter 4, aiming at reconstruct more parts of a subject's head and totally free the subject out of the capturing session, we designed a new pipeline of head reconstruction from Internet photos. We divided a subject's photos into several pose clusters. A depth map of the frontal cluster was first reconstructed assuming the head is a lambertian surface. The method gradually growed the reconstruction by estimating surface normals per-view cluster and then constraining the depth of current cluster with boundary conditions, resulting a head mesh of the subject that combines all the views. Results of several celebrities were shown using their online photo collections.

Finally, to solve the limitations in Chapter 4 and reconstruct a complete head model with hairs, we developed a fully automatic pipeline to reconstruct a subject's head from a in-the-wild video in Chapter 5. We start with the visual hull of the head shape from the video frames and automatically extracted hair directional informations. A best matching hairstyle was retrieved from a synthetic hairstyle database and deformed to fit the subject's personalized hairstyle. We collected 8 celebrities videos, added 9 subjects from mobile selfie

videos and showed the reconstruction results.

The key contribution of this work is that we showed 3 pipelines that led to the personalized reconstruction of a subject's face or hair from easy inputs, taking advantage of some pre-collected 3D shape databases. For future work, an important extension is to add personalized dynamics to both the face and hair of the subject. From a video sequence, we could extract a personalized facial blend shape to model the expression. Personalized hair motion could also be extracted and analyzed.

Furthermore, we could potentially improve the skin and hair textures from the video. In our current pipeline, we simply take the per-vertex color from the frames as textures regardless of the lighting conditions. In the future, we want to be able to model the environment lighting from the video and extract true albedos for both the skin and hair regions. Moreover, in addition to using SIFT features for frame alignments, we want to investigate on face and hair specific features to get a better correspondence for head pixels across video frames.

BIBLIOGRAPHY

[1] Adobe project animal. `http://blogs.adobe.com/aftereffects/2014/10/weve-created-an-animal-2.html`.

[2] Sameer Agarwal, Keir Mierle, and Others. Ceres solver. `https://code.google.com/p/ceres-solver/`.

[3] Antonio Agudo, JMM Montiel, Lourdes de Agapito, and Begoña Calvo. Online dense non-rigid 3d shape and camera motion recovery. In *BMVC*, 2014.

[4] Oleg Alexander, Graham Fyffe, Jay Busch, Xueming Yu, Ryosuke Ichikari, Andrew Jones, Paul Debevec, Jorge Jimenez, Etienne Danvoye, Bernardo Antionazzi, et al. Digital ira: creating a real-time photoreal digital actor. In *ACM SIGGRAPH 2013 Posters*, page 1. ACM, 2013.

[5] Oleg Alexander, Mike Rogers, William Lambeth, Matt Chiang, and Paul Debevec. The digital emily project: photoreal facial modeling and animation. In *ACM SIGGRAPH 2009 Courses*, page 12. ACM, 2009.

[6] B. Allen, B. Curless, and Z. Popović. The space of human body shapes: reconstruction and parameterization from range scans. In *ACM Transactions on Graphics (TOG)*, volume 22, pages 587–594. ACM, 2003.

[7] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. Scape: shape completion and animation of people. In *ACM Transactions on Graphics (TOG)*, volume 24, pages 408–416. ACM, 2005.

[8] Indriyati Atmosukarto, Linda G Shapiro, and Carrie Heike. The use of genetic programming for learning 3d craniofacial shape quantifications. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 2444–2447. IEEE, 2010.

[9] Tali Basha, Yael Moses, and Nahum Kiryati. Multi-view scene flow estimation: A view centered variational approach. *International journal of computer vision*, 101(1):6–21, 2013.

[10] Ronen Basri, David Jacobs, and Ira Kemelmacher. Photometric stereo with general, unknown lighting. *International Journal of Computer Vision*, 72(3):239–257, 2007.

[11] Ronen Basri and David W Jacobs. Lambertian reflectance and linear subspaces. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(2):218–233, 2003.

[12] Bruce Guenther Baumgart. Geometric modeling for computer vision. Technical report, STANFORD UNIV CA DEPT OF COMPUTER SCIENCE, 1974.

[13] Thabo Beeler, Bernd Bickel, Paul Beardsley, Bob Sumner, and Markus Gross. High-quality single-shot capture of facial geometry. *ACM Transactions on Graphics (TOG)*, 29(4):40, 2010.

[14] Thabo Beeler, Fabian Hahn, Derek Bradley, Bernd Bickel, Paul Beardsley, Craig Gotsman, Robert W Sumner, and Markus Gross. High-quality passive facial performance capture using anchor frames. In *ACM Transactions on Graphics (TOG)*, volume 30, page 75. ACM, 2011.

[15] Stefano Berretti, Alberto Del Bimbo, and Pietro Pala. 3d face recognition using isogeodesic stripes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(12):2162–2177, 2010.

[16] Dmitri Bitouk, Neeraj Kumar, Samreen Dhillon, Peter Belhumeur, and Shree K Nayar. Face swapping: automatically replacing faces in photographs. In *ACM Transactions on Graphics (TOG)*, volume 27, page 39. ACM, 2008.

[17] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 187–194. ACM Press/Addison-Wesley Publishing Co., 1999.

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

[19] Derek Bradley, Wolfgang Heidrich, Tiberiu Popa, and Alla Sheffer. High resolution passive facial performance capture. *ACM Transactions on Graphics (TOG)*, 29(4):41, 2010.

[20] Matthew Brand. A direct method for 3d factorization of nonrigid motion observed in 2d. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 122–128. IEEE, 2005.

[21] Christoph Bregler, Aaron Hertzmann, and Henning Biermann. Recovering non-rigid 3d shape from image streams. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, volume 2, pages 690–696. IEEE, 2000.

[22] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. In *European Conference on Computer Vision (ECCV)*, volume 3024 of *Lecture Notes in Computer Science*, pages 25–36. Springer, May 2004.

[23] Thomas Brox, Andrés Bruhn, Nils Papenberg, and Joachim Weickert. High accuracy optical flow estimation based on a theory for warping. In *Computer Vision-ECCV 2004*. 2004.

[24] Andrés Bruhn, Joachim Weickert, and Christoph Schnörr. Lucas/kanade meets horn/schunck: Combining local and global optic flow methods. *International Journal of Computer Vision*, 61(3):211–231, 2005.

[25] John D Bustard and Mark S Nixon. 3d morphable model construction for robust ear and face recognition. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2582–2589. IEEE, 2010.

[26] Chen Cao, Qiming Hou, and Kun Zhou. Displaced dynamic expression regression for real-time facial tracking and animation. *ACM Transactions on Graphics (TOG)*, 33(4):43, 2014.

[27] Chen Cao, Yanlin Weng, Stephen Lin, and Kun Zhou. 3d shape regression for real-time facial animation. *ACM TOG (Proc. SIGGRAPH)*, 32(4):41, 2013.

[28] Chen Cao, Yanlin Weng, Shun Zhou, Yiying Tong, and Kun Zhou. Facewarehouse: a 3d facial expression database for visual computing. 2013.

[29] Chen Cao, Yanlin Weng, Shun Zhou, Yiying Tong, and Kun Zhou. Facewarehouse: a 3d facial expression database for visual computing. *Visualization and Computer Graphics, IEEE Transactions on*, 20(3):413–425, 2014.

[30] Chen Cao, Hongzhi Wu, Yanlin Weng, Tianjia Shao, and Kun Zhou. Real-time facial animation with image-based dynamic avatars. *ACM Transactions on Graphics (TOG)*, 35(4):126, 2016.

[31] Xudong Cao, Yichen Wei, Fang Wen, and Jian Sun. Face alignment by explicit shape regression. *International Journal of Computer Vision*, 107(2):177–190, 2014.

[32] Yong Cao, Wen C Tien, Petros Faloutsos, and Frédéric Pighin. Expressive speech-driven facial animation. *ACM Transactions on Graphics (TOG)*, 24(4):1283–1302, 2005.

[33] Menglei Chai, Linjie Luo, Kalyan Sunkavalli, Nathan Carr, Sunil Hadap, and Kun Zhou. High-quality hair modeling from a single portrait photo. *ACM Transactions on Graphics (TOG)*, 34(6):204, 2015.

[34] Menglei Chai, Tianjia Shao, Hongzhi Wu, Yanlin Weng, and Kun Zhou. Autohair: Fully automatic hair modeling from a single image. *ACM Transactions on Graphics (TOG)*, 35(4):116, 2016.

