Audiovisual Persona Reconstruction

Supasorn Suwajanakorn

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

Doctor of Philosophy

University of Washington

2017

Reading Committee:

Steve M. Seitz, Chair

Ira Kemelmacher-Shlizerman, Chair

Richard Szeliski

Program Authorized to Offer Degree:
Computer Science & Engineering

University of Washington

Abstract

Audiovisual Persona Reconstruction

Supasorn Suwajanakorn

Co-Chairs of the Supervisory Committee:
Professor Steve M. Seitz
Computer Science & Engineering

Assistant Professor Ira Kemelmacher-Shlizerman
Computer Science & Engineering

How can Tom Hanks come across so differently in "Forrest Gump" and "Catch Me If You Can"? What makes him unique in each of his roles? Is it his appearance? The way he talks? The way he moves? In my thesis, I introduce the problem of persona reconstruction. I define it as a modeling process that accurately represents the likeness of a person, and propose solutions to address the problem with the goal of creating a model that looks, talks, and acts like the recorded person. The specific aspects of persona modelled in this thesis include facial shape and appearance, motion and expression dynamics, the aging process, the speaking style and how a person talks through solving the visual speech synthesis problem.

These goals are highly ambitious. The key idea of this thesis is that the utilization of a large amount of unconstrained data enables overcoming many of the challenges. Unlike most traditional modeling techniques which require a sophisticated capturing process, my solutions to these tasks operate only on unconstrained data such as an uncalibrated personal photo and video collection, and thus can be scaled to virtually anyone, even historical figures, with minimal efforts.

In particular, I first propose new techniques to reconstruct time-varying facial geometry equipped with expression-dependent texture that captures even minute shape variations such

as wrinkles and creases using a combination of uncalibrated photometric stereo, novel 3D optical flow, dense pixel-level face alignment, and frequency-based image blending. Then I demonstrate a way to drive or animate the reconstructed model with a source video of another actor by transferring the expression dynamics while preserving the likeness of the person. Together these techniques represent the first system that allows reconstruction of a controllable 3D model of any person from just a photo collection.

Next, I model facial changes due to aging by learning the aging transformation from unstructured Internet photos using a novel illumination-subspace matching technique. Then I apply such a transformation in an application that takes as input a photograph of a child and produces a series of age-progressed outputs between 1 and 80 years of age. The proposed technique establishes a new state of the art for the most difficult aging case of babies to adults. This is demonstrated by an extensive evaluation of age progression techniques in the literature.

Finally, I model how a person talks via a system that can synthesize a realistic video of a person speaking given just an input audio. Unlike prior work which requires a carefully constructed speech database from many individuals, my solution solves the video speech problem by requiring only existing video footage of a single person. Specifically, it focuses on a single person (Barack Obama) and relies on an LSTM-based recurrent neural network trained on Obama's footage to synthesize a high-quality mouth video of him speaking. My approach generates compelling and believable videos from audio that enable a range of important applications such as lip-reading for hearing-impaired people, video bandwidth reduction, and creating digital humans which are central to entertainment applications like special effects in films.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

I wish to thank Steve Seitz and Ira Kemelmacher for their guidance and support throughout my graduate studies. Their perspectives on problems and directions were truly insightful and inspirational. I have learned a great deal not only through their advising but through examples that they set, and I'm deeply grateful for the opportunity they had given me. I wish to thank my committee members: Rick Szeliski for his valuable feedback and technical advice which led to many new ideas and solutions, and Duane Storti for his great support and attentiveness on my progress and career. I am also thankful for my mentors during two different internships at Google who made the experience very memorable and valuable for my future career path: Carlos Hernandez, Sameer Agarwal, Ce Liu, Michael Rubenstein, Bill Freeman. I would like to thank Noah Snavely who first excited me with his inspiring work and computer vision class back when I was an undergrad at Cornell.

I wish to thank numerous colleagues, former students, postdoctoral fellows who made my time here very enjoyable. I've also learned many things from these people which always helped spark new ideas – Ricardo Martin, Kanit Wongsuphasawat, Aditya Sankar, Kathleen Tuite, Qi Shan, Rahul Banerjee, Aleksander Holynski, Xuan Luo, Edward Zhang, Roy Or-el, Jeong Joon Park, Keunhong Park, Isaac Tian, Chung-Yi Weng, Min Sun, Richard Newcombe, Konstantinos Rematas, Chris Sweeney, Eric Kuo, and Fisher Yu.

This thesis wouldn't be possible without my family's love and sacrifice – my father, mother, and brother. Last but not least, I would like thank Pasita Chaijaroen for her love and support throughout my time at UW.

Chapter 1

INTRODUCTION

When we think of someone we know, the first thing that comes to mind tends to be the person's face. The face is arguably the strongest cue we use to identify and differentiate a person from others. Yet, a single person may appear very different under different circumstances. For example, Tom Hanks in the movie "Forrest Gump" comes across as a simple man with a slow, absent-minded, yet heart-warming persona, whereas in "Catch Me If You Can," he appears as a serious FBI detective with a smart, solemn, and determined persona. Can we capture and recreate these different aspects of a person? Moreover, can we capture anyone's *persona* just by analyzing their casual, day-to-day photos and videos of themselves such as those posted on social networks? In the case of Tom Hanks, can we reconstruct his different personas by analyzing just video frames from his movies?

Let us start by defining "persona reconstruction." The idea is to build a model that accurately represents the likeness of a person, i.e., their visual appearances such as face, hair, skin, and expressions, as well as their behaviors and mannerisms such as how they talk, react, and interact in different situations. In this thesis, I propose solutions for capturing specific aspects of persona by 1) building a 3D face model of a person that captures their shape and appearances, 2) modeling facial expressions and their associated motions from videos, and using them to drive a model for facial puppetry, 3) modeling facial changes due to aging and applying such a transformation to "age-progress" any photo, and 4) learning how to map a person's voice to mouth video and face motion to create realistic speech videos.

Reconstructing a model that looks, talks, and acts like the person has only been done in highly calibrated settings such as production studios. For example, in the movie "Benjamin Button," a model of Brad Pitt was captured using an array of hundreds of synchronized

flashing lights and cameras [3, 7]. Even in this setup, the model captured Brad Pitt only in one personality dimension, i.e., acting for a particular role. Capturing the multidimensionality of persona requires an integration of various components across many areas such as computer graphics, computer vision, natural language processing, and artificial intelligence. More importantly, one missing capability that holds the key to learning those many aspects is the ability to utilize large amount of unconstrained data. It is only recently that such techniques, including those proposed in this thesis, emerge and produce promising and practical results.

Figure 1.1: My thesis proposes techniques for capturing persona that operate on unconstrained data such as this photo collection of President Barack Obama as returned by Google Image Search.

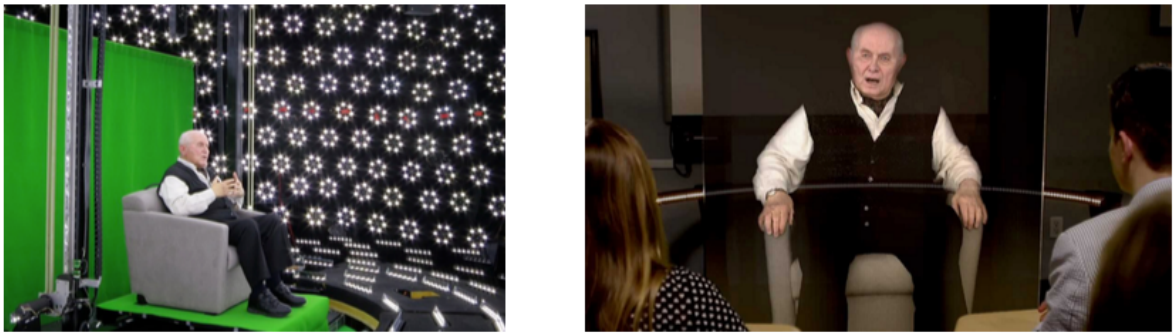
Unlike in most traditional modeling techniques which require a sophisticated capturing process, all of my proposed tasks will be achieved by operating only on unconstrained data such as a general personal photo and video collection or a photo collection of a celebrity as returned by a search engine (Figure 1.1). For these modeling tasks, unconstrained data is particularly well suited and has many appeals over controlled or carefully constructed laboratory datasets: 1) Unconstrained data can be easily gathered in a large amount and its content is constantly growing thanks to the popularity of digital photography and the Internet as a sharing and storage medium. As a result, it provides a rich source for learning

a wide variety of aspects which are challenging to obtain traditionally in a lab such as how we react in various situations or how we age through the years. 2) The ability to operate on just unconstrained data allows the method to scale to anyone with minimal efforts, such as those we do not physically have access to or historical figures.

Figure 1.2: Snavely et al.'s 3D reconstructions of landmarks from Internet photo collections using a Structure-from-Motion technique. The goal of this work is to provide the experience of being at those places.

The idea of leveraging vast unconstrained photo collections to better make sense of things around us has been pioneered in Snavely et al.'s Photo Tourism [153] where thousands of tourists' photographs of a certain landmark are combined to create an intuitive 3D visualization with the ultimate goal of providing the experience of being there (Figure 1.2). Analogous to the goal of being at a place, this thesis is about building a model of a person, out of all available imagery, that can act and talk just like them and gives us the feeling and experience of knowing them as a person. Unlike modeling places, reconstructing even basic aspects of persona such as facial shape is highly challenging due to its dynamics and the large variability in expression.

Another inspiration is a project called "New Dimensions in Testimony" [167] where a



(a) (b)

Figure 1.3: a) 3D capturing and recording of a Holocaust survivor answering predetermined questions. b) An interactive visualization of the survivor that can answer questions from users.

WWII holocaust survivor was recorded in a capture studio answering thousands of predetermined questions related to the event (Figure 1.3). Later on, a user can ask any questions to a computer visualization of him and listen to his answers. Even though the system joins together and replays pre-recorded videos related to the questions being asked, it provides one of a kind experience that allows the user to feel as if they are having a conversation with the survivor. Taking this further, a new set of uncalibrated techniques in this thesis would provide the groundwork for *automatically creating such an experience of anyone from their existing footage*. The survivor's responses given as video playbacks could also be replaced with a more flexible form of visualization such as a realistic 3D avatar that can interactively talk and express itself freely while exhibiting the likeness and persona of him. Apart from building digital avatars which finds wide applications in the movie and entertainment industries, the ability to learn persona of many people at scale could lead to a better understanding of human qualities and could one day give human qualities to robots or virtual assistants to make them more personal and emotionally connected.

1.1 Contributions

This thesis addresses the problem of capturing and modeling persona of a person from unconstrained data. In particular, my contributions include 1) reconstruction of time-varying 3D geometry using uncalibrated photometric stereo and 3D optical flow, 2) synthesis of expression-dependent texture using dense alignment and frequency-based blending, 3) a technique to drive a reconstructed model for facial puppetry by transferring facial motion from an input video, 4) a technique that generates age-progressed results by learning the transformation from Internet photos, 5) visual speech synthesis that generates a lip sync video given an input audio using a recurrent neural network. Each contribution will be described next.

1.1.1 Reconstructing Time-Varying 3D Geometry

Reconstructing 3D geometry of a face from unconstrained data is highly challenging due to the non-rigid nature of human face. Many traditional 3D reconstruction techniques reconstruct a single 3D model of a scene or an object by analyzing multiple photos from different viewpoints [147]. However, general photo collections contain photos that are taken at different moments with varying facial expressions and thus cannot be represented with a single model. In fact, it requires a different model for each and every photo. Existing face reconstruction techniques that operate on a single photo rely on a linear combination of blend shapes [28] which limits the expressiveness and the ability to capture fine details. Other techniques such as non-rigid structure from motion [69] can only operate on video sequences with a sufficient motion.

Rather than solving for a model for every photo in isolation, my solution leverages a vast photo collection of a person to serve as prior and help reconstruct any expressions in any given photos or video frames. In particular, it first applies an unconstrained photometric stereo technique as pioneered by Basri et al.[18] and Kemelmacher et al. [95] which exploits the shading cues to reconstruct an initial average 3D model via a factorization. This average

Figure 1.4: Given a YouTube video of a person's face our method estimates time-varying high detail geometry in each video frame completely automatically.

will serve as a base shape that will be deformed to match other expressions of any given photos. To do this, an accurate alignment of the base shape to an input photo is first needed. In Chapter 2, I propose an accurate pose estimation technique based on optical flow and one key property of the base shape, i.e., it can be relit to match the illumination of a reference image. After aligning the pose, I propose a novel 3D optical flow technique used to transform each vertex of the base shape so that its relit rendering looks as close as possible to the input photo. Finally, facial details such as wrinkles are reconstructed using a shading refinement technique similar to Kemelmacher et al. [93].

The output is a high detailed time-varying 3D mesh that captures even minute shape variations (e.g., dimples, wrinkles, and pimples.) over the video sequence or in a photo (Figure 1.4). I show results on a variety of video sequences that include various lighting conditions, head poses, and facial expressions in Chapter 2. This ability enables a reconstruction of a person's face without requiring the person to participate in a training or capturing process, unlike prior work, and produces accurate reconstruction even under degenerate motions, e.g., when there is no head rotation, that foil nonrigid structure-from motion methods [69].

1.1.2 Synthesizing Expression-Dependent Textures

In addition to the reconstructed geometry described in Chapter 2, we need to recover the color or the "texture" of the reconstructed model which captures the appearance such as the skin and eye color. The goal of this part is to synthesize a sharp, dynamic texture that changes according to the expression of the reconstructed model. A standard approach to recover, say a smile texture, would be to compute an average of smiling photos that are warped and aligned to some canonical view based on a facial tracker (e.g., [188]). Alternatively, we can compute an average of all photos weighted by how "smiling" each photo is to obtain a smooth interpolation between expressions. However, these approaches yield a very blurry texture with inconsistent color due to the large variations in pose, expression, and illumination in the photo collection.

Chapter 3 addresses these issues and recovers facial textures that are 1) sharp and detailed, 2) expression-dependent, and 3) consistent in the overall color shown in Figure 1.5. The main ideas are first to perform a pixel-level alignment of face images using a sparse thin plate splines followed a novel dense warping technique based on relightable average and optical flow. Then, image averages are taken separately across different levels of image frequencies, i.e., those constructed from a Laplacian pyramid. These weights are spatially- and frequency-varying and are designed such that the color of the texture, a low-frequency component, appear consistent across all expressions while enhancing and preserving high-frequency details specific to the target expression.

Together with the time-varying geometric reconstruction, this process represents the first system capable of building a dynamic, textured model automatically from large photo collections. Such a model can be rendered in any expressions from any angles with realistic appearance changes such as creases and wrinkles (Figure 1.6).

Figure 1.5: Our texture synthesis produces consistent, sharp textures with expression-dependent details according to the expressions in the reference images. Note the consistency in color in columns 2, 4, 6.

1.1.3 Facial Puppetry

To use the reconstructed model as a controllable CG character, we also need the ability to drive or animate the model. While prior approaches can reconstruct and transfer expressions of an actor to a CG character with high fidelity [66, 41] they rely on a sophisticated capturing process with multiple cameras, controlled lighting [55], or facial markers [79] which are inapplicable in our setting where we do not have access to the actor for training purposes. Automatic methods for expression transfer such as those that rely on multilinear models created from 3D scans [29, 178], or a direct deformation transfer [154, 179, 99] either account for only large scale deformations or do not handle texture changes on the puppet. Unlike existing work that transfers wrinkles from another person [115], our goal is to transfer

Figure 1.6: A consistent textured model of Daniel Craig rendered at the same poses and expressions as reference photos in the top row.

expressions from an actor to a puppet while preserving the puppet's appearance.

Chapter 3 proposes a way to realistically drive the reconstructed model using only an input video of the actor or of other people for "facial puppetry." My solution first finds correspondence between the driver input video and an illumination-matched "puppet" model by utilizing the relightable averages. The expression of the driver represented as a deformation field is then globally denoised and transferred to the puppet while taking into account the difference in facial features of the driver and puppet. Finally, by independently learning puppet's appearance from their photo collection using the novel synthesis, the final texture of the puppet retains puppet-specific qualities such as their wrinkles and creases even though the model is being driven by another person.

This ability to construct and drive a puppet of anyone given only their photo collections opens up many applications such as creating personal CG characters for movies and video games [8] and driving the facial motion of an animated character in Virtual Reality applications, e.g. AltspaceVR [2]. I demonstrate an example of facial puppetry on eight different celebrities driven by George W. Bush's interview video in Figure 1.7.

Figure 1.7: 3D textured models of 8 different celebrities being driven by a YouTube video of George W. Bush.

1.1.4 Age Progression

Modeling facial changes due to aging is challenging for a number of reasons. First, the aging process is non-deterministic, depending on environmental and genetic factors that may not be evident in the input photos. Second, there is little high-quality training data that spans over many years of individuals from which to build effective models. To solve this problem, in Chapter 4, we propose an illumination-aware age progression technique that learns the aging transformation from Internet photos and can be applied to any photos under any poses and lighting conditions to produce an age-progressed result. The key components include a new database consisting of thousands of face photos collected from the Internet clustered by ages, and a novel technique that utilizes relightable clusters' averages to learn the transformations between clusters, i.e., how shape and appearance changes across age groups. By applying the transformations to a photograph of a child, we can automatically produce a series of age-progressed outputs (Figure 1.8).

One of the most difficult cases of age progression is for very young children where facial structure changes significantly. Using the proposed method, I show the first compelling

Figure 1.8: Given a single input photo of a child (far left) our method renders an image at any future age range between 1 and 80. Note the change in shape (e.g., nose gets longer, eyes narrow) and texture, while preserving the identity of the input person.

results for aging babies to adults, establishing a new state-of-the-art in this category as demonstrated by comprehensive evaluations against prior work and ground-truth photos. Such a tool could be invaluable for solving missing children cases.