[35] Menglei Chai, Lvdi Wang, Yanlin Weng, Xiaogang Jin, and Kun Zhou. Dynamic hair manipulation in images and videos. *ACM Transactions on Graphics (TOG)*, 32(4):75, 2013.

[36] Menglei Chai, Lvdi Wang, Yanlin Weng, Yizhou Yu, Baining Guo, and Kun Zhou. Single-view hair modeling for portrait manipulation. *ACM Transactions on Graphics (TOG)*, 31(4):116, 2012.

[37] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *arXiv preprint arXiv:1606.00915*, 2016.

[38] Timothy F Cootes, Gareth J Edwards, and Christopher J Taylor. Active appearance models. In *Computer Vision—ECCV'98*, pages 484–498. Springer, 1998.

[39] Yuchao Dai, Hongdong Li, and Mingyi He. A simple prior-free method for non-rigid structure-from-motion factorization. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2018–2025. IEEE, 2012.

[40] Kevin Dale, Kalyan Sunkavalli, Micah K Johnson, Daniel Vlasic, Wojciech Matusik, and Hanspeter Pfister. Video face replacement. In *ACM Transactions on Graphics (TOG)*, volume 30, page 130. ACM, 2011.

[41] Paul Debevec. The light stages and their applications to photoreal digital actors. *SIGGRAPH Asia*, 2012.

[42] Mathieu Desbrun, Mark Meyer, Peter Schröder, and Alan H Barr. Implicit fairing of irregular meshes using diffusion and curvature flow. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 317–324. ACM Press/Addison-Wesley Publishing Co., 1999.

[43] Carl Doersch, Saurabh Singh, Abhinav Gupta, Josef Sivic, and Alexei A. Efros. What makes paris look like paris? *ACM Trans. Graph.*, 31(4):101:1–101:9, July 2012.

[44] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(2):295–307, 2016.

[45] J. Wu S. Liang L. G. Shapiro E. Mercan, I. Atmosukarto. *Comprehensive Health Monitoring and Personalized Feedback using Multimedia Data*, chapter Craniofacial Image Analysis. Springer, In Press.

[46] Paul Ekman and Wallace V Friesen. Facial action coding system. 1977.

[47] Tony Ezzat and Tomaso Poggio. Facial analysis and synthesis using image-based models. In *Automatic Face and Gesture Recognition, 1996., Proceedings of the Second International Conference on*, pages 116–121. IEEE, 1996.

[48] Haoqiang Fan, Zhimin Cao, Yuning Jiang, Qi Yin, and Chinchilla Doudou. Learning deep face representation. *arXiv preprint arXiv:1403.2802*, 2014.

[49] Gabriele Fanelli, Matthias Dantone, Juergen Gall, Andrea Fossati, and Luc Van Gool. Random forests for real time 3d face analysis. *International Journal of Computer Vision*, 101(3):437–458, 2013.

[50] Andrew Fitzgibbon, Geoff Cross, and Andrew Zisserman. Automatic 3d model construction for turn-table sequences. *3D Structure from Multiple Images of Large-Scale Environments*, pages 155–170, 1998.

[51] Hongbo Fu, Yichen Wei, Chiew-Lan Tai, and Long Quan. Sketching hairstyles. In *Proceedings of the 4th Eurographics workshop on Sketch-based interfaces and modeling*, pages 31–36. ACM, 2007.

[52] Simon Fuhrmann, Fabian Langguth, and Michael Goesele. Mve-a multi-view reconstruction environment. In *GCH*, pages 11–18, 2014.

[53] Ravi Garg, Anastasios Roussos, and Lourdes Agapito. Dense variational reconstruction of non-rigid surfaces from monocular video. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 1272–1279. IEEE, 2013.

[54] Pablo Garrido, Levi Valgaerts, Ole Rehmsen, Thorsten Thormaehlen, Patrick Perez, and Christian Theobalt. Automatic face reenactment. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 4217–4224. IEEE, 2014.

[55] Pablo Garrido, Levi Valgaerts, Chenglei Wu, and Christian Theobalt. Reconstructing detailed dynamic face geometry from monocular video. *ACM Transactions on Graphics (TOG)*, 32(6):158, 2013.

[56] Abhijeet Ghosh, Graham Fyffe, Borom Tunwattanapong, Jay Busch, Xueming Yu, and Paul Debevec. Multiview face capture using polarized spherical gradient illumination. *ACM Transactions on Graphics (TOG)*, 30(6):129, 2011.

[57] Michael Goesele, Noah Snavely, Brian Curless, Hugues Hoppe, and Steven M Seitz. Multi-view stereo for community photo collections. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.

[58] John C Gower. Generalized procrustes analysis. *Psychometrika*, 40(1):33–51, 1975.

[59] Gravis Research Group. Vizago. `http://www.vizago.ch`.

[60] Brian Guenter, Cindy Grimm, Daniel Wood, Henrique Malvar, and Fredric Pighin. Making faces. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 55–66. ACM, 1998.

[61] Hairbobo. Hairbobo. `http://www.hairbobo.com/faxingtupian`, September 2017.

[62] Tal Hassner. Viewing real-world faces in 3d. ICCV, 2013.

[63] Tal Hassner and Ronen Basri. Example based 3d reconstruction from single 2d images. In *Computer Vision and Pattern Recognition Workshop, 2006. CVPRW'06. Conference on*, pages 15–15. IEEE, 2006.

[64] James Hays and Alexei A Efros. Scene completion using millions of photographs. In *ACM Transactions on Graphics (TOG)*, volume 26, page 4. ACM, 2007.

[65] Pei-Lun Hsieh, Chongyang Ma, Jihun Yu, and Hao Li. Unconstrained realtime facial performance capture. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1675–1683, 2015.

[66] Liwen Hu, Chongyang Ma, Linjie Luo, and Hao Li. Robust hair capture using simulated examples. *ACM Transactions on Graphics (TOG)*, 33(4):126, 2014.

[67] Liwen Hu, Chongyang Ma, Linjie Luo, and Hao Li. Single-view hair modeling using a hairstyle database. *ACM Transactions on Graphics (TOG)*, 34(4):125, 2015.

[68] Liwen Hu, Chongyang Ma, Linjie Luo, Li-Yi Wei, and Hao Li. Capturing braided hairstyles. *ACM Transactions on Graphics (TOG)*, 33(6):225, 2014.

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

[70] Gary B Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical report, Technical Report 07-49, University of Massachusetts, Amherst, 2007.

[71] Hayley Iben, Mark Meyer, Lena Petrovic, Olivier Soares, John Anderson, and Andrew Witkin. Artistic simulation of curly hair. In *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 63–71. ACM, 2013.

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

[73] Megvii Inc. Face++ research toolkit. `http://www.faceplusplus.com/`, December 2013.

[74] Itseez. Open source computer vision library. `https://github.com/itseez/opencv`, 2015.

[75] Wenzel Jakob, Jonathan T Moon, and Steve Marschner. Capturing hair assemblies fiber by fiber. In *ACM Transactions on Graphics (TOG)*, volume 28, page 164. ACM, 2009.

[76] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678. ACM, 2014.

[77] Evangelos Kalogerakis, Aaron Hertzmann, and Karan Singh. Learning 3d mesh segmentation and labeling. *ACM Transactions on Graphics (TOG)*, 29(4):102, 2010.

[78] Vahid Kazemi, Cem Keskin, Taylor Jonathan, Kholi Pushmeet, and Izadi Shahram. Real-time face reconstruction from a single depth image. 2014.

[79] Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *ACM Transactions on Graphics (TOG)*, 32(3):29, 2013.

[80] Ira Kemelmacher-Shlizerman. Internet based morphable model. In *International Conference on Computer Vision (ICCV)*, 2013.

[81] Ira Kemelmacher-Shlizerman. Transfiguring portraits. *ACM Transactions on Graphics (TOG)*, 35(4):94, 2016.

[82] Ira Kemelmacher-Shlizerman and Ronen Basri. 3d face reconstruction from a single image using a single reference face shape. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(2):394–405, 2011.

[83] Ira Kemelmacher-Shlizerman, Aditya Sankar, Eli Shechtman, and Steven M Seitz. Being john malkovich. In *Computer Vision–ECCV 2010*, pages 341–353. Springer, 2010.

[84] Ira Kemelmacher-Shlizerman and Steven M Seitz. Face reconstruction in the wild. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1746–1753. IEEE, 2011.

[85] Ira Kemelmacher-Shlizerman and Steven M Seitz. Collection flow. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1792–1799. IEEE, 2012.

[86] Ira Kemelmacher-Shlizerman, Eli Shechtman, Rahul Garg, and Steven M Seitz. Exploring photobios. In *ACM Transactions on Graphics (TOG)*, volume 30, page 61. ACM, 2011.

[87] Ira Kemelmacher-Shlizerman, Supasorn Suwajanakorn, and Steven M Seitz. Illumination-aware age progression. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 3334–3341. IEEE, 2014.

[88] Natasha Kholgade, Iain Matthews, and Yaser Sheikh. Content retargeting using parameter-parallel facial layers. In *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 195–204. ACM, 2011.

[89] Artiom Kovnatsky, Michael M Bronstein, Alexander M Bronstein, Klaus Glashoff, and Ron Kimmel. Coupled quasi-harmonic bases. In *Computer Graphics Forum*, volume 32, pages 439–448. Wiley Online Library, 2013.

[90] Kiriakos N Kutulakos and Steven M Seitz. A theory of shape by space carving. *International Journal of Computer Vision*, 38(3):199–218, 2000.

[91] Aldo Laurentini. The visual hull concept for silhouette-based image understanding. *IEEE Transactions on pattern analysis and machine intelligence*, 16(2):150–162, 1994.

[92] Chan-Su Lee and Ahmed Elgammal. Facial expression analysis using nonlinear decomposable generative models. In *Analysis and Modelling of Faces and Gestures*, pages 17–31. Springer, 2005.

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

[94] Hao Li, Jihun Yu, Yuting Ye, and Chris Bregler. Realtime facial animation with on-the-fly correctives. *ACM Trans. Graph.*, 32(4):42–1, 2013.

[95] Shu Liang, Ira Kemelmacher-Shlizerman, and Linda G Shapiro. 3d face hallucination from a single depth frame. In *3D Vision (3DV), 2014 2nd international conference on*, volume 1, pages 31–38. IEEE, 2014.

[96] Shu Liang, Linda G Shapiro, and Ira Kemelmacher-Shlizerman. Head reconstruction from internet photos. In *European Conference on Computer Vision*, pages 360–374. Springer, 2016.

[97] Shu Liang, Jia Wu, Seth M Weinberg, and Linda G Shapiro. Improved detection of landmarks on 3d human face data. In *Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE*, pages 6482–6485. IEEE, 2013.

[98] C. Liu. *Beyond Pixels: Exploring New Representations and Applications for Motion Analysis*. PhD thesis, MIT, 2009.

[99] Sifei Liu, Jianping Shi, Ji Liang, and Ming-Hsuan Yang. Face parsing via recurrent propagation. *arXiv preprint arXiv:1708.01936*, 2017.

[100] Sifei Liu, Jimei Yang, Chang Huang, and Ming-Hsuan Yang. Multi-objective convolutional learning for face labeling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3451–3459, 2015.

[101] Zicheng Liu, Ying Shan, and Zhengyou Zhang. Expressive expression mapping with ratio images. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 271–276. ACM, 2001.

[102] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[103] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[104] Yao Lu, Aakanksha Chowdhery, and Srikanth Kandula. Optasia: A relational platform for efficient large-scale video analytics. In *Proceedings of the Seventh ACM Symposium on Cloud Computing*, pages 57–70. ACM, 2016.

[105] Yao Lu, Srikanth Kandula, and Surajit Chaudhuri. Interactive demonstration of probabilistic predicates. In *Proceedings of the 2018 International Conference on Management of Data*, pages 1669–1672. ACM, 2018.

[106] Linjie Luo, Hao Li, and Szymon Rusinkiewicz. Structure-aware hair capture. *ACM Transactions on Graphics (TOG)*, 32(4):76, 2013.

[107] Ping Luo, Xiaogang Wang, and Xiaoou Tang. Hierarchical face parsing via deep learning. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2480–2487. IEEE, 2012.

[108] Wan-Chun Ma, Andrew Jones, Jen-Yuan Chiang, Tim Hawkins, Sune Frederiksen, Pieter Peers, Marko Vukovic, Ming Ouhyoung, and Paul Debevec. Facial performance synthesis using deformation-driven polynomial displacement maps. In *ACM Transactions on Graphics (TOG)*, volume 27, page 121. ACM, 2008.

[109] Fabio Maninchedda, Christian Häne, Bastien Jacquet, Amaël Delaunoy, and Marc Pollefeys. Semantic 3d reconstruction of heads. In *European Conference on Computer Vision*, pages 667–683. Springer, 2016.

[110] Ezgi Mercan, Linda G Shapiro, Seth M Weinberg, and Su-In Lee. The use of pseudo-landmarks for craniofacial analysis: A comparative study with l 1-regularized logistic regression. In *Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE*, pages 6083–6086. IEEE, 2013.

[111] Microsoft. Kinect for windows software development kit v1.8. `http://www.microsoft.com/en-us/kinectforwindows/`.

[112] Microsoft. Microsoft face api. https://www.microsoft.com/cognitive-services/en-us/face-api. Accessed: 2016-03-30.

[113] Diego Nehab, Szymon Rusinkiewicz, James Davis, and Ravi Ramamoorthi. Efficiently combining positions and normals for precise 3d geometry. In *ACM Transactions on Graphics (TOG)*, volume 24, pages 536–543. ACM, 2005.

[114] Richard A Newcombe, Andrew J Davison, Shahram Izadi, Pushmeet Kohli, Otmar Hilliges, Jamie Shotton, David Molyneaux, Steve Hodges, David Kim, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on*, pages 127–136. IEEE, 2011.

[115] Richard A Newcombe, Dieter Fox, and Steven M Seitz. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 343–352, 2015.

[116] Qiang Ning, Kan Chen, Li Yi, Chuchu Fan, Yao Lu, and Jiangtao Wen. Image super-resolution via analysis sparse prior. *IEEE Signal Processing Letters*, 20(4):399–402, 2013.

[117] Sylvain Paris, Hector M Briceño, and François X Sillion. Capture of hair geometry from multiple images. In *ACM Transactions on Graphics (TOG)*, volume 23, pages 712–719. ACM, 2004.

[118] Sylvain Paris, Will Chang, Oleg I Kozhushnyan, Wojciech Jarosz, Wojciech Matusik, Matthias Zwicker, and Frédo Durand. Hair photobooth: geometric and photometric acquisition of real hairstyles. In *ACM Transactions on Graphics (TOG)*, volume 27, page 30. ACM, 2008.

[119] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter. A 3d face model for pose and illumination invariant face recognition. Genova, Italy, 2009. IEEE.

[120] Frédéric Pighin, Jamie Hecker, Dani Lischinski, Richard Szeliski, and David H Salesin. Synthesizing realistic facial expressions from photographs. In *ACM SIGGRAPH 2006 Courses*, page 19. ACM, 2006.

[121] Ulrich Pinkall and Konrad Polthier. Computing discrete minimal surfaces and their conjugates. *Experimental mathematics*, 2(1):15–36, 1993.

[122] Jonathan Pokrass, Alexander M Bronstein, and Michael M Bronstein. Partial shape matching without point-wise correspondence. *Numerical Mathematics: Theory, Methods & Applications*, 6(1), 2013.

[123] Elad Richardson, Matan Sela, and Ron Kimmel. 3d face reconstruction by learning from synthetic data. In *3D Vision (3DV), 2016 Fourth International Conference on*, pages 460–469. IEEE, 2016.

[124] Elad Richardson, Matan Sela, Roy Or-El, and Ron Kimmel. Learning detailed face reconstruction from a single image. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5553–5562. IEEE, 2017.

[125] Robert E Rosenblum, Wayne E Carlson, and Edwin Tripp. Simulating the structure and dynamics of human hair: modelling, rendering and animation. *Computer Animation and Virtual Worlds*, 2(4):141–148, 1991.

[126] Joseph Roth, Yiying Tong, and Xiaoming Liu. Unconstrained 3d face reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2606–2615, 2015.

[127] Joseph Roth, Yiying Tong, and Xiaoming Liu. Adaptive 3d face reconstruction from unconstrained photo collections. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4197–4206, 2016.