1.1.5 Visual Speech Synthesis

So far, we have modelled the basic building blocks of persona such as the facial shape, appearances, and expressions of a person from their photos and videos. In the next step, we're interested in modeling higher level components such as behaviors or how and when certain expressions and actions are performed. In particular, I first focus on the visual speech synthesis problem: given an audio track of a person speaking, synthesize a talking video with an accurate lip sync that captures the person's style of speaking. The problem of generating realistic mouth video from audio is highly difficult due to the fact that humans are extremely attuned to subtle details in the mouth region. Learning to perform this task from general unlabelled, unconstrained videos adds yet another level of complexity and has never been attempted before in the literature. Prior work that synthesizes mouth from audio requires a speech database constructed manually from video recordings of a number of subjects speaking predetermined sentences [119, 60]. And many attempts at rendering talking head produce results that look uncanny with various types of artifacts on the mouth including flickering,

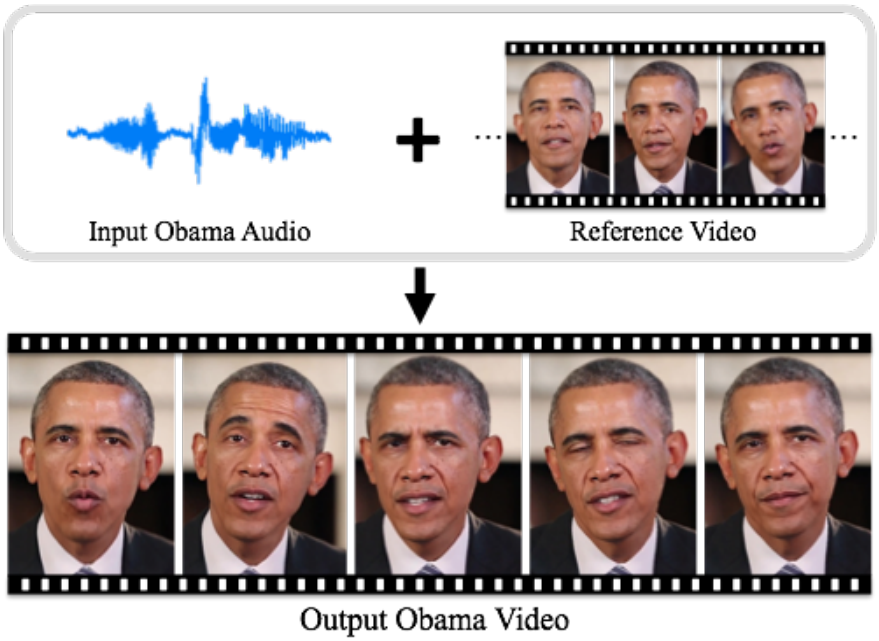


Figure 1.9: Given input Obama audio and a reference video, I synthesize photo-realistic, lip-synced video of Obama speaking those words.

ghosting, and blurry or non-rigid teeth [14, 60].

In Chapter 5, I propose a solution to synthesize a realistic high-quality video of a person speaking from input audio with accurate lip sync. The technique is demonstrated with a case study on President Barack Obama. The solution represents the first attempt to solve the audio speech to video speech problem by analyzing a large corpus of existing video data of a single person. In particular, it combines computer graphics techniques and an LSTM-based recurrent neural network trained on 14-hour of Obama footage to synthesize a high-quality mouth video of him speaking. The mouth video is then composited into a reference video that is retimed to match the flow and pauses of the given speech. The final output is a novel realistic video of Obama talking synced with the input audio (Figure 1.9).

The ability to generate high quality video just from audio has a wide range of practical applications. For example, video synthesis could significantly reduce the amount of bandwidth needed in video call transmission, enable lip-reading from over-the-phone audio

for hearing-impaired people, and replacing or complementing lip tracking used in motion capture.

The rest of the thesis is structured as follows. Chapter 2 describes the algorithm to reconstruct time-varying geometry from unconstrained photo collections. Chapter 3 describes the expression-dependent texture synthesis algorithm as well as the facial mapping technique used to drive the reconstructed model. Chapter 4 describes the age progression algorithm. Chapter 5 describes the visual speech synthesis system. Chapter 6 concludes and discusses several directions for future work.

Chapter 2

TIME-VARYING FACIAL RECONSTRUCTION

Figure 2.1: Given a YouTube video of a person's face our method estimates high detail geometry (full 3D flow and pose) in each video frame completely automatically.

This chapter is based on "Total Moving Face Reconstruction" [157] published in the proceedings of the European Conference on Computer Vision 2014. The project webpage can be found at `http://grail.cs.washington.edu/projects/totalmoving/` with video results `https://www.youtube.com/watch?v=C1iLVAUiC7s`.

Reconstructing the time-varying geometry of a person's face from a video is extremely challenging. Indeed, the highly nonrigid nature of the human face, coupled with our ability to discern even minute facial details and geometry flaws, make it very difficult to achieve high quality results. Operating on free-form video captured "in the wild" adds another level of complexity; only a handful of such results have been reported in the literature

[35, 33, 69, 53, 43].

Rather than reconstruct the input video in isolation, suppose that we had access to a large collection of other photos of the same person captured at different times, with varying pose, expression and lighting. Indeed, most people are captured in numerous photos and videos over their lifetimes; we propose to leverage the *total corpus of available imagery* of the same person to help reconstruct his/her face in an input video. We call this problem *total* moving face reconstruction.

Virtually all modern 3D face tracking and video reconstruction approaches leverage an assumption that the human face is well represented by a linear combination of *blend shapes*, e.g., Morphable models [28, 29, 178], AAMs [59, 52], and Nonrigid Sfm [35, 33, 69, 53]. The advantage of the blend-shape model is that it makes the problem more constrained, as the number of parameters (blend shapes and/or coefficients) is less than the number of measurements (pixels in the video). The main disadvantage is the low-rank model limits expressiveness and the ability to capture fine details.

Instead, our approach is based on deriving a person-specific face model (from all available imagery), and fitting it to each image in the video using a novel 3D optical flow approach coupled with shading cues. The combination of flow and shading enables capturing even minute shape variations (e.g., dimples, wrinkles, pimples, etc.) over the sequence.

We leverage the corpus of images to compute a person-specific face model that captures both the average 3D shape and the illumination-dependent appearance subspace. One key property of this model is that it enables appearance matching of any new image, and solving for dense correspondence via a 3D optical flow approach, yielding more precise alignment and robust 3D tracking than are possible by matching sparse fiducials, e.g., [43]. Another key property is that our use of previously captured photos enables accurate reconstruction even under degenerate motions (e.g., no head rotation) that foil nonrigid structure-from-motion methods [35, 33, 69, 53]. Finally, we incorporate shading cues to obtain higher resolution details than are possible to capture with any other method.

2.1 Related Work

High quality time-varying 3D face geometry capture is extremely challenging due to highly non rigid nature of the human face–ultimately we would like to capture wrinkles, eye and muscle movement, dimples, detailed mouth expressions, eye lid details, and so forth. All these together form our perception of a person's face and are highly important for further face analysis.

Early methods in 3D facial performance capture use marker-based motion capture systems, e.g., [79], that track a sparse set of markers on a person's face. This requires the person to spend hours in a lab, and tracks only a sparse set of points. In contrast, modern high detail reconstruction methods use multi-view stereo approaches on input coming from multiple high resolution synchronized cameras which does not require markers but assumes calibration of the cameras and controlled lighting [20, 21, 30] or uncontrolled lighting[183]. Structured light [194] and light stages [40, 11, 13, 73] provide the ability to use multiple synchronized and calibrated lights for reconstruction.

Recently, RGBD cameras were proven to be extremely successful in face and expression tracking [29, 178, 108]. The idea is to fit raw depth camera output to a deformable facial expression model (blend shapes) created by an artist for facial expression retargeting, puppeteering, and high quality face tracking. Similarly, [43] showed that it is possible to achieve high quality tracking via 3D regression and fitting to a blend shape model extracted from large number of face shapes captured via kinect fusion method [44]. These methods achieve very impressive face tracking results, however 1) require the person to participate in the training stage or be present in front of a depth camera, and 2) assume that face shapes can be represented by a linear combination of blend shapes. Representing face shapes using linear combinations of laser scans of other people's faces and artist created blend shapes goes back to the classical work by Blanz and Vetter [28] as well as more recent works by [54, 172]. These however only enable capture of large scale deformations and tend to miss the fine details (wrinkles, dimples, etc.) that distinguish individuals.

Non-rigid structure from motion methods enable reconstruction from a single video by creating a linear basis for the non rigid motion that appears in the particular video; correspondence between the frames is typically given [35, 33, 53] or estimated via optical flow [69]. The major drawback of these methods is that the basis is extracted from the video itself which not only limits the ability to capture fine details, but also requires head pose to change significantly throughout the video to enable basis reconstruction.

Most related to our work are single view methods, particularly [93, 82, 81, 92]. These methods can produce detailed reconstructions, but do not estimate how the scene deforms over time. Similar to scene flow methods [171], we reconstruct a dense 3D flow field; key differences include our illumination invariance model, and that we compute 3D to 2D correspondence rather than 3D to 3D. Furthermore, recent scene flow methods either assume availability of a stereo pair of photos taken in the same rigid configuration (e.g., same expression) [168, 169] or rigid motion throughout the video [17]. The most relevant to our work is [72], who also operate on monocular video and leverage motion and shading cues to reconstruct a moving face model. However, whereas we simply fit a rigid 3D model independently to each frame, their technical approach involves several additional steps including blend-shape coefficient fitting, keyframe selection, feature-point refinement, multi frame optical flow, and temporal shape filtering (we filter only pose, not shape or flow). We believe the success of our much simpler approach stems from our 3D flow model ([72] move mesh vertices only parallel to the image plane), and our use of *all available imagery* to build an illumination-invariant appearance model. Most importantly, their approach requires a prior, lab-captured model of each actor (requiring a stereo rig and manual work), and hence is *not* applicable to videos of celebrities or other content in personal photo and video collections.

In this chapter, we target high detail reconstruction from a single video captured *in the wild*, i.e., under uncontrolled imaging conditions. Instead of requiring the person to be scanned in the lab or participate in the reconstruction process (as many other methods require [72, **?**, 29, 178, 108]), we leverage whatever existing imagery is available online or in personal photo collections. This enables applying our approach on YouTube videos of

celebrities (e.g., video of Prince Charles[1] as in Figure 2.1), for which we produce arguably the best reconstructions to date.

Our approach is based on foundation work of Kemelmacher and Seitz [95] who first apply uncalibrated photometric stereo techniques [191, 18] to the face reconstruction problem. They attempt to reconstruct a single canonical shape of a person's face based on a random personal photo collection. Their method works by first estimating facial poses and warping each face photo to frontal based on a 3D face template. Then a factorization method based on singular value decomposition and Tikhonov regularization is performed on an image matrix where each row represents a vectorized version of an image in the collection. This factorization produces a set of normal vectors which is later integrated to form a 3D surface.

2.2 Overview

Given a video of a person, we seek to reconstruct a moving 3D model of his/her face that captures apparent motion and fine-scale shape details as well as possible. Specifically, we compute a 3D reconstruction that optimally fits both the *image motion* and *shading* in each frame. Because the problem is not fully constrained (we have only one view of the deforming face at each time instant), we leverage *all available imagery* of the person's face (e.g., photos on the Internet or in personal collections) to compute a reference model of that person (Section 2.3), capturing both their average shape and appearance under a subspace of illuminations. The reference model is used to constrain the gross shape of the sought reconstruction.

To compute the 3D facial deformation in each frame, we formulate a novel 3D optical flow problem (Section 2.4.1) that computes dense correspondence between the 3D model and each video frame, and optimally deform the reference mesh to fit. Similarly, to capture wrinkles and other high frequency structures, we introduce a novel approach to deform the reference mesh so that, when rendered, the mesh shading fits the image shading as accurately

[1] http://www.youtube.com/watch?v=s89KEI2AfBU

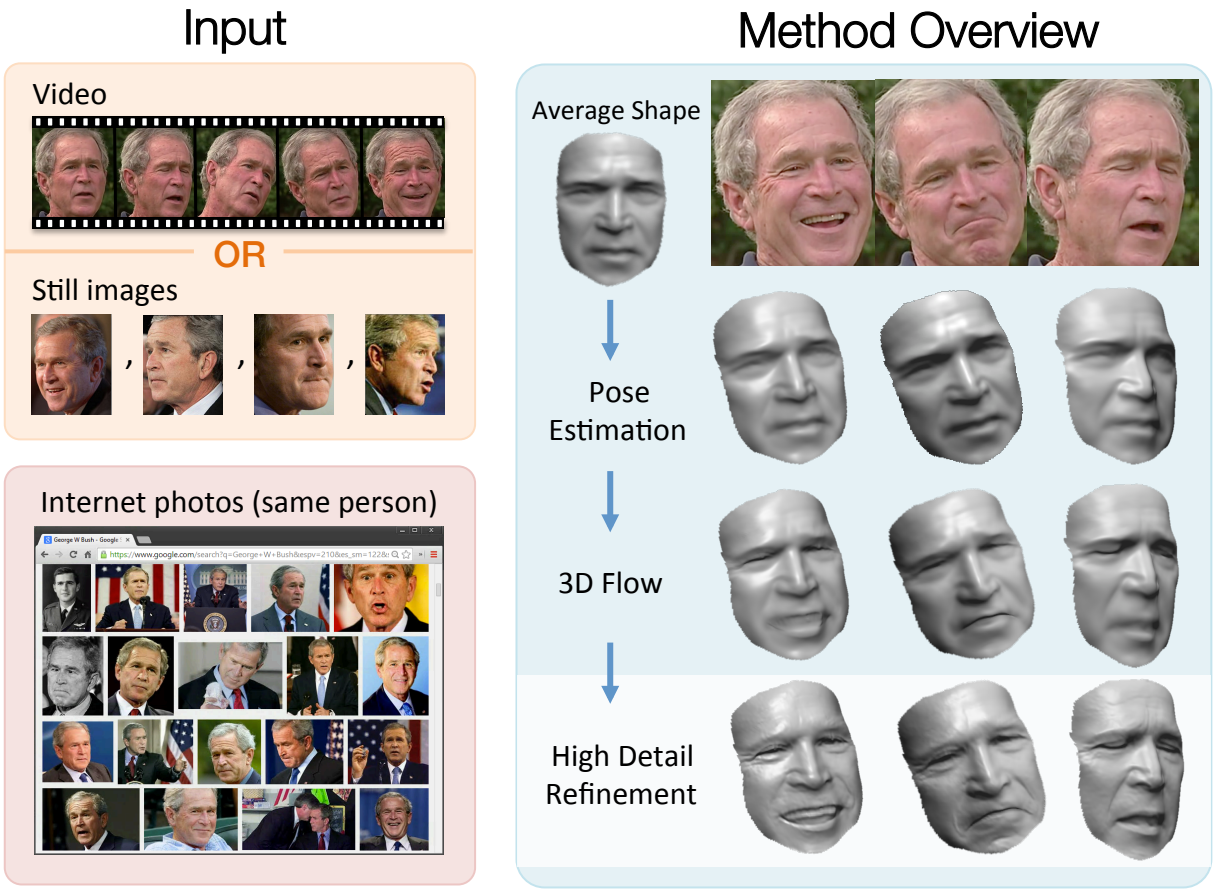


Figure 2.2: Overview of our method. Given a video sequence we estimate 3D pose (average shape is rotated to the input pose for each of the 3 examples), followed by estimate of dense 3D flow of the average model to fit the input expression, and final refinement using shading cues (note the appearance of teeth, details in eyes, and so forth.)

as possible.

We note that our method does not guarantee an accurate fit to ground-truth geometry, as the shape of the face may change in each frame and single-image cues are not sufficient for this purpose. Rather, we seek to produce a reasonably convincing model (leveraging all available imagery) which optimally fits the image information in each frame.

2.3 Reconstructing Average Shape and Appearance

While a person's face shape may be slightly different at each time instant, their rough shape (e.g., distance between eyes, nose length, overall geometry), tends to be consistent over time. Hence, we leverage all available imagery (photos and/or video frames) to reconstruct a shape and appearance model of the person that captures their average shape and appearance under a subspace of illuminations.

In principle, this shape could be acquired in a number of different ways, e.g., a laser scan, kinect fusion model, stereo reconstruction, ohotometric stereo, etc. Given registered or rendered imagery of the same person under many different illuminations, we can construct an illumination subspace by projecting onto the first four singular vectors [18].

In practice, such 3D data with registered imagery is seldom available. Hence, we leverage Kemelmacher et al's Face Reconstruction in the Wild approach [95] to obtain an average shape and appearance model (rank-4 linear basis of the aligned image set). In practice, we find that aligning the images using Collection Flow [96] prior to reconstruction yields slightly sharper reconstructions. We will assume that as a result of this process we have obtained an average shape of the person v_{avg}, texture basis I_{avg}, and initial 3D pose estimate P_0.

2.4 Total Moving Reconstruction

We now describe our approach for reconstructing a moving 3D face shape by deforming an average model to fit the motion and shading cues in each video frame. The face in any given frame may have unknown and possibly changing lighting conditions, arbitrary facial expressions, and varying head orientation (even profile or other highly non-frontal poses are supported–see supplementary video).

Key to our approach is a metric based on *photo consistency*, i.e., comparing mesh renderings with input video frames. This capability depends critically on being able to match the illumination and shading in each input frame to that of the rendered mesh, a property achieved by our appearance subspace representation (Section 2.3). We recover shape in two

steps: first, we deform the average shape to fit the image motion, and second, we deform the resulting shape to fit the shading cues in each frame. We now formulate each problem in turn.