[128] Shunsuke Saito, Lingyu Wei, Liwen Hu, Koki Nagano, and Hao Li. Photorealistic facial texture inference using deep neural networks. *arXiv preprint arXiv:1612.00523*, 2016.

[129] Jason M Saragih, Simon Lucey, and Jeffrey F Cohn. Face alignment through subspace constrained mean-shifts. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1034–1041. IEEE, 2009.

[130] Ari Shapiro, Andrew Feng, Ruizhe Wang, Hao Li, Mark Bolas, Gerard Medioni, and Evan Suma. Rapid avatar capture and simulation using commodity depth sensors. *Computer Animation and Virtual Worlds*, 25(3-4):201–211, 2014.

[131] Jie Shen, Stefanos Zafeiriou, Grigorios G Chrysos, Jean Kossaifi, Georgios Tzimiropoulos, and Maja Pantic. The first facial landmark tracking in-the-wild challenge: Benchmark and results. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 50–58, 2015.

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

[133] Olga Sorkine, Daniel Cohen-Or, Yaron Lipman, Marc Alexa, Christian Rössl, and H-P Seidel. Laplacian surface editing. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 175–184. ACM, 2004.

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

[135] Supasorn Suwajanakorn, Ira Kemelmacher-Shlizerman, and Steven M Seitz. Total moving face reconstruction. In *Computer Vision–ECCV 2014*. 2014.

[136] Supasorn Suwajanakorn, Steven M Seitz, and Ira Kemelmacher-Shlizerman. What makes tom hanks look like tom hanks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3952–3960, 2015.

[137] J Rafael Tena, Fernando De la Torre, and Iain Matthews. Interactive region-based linear 3d face models. In *ACM Transactions on Graphics (TOG)*, volume 30, page 76. ACM, 2011.

[138] Justus Thies, Michael Zollhoefer, Matthias Niessner, Levi Valgaerts, Marc Stamminger, and Christian Theobalt. Real-time expression transfer for facial reenactment. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 2015.

[139] Anh Tuan Tran, Tal Hassner, Iacopo Masi, and Gérard Medioni. Regressing robust and discriminative 3d morphable models with a very deep neural network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1493–1502. IEEE, 2017.

[140] Levi Valgaerts, Andrés Bruhn, Henning Zimmer, Joachim Weickert, Carsten Stoll, and Christian Theobalt. Joint estimation of motion, structure and geometry from stereo sequences. In *Computer Vision–ECCV 2010*, pages 568–581. Springer, 2010.

[141] Levi Valgaerts, Chenglei Wu, Andrés Bruhn, Hans-Peter Seidel, and Christian Theobalt. Lightweight binocular facial performance capture under uncontrolled lighting. *ACM Trans. Graph.*, 31(6):187, 2012.

[142] Nuttapon Vanakittistien, Attawith Sudsang, and Nuttapong Chentanez. 3d hair model from small set of images. In *Proceedings of the 9th International Conference on Motion in Games*, pages 85–90. ACM, 2016.

[143] Sundar Vedula, Simon Baker, Peter Rander, Robert Collins, and Takeo Kanade. Three-dimensional scene flow. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 722–729. IEEE, 1999.

[144] Daniel Vlasic, Matthew Brand, Hanspeter Pfister, and Jovan Popović. Face transfer with multilinear models. In *ACM Transactions on Graphics (TOG)*, volume 24, pages 426–433. ACM, 2005.

[145] Grace Wahba. *Spline models for observational data*, volume 59. Siam, 1990.

[146] Dan Wang, Xiujuan Chai, Hongming Zhang, Hong Chang, Wei Zeng, and Shiguang Shan. A novel coarse-to-fine hair segmentation method. In *Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, pages 233–238. IEEE, 2011.

[147] Yang Wang, Xiaolei Huang, Chan-Su Lee, Song Zhang, Zhiguo Li, Dimitris Samaras, Dimitris Metaxas, Ahmed Elgammal, and Peisen Huang. High resolution acquisition, learning and transfer of dynamic 3-d facial expressions. In *Computer Graphics Forum*, volume 23, pages 677–686. Wiley Online Library, 2004.

[148] Kelly Ward, Florence Bertails, Tae-Yong Kim, Stephen R Marschner, Marie-Paule Cani, and Ming C Lin. A survey on hair modeling: Styling, simulation, and rendering. *IEEE Transactions on Visualization and Computer Graphics*, 13(2), 2007.

[149] Seth M Weinberg, Zachary D Raffensperger, Matthew J Kesterke, Carrie L Heike, Michael L Cunningham, Jacqueline T Hecht, Chung How Kau, Jeffrey C Murray, George L Wehby, Lina M Moreno, et al. The 3d facial norms database: Part 1. a web-based craniofacial anthropometric and image repository for the clinical and research community. *The Cleft Palate-Craniofacial Journal*, 53(6):185–197, 2016.

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

[151] Thibaut Weise, Hao Li, Luc Van Gool, and Mark Pauly. Face/off: Live facial puppetry. In *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 7–16. ACM, 2009.

[152] Yanlin Weng, Lvdi Wang, Xiao Li, Menglei Chai, and Kun Zhou. Hair interpolation for portrait morphing. In *Computer Graphics Forum*, volume 32, pages 79–84. Wiley Online Library, 2013.

[153] Changchang Wu. Visualsfm: A visual structure from motion system. 2011.

[154] Xuehan Xiong and Fernando De la Torre. Supervised descent method and its applications to face alignment. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 532–539. IEEE, 2013.

[155] Xuehan Xiong and Fernando De la Torre. Supervised descent method and its applications to face alignment. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[156] Feng Xu, Jinxiang Chai, Yilong Liu, and Xin Tong. Controllable high-fidelity facial performance transfer. *ACM Transactions on Graphics (TOG)*, 33(4):42, 2014.

[157] Yaser Yacoob and Larry S Davis. Detection and analysis of hair. *IEEE transactions on pattern analysis and machine intelligence*, 28(7):1164–1169, 2006.

[158] Fei Yang, Lubomir Bourdev, Eli Shechtman, Jue Wang, and Dimitris Metaxas. Facial expression editing in video using a temporally-smooth factorization. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 861–868. IEEE, 2012.

[159] Matthew D Zeiler, Graham W Taylor, Leonid Sigal, Iain Matthews, and Rob Fergus. Facial expression transfer with input-output temporal restricted boltzmann machines. In *Advances in Neural Information Processing Systems*, pages 1629–1637, 2011.

[160] Li Zhang, Noah Snavely, Brian Curless, and Steven M Seitz. Spacetime faces: High-resolution capture for˜ modeling and animation. In *Data-Driven 3D Facial Animation*, pages 248–276. Springer, 2007.

[161] Meng Zhang, Menglei Chai, Hongzhi Wu, Hao Yang, and Kun Zhou. A data-driven approach to four-view image-based hair modeling. *ACM Transactions on Graphics (TOG)*, 36(4):156, 2017.

[162] Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip HS Torr. Conditional random fields as recurrent neural networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1529–1537, 2015.

[163] Xiangxin Zhu and Deva Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2879–2886. IEEE, 2012.

[164] Michael Zollhöfer, Michael Martinek, Günther Greiner, Marc Stamminger, and Jochen Süßmuth. Automatic reconstruction of personalized avatars from 3d face scans. *Computer Animation and Virtual Worlds*, 22(2-3):195–202, 2011.

[165] Michael Zollhöfer, Matthias Nießner, Shahram Izadi, Christoph Rehmann, Christopher Zach, Matthew Fisher, Chenglei Wu, Andrew Fitzgibbon, Charles Loop, Christian Theobalt, et al. Real-time non-rigid reconstruction using an rgb-d camera. *ACM Transactions on Graphics, TOG*, 2014.

Appendix A

HAIR CLASSIFIERS

Hair segmentation is an important part of face parsing and full head reconstruction. Yacoob et al. [157] detected hair based on the position relationship between face and hair and a simple color model. Wang et al. [146] proposed a coarse-to-fine hair segmentation method that starts from a coarse candidate region and performs graph-cuts to segment the hair. A CNN-based face parsing method by Luo et al. [107] hierarchically combined several detectors to detect face components. Liu et al. [100, 99] proposed multi-objective learning frameworks that could parse facial components as well as hair regions, but this model requires facial landmarks as prior inputs and can only handle simple hairstyles. To allow robust hair segmentation on various hairstyles, Chai et al. [34] trained a deep network specifically for the hair regions. However, their method requires pre-alignment of the face to detect the hair region. In recent years, fully convolutional networks (FCN) by Long et al. [102] have been widely used for pixel-level segmentation. We adopted the FCN model for robust hair segmentation and trained a network to segment hair regions across various poses.

We collected $15,977$ hair salon images from a hairstyle design website [61] [1] and $3,923$ Internet images of celebrities from Google image search in various head poses and different hairstyles total of $19,900$. Each pixel in those images was labeled manually as hair or non-hair. To preserve continuity of the hair region, hair accessories were labeled as hair. The training images were not cropped nor aligned to increase robustness.

A.1 Hair Segmentation Classifier

We trained the hair segmentation classifier using a fully convolutional network with $13,900$ images out of all the collected photos, and the remaining were used for testing. Specifically,

[1] `http://www.hairbobo.com/faxingtupian`

the FCN-32s model [102] was used, and it was fine-tuned with PASCAL VOC data from the ILSVRC-trained VGG-16 model. To only detect the hair category, we changed the output number of the last convolution layer to one and add a sigmoid layer to get scores between 0 and 1. The output score represents the probability that the pixel belongs to hair. In our implementation, we resized all our input images to 500×500. With FCN-32s, we downsampled the output with a factor of 32. We later used bilinear interpolation to upsample the output heatmap to obtain the final segmentation result. We fine-tuned the pretrained FCN-32s with the following parameters: minibatch size 1, learning rate 10^{-9}, momentum 0.99, and weight decay 0.0005. We froze all the layers except the last score layer in the first $100,000$ iterations. Then we fine-tuned all the layers in the next $100,000$ iterations.

We tested our segmenter on the $6,000$ test images. The hair segmentation network was implemented with Caffe [76] and C++ and ran on a NVIDIA GTX 1080 GPU with an inference time of 150ms for a 500×500 input image. The accuracy on the test images reached 0.9613, and the IOU rate reached 0.8585.

Using automatic hair segmentation (that works well across views, even back views) is the key to enabling a fully automatic hair modeling system. Our algorithm is capable of segmenting the hair region successfully in different views and head poses. However, it still fails to segment some hairstyles correctly when the hair color is too close to the background or when the image has a large motion blur.

We compared our classifier to two methods: Liu et al.'s work [99] and DeepLab [37] (which was compared to by Chai et al. [34], but Chai et al. [34] does not provide code so we compared with DeepLab). We fine tuned and ran DeepLab on all of our test images and got a pixel accuracy of 0.8892 and IOU rate of 0.7173. For the method of Liu et al. [99], we used the pre-trained model to run on only 41.3% of our test images, since it requires pre-detection of the face and fails on back and side views. The pixel accuracy for the 41.3% test images was 0.8462, and IOU rate was 0.5573.

A.2 Hair Directional Classifier

Hair areas were divided to regions based on their general growing trends. One of four directional labels was assigned in the labeling stage: $[0, 0.5\pi)$, $[0.5\pi, \pi)$, $[\pi, 1.5\pi)$, $[1.5\pi, 2\pi)$. Hair accessories and hair occlusions were labeled as undetermined region, and background pixels were labeled as background. We trained a modified VGG16 network as proposed by Chai et al. [34] on the hair regions, using a multi-class approach with 6 classes (4 directions, 1 background, 1 undetermined).

To train our hair directional classifier, we cropped and extracted the hair region, resized each image to 256×256 and downsampled 8 times with a bilinear filter. The output result was then upsampled to the original image size with bilinear interpolation and then followed by a CRF for per-pixel labels. In the training stage, we utilized the same set of $13,900$ images and augmented the dataset to $20,000$ images by image rotation, translation, and mirroring.

We implemented the classifier in the same environment as the segmenter and set the minibatch size to 32 and the initial learning rate to 0.0001 with exponential decay. Our network converged after 50k steps. The inference time was 59ms for a 256×256 input image. We ran the directional classifier on the same test set of $6,000$ images and got an accuracy of 0.9425.

Our classifier typically fails to generate a correct direction label for some small regions on the side of the face. However, in our pipeline, since we have video sequences, we can still get a correct direction from a different view. The availability of many views compensates for individual failure cases.

Appendix B

AMAZON MECHANICAL TURK SURVEYS

B.1 Comparison to Four-view Method [161]

To ensure the diversity of the answers, we separated each result into 6 surveys and assigned each survey to 20 different mechanical turks. The result of each subject is evaluated by 120 different mechanical turks. The instruction of each survey is as following:

In the photo you see three columns:

Input || Hair model #1 || Hair model #2

Which of the two hair models resembles in 3D hair shape more to the input? Ignore: face, shoulders, and rendering quality.