2.4.1 3D Flow Objective

Given an average shape, we seek a 3D flow field mapping it to the reconstructed shape in a given input image (video frame). Denote by $\mathbf{v} := (x, y, z)^\top$ a vertex on the average mesh we wish to deform, and $\vec{f}(\mathbf{v}) \in \mathbb{R}^3$ is the desired per vertex 3D flow (3D displacement to the reconstruction). As the average shape is provided as a depth map $\vec{d}(u, v)$, vertices are connected to form triangle meshes over 4 neighbor pixels in a regular 4-connected grid of the depth map and flow $\vec{f}(\mathbf{v})$ can also be parametrized on 2D image plane as $\vec{f}(u, v) = \vec{f}(u, v, \vec{d}(u, v))$. $I(u, v)$ gives the input image intensity at pixel (u, v), and denote $C(\mathbf{v})$ to be the intensity of vertex $\mathbf{v}$ in the rendering of the average shape from the viewpoint of the input image. Define the camera function as $\mathbb{P} : \mathbb{R}^3 \to \mathbb{R}^2$ which takes a vertex as input and applies a rigid transformation and weak-perspective projection to produce 2D point on the image plane. We therefore cast 3D flow as an optimization problem with the following objective:

$$E_{flow3d}(\vec{f}) = \sum_{\mathbf{v}} |I(\mathbb{P}(\mathbf{v} + \vec{f}(\mathbf{v}))) - C(\mathbf{v})|^2 + \alpha \left(|\nabla \vec{f}_x|^2 + |\nabla \vec{f}_y|^2 + |\nabla \vec{f}_z|^2 \right) \tag{2.1}$$

where $|\nabla \vec{f}_x|^2 = \left(\frac{\partial \vec{f}_x}{\partial u}\right)^2 + \left(\frac{\partial \vec{f}_x}{\partial v}\right)^2$ is the gradient magnitude of the x component of flow parametrized on 2D image plane and $|\nabla \vec{f}_y|^2, |\nabla \vec{f}_z|^2$ along y and z and are defined similarly. $\alpha > 0$ is the smoothness weight that serves as a regularization parameter. We will describe how to optimize this function shortly.

2.4.2 Shape-from-Shading Objective

Applying the estimated 3D flow field $\vec{f}$ yields a new mesh $\mathbf{v}' = \mathbf{v} + \vec{f}$ that deforms the average shape to match the input image. While the resulting reconstruction captures dense nonrigid correspondence, it does not model the impact of the deformation on surface normals and their resulting shading effects. Hence, we introduce a second step to optimize the reconstruction to best fit the *shading* of the input image, by iteratively deforming the mesh vertices and re-rendering.

Specifically, we optimize for new z-coordinate $z(\mathbf{v}')$ of each vertex $\mathbf{v}'$ by minimizing the sum of photometric and position error terms:

$$E_{shading}(z) = \sum_{\mathbf{v}'} |I\left(\mathbb{P}(\mathbf{v}')\right) - \vec{l}^{\top}\vec{h}_{\mathbf{v}'}\left(z(\mathbf{v}')\right)|^2 + \beta|z(\mathbf{v}') - \mathbf{v}'_z|^2 \tag{2.2}$$

$\mathbf{v}'_z$ is the original z-coordinate of $\mathbf{v}'$ after 3D flow, $\vec{h}_{\mathbf{v}'}$ is a 4D spherical harmonics approximation to surface reflectance at new vertex mesh $(\mathbf{v}'_x, \mathbf{v}'_y, z(\mathbf{v}'))$ and $\vec{l}$ is a 4D vector of spherical harmonics coefficients. β is a regularization weight for the second position error term that constrains final z to be close to the original shape. We describe in detail each of the optimization steps in the following subsections.

2.4.3 Optimization

We now describe our optimization approach for computing 3D flow and shading-based mesh refinement. Our approach requires an initial estimate of 3D head pose and lighting (described in Section 2.4.6).

2.4.4 3D Flow estimation

Minimizing Eq. 2.1 is a non-linear optimization task even if we assume weak-perspective projection with L2 norm because $I\left(\mathbb{P}(\mathbf{v} + \vec{f}(\mathbf{v}))\right)$ is generally non-linear in the image coordinate. To optimize this objective, we use Levenberg-Marquardt (LM) implemented in

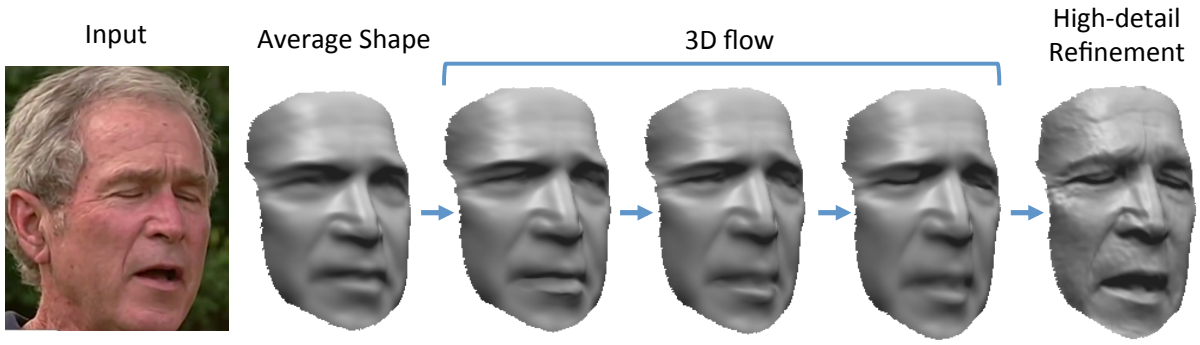


Figure 2.3: 3D flow convergence example. The optimization starts from an average model of Bush with closed mouth, the mouth opens with 3D flow estimation iterations and gets refined at the shading step. This computation is done independently for each single frame in the video (temporal constraint is applied only at the rigid pose estimation step).

the Ceres Solver [10]. This requires a calculation of the Jacobian matrix in which the variables are x, y, and z for each flow value. To compute the derivatives of $I\left(\mathbb{P}(\mathbf{v}+\vec{f}(\mathbf{v}))\right)$ with respect to each flow component x, y and z, let us denote $\mathbb{P}(\mathbf{v}+\vec{f}(\mathbf{v})) = (u, v)^\top$ and $\vec{f}(\mathbf{v}) = (x, y, z)^\top$. By applying the chain rule with respect to x we get:

$$\frac{\partial}{\partial x} I\left(\mathbb{P}(\mathbf{v}+\vec{f}(\mathbf{v}))\right) = I_u \frac{\partial u}{\partial x} + I_v \frac{\partial v}{\partial x} \tag{2.3}$$

where I_u and I_v denote image derivatives along the horizontal and vertical axis and are computed using the 5-point derivative filter $\frac{1}{12}[-1 \;\; 8 \;\; 0 \;\; -8 \;\; 1]$. Let us further define the camera function as

$$\mathbb{P}(\vec{q}) = \pi(\mathbf{R}_{3\times3}\vec{q} + \mathbf{T}_{3\times1}) \tag{2.4}$$

$$\pi(\vec{r}) = (f \cdot \vec{r}_x/\bar{z}, f \cdot \vec{r}_y/\bar{z})^\top \tag{2.5}$$

where $\mathbf{R}_{3\times3}$ is a rotation matrix and $\mathbf{T}_{3\times1}$ is a translation vector. π is a weak-perspective projection with $\bar{z}$ being the constant average of vertex z-coordinate; $\frac{\partial u}{\partial x}$ and $\frac{\partial u}{\partial y}$ are evaluated

using automatic differentiation. This provides a derivative with respect to x, derivatives with respect to y and z are computed similarly.

To differentiate the smoothness term, we approximate the partial derivatives of $\nabla\vec{f}_x, \nabla\vec{f}_y, \nabla\vec{f}_z$ by forward differences (i.e., re-parametrize flow on 2D image plane $\frac{\partial\vec{f}_x}{\partial u} = \vec{f}_x(u+1,v) - \vec{f}_x(u,v)$, $\frac{\partial\vec{f}_x}{\partial v} = \vec{f}_x(u,v+1) - \vec{f}_x(u,v)$), and then take the derivatives. Similar computation is done for y and z components.

We implement this in a coarse-to-fine multi-resolution scheme [36] to deal with large flow displacements, i.e., we construct a Gaussian pyramid of the input image with down sampling rate of 0.75, and use the output flow in a coarser level as an initialization for the next finer level.

2.4.5 Shading-based refinement

We deform the average mesh to fit the input face according to the estimated 3D flow and use this new mesh as initialization to shading based mesh refinement. The idea is to capture high frequency details, e.g., wrinkles, folds, etc. We assume Lambertian reflectance and use the 1st order spherical harmonics (SH) approximation to Lambertian reflectance [19] to model the relationship between surface normals and image intensities. From Eq. 2.2, we define the SH approximation to surface reflectance at each new vertex $\mathbf{w} = (\mathbf{v}'_x, \mathbf{v}'_y, z)$ as

$$\vec{h}_{\mathbf{v}'} = \left(1,\ \frac{(\mathbf{w}_u - \mathbf{w}) \times (\mathbf{w}_v - \mathbf{w})}{\|(\mathbf{w}_u - \mathbf{w}) \times (\mathbf{w}_v - \mathbf{w})\|}\right)^\top \tag{2.6}$$

where $\mathbf{w}_u$ and $\mathbf{w}_v$ are vertices adjacent to $\mathbf{w}$ in the mesh structure along the positive horizontal and vertical directions. We estimate the SH coefficients $\vec{l}$ by finding the best coefficients that fit the deformed mesh after 3D flow to the input via:

$$\min_{\vec{l}} \sum_{\mathbf{v}'} |I\left(\mathbb{P}(\mathbf{v}')\right) - \vec{l}^\top \vec{h}_{\mathbf{v}'}\left(\mathbf{v}'_z\right)|^2. \tag{2.7}$$

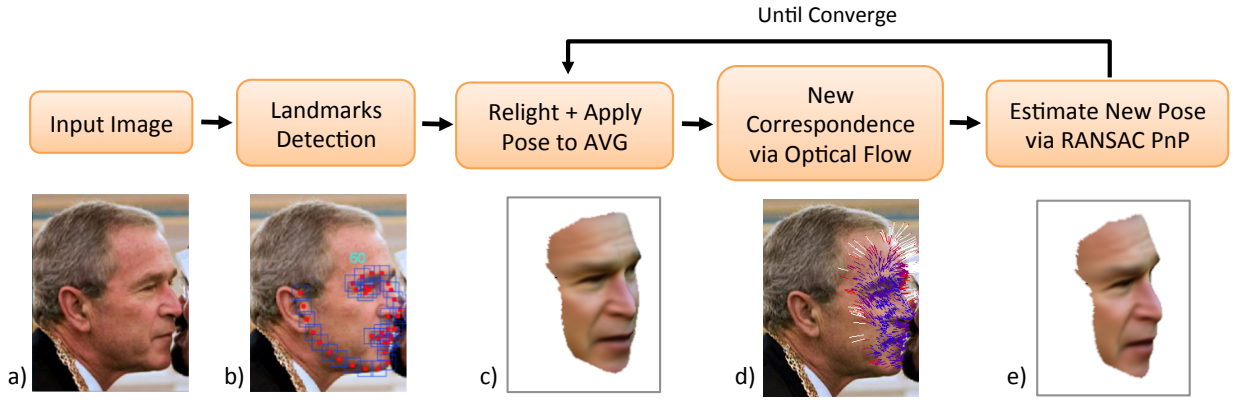


Figure 2.4: Pose refinement algorithm. (a) non-frontal photo–challenging for current methods, (b) landmarks detection and (c) pose estimation using landmarks (slightly off) which is used to initialize our refinement. (d) optical flow matching between an average model rendering in the initial pose and input image. (e) final pose estimation result using PnP on dense point sets chosen via RANSAC.

To finally optimize Eq. 2.2, we pre-compute $I\left(\mathbb{P}(\mathbf{v}')\right)$ and further linearize by precomputing the normalizing factor $\|(\mathbf{w}_u - \mathbf{w}) \times (\mathbf{w}_v - \mathbf{w})\|$ as suggested in [93] using the deformed mesh. The resulting formulation becomes linear in z and solved efficiently using linear least squares optimization.

2.4.6 Pose and Lighting

Faces in input frames/photos may appear in an arbitrary 3D pose, and often in highly non-frontal poses, e.g., 90 degrees out of plane rotation. To estimate 3D flow we first need to compute the 3D rigid transformation $P = [\mathbf{R} \mid \mathbf{T}]$ that takes the average mesh $\vec{v}$ and transforms it to the position of the face in the image. While we obtain an initial estimate from the warping process in Section 2.3, it is performed using a 3D reference model of a different individual (see [95] for more details), thus pose estimation error increases with larger angles of rotation, e.g., due to difference in nose shape across people. We propose the following process (Alg. 1) to recover accurate face pose in a single photo, and we further show how to leverage temporal information in videos to achieve accurate pose estimates. We

Data: $P_0 = P_{ref}$ initialize pose from Sec. 2.3;
I: input image;
A_P^L: rendering of an average shape v_{avg} with texture in pose P and lighting L;
$i = 0$;
Result: 3D pose P
while *until convergence* **do**
    estimate lighting L_i of input I using process described in Sec. 2.3;
    render v_{avg} in pose P_i and input lighting L_i;
    run 2D optical flow between $A_{P_i}^{L_i}$ and I;
    generate 3D-to-2D correspondences from v_{avg} to I through 2D flow ;
    solve PnP using RANSAC on subset of correspondences:;
    solve PnP on all inliers to compute new estimate of pose P_{i+1};
end

Algorithm 1: Out of plane pose estimation in a single photo.

solve the Perspective-n-Point problem (PnP) using OpenCV's implementation of Levenberg-Marquardt [31]. Following the optimization in Alg. 1 we get high quality pose estimates for challenging poses. To achieve temporal coherence across the video, we refine the individual pose estimates using nearby frames. Specifically, we use each frame's 12 neighbors and their corresponding poses for refinement, as follows. We compute bi-directional 2D optical flow between every consecutive pair of frames, then we concatenate them to produce flows between frame j and all its neighbors. Once these flows are available, we project 3D points of v_{avg} onto the image plane using pose estimate of frame $j+1$ then follow 2D flow from frame $j+1$ to j to produce dense 3D-to-2D correspondences between v_{avg} and the image pixels of frame j. Then we solve RANSAC PnP problem as in Alg. 1 to get another pose estimate for frame j. Performing this for all its neighbors will produce 12 additional estimates for frame j which are averaged together using quarternion average for rotations and linear average for translations. While we did not rigorously evaluate our method in comparison to state of the art [188, 197, 143], we have found that our pose estimation is comparable to these methods and gives temporally smooth, drift-free pose estimates, as can be observed from the accompanying videos. This process is completely automatic and the same for all video sequences.

2.5 Experiments

We evaluate the performance of our approach on a variety of videos downloaded from the Internet. Figure 2.8 shows example frames from four different videos (Tom Hanks, George Bush, Arnold Schwarzenegger, and Thaksin Shinawatra) downloaded from YouTube.com[2] and the corresponding per-frame 3D shape reconstructions obtained using our algorithm. On the left of Figure 2.8, we also present the average shapes (that are used to initialize the 3D flow estimation) for each person; these were obtained using [95]. The level of detail in the reconstructions is remarkable; the algorithm succeeds in capturing very fine details such as wrinkles and subtle expressions. Note the change in facial expression (compared to the average shape) in each frame, e.g., mouth opening, eyes close and open, wrinkles appear and disappear, detail in eye region, and so forth. The approach is robust to very large changes in pose, providing high quality results even for profile views (e.g., supplementary video of Tom Hanks). The stability of our results without any temporal smoothing other than pose filtering is evidence for the strength of the photo-based illumination subspace approach. Specifically, the illumination matching process makes the flow more accurate and thus stable. We strongly encourage the readers to watch the accompanying videos where we show per frame reconstructions for full length videos. Specifically, the lengths are: Tom Hanks: 20s (591 frames), George Bush 20s (610 frames), Arnold Schwarzenegger 24s (719frames), and Thaksin Shinawatra 20s (600 frames). Note that unlike non-rigid SfM methods [69], our reconstruction quality is independent of input video length (we can produce good results from even a single frame). And since we estimate pose independently in each frame (and then average) by matching to an illumination-matched reference, the approach is not susceptible to drift problems that plague many tracking methods. We show long and short sequences to illustrate the quality of the reconstruction under a large variety of imaging conditions, non

[2]URLs of input videos:
Hanks: https://www.youtube.com/watch?v=emLpj38huDA
Bush: https://www.youtube.com/watch?v=BJbUXw87j0A
Schwarzenegger: https://www.youtube.com/watch?v=wH8VtPG-okI
Shinawatra: https://www.youtube.com/watch?v=dZdhr1WcYEM

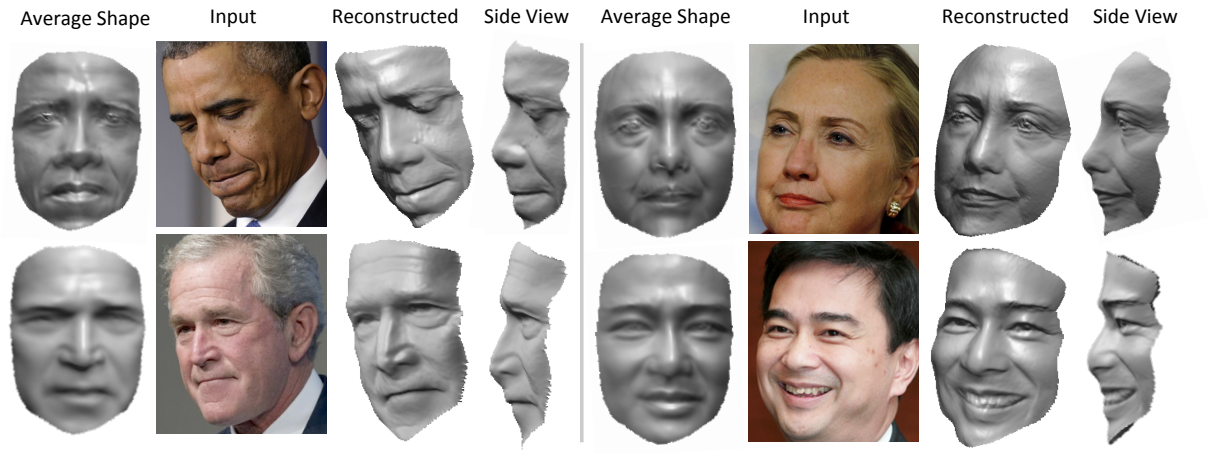


Figure 2.5: Example results on still images in non-frontal views. Single view methods typically fail on such extreme poses.

rigid motion, pose and lighting.

In addition to handling videos, we can also estimate 3D shapes from single still images, and we show a number of results in Figure 2.5. We chose to show photos of faces that appear in highly non-frontal poses, these are typically the hardest cases for any state of the art single view method. The algorithm's ability to handle such extreme poses stems from our use of a person-specific template and appearance model that can be relit to match the input photo. In contrast, most other face tracking methods use generic face models which produce less reliable pose estimates, particularly for non-frontal poses.

Implementation details. We use the Ceres solver [10] for optimization in the 3D flow estimation stage with $\alpha = 0.03$. For pose refinement we used the 2D optical flow code of [113] with the following parameters: α=0.02, ratio=0.75, nOuterFPIterations=4, nSORIterations=40. The regularization weight in shading-based refinement step is $\beta = 2$ for all videos. The running times are 35s for pose estimation (incl. 15s for temporal refinement), 70s for 3D flow, and 0.1s for shading, for a 350×350 frame size (face size 220×260 pixels).

Comparisons. We provide qualitative comparisons to calibrated results captured in the lab using range sensing and multi view stereo. We run our algorithm on data from [21]

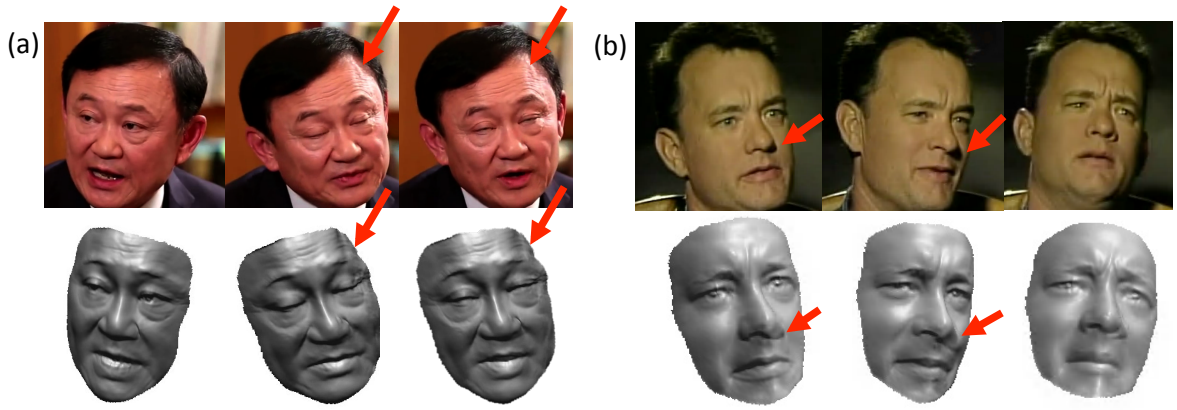


Figure 2.6: Limitations of our reconstruction due to (a) specular highlight (b) cast shadows. We show a few frames from a video where the method introduces artifacts on the forehead in case of specularities or near the nose in case of cast shadows. This is due to violations of the Lambertian assumption. The full video and per frame reconstruction is shown in the accompanying video at 30fps.

and compare their capture with our reconstruction in Figure 2.9, note the resemblance to the model captured by [21] (acquired by stereo setup) and our single view reconstruction. The base shape was acquired using the method described in Sec. 2.3 on 100 renders under different random lightings of frame 390 (neutral expression). The input photos are renderings of frames 80 and 340 in the dataset provided by [21]. We have also compared with Kinect Fusion [125] and present the results in Figure 2.10. The input to our reconstruction is a single frame; to obtain the Kinect Fusion result the person had to stay still while the depth camera captures him from a number of different viewpoints. To preserve consistency we ran our method on the direct RGB stream of kinect camera (lower in resolution than typical videos). We also compare to single view reconstruction methods, see results in the supp. material. The comparisons are qualitative since our method currently does not **guarantee** an accurate fit to ground-truth geometry due to gauge and bas-relief ambiguities. Any monocular uncalibrated approach will have this limitation, unless they assume a 3D model of the person a priori, e.g., [21, 29, 43, 72, 178] or sufficient 3D head rotation [69]. Rather, we seek to produce a reasonably accurate model (leveraging all available imagery) which

optimally fits the image information in each frame. It is our future work to conduct a quantitative evaluation once time-varying 3D datasets exist with the level of detail we are attempting to capture and extend our work to handle shadows and specularities, and account for non-uniform albedo as introduced by earlier work [184].

We compare to single view method in Figure 2.7. Our method (outlined in red) obtains higher quality coarse as well as detailed reconstruction due to estimation of 3D flow coupled with extreme non frontal pose and shading. State of the art single view methods do not account for 3D flow. Additionally, while we estimate an initial template (average shape) of the individual we would like to reconstruct, state of the art single view methods use models of other individuals as the template; this also decreases the quality of the results.

All the examples are viewed best as videos, so we strongly encourage you to watch the supplementary video!

2.6 Discussion

While we found our method to be extremely robust to a variety of lighting conditions, individuals and poses, there are a number of limitations that we would like to discuss. The first are due to the use of spherical harmonics approximation to reflectance modeling, and the Lambertian assumption. In Figure 2.6 we present a number of frames where (a) the person rotates the head and specularities appear on the forehead, and (b) cast shadows appear around the nose area. These are not covered by our reflectance model and therefore the algorithm will produce slightly erroneous results in the specular and shadow vertices.

Our initial base shape model still suffers from the bas-relief ambiguity inherent to unconstrained photometric stereo techniques. It's an interesting future direction to combine photometric stereo with multiview stereo to resolve this ambiguity or estimate the bas-relief parameters from additional side profile photos.

Figure 2.7: A comparison to single view method by Kemelmacher et al. [30].

Figure 2.8: Results of our reconstruction on four video sequences. Average shape per individual are presented on the left. The video reconstruction results illustrate variety in facial expressions, head pose, appearance of wrinkles, eye detail, and even partial teeth. Take a look at the full videos in the supplementary material for the full experience!

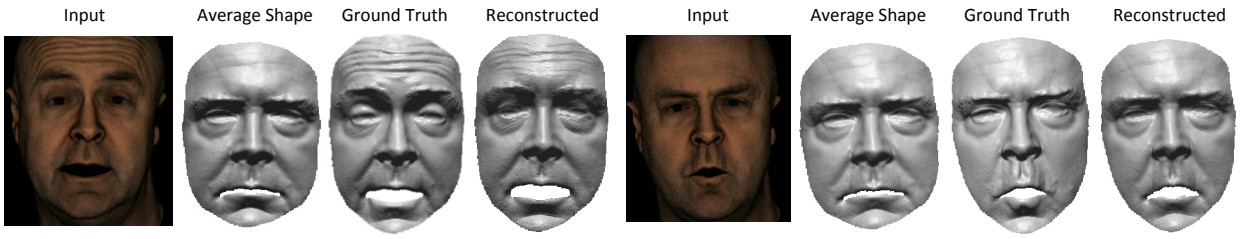


Figure 2.9: Comparison to ground truth meshes [21]. Given the input photo (left) we show our reconstruction and the original shape captured by [21] for this particlar expression.

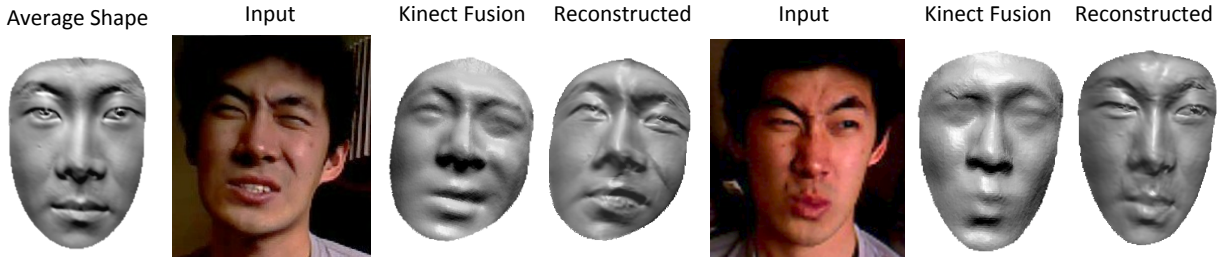


Figure 2.10: Comparison to KinectFusion [125]. Two input photos, our reconstructions and results obtained using Kinect Fusion. The input photos are of lower quality than typical video sequences (taken from RGB kinect stream).

Chapter 3

FACIAL TEXTURE SYNTHESIS & PUPPETRY

Figure 3.1: Model of Tom Hanks (bottom), derived from Internet Photos, is controlled by his own photos or videos of other celebrities (top). The Tom Hanks model captures his appearance and behavior, while mimicking the pose and expression of the controllers.

This chapter is based on "What Makes Tom Hanks Look Like Tom Hanks" [158] published in the proceedings of the International Conference on Computer Vision 2015. The project webpage can be found at `http://grail.cs.washington.edu/projects/3DPersona/` with video results `https://www.youtube.com/watch?v=ladqJQLR2bA`.

In addition to the reconstructed geometry described in the previous chapter, we need to recover the color or the "texture" of the reconstructed model which captures the appearance such as the skin and eye color. We address this with a new technique to recover facial textures that are 1) sharp and detailed, 2) expression-dependent, and 3) consistent in the overall color. And with this complete, textured model, we propose a facial puppetry technique to drive the model of an actor using a source video of another actor. Specifically, we define the following problem:

Input: 1) a photo collection of actor B, and 2) a photo collection and a single video V of actor A

Output: a video V′ of actor B performing the same role as actor A in V, but with B's personality and character.

Figure 3.1 presents example results with Tom Hanks as actor B, and two other celebrities (Daniel Craig and George Bush) as actor A.

The problem of using one face to drive another is a form of *puppetry*, which has been explored in the graphics literature e.g., [154, 179, 99]. The term *avatar* is also used sometimes to denote this concept of a puppet. Making facial puppetry work well is challenging, as we need to determine what aspects are preserved from actor A's performance and actor B's personality. For example, if actor A smiles, should actor B smile in the exact same manner? Or use actor B's own particular brand of smile? After a great deal of experimentation, we obtained surprisingly convincing results using the following simple recipe: use actor B's shape, B's texture, and A's motion (adjusted for the geometry of B's face). Both the shape and texture model are derived from large photo collections of B, and A's motion is estimated using a 3D optical flow technique.

3.1 Related Work

Creating a realistic controllable model of a person's face is challenging due to the high degree of variability in the human face shape and appearance. Moreover, the shape and texture are highly coupled: when a person smiles, the 3D mouth and eye shape changes, and wrinkles and creases appear and disappear which changes the texture of the face.

Most research on avatars focuses on non-human faces [99, 178]. The canonical example is that a person drives an animated character, e.g., a dog, with his/her face. The drivers face can be captured by a webcam or structured light device such as Kinect, the facial expressions are then transferred to a blend shape model that connects the driver and the puppet and then coefficients of the blend shape model are applied to the puppet to create a similar

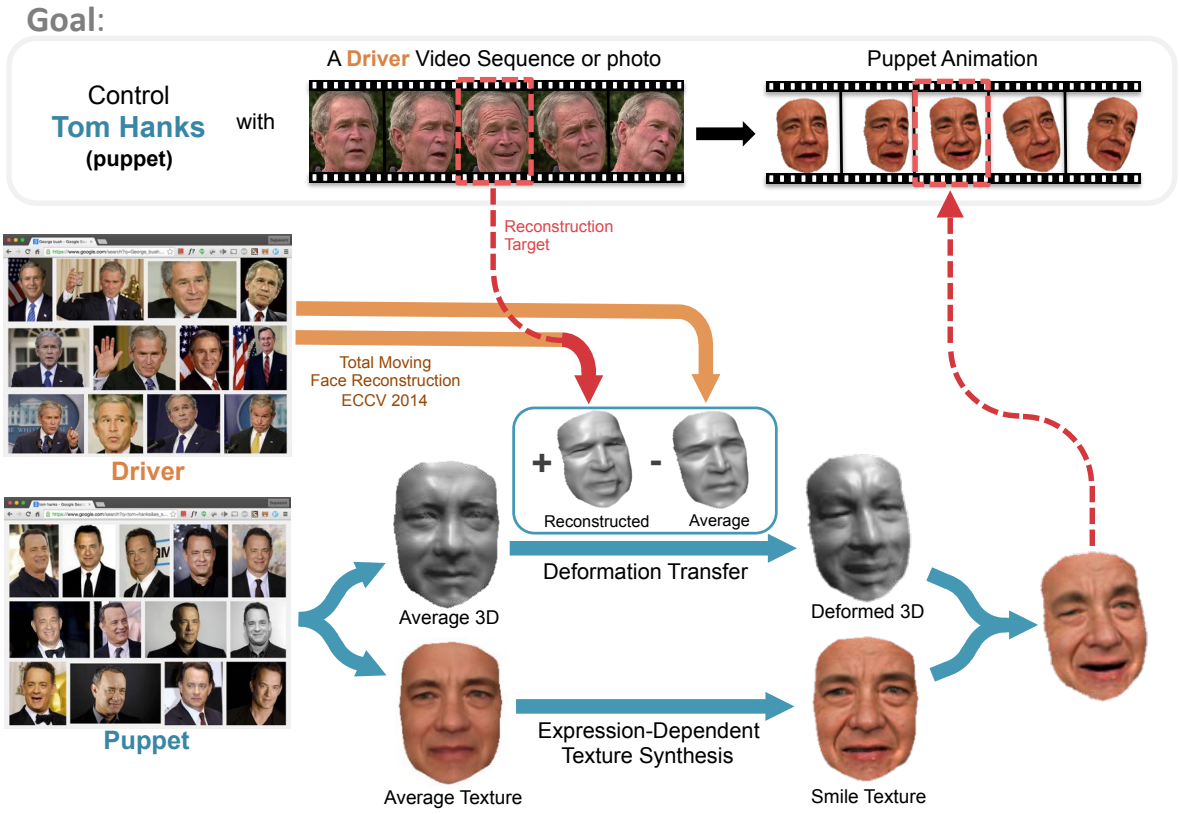


Figure 3.2: Our system aims to create a realistic puppet of any person which can be controlled by a photo or a video sequence. The driver and puppet only require a 2D photo collection. To produce the final textured model, we deform the average 3D shape of the puppet reconstructed from its own photo collection to the target expression by transfering the deformation from the driver. The texture of the final model is created separately for each frame via our texture synthesize process which produces detailed, consistent, and expression-dependent textures.

facial expression. Recent techniques can operate in real-time, with a number of commercial systems now available, e.g., faceshift.com (based on [178]), and Adobe Project Animal [1].

The blend shape model typically captures large scale expression deformations. Capturing fine details remains an open challenge. Some authors have explored alternatives to blend shape models for non-human characters by learning shape transfer functions [193], and dividing the shape transfer to several layers of detail [189, 99].

Creating a model of a real person, however, requires extreme detail. One way of capturing fine details is by having the person participate in lab sessions and use multiple synchronized and calibrated lights and camera rigs [11]. For example, light stages were used for creation of the Benjamin Button movie–to create an avatar of Brad Pitt in an older age [55] Brad Pitt participated in many sessions where his face was captured making expressions according to the Facial Action Coding System [57]. The expressions were later used to create a personalized blend shape model and transferred to an artist created sculpture of an older version of him. This approach produces amazing results, however, requires actor's active participation and takes months to execute.

Automatic methods, for expression transfer, explored multilinear models created from 3D scans [172] or structured light data [44], and transfered differences in expressions of the driver's mesh to the puppet's mesh through direct deformation transfer, e.g., [154, 179, 99], coefficients that represent different face shapes [178, 172], decomposable generative models [106, 177], or driven by speech [46]. These approaches either account only for large scale deformations or do not handle texture changes on the puppet.

This chapter is about creating expression transfer in 3D with high detail models and accounting for expression related texture changes. Change in texture was previously considered by [115] via image based wrinkles transfer using ratio images, where editing of facial expression used only a single photo [190], face swapping [27, 54], reenactment [70], and age progression [98]. These approaches changed a person's appearance by transferring changes in texture from another person, and typically focus on a small range of expressions. Prior work on expression-dependent texture synthesis has been proposed in [116] focusing on skin

deformation due to expressions. Note that our framework is different since it is designed to work on uncalibrated (in the wild) datasets and can synthesize textures with generic modes of variations. Finally, [94] showed that it is possible to create a puppetry effect by simply comparing two youtube videos (of the driver and puppet) and finding similarly looking (based on metrics of [97]) pairs of photos. However, the results simply recalled the best matching frame at each time instance, and did not synthesize continuous motion. In this chapter, we show that it is possible to leverage a completely unconstrained photo collection of the person (e.g., Internet photos) in a simple but highly effective way to create texture changes, applied in 3D.

3.2 Overview

Given a photo collection of the driver and the puppet, our system (Fig. 5.2) first reconstructs rigid 3D models of the driver and the puppet. Next, given a video of the driver, it estimates 3D flow from each video frame to the driver's model. This flow is then transfered onto the model of the puppet creating a sequence of shapes that move like the driver (Sec. 3.3). In the next stage, high detail consistent texture is generated for each frame that accounts for changes in facial expressions (Sec. 3.4).

3.3 3D Dynamic Mesh Creation

By searching for "Tom Hanks" on Google's image search we get a large collection of photos that are captured under various poses, expressions, and lightings. In this section, we describe how we estimate a 3D model of the driver and the puppet, and deform it according to a video or a sequence of photos of the driver. Figure 3.3 illustrates the shape creation process.