◦ Hair Column 2

◦ Hair Column 3

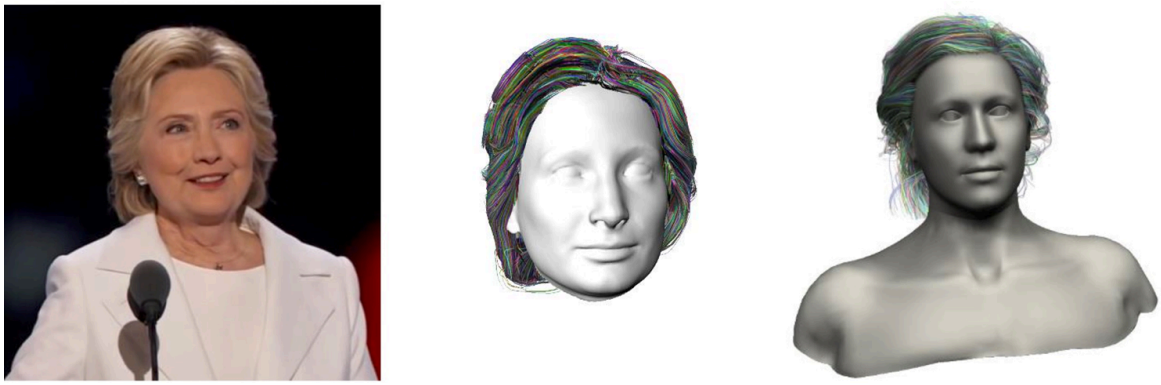


Figure B.1: Survey 1.1

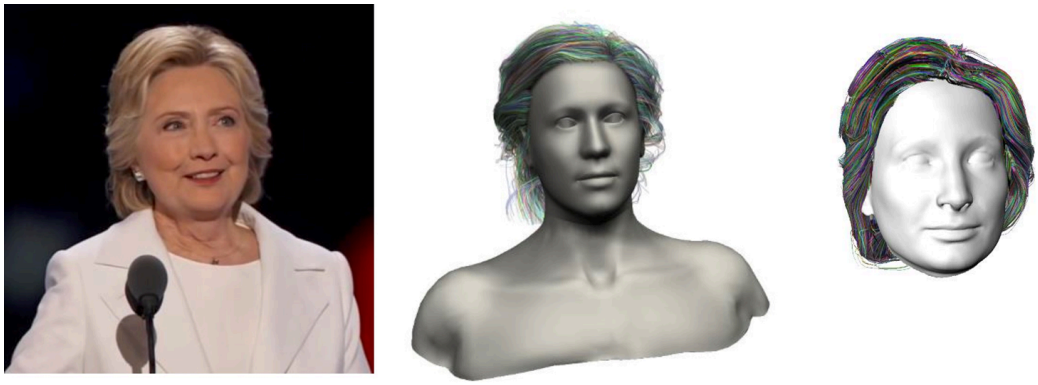


Figure B.2: Survey 1.2

Figure B.3: Survey 1.3

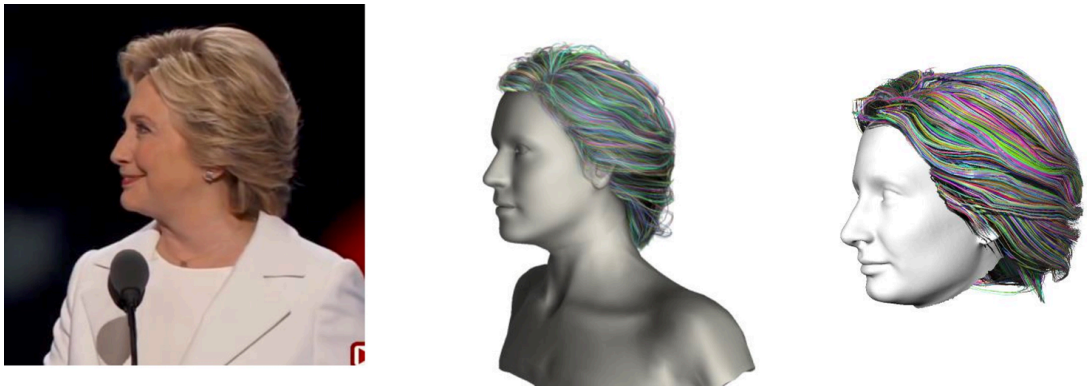


Figure B.4: Survey 1.4

Figure B.5: Survey 1.5

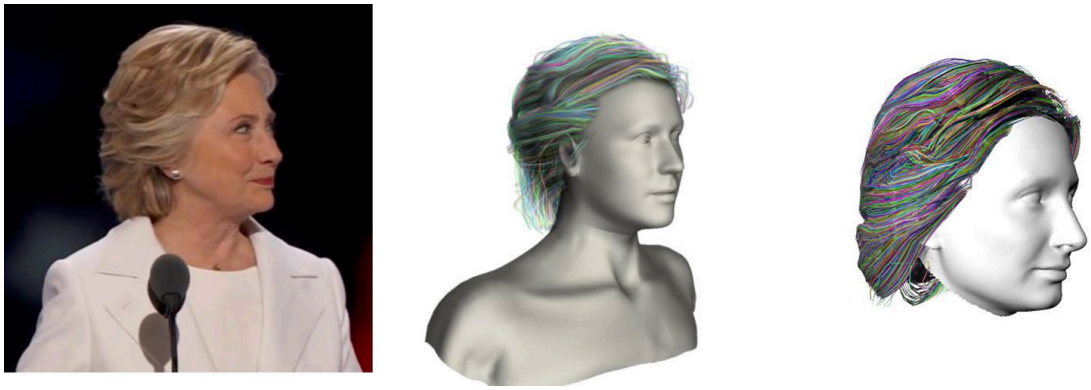


Figure B.6: Survey 1.6

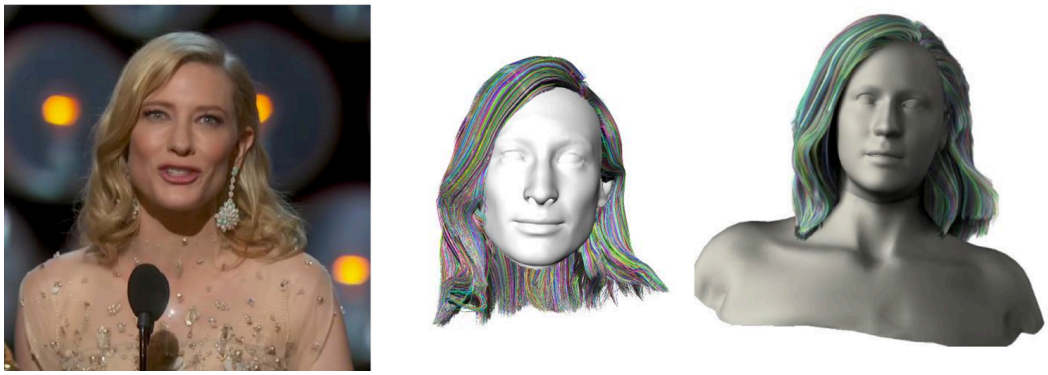


Figure B.7: Survey 1.7

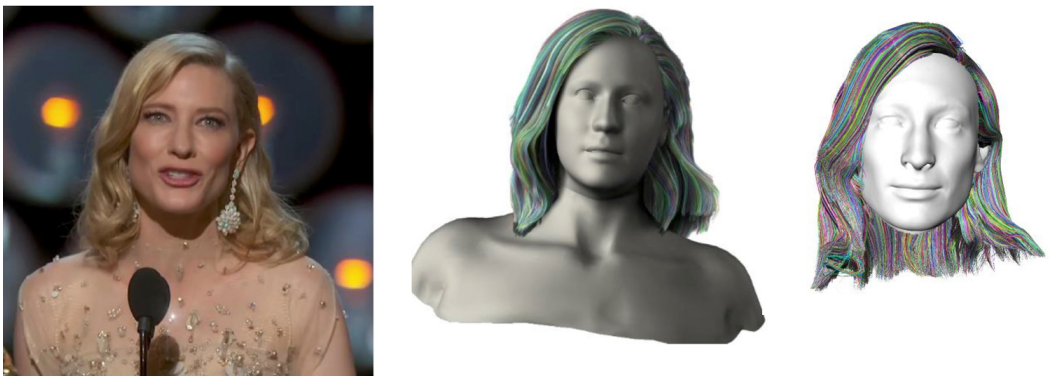


Figure B.8: Survey 1.8

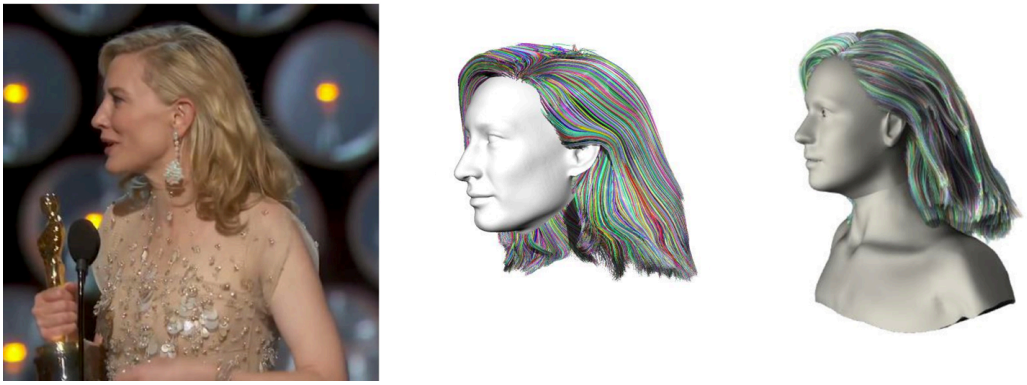


Figure B.9: Survey 1.9

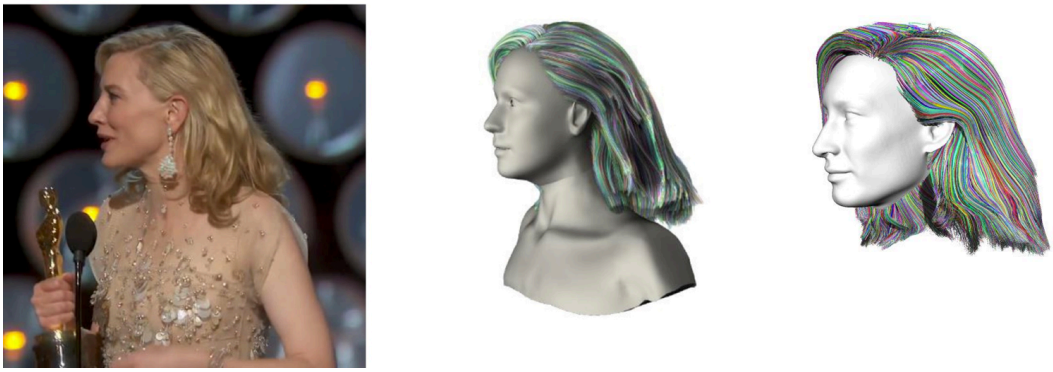


Figure B.10: Survey 1.10

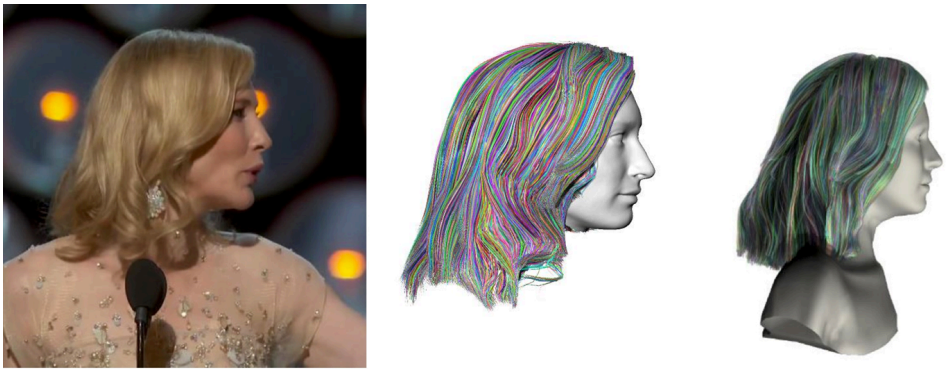


Figure B.11: Survey 1.11

Figure B.12: Survey 1.12

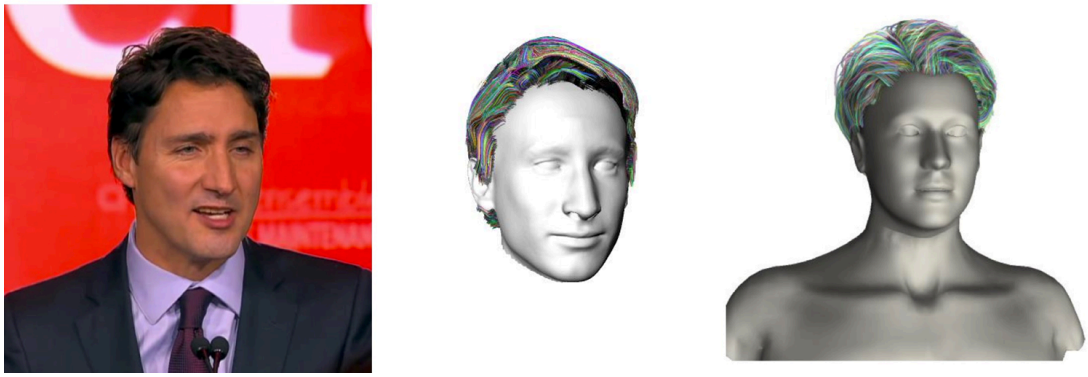


Figure B.13: Survey 1.13

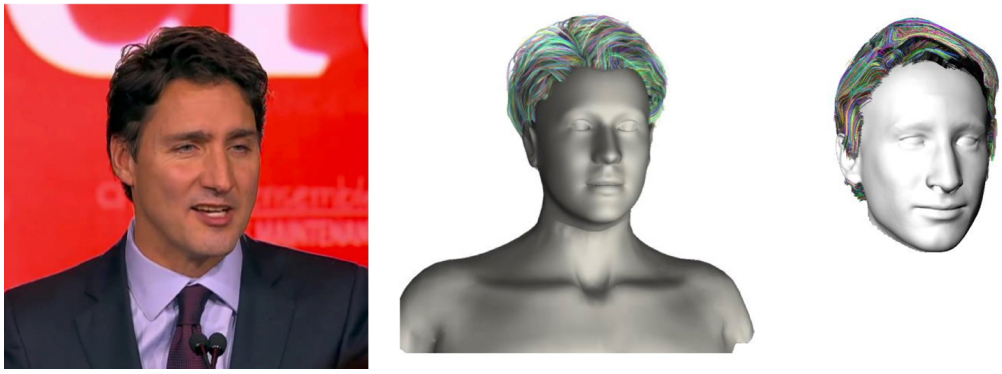


Figure B.14: Survey 1.14

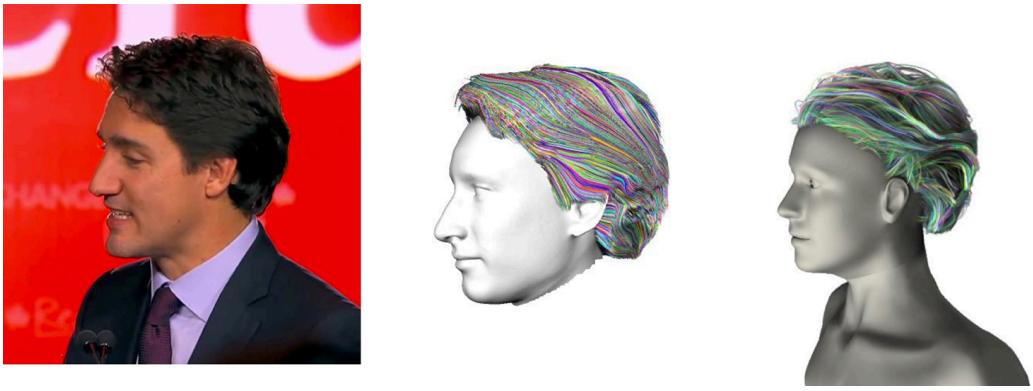


Figure B.15: Survey 1.15