3D Average Model Estimation. We begin by detection of face and fiducial points (corners of eyes, mouth, nose) in each photo using [188]. We next align all the faces to a canonical coordinate frame and reconstruct an average rigid 3D shape of person's face. For 3D average shape reconstruction we follow Kemelmacher-Shlizerman and Seitz [95] with the modifica-

tion of non-rigidly aligning photos prior to 3D reconstruction. We describe the non-rigid alignment step in Section 3.4. The same reconstruction pipeline is applied on the driver and the puppet photo collections, resulting in two average 3D rigid models.

Dynamic 3D Model. Next, we create a dynamic model of the puppet that is deformed according to the driver's non-rigid motions. For the driver, we are given a video or sequence of photos. The first step is to reconstruct the 3D flow that deforms the driver's 3D average model to the expression of the driver in every single frame of the input video, using the method of [157]. The geometric transformation is given as a 3D translation field $T : \mathbb{R}^3 \to \mathbb{R}^3$ applied on a driver's average shape.

Given a reconstructed mesh at frame i of a driver $M_D^i(u, v) : \mathbb{R}^2 \to \mathbb{R}^3$ parametrized on an image plane (u, v) from a depth map, and the average mesh over the entire frame sequence $\overline{M}_D$, the goal is to transfer the translation field $M_D^i - \overline{M}_D$ to the puppet's base mesh M_P to produce M_P^i. To transfer the deformation, we first establish correspondence between M_D and M_P through a 2D optical flow algorithm between the puppet's and driver's 2D averages from their photo collections.

The "collection flow" work [96] has shown that we can obtain correspondence between two very different people by projecting one average onto the appearance subspace of the other by this matching illumination, and then run an optical flow algorithm between the resulting projections. With this flow, we can apply the deformation of the driver on the same facial features of the puppet. However, the direct deformation from the driver may not be suitable for the puppet, for example, if their eye sizes are different, the deformation needed to blink will be different. We solve this by scaling the magnitude of the deformation to fit each puppet as follows (Figure 3.3): Let the deformation vector from the driver at vertex $M_D(u, v)$, be $\Delta(u, v)$. We first find the nearest vertex to $M_D(u, v) + \Delta(u, v)$ in euclidean distance on the driver mesh, denoted by $M_D(s, t)$. Through the flow between M_D and M_P we computed earlier, we can establish a corresponding pair $(M_P(u', v'), M_P(s', t'))$ on the puppet mesh. The magnitude-adjusted deformation at $M_P(u', v')$ is then computed

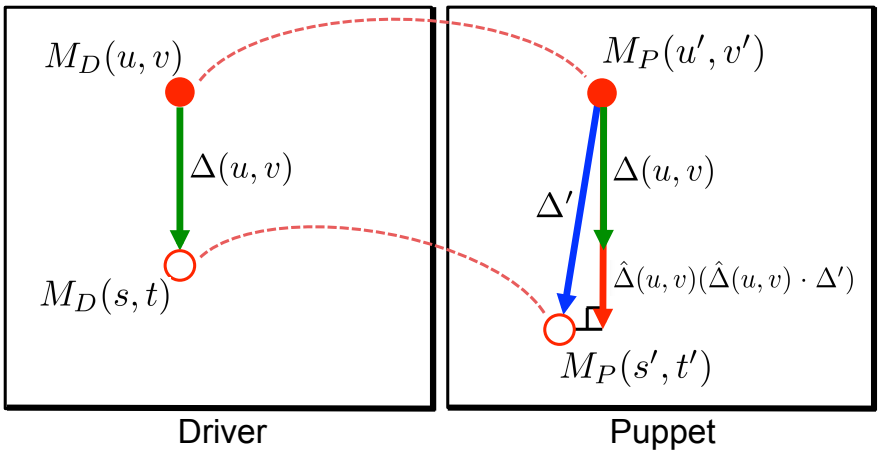


Figure 3.3: Magnitude adjustment in deformation transfer. Let's take an example of a blinking eye, and denote by $M_D(u,v)$ a vertex on a driver's upper eye lid. The vertex is moving down by $\Delta(u,v)$ toward $M_D(s,t)$ in order to blink. Let's denote the corresponding vertex on the puppet mesh $M_P(u',v')$. Our goal is to apply $\Delta(u,v)$ to $M_P(u',v')$, it could happen, however, that the puppet's eyes are bigger, thus we adjust the deformation and instead use $\hat{\Delta}(u,v)(\hat{\Delta}(u,v)\cdot\Delta')$.

by $\hat{\Delta}(u,v)(\hat{\Delta}(u,v)\cdot\Delta')$ where $\hat{\Delta}=\frac{\Delta}{\|\Delta\|}$ and $\Delta'=M_P(s',t')-M_P(u',v')$. In addition, since the flow between the driver and puppet can be noisy around ambiguous, untextured regions, we perform the standard denoising on the term $f(u,v)=(\hat{\Delta}(u,v)\cdot\Delta')$ to obtain a regularized field $f^*(u,v)$. The particular denoising algorithm we use is ROF denoising with the Huber norm and TV regularization. The final puppet's mesh is constructed as $M_P^i(u,v)=M_P(u,v)+\hat{\Delta}(u,v)f^*(u,v)$.

3.4 High detail Dynamic Texture Map Creation

In the previous section, we have described how to create a dynamic mesh of the puppet. This section will focus on creation of a dynamic texture. The ultimate set of texture maps should be consistent over time (no flickering, or color change), have the facial details of the puppet, and change according to the driver's expression, i.e., when the driver is laughing, creases around the mouth and eye wrinkles may appear on the face. For the latter it is particularly

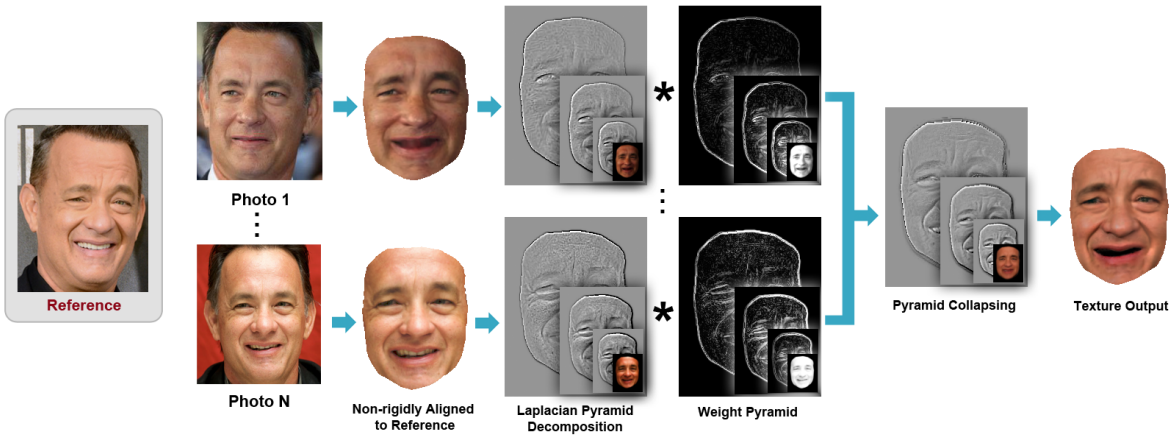


Figure 3.4: A diagram for synthesizing a texture for a given reference photo shown on the left. Each photo is non-rigidly aligned to the reference and decomposed into a Laplacian pyramid (For visualizing purpose, the Laplacian images are shifted by 0.5 so that a gray pixel corresponds to 0.0). The final output shown on the right is produced by computing a weighted average pyramid of all the pyramids and collapsing it.

important to account for the puppet's identity–some people may have wrinkles while others won't. Thus, a naive solution of copying the expression detail from the driver's face will generally not look realistic. Instead, we leverage a large unconstrained photo collection of the puppet's face. The key intuition is that to create a texture map of a smile, we can find many more smiles of the person in the collection. While these smiles are captured under different pose, lighting, white balance, etc. they have a common high detail that can be transfered to a new texture map.

Our method works as follows. Given the target expression which is either the configuration of fiducials on the driver's face (that represents e.g., a rough shape of a smile) or by a reference photo if the driver is the same person as the puppet, we first warp all the photos in the puppet's collection to the given expression. We then create a multi-scale weighted average that perserves a uniform illumination represented by the lower frequency bands and enhances details in the higher frequency bands. Next we explain each of these in more detail.

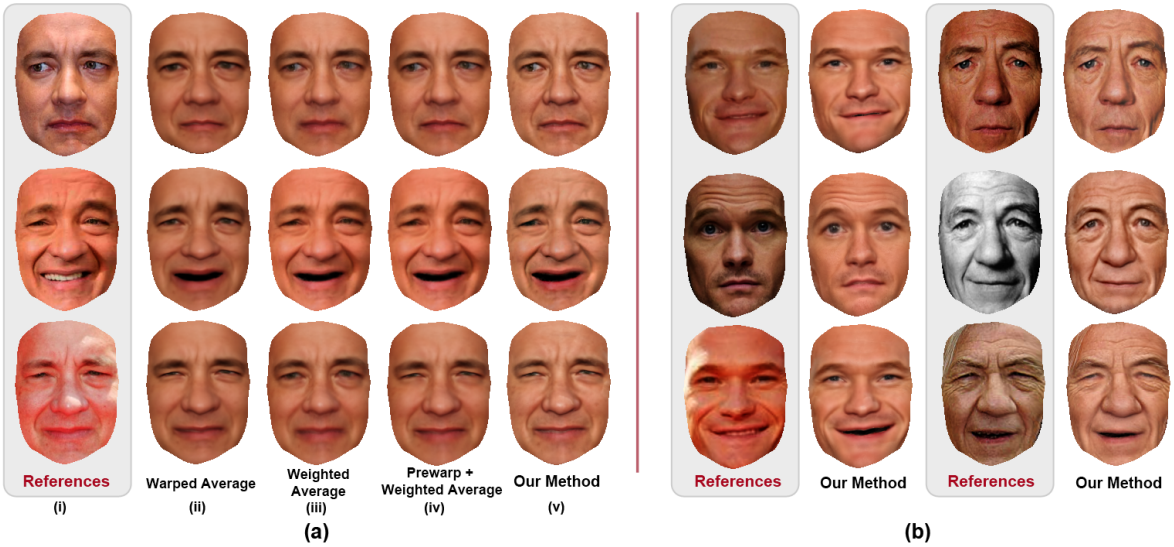


Figure 3.5: a) A comparison between our method (column v) and 3 baseline methods (columns ii-iv) to produce a texture that matches the target expressions given in the column i. Baseline results in column (ii) are produced by warping a single average texture to the target which lack details such as creases around the mouth when the subject is smiling in the second row. Baseline results in column (iii) is produced by taking a weighed average of the photo collection with identical weights used in our method (Eq. 3.3). The facial features such as mouth appear blurry and the colors of the faces appear inconsistent. Baseline results in column (iv) are produced similarly to column (iii), but each photo is warped using thin plate spline and dense warping to the reference before taking the average. The textures appear sharper but still have inconsistent colors. Our method in column v and image b) produces consistent, sharp textures with expression-dependent details.

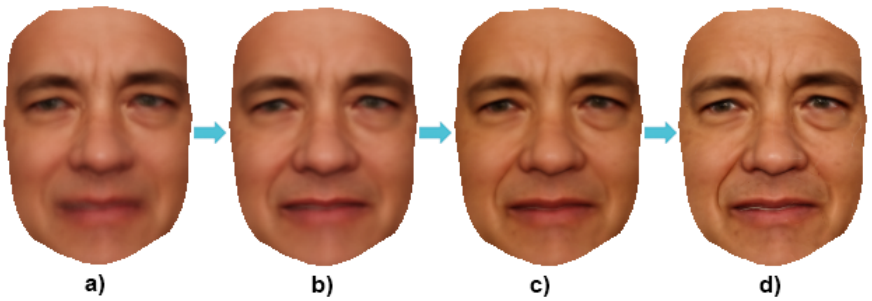


Figure 3.6: A visualization of the results after each step of the texture synthesis process to generate an average face of Tom Hanks. a) shows an average after all photos in the collection are frontalized by a 3D face template, b) after TPS warping, c) after dense warping, and d) the final texture after the multi-scale weighted average which enhances facial details.

Non-rigid warping of the photos. Each photo in the puppet's photo collection has 49 fiducial points that we detected. Next we frontalize the face by marking the same fiducials on a generic 3D face model and solve a Perspective-n-Point problem to estimate the 3D pose of the face in the photo. The model is then back-projected to produce a frontal-warp version of each photo. Let the rigid pose-corrected fiducials in each photo be $F^i \in \mathbb{R}^{2\times 49}$ and the target fiducials be F^T. Given two sets of fiducials we estimate a smooth mapping r that transforms the i-th photo to the target expression using a smooth variant of thin-plate splines [173] which minimizes the following objective:

$$\min_r \sum_{i=1}^{n} \|F^i - r(F^T)\|^2 + \lambda \iint r_{xx}^2 + 2r_{xy}^2 + r_{yy}^2 \,\mathrm{d}x\,\mathrm{d}y \tag{3.1}$$

The optimal mapping r satisfying this objective can be represented with a radial basis function $\phi(x) = x^2 \log x$ and efficiently solved with a linear system of equations [173]. Given the optimal r, we can then warp each face photo to the target expression by backward warping. However, this warping relies only on a sparse set of fiducials and the resulting warp field can be too coarse to capture shape changes required to match the target expression which results in a blurry average around eyes and mouth (Figure 3.6 b). To refine the alignment, we perform an additional dense warping step by exploiting appearance subspaces based on [157, 96]. The idea is to warp all photos to their average, which now has the target expression, through optical flow between illumination-matched pairs. Specifically, let the i^{th} face image after TPS warping be I^i, its projection onto the rank 4 appearance subspace of the TPS warped photos be $\hat{I}^i$. The refined warp field is then simply the flow from I^i to $\hat{I}^i$. In the case where a reference photo is available, (Figure 3.5), we can further warp the entire collection to the reference photo by computing an optical flow that warps $\hat{I}^T$ to I^T, denoted by $F_{\hat{I}^T \to I^T}$, and compute the final warp field by composing $F_{I^i \to \hat{I}^i} \circ F_{\hat{I}^i \to \hat{I}^T} \circ F_{\hat{I}^T \to I^T} = F_{I^i \to \hat{I}^i} \circ F_{\hat{I}^T \to I^T}$ from the fact that $F_{\hat{I}^i \to \hat{I}^T}$ is an identity mapping.

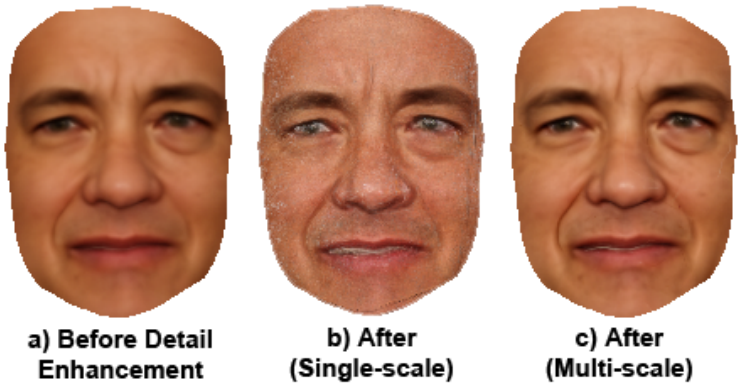


Figure 3.7: a) shows Tom Hanks' average before detail enhancement. b) and c) show the average after single-scale and multi-scale blending.

Adding high-detail. Given the set of aligned face photos, we compute a weighted average of the aligned photos where the weights measure the expression similarity to the target expression and the confidence of high-frequency details. We measure expression similarity by L_2 norm of the difference between the source and target fiducial points, and high-frequency details by the response of a Laplacian filter. A spatially-varying weight W^i_{jk} for face i at pixel (j,k) is computed as:

$$W^i_{jk} = \exp\left(\frac{-\|F^T - F^i\|^2}{2\sigma^2}\right) \cdot (L^i_{jk})^\alpha \tag{3.2}$$

where L^i_{jk} is the response of a Laplacian filter on face image i at pixel (j,k). An average produced with this weighting scheme produces blending artifacts, for example if high-frequency details from many photos with various illuminations are blended together (Figure 3.7). To avoid this problem, the blending is done in a multi-scale framework, which blends different image frequency separately. In particular, we construct a Laplacian pyramid for every face photo and compute the weighted average of each level from all the pyramids according to the normalized W^i_{jk}, then collapse the average pyramid to create a final texture.

With real photo collections, it is rarely practical to assume that the collection spans any expression under every illumination. One problem is that the final texture for different

expressions may be averaged from a subset of photos that have different mean illuminations which results in an inconsistency in the overall color or illumination of the texture. This change in the color, however, is low-frequency and is mitigated in the multi-scale framework by preferring a uniform weight distribution in the lower frequency levels of the pyramid. We achieve this is by adding a uniform distribution term, which dominates the distribution in the coarser levels:

$$W_{jk}^{i} = \left(\exp\left(\frac{-\|F^{T} - F^{i}\|^{2}}{2\sigma^{2}} \right) + \tau l^{-\beta} \right) \cdot (L_{jk}^{i})^{\alpha} \tag{3.3}$$

where $l \in \{0, \ldots, p-1\}$ and $l = 0$ represents the coarsest level of a pyramid with p levels, and τ and β are constants.

3.5 Experiments

In this section, we describe implementation details, runtime, and our results.

Implementation details In Section 3.3, the 3D average models for both driver and puppet are reconstructed using [157] which outputs meshes as depth maps with a face size around 194 x 244 (width x height) pixels. To find correspondence between the driver and puppet for deformation transfer purpose, we project the 2D average of the puppet onto the rank-4 appearance subspace of the driver, then compute an optical flow based on Brox et al.[36] and Bruhn et al. [37] with parameters (α, ratio, minWidth, outer-,inner-,SOR-iterations) = $(0.02, 0.85, 20, 4, 1, 40)$. The ROF denoising algorithm used for adjusting deformation magnitude has only two parameters: The weight constant for the TV regularization which is set to 1, and the Huber epsilon to 0.05. In Section 3.4, λ in TPS warping objective is set to 10, and the dense warping step uses the same optical flow implementation but with $\alpha = 0.3$. For Eq. 3.3, $(\alpha, \beta, \tau) = (1, 20, 1)$, σ is typically set to 10 but can vary around $6 - 12$ for different sizes and qualities of the photo collections (See Sec. 3.6).

Runtime We test our system on a single CPU core of a quad-core Intel i7-4770@3.40GHz. Computing a deformed puppet mesh based on the driver sequence takes 0.2 second per frame with 0.13 second spent on denoising. Synthesizing a texture which includes TPS warping,

Figure 3.8: The first row contains two frames from YouTube videos of Tom Hanks and George W. Bush used as references for puppets of many celebrities in the following rows.

Figure 3.9: We show 3 example subjects for 3D shape and texture reconstruction. The input is a set of photos with varying expressions and appearances, and the output is 3D textured shapes in the same expressions as the input.

dense warping, and multi-scale pyramid blending takes 0.34 second per frame on average.

Evaluating a puppetry system objectively are extremely hard, and there exists no accuracy metric or benchmark to evalute such system. Ground-truth shapes for evaluating deformation transfer across two people cannot be captured as this requires the puppet person whose shape will be captured, to perform exactly like a driver sequence, which is not possible unless the person is the driver themselves. However, such a setup of self puppetry to evalute the reconstructed geometry requires no deformation transfer and does not evaluate our system. Evaluating the synthesized textures is also qualitative in nature as the average texture we generate cannot be pixel-wise compared to the reference. We provide results and input references for qualitative comparisons and point out areas where further improvement can be done.

From Google Images, we gathered around 200 photos for celebrities and politicians in Figure 3.8. We generated output puppetry sequences of those people performing various facial expressions driven by YouTube Videos of Tom Hanks and George W. Bush in the top row. These 3D models are generated by warping an average model of each person with 3D optical flow transfered from the driver (top). So, to render these texture-mapped models, we only synthesize textures in their neutral expressions for the average models but use the target expressions to calculate the blending weights. The identities of these puppets are well-preserved and remain recognizable even when driven by the same source, and the transformation provides plausible output for puppets with different genders, ethnicities, skin colors, or facial features. Facial details are enhanced and change dynamically according to the reference expressions, for example, in creases around the mouth in the last column (supplementary video shows the results in detail).

In Figure 3.5, we show the capability to recreate consistent textures with similar expressions as reference photos in the photo collection. In other words, we are able to "regenerate" each photo in the entire collection so that they appear as if the person is performing different expressions within the same video or photograph captures. Note that each reference here is part of the photo collection used in the averaging process. Texture results for references out-

side the photo collection is in Figure 3.9. We compare our method with 3 baseline approaches: 1. A single static average is TPS warped to the reference. This approach produces textures that lack realistic changes such as wrinkles and creases, and shapes that only roughly match the reference (e.g. eyes in column (ii) second row which appear bigger than the reference) because the warping can only rely on sparse fiducial points. 2. A weighted average of the photo collection using identical weights as our method. With this approach, creases can be seen, but the overall texture colors appear inconsistent when there is a variation in the mean color of different high-weighted sets of photos. The overall textures look blurry as there is no alignment done for each photo, and the shapes (eyes in the third row) do not match the reference when the number of similar photos in the collection is small. 3. An improved weighted average with prewarping step which includes TPS and dense warping similar to our pipeline. The prewarping step improves the shapes and the sharpness of the faces, but the textures remain inconsistent. Our method in column (v) produces sharp, realistic, and consistent textures with expression-dependent details and is able to match references with strong illuminations (diffuse-dominated, visible shadows) or in black-and-white in Figure 3.5 (b). Since the references are part of the averaging process, some high-frequency details such as wrinkles are transfered to the output texture. However, the low-frequency details such as shading effects, soft shadow under the nose (in the last example, middle row), or highlights (in the second example, last row) are averaged out in the multi-scale blending and are not part of the final textures.

In Figure 3.9, we show self-puppetry results where we render output 3D models from [157] with our textures. Similarly to Figure 3.8, we only synthesize textures in neutral expressions for the average models with blending weights calculated based on the target expressions. The reference photos are *excluded* from the photo collection in the averaging process. Our textures remain consistent when the references have different lightings and look realistic from various angles. In the fourth reference in the last row, our textures have wrinkles but are less pronounced than the input reference, which is due partly to the fact that the number of photos with wrinkles in the collection is less than 5%.

3.6 Discussion

The quality of the synthesized textures highly depends on many aspects of the photo collection which include the number and resolutions of the photos, expression and light variations. Since the textures are synthesized based on the assumption that we can find photos with similar expressions, the results will degrade with smaller photo collection (less expression variation). In that situation, the method needs to take into account less-similar photos with a larger standard deviation in Equation 3.3 resulting in a less pronounced expression. If the standard deviation is kept small, high-frequency details can flicker when the rendered models from video input are played in sequence. Higher resolution photos directly contribute to a sharper average. Our method is less sensitive to having small light variations, in contrast to expression variations, because the shading differences are of low-frequency and can be shared across a wider range of photos in the coarser levels of pyramid. The final shape is derived from [157], so any shape inaccuracy will remain as we do not correct the geometry.

When a photo collection contains in the order of thousands photos such as when we extract frames from all movies starring a particular actor, additional characteristics of photos can be used to fine-tune the similarity measure in the averaging process such as the directions of lights in the scene to enable a relighting capability or the age of the person (e.g. from a regressor) to synthesize textures at different ages. Only a small modification is needed to implement these changes in our framework. It is also useful to learn the association between the appearance of facial details and facial motions to help with unseen expressions that may share common facial details with already existing photos in the collection.

We presented the first system that allows reconstruction of a controllable 3D model of any person from a photo collection toward the goal of capturing persona. The reconstructed model has time-varying, expression-dependent textures and can be controlled by a video sequence of a different person. This capability opens up the ability to create puppets for any photo collection of a person, without requiring them to be scanned. Furthermore, we believe that the insights from this approach (i.e., using actor B's shape and texture but A's

motion), will help drive future research in this area.

Chapter 4

AUTOMATIC AGE PROGRESSION

Figure 4.1: Given a single input photo of a child (far left) our method renders an image at any future age range between 1 and 80. Note the change in shape (e.g., nose gets longer, eyes narrow) and texture, while keeping the identity (and milk mustache!) of the input person.

This chapter is based on ''Illumination-Aware Age Progression" [98] published in the proceedings of the Conference on Computer Vision and Pattern Recognition 2017. The project webpage can be found at `http://grail.cs.washington.edu/aging/`.

Modeling facial changes due to aging or "age progression" is one of the most intriguing of digital image processing operations. It is also one of the most challenging for a variety of reasons. First, the aging process is non-deterministic, depending on environmental as well as genetic factors that may not be evident in the input photos. Second, facial appearance and recognizability is strongly influenced by hair style, glasses, expression, and lighting, which is variable and unpredictable. Finally, there is relatively little data available from which to build effective models, as existing age analysis databases are relatively small, low resolution, and/or limited in age range.

Nevertheless, age progression techniques have enjoyed significant success in helping to

solve missing children cases, where subjects have been recognized many years later based on age progressed images. Described as "part art, part science, and a little intuition" [132], these images are produced by forensic artists who combine a background in art, physical anthropology, and expertise with image editing software to simulate the appearance of a person later in life [83]. Aging photos of very young children from a single photo is considered the most difficult case of all, where age progression beyond a few years is considered impractical [124]. We focus specifically on this very challenging case.

Our approach takes a single photo as input and automatically produces a series of age-progressed outputs between 1 and 80 years of age. Figure 2.1 shows an example result. Our approach has three primary contributions. First, we present the first fully-automated approach for age progression that operates "in the wild", i.e., without strong constraints on lighting, expression, or pose. Second, we present some of the first compelling (and most extensive) results for aging babies to adults. And third, we introduce a novel illumination-aware age progression technique, leveraging illumination modeling results [19, 133], that properly account for scene illumination and correct surface shading without reconstructing 3D models or light source directions.

4.1 Related Work

We build on prior work on age progression, notably, the seminal work of Burt and Perrett [38], who created convincing average male faces for several ages (in the range of 20-54) by aligning and averaging photos together. A new query photo was then age progressed by adding to it the difference in shape and texture between the average of the desired target age, and the average for the age corresponding to the query. Their approach required manual alignment. Subsequent aging work in the computer vision literature introduced more automation, often using Active Appearance Models [105] or detecting fiducials [134]. Additional improvements included texture modeling for wrinkles [155] and person-specific models [145, 130]. More details can be found in these excellent survey papers [65, 135]. There are now several commercial programs that will age photos taken with a webcam or

mobile phone. Typically, however, these programs operate effectively only for photos of adults or older children; [122] requires a minimum age of 18, ageme.com lists 7 as the low range, the popular AgingBooth iphone app suggests a minimum age of 15. Furthermore, both commercial offerings and state-of-the-art methods from the research literature still require frontal, simply-lit faces, with neutral expression [65].

There is a body of work on automatic age estimation, e.g., [126, 80, 107]. They, however, did not pursue age progression or other synthesis applications.

Our results set a new bar for age-progression research, demonstrated by a comprehensive evaluation of prior art (the first of its kind in the age progression literature), and an extensive comparison to "ground truth," via large scale user studies on Amazon Mechanical Turk, as described in Section 4.4. The key components that make this advance possible are first, a new database consisting of thousands of photos of people spanning age (0 to 100), variable lighting, and variable pose and expression (Section 4.2.1). Second, *relightable* average images that capture changes in facial appearance and shape across ages, in an illumination invariant manner (Section 4.2.2). And third, a novel technique for aligning illumination subspaces that enables capturing and synthesizing age transformations (Section 4.3).

4.2 Overview

As we age, our faces undergo changes in shape and appearance. The transformation from child to adult is dominated by craniofacial growth, in which the forehead slopes backward, the head expands, and the lower portion of the face extends downward [62]. Changes in later years are dominated by growth of the nose, narrowing of the eyes, the formation of wrinkles and other textural changes.

One of the most compelling ways to model and view these changes across people is by creating a sequence of composite faces, where each composite is the average of several faces of the same gender and age. This idea dates back more than two centuries; Galton [68] generated average images by taking several exposures of portraits on the same photographic plate. Bensen and Perrett [24] showed that dramatically better composites can be obtained

by first aligning facial features (208 fiducials) and warping the images to a reference prior to averaging. Producing composites for aging studies is hampered, however, by the lack of good photographic data for young children, as existing databases are relatively small, low resolution, and limited in age range [65]. In the remainder of this section, we introduce an approach for creating and analyzing a large dataset of human faces across ages, based on thousands of photos from the Internet.

4.2.1 Data collection

To analyze aging effects we created a large dataset of people at different ages, using Google image search queries like "Age 25", "1st grade portrait," and so forth. We additionally drew from science competitions, soccer teams, beauty contests, and other websites that included age/grade information. The resulting databases spans 0 to 100, pooled into 14 age groups (we call them *clusters*), separated by gender. The clusters correspond to ages 0, 1, 2-3, 4-6, 7-9, 10-12, 13-15, 16-24, 25-34, 35-44, 45-56, 57-67, 68-80 and 81-100. The total number of photos in the dataset is 40K and each cluster includes, on average, 1500 photos of different people in the same age range. This database captures people "in the wild" and spans a large range of ages.

4.2.2 Aligned, re-lightable averages

To obtain dense correspondence between the photos in each cluster, we use the "collection flow" method [96], which enables accurate dense correspondence across images with large illumination variation. The input to the collection flow method are aligned and warped to frontal photos for which we use the pipeline of [97]. Figure 4.2 shows the average image for each age, and the average of flow-warped photos using collection flow. Note how much sharper the flow-aligned averages look. While these aligned averages can appear remarkably lifelike, the lighting is dull and unrealistic, as it is averaged over all images in the collection. We instead produce *relightable* average images, which may be re-illuminated from any direction with realistic shading effects. We propose to *match* the lighting of any new input image I

by first pose-aligning the image [97], and projecting that image onto every age subspace. Specifically, for an age cluster j with flow-aligned average A_j, we compute a rank-4 basis via singular value decomposition on the *flow-aligned* images, i.e., $M_j = U_j D_j V_j^T$ where M_j is $f \times p$ the matrix representation of the cluster's flow-aligned photos (f is the number of photos and p number of pixels in each photo). As described in [96], this rank-4 approximation retains the lighting and shading of the input photos, but *neutralizes* the changes due to identity and facial expression, producing a set of images in nearly perfect alignment with a common, average face pose. Next solving

$$\min_{\alpha} ||I - \alpha V_j^T||^2 \tag{4.1}$$

for the coefficients α yields a *re-lit* average that matches the illumination of I:

$$A_j^I = \alpha V_j^T \tag{4.2}$$

(V_j is truncated to rank=4). Figure 4.2 (rows 3-4) shows this capability. Two key advantages of relighting are that 1) it generates a more realistic set of average images, bringing out fine details that are only visible with proper shading, and 2) we can *align* the lighting across the set of averages, to enable comparing changes at different ages. We use this relighting capability to estimate flow across clusters as described below.

4.2.3 Illumination Subspace Flow

We have so far focused on aligning photos within each age cluster. Next, we show how to estimate flow across age clusters, to measure face shape changes over time. Each cluster has many photos under different illumination conditions and thus captures an *illumination subspace*, representing how an average person at a particular age appears under all illuminations [19]. A key contribution is how to align two such illumination subspaces V_i and V_j.

We seek the (single) optical flow field that aligns V_i and V_j. Our insight is to use relighting for flow estimation. As shown in Fig. 4.2 (last column), relighting brings out 3-dimensional

shape differences that are otherwise invisible when averaging many photos. We therefore propose an optical flow method that optimizes over many different lighting conditions. The challenge here is twofold: 1) each illumination subspace represents a continuum of different images, and 2) their coefficient space is not aligned, i.e., any physical lighting direction may map to different lighting coefficients in each illumination subspace.

We introduce a solution that can be easily implemented within the traditional two-view optical flow framework. Let K be the number of database images in the union of clusters i and j. For each image I_k in this union, we project it to each of the two illumination subspaces resulting in an average image A_i^k and A_j^k. The resulting set of images $\{A_i^k\}_{k=1}^K$ can be represented as a single K-channel image $\mathbf{A}_i$, and similarly for $\mathbf{A}_j$. Unlike the original illumination subspaces V_i and V_j, these two multi-channel images are *illumination-aligned*; the k^{th} channel of $\mathbf{A}_i$ and $\mathbf{A}_j$ have the same lighting. Hence, any optical flow algorithm that supports multiple channel images can be used to compute the flow field between them.

When K is large, a smaller representative set of images can be chosen using either discrete sampling, clustering, or dimensionality reduction techniques. We leveraged the fact that the illumination subspaces are low-dimensional [95] (V_i is 4D) and computed an orthogonal 24D basis (two 4D clusters times 3 color channels) for the K images using PCA. Each basis vector (mean + principle vector) was weighted in proportion to its principle value (we modified [113] to support weighted multi-channel images).

4.2.4 Age Transformations

To align all age clusters, we compute subspace flow between each pair of successive age clusters i and $i+1$. Longer range flows between more disparate ages i and j are obtained by concatenation of the flow fields between i and $i+1$, $i+1$ and $i+2$, ..., $j-1$ and j. This concatenation approach gives more reliable flow fields than direct, pairwise flow computation between i and j. These flows enable estimating differences in texture and shape between the different age groups, as we describe in the next section.

4.3 Illumination-Aware Age Progression

Given an input photo of a 2 year old, we can render her at age 60 by computing the difference in flow and texture between the cluster of ages 2-3 (source) and cluster of ages 57-67 (target) and applying it to the input photo. This task is challenging for images "in the wild," as it requires taking into account variations in lighting, pose, and identity. Illumination and shading are inherently 3D effects that depend upon light source direction and surface shape, e.g., as the nose becomes more angular, its shading should change in a manner that depends on light source direction. We show, however, that it is possible to utilize our rank-4 relightable aging basis to work entirely in the 2D domain, without reconstructing 3D models.

To age progress a face photo we perform the following steps, as illustrated in Figure 4.4.

Pose correction: the input face is warped to approximately frontal pose using the alignment pipeline of [97] (step 1 in the figure). Denote the aligned photo I.

Texture age progress: Relight the source and target age cluster averages to match the lighting of I as described in Section 4.2.2, yielding A_s^I and A_t^I. Compute flow $F_{\text{source–input}}$ between A_s^I and I and warp A_s^I to the input image coordinate frame, and similarly for $F_{\text{target–input}}$. This yields a pair of illumination matched projections, J_s and J_t both warped to input. The texture difference $J_t - J_t$ is added to the input image I.

Flow age progress: Apply flow from source cluster to target cluster $F_{\text{target–source}}$ mapped to the input image, i.e., apply $F_{\text{target–input}} \circ F_{\text{target–source}}$ to the texture-modified image $I + J_t - J_s$. For efficiency, we precompute bidirectional flows from each age cluster to every other age cluster.

Aspect ratio progress: Apply change in aspect ratio, to account for variation in head shape over time. Per-cluster aspect ratios were computed as the ratio of distance between the left and right eye to the distance between the eyes and mouth, averaged over the fiducial point locations of images in each of the clusters.

We also allow for differences in skin tone (albedo) by computing a separate rank-4 subspace and projection for each color channel.

4.4 Experiments

We now describe implementation details, results, and evaluation based on a large scale user study.