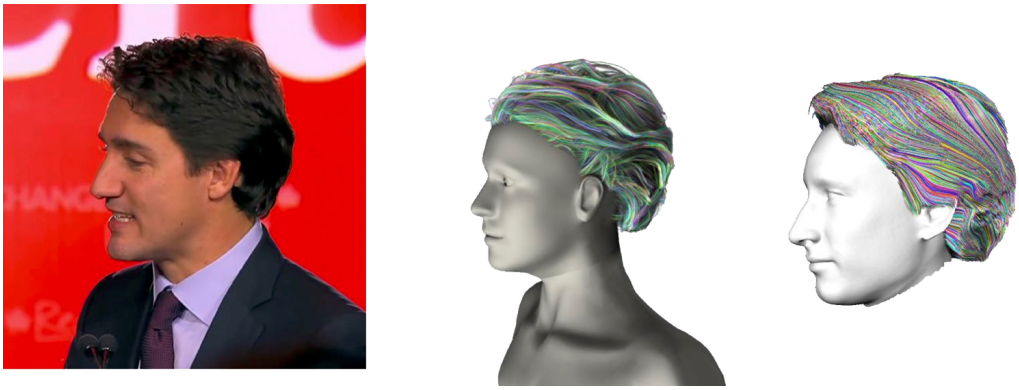


Figure B.16: Survey 1.16

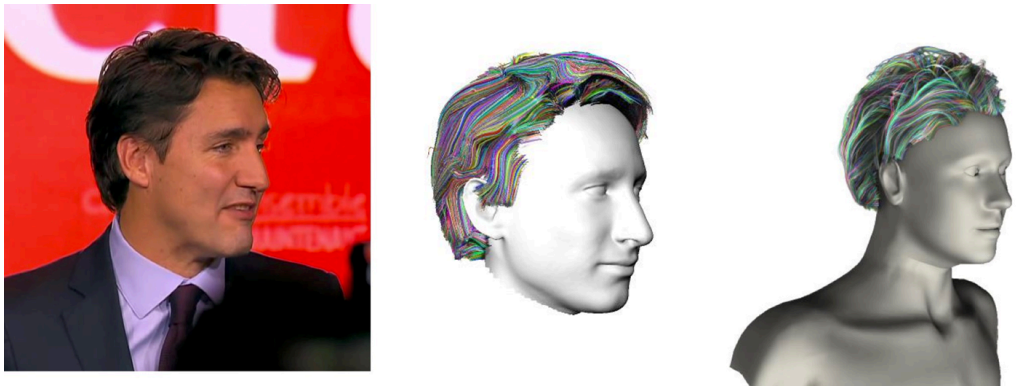


Figure B.17: Survey 1.17

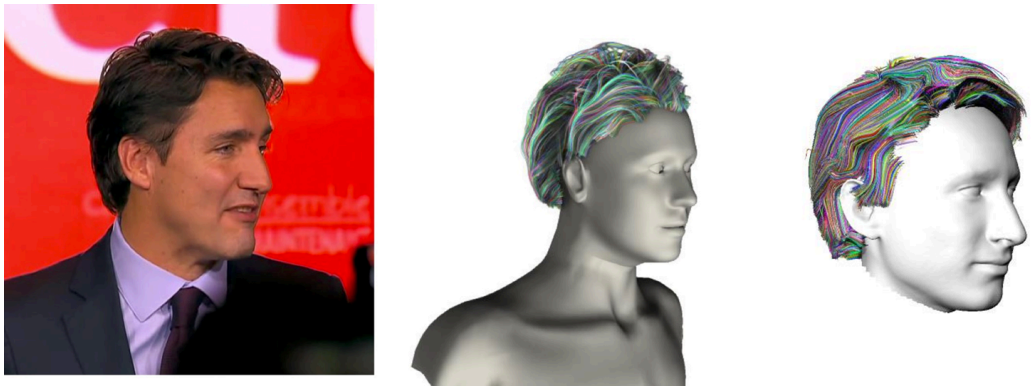


Figure B.18: Survey 1.18

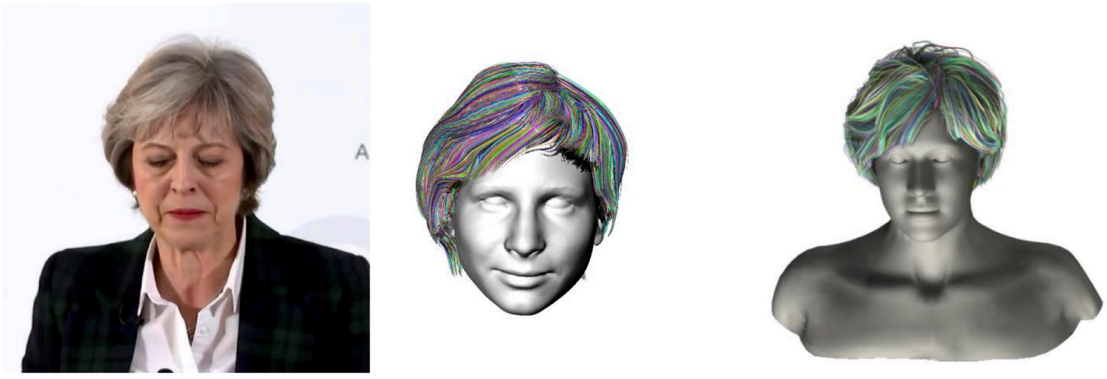


Figure B.19: Survey 1.19

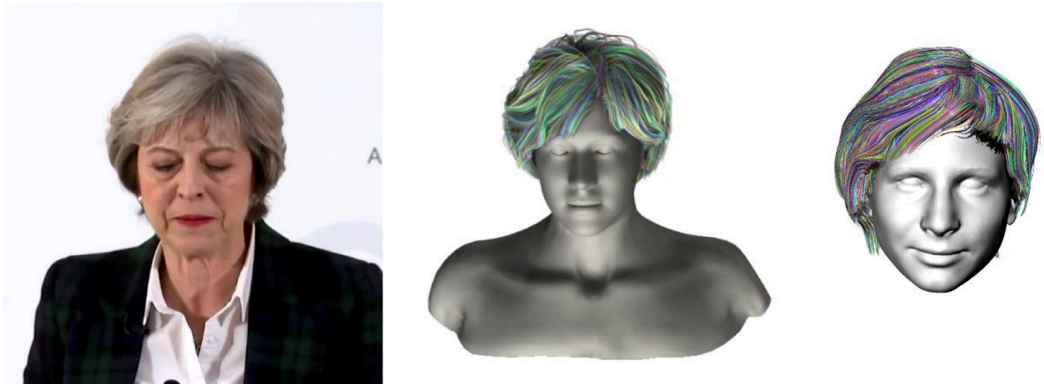


Figure B.20: Survey 1.20

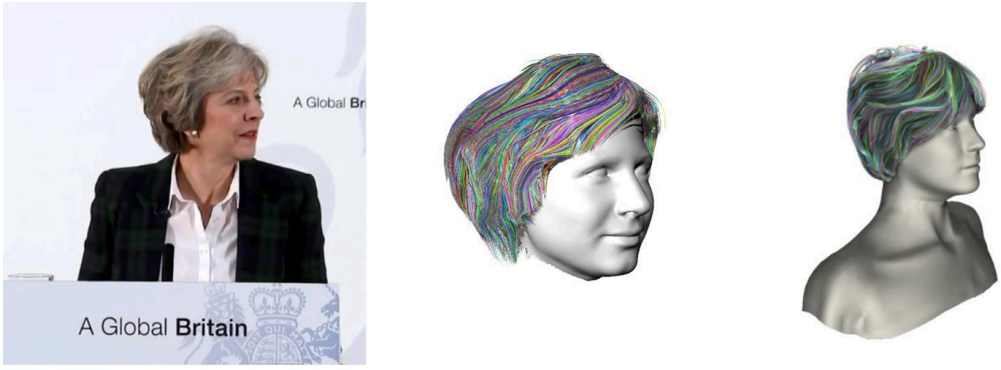


Figure B.21: Survey 1.21

Figure B.22: Survey 1.22

Figure B.23: Survey 1.23

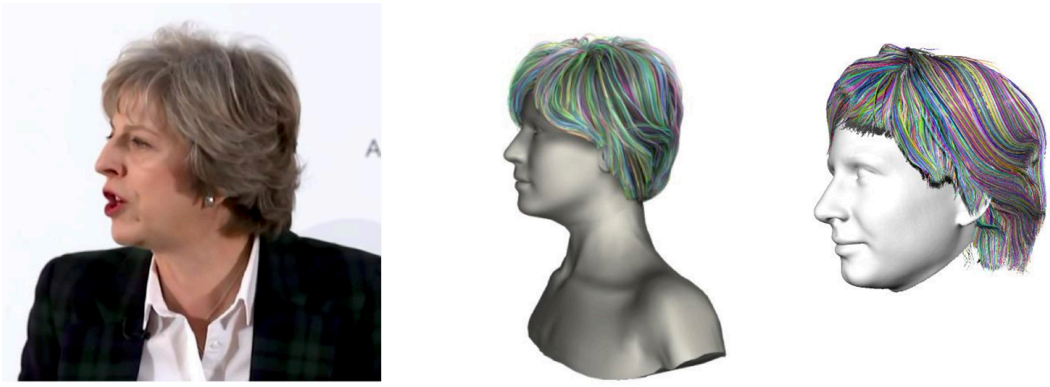


Figure B.24: Survey 1.24

B.2 Comparison to Digital Avatar Method [69]

To ensure the diversity of the answers, we separated each result into 3 surveys and assigned each survey to 20 different mechanical turks. The result of each subject is evaluated by 60 different mechanical turks. The instruction of each survey is as following:

In the photo you see three colums:

Input || Hair model #1 || Hair model #2