Implementation details For all flow computations, we modified Ce Liu's [113] implementation (based on Brox et al. [36] and Bruhn et al. [37]) to work with weighted multi-channel photos. We used the following parameters $\alpha = 0.005$, ratio= 0.85, minWidth= 20, nOuterFPIterations= 10, nInnerFPIterations= 1, nSORIterations= 20. We used random SVD [138] for fast low rank computations. Processing the photo database required 30 minutes (on 14 compute nodes) per age cluster of 300 photos, including flow, averages, and subspace computation. Given the precomputed aging basis, age progression of a new input photo takes 0.1 seconds. For blending aged faces into adult heads we estimate fiducials in the adult head photo (computed during pose correction), to match fiducials between the input and target photos, and then run graph cuts to find an optimal seam followed by poisson blending to blend the aged face into the adult head photo [27].

Cropped progression results Figures 2.1 and 4.3 show age progressed images generated automatically using our method. The input images were taken from the FGNET database [105] and were not part of the training set used to create the flow and texture age differences. The results shown here focus on extremely challenging photos of children, with examples that cover a wide range of face types and imaging conditions: neutral, smiling, and laughing facial expressions, frontal and non-frontal pose, lower quality scans as well as higher quality photos, female and male children and a variety of lighting conditions. All results are cropped to the face area to show the raw output of the method. Note how the face shape changes with age in these sequences, e.g., the nose stretches, eyes narrow, and wrinkles appear. Textural changes include facial hair, "shadows" in male faces, eye makeup in female faces, and stronger eyebrows. Many more examples can be found in the supplementary material.

4.4.1 Evaluation

We performed a large scale user study on Mechanical Turk, the most extensive of its kind in the age progression literature. In particular, we had human subjects compare our results to *every prior age progression result* we could find in the literature, and to ground truth (photos of 82 people at different ages). Each subject was shown a photo of a person at age X (e.g., 4), and two additional photos: A) a photo of the same person at an older age Y (e.g., 25), and B) our age-progressed result. The user was asked which of A or B is more likely to be the same person at age Y. They also had the option of selecting "both are equally likely" or "neither is likely." Please refer to the supplementary material for a screenshot of the interface and exact wording. The order of our result and the ground truth was randomly chosen to prevent order bias. All photos were cropped to the face area only. If the progressed image at age Y is generated from the reference at age X, it will have the same lighting and expression. To avoid this similarity bias, our age progression was generated not from the reference, but instead from a photo of the same person at the closest age to the reference.

Comparison with ground-truth We ran our method on every photo in the FGNET dataset, and compared to every older photo available for each person. FGNET consists of photos of the same person over time, and several span baby to adult, resulting in a total of 2976 comparisons. Each user was presented three images: a photo of the subject at age X, an older photo at age Y, and an age progressed photo at age Y. They were asked to specify which of the latter two photos was more likely the same person at age Y by choosing: photo A, photo B, both are equally likely, or neither is likely to be the same person at age Y. Each comparison was evaluated by 3 different people, and 12 comparisons were left blank, making the total number of comparisons we received 8916. The number of unique workers was 72. The results are as following: we received 3288 votes (out of 8916, i.e., 37%) that our result is more likely, 3901 (44%) that ground truth is more likely, 1303 (15%) that both are equally likely, and 424 (5%) that neither is likely. Surprisingly, users could correctly distinguish our results from ground truth only 44% of the time, i.e., no better than chance. I.e., on average,

participants were not able to distinguish our results from ground truth.

This result is so surprising that it led us to question how proficient humans are at this task, i.e., maybe we are just not good at face recognition across large age differences. To test this hypothesis, we conducted a perceptual study in which each user was shown two real (ground truth) images of the same person, separated by at least 5 years, and asked to specify if it is the same or a different person. We used all pairs (at least 5 years apart) of each person on FGNET, and repeated each test three times on Mechanical Turk (8928 tests in total). The results indicate that people are generally good at recognizing adults across different age ranges, but poor at recognizing children after many years. In particular, across children aged 0-7, participants performed barely better than chance (57%) at recognition for roughly 10 year differences, at chance for 20 years (52%), and worse than chance for 50 years (33%). See supplementary material for the full details of the experiment and results.

These studies point to the limits of human evaluation for assessing age progression results. However, the main conclusion that we can draw is that our results are *no less realistic* than the ground truth. This is significant because humans are very adept at identifying errors in synthesized face images.

Ground-truth-blended comparisons While the Mechanical Turk study focuses on cropped faces, we also experimented with blending age progressed faces onto the ground truth head; representative results are shown in Figure 4.7 (additional results appear in the supplementary material). In each case, we take an input photo in the 0-3 age range and compare the ground truth image at each age (right) with our result (left). We blended our result into the ground truth head, using the process described earlier in this section. (We also include unblended results cropped to only the face area in the supplementary material.) The similarity is impressive, especially given that each sequence (column) is produced from a single baby photo. Note that the facial expression and lighting are fixed from the baby photo and therefore differ from the ground truth. As a strawman, we also blended the input child's face onto the older ground truth for comparison (Figure 4.6 (b)); clearly age progressing the input face prior to blending yields much more realistic composites.

Comparison to prior work We compared our results to *all* prior papers that demonstrate age progression results, with the exception of Lanitis et al. [105] whose results do not specify ages. These papers are: (p1) [155], (p2) [145], (p3) [135], (p4) [128], (p5) [129], (p6) [111], (p7) [112], (p8) [149].

While we're most interested in long range age progression of very young children, for comparison we ran our method on every result we found in these papers (including adults and older children). The number of age progression results in papers p1-p8 was: 56, 2, 8, 5, 7, 4, 30 and 8 respectively, for a total of 120 comparisons. Each comparison was performed by 10 workers, and there were on average 13 unique workers per paper. Figure 4.5 plots the results of the user study: the x-axis is the input age group and the y-axis is the output age group. The score is calculated as follows: as in the ground-truth experiment, workers were asked to choose one of the four options. We added 1 point when our result was chosen, 0.5 when "both are likely" was chosen, and 0 when a result from prior work was chosen. The score was then normalized by number of responses per cell (we did not include examples for which the option "neither" was chosen here, as the ground truth evaluation captures similar statistics). As can be seen from Figure 4.5, our approach almost uniformly outperforms prior work for aging young children, and clearly dominates for aging children to adult. The one "red" box corresponds to an age change of only three years (a less interesting case). Note, that there are no prior results in the literature for aging children beyond age 25; we are the first to attempt this task. On the other hand, techniques that focus on modeling older people (modeling wrinkles, hair color, etc.) do better for that category. Note that all previous works typically focus on one of the two age ranges: child to teenager or adult to older person, while our method is general and spans ages 0 to 100 (e.g., Fig. 4.3). While beyond the scope of this chapter, incorporating wrinkles or hair lightening models could yield further improvements in the upper age ranges.

Very few age progression papers address young children [105, 111, 134, 145, 149], and those that do include only a handful of results. See supplementary material for a figure that compares our results to all results in the literature for children under 9 years of age.

Figure 4.6 (a) compares our results to Perrett et al.'s FaceTransformer tool at http://morph.cs.st-andrews.ac.uk/Transformer/ and the PsychoMorph tool by Tiddeman et al. [166] at the Face Research Lab website http://www.faceresearch.org/demos/. As can be seen, they do not perform well on young children. As a baseline we also compare to applying only the aspect ratio change to the input face (compare columns 2 and 5). Both of these tools require manual placement of facial features, whereas our approach is fully automated.

4.5 Discussion

We presented a method for automatic age progression of a child's photo to any age between 1 and 80, by leveraging thousands of photos across age groups and illuminations collected from the Internet. Despite its simplicity, this method works remarkably well, in particular for the most challenging case of very young children, for which few prior results have been demonstrated. A key contribution is the ability to handle photos "in the wild," with variable illumination, pose, and expression.

Our results could be further improved by modeling wrinkles and hair whitening [155] to enhance realism for older subjects. Rather than output a single face, it could be useful to output a set of different candidates to account for variations in appearance, building on face editing techniques, e.g., [123]. And, while we focus on the face region, having a database of heads and upper torsos of different ages to composite onto with a face swapping technique [27] could yield more natural looking results.

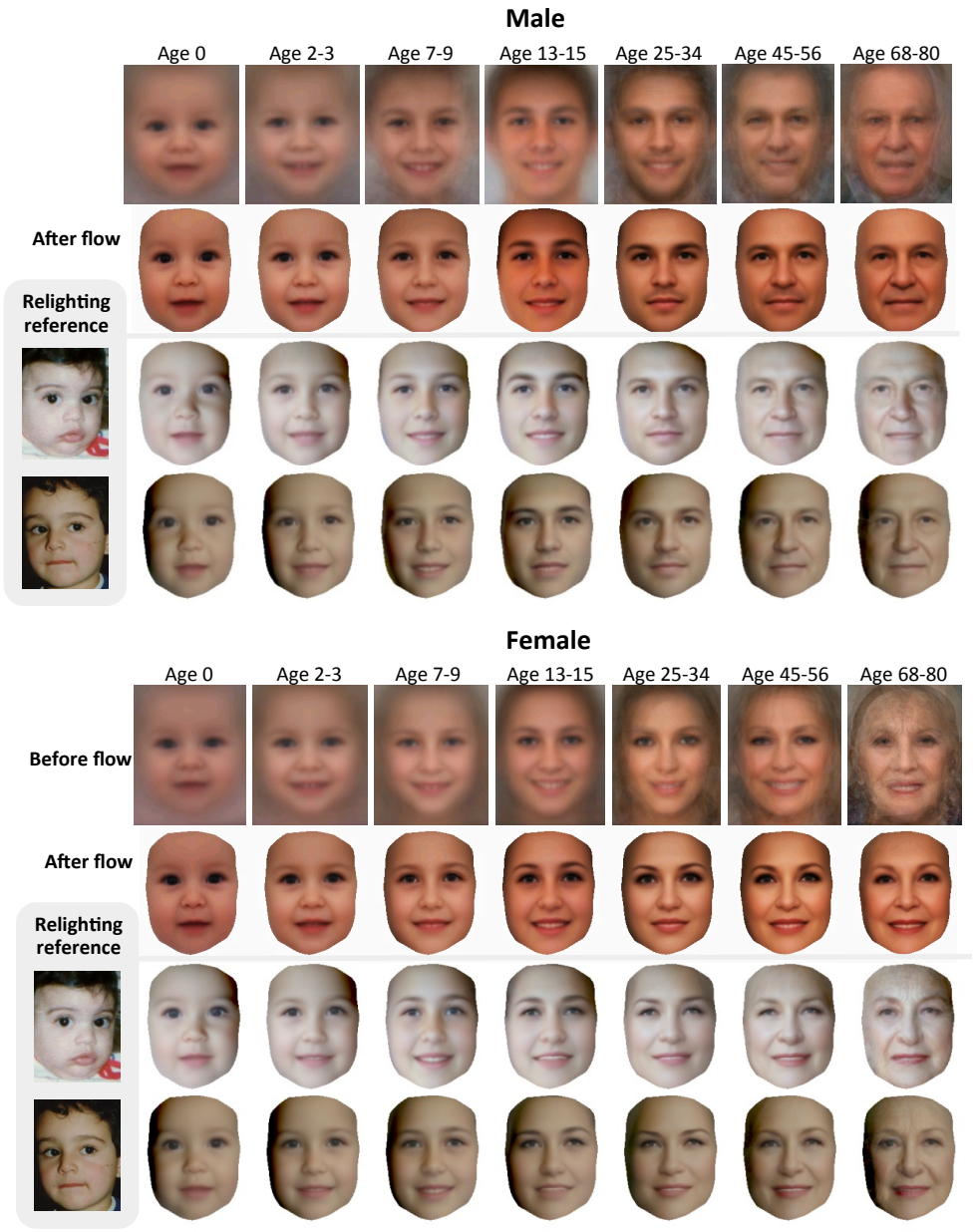


Figure 4.2: Average images of people at different ages. Each image represents an average of about 1500 individuals. Results in the top row are aligned only to place the eyes, nose, and mouth in rough correspondence. The second row shows averages after pixel-to-pixel alignment. These are much sharper, but the tone is variable, the lighting is unnatural, and subtle shape differences (e.g., wrinkles) are averaged out (to see it zoom-in to the last column). The bottom two rows show *re-lit* averages, matched to two reference frames (far left) with opposite lighting directions. The re-lit results have proper shading, are tone-matched to allow easier comparison across ages, and reveal 3D shape changes (note the nose and forehead).

Figure 4.3: Age progression results. For each input image we automatically generate age progressed images for a variety of ages. Note the realistic progression results even with strong directional lighting, non-frontal pose, and non-neutral expressions.

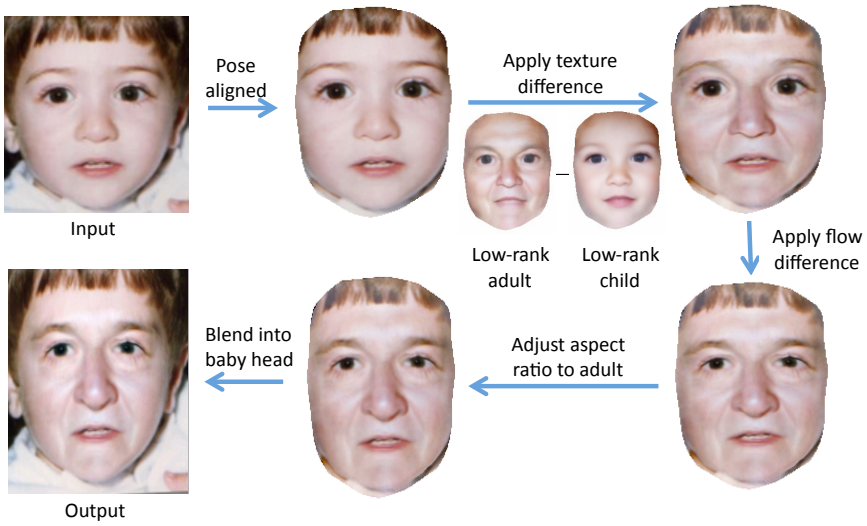


Figure 4.4: Steps of illumination-aware age progression.

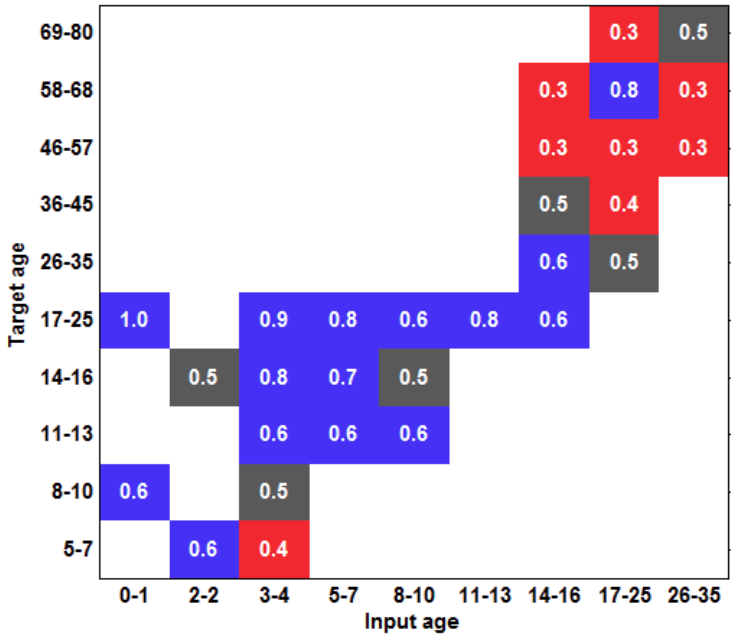


Figure 4.5: Comprehensive comparison to prior work, plotting user study ratings of our method vs. all 120 results from prior work. Blue cells (> 0.55) are where our method scored higher, red cells (< 0.45) have prior method(s) scoring higher, and gray cells are ambiguous. Our method excels for aging children, while prior techniques that target adults perform better for that category.

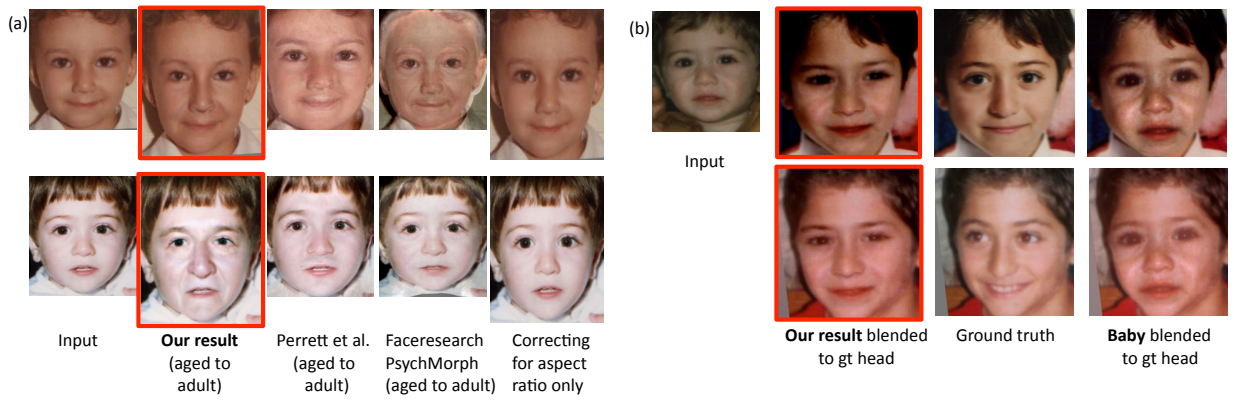


Figure 4.6: Comparison to other methods: (a) to Perrett et al. and FaceResearch online tool, (b) to mapping the baby's face (far left) onto the ground truth (column 3) to produce a blended result (far right). The aged results (column 2) look much more similar to the ground truth, indicating that simply blending a face into a head of an older person does not produce a satisfactory age progression, additional shape and texture changes must be added.