Which of the two hair models resembles in 3D hair shape more to the input? Ignore: face, shoulders, texture and rendering quality.

◦ Hair Column 2

◦ Hair Column 3

Figure B.25: Survey 2.1

Figure B.26: Survey 2.2

Figure B.27: Survey 2.3

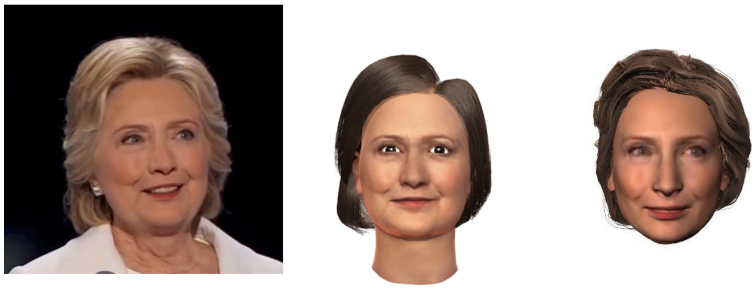


Figure B.28: Survey 2.4

Figure B.29: Survey 2.5

Figure B.30: Survey 2.6

Figure B.31: Survey 2.7

Figure B.32: Survey 2.8

Figure B.33: Survey 2.9

Figure B.34: Survey 2.10

Figure B.35: Survey 2.11

Figure B.36: Survey 2.12

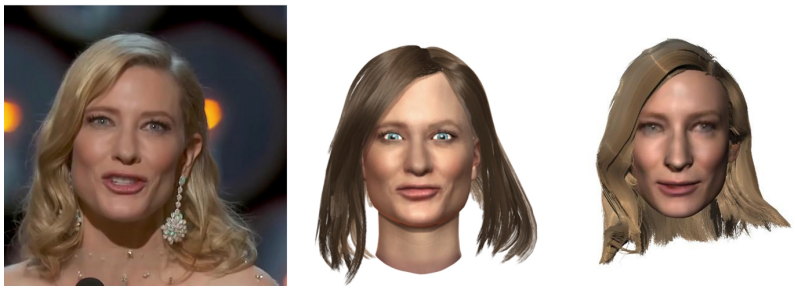


Figure B.37: Survey 2.13

Figure B.38: Survey 2.14

Figure B.39: Survey 2.15

VITA

Shu Liang recieved her Bachelor of Science in Electronic Engineering from Tsinghua University in China, her Master of Science in Computer Science and Engineering from University of Washington in Seattle, WA. She is currently a PhD candidate in Computer Science and Engineering at the University of Washington.