Figure 4.7: Comparison to ground truth images. In each case a single photo of a child (top) is age progressed (left) and compared to photos of the same person (right) at the corresponding age (labeled at left). The age progressed face is composited into the ground truth photo to match the hairstyle and background (see supplementary material for comparisons of just the face regions). Facial expression and lighting are not matched to the ground truth, but retained from the input photo. Note how well the progressed photo matches the ground truth, given that the full sequence is synthesized from a single baby photo.

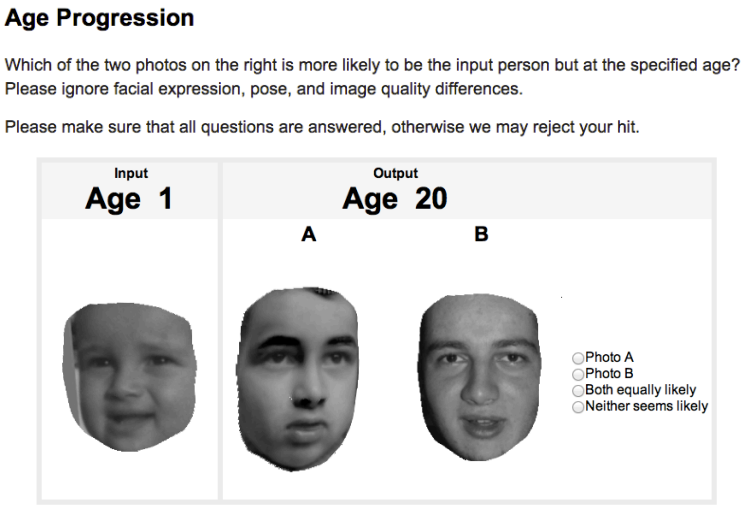


Figure 4.8: Our Mechanical Turk test to compare with ground-truth. We show the input image (far left), our result (middle), and ground truth (right). Note that if the progressed image at age Y is generated from the reference at age X, it will have the same lighting and expression. To avoid this similarity bias, we show to the user a different input photo of the same person at the closest age to the input. Also, the order of our and ground truth was randomly chosen to prevent order bias.

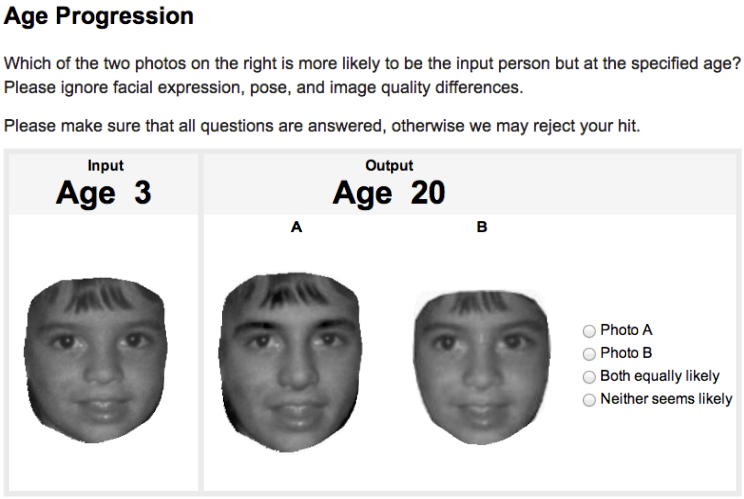


Figure 4.9: Our Mechanical Turk test to compare with previous work. We show the input image (far left), our result (in this case A; we randomize the order of ours and previous to prevent bias), and previous result (in this case B).

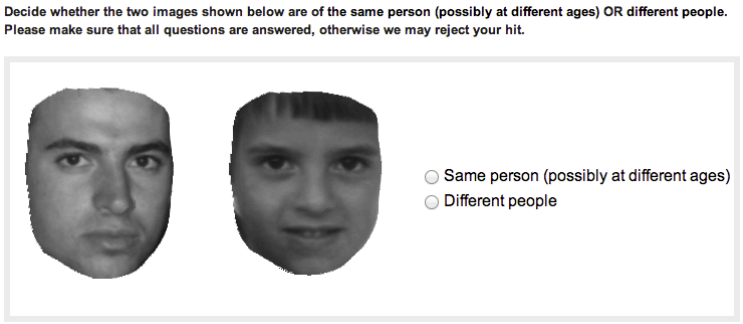


Figure 4.10: Our Mechanical Test to evaluate human proficiency at recognizing the same person across different ages. In each test two real (ground truth) images of the same person, separated by at least 5 years, are shown.

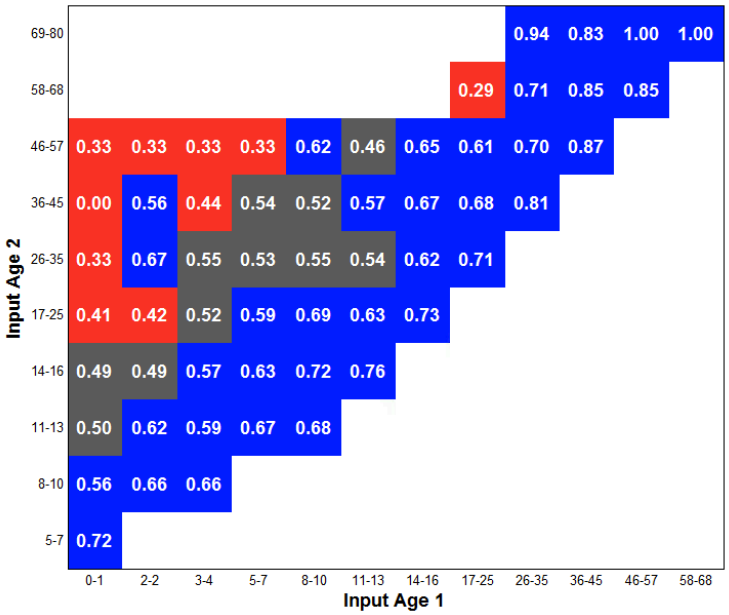


Figure 4.11: Results of human study in Fig. 4.10. The results indicate that people are generally good at recognizing adults across different age ranges, but poor at recognizing children after many years. In particular across children aged 0-7, participants performed barely better than chance (57%) at recognition for roughly 10 year differences, at chance for 20 years (52%), and worse than chance for 50 years (33%).

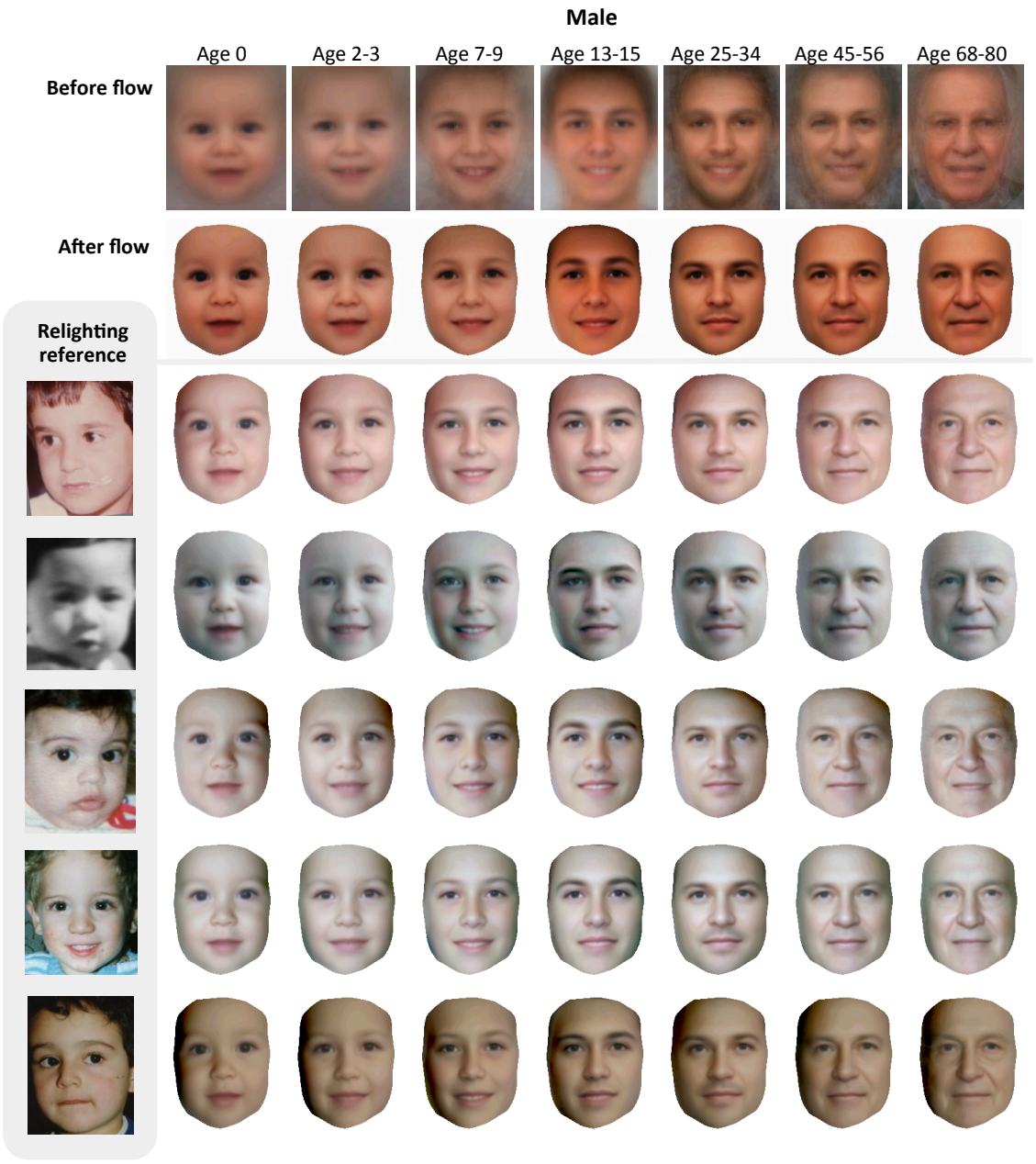


Figure 4.12: Higher resolution averages of people at different ages, and additional re-lit averages and corresponding relighting references (left). These are for the dataset of males.

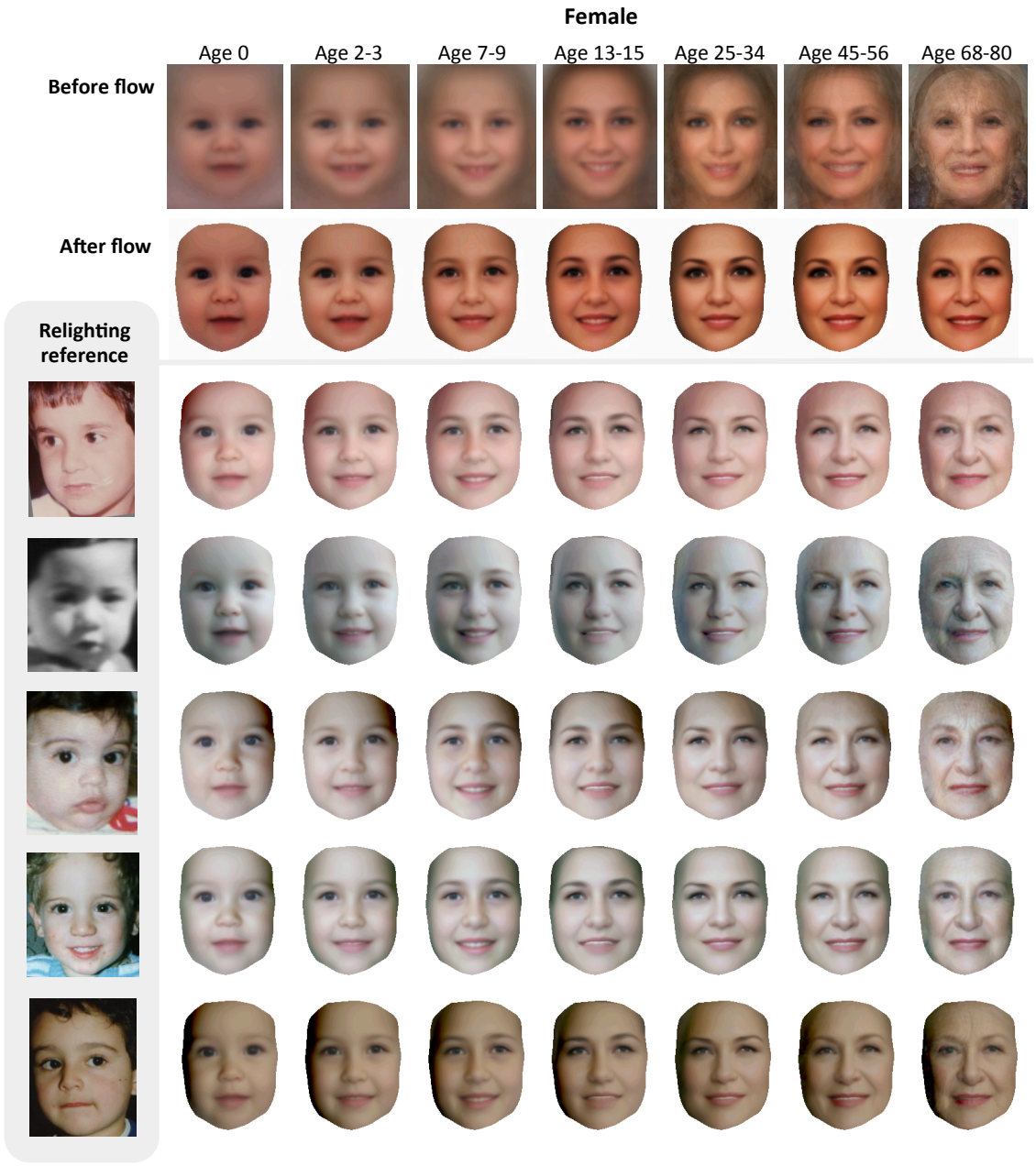


Figure 4.13: Higher resolution averages of people at different ages, and additional re-lit averages and corresponding relighting references (left). These are for the dataset of females.

Figure 4.14: Age progression results. For each input image (left) we automatically generate age progression photos in different ages.

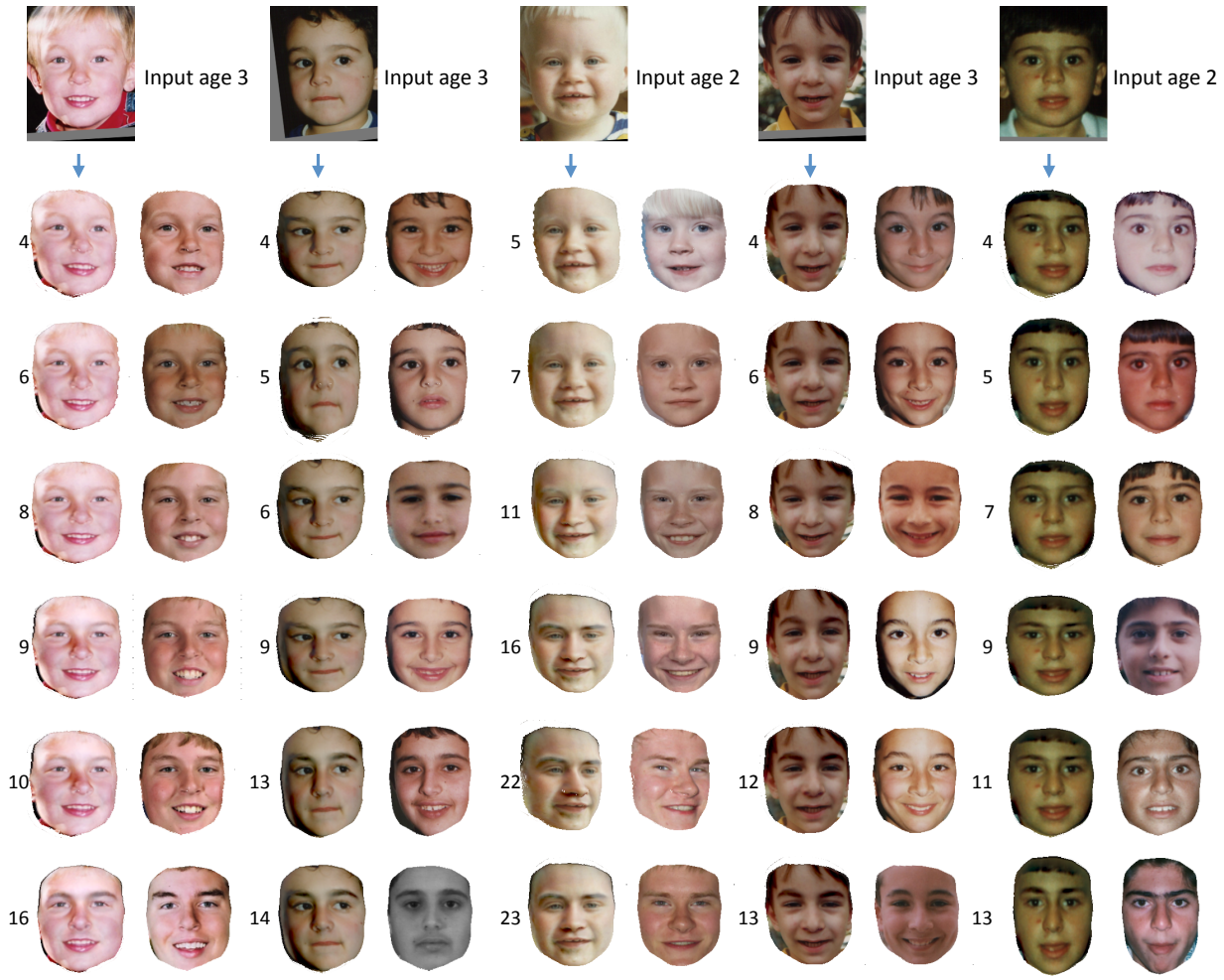


Figure 4.15: Age progression and comparison to cropped ground truth images. In each case a single photo of a child (top) is age progressed (left) and compared to photos of the same person (right) at the corresponding age (labeled at left). Note that lighting and facial expression are not designed to match.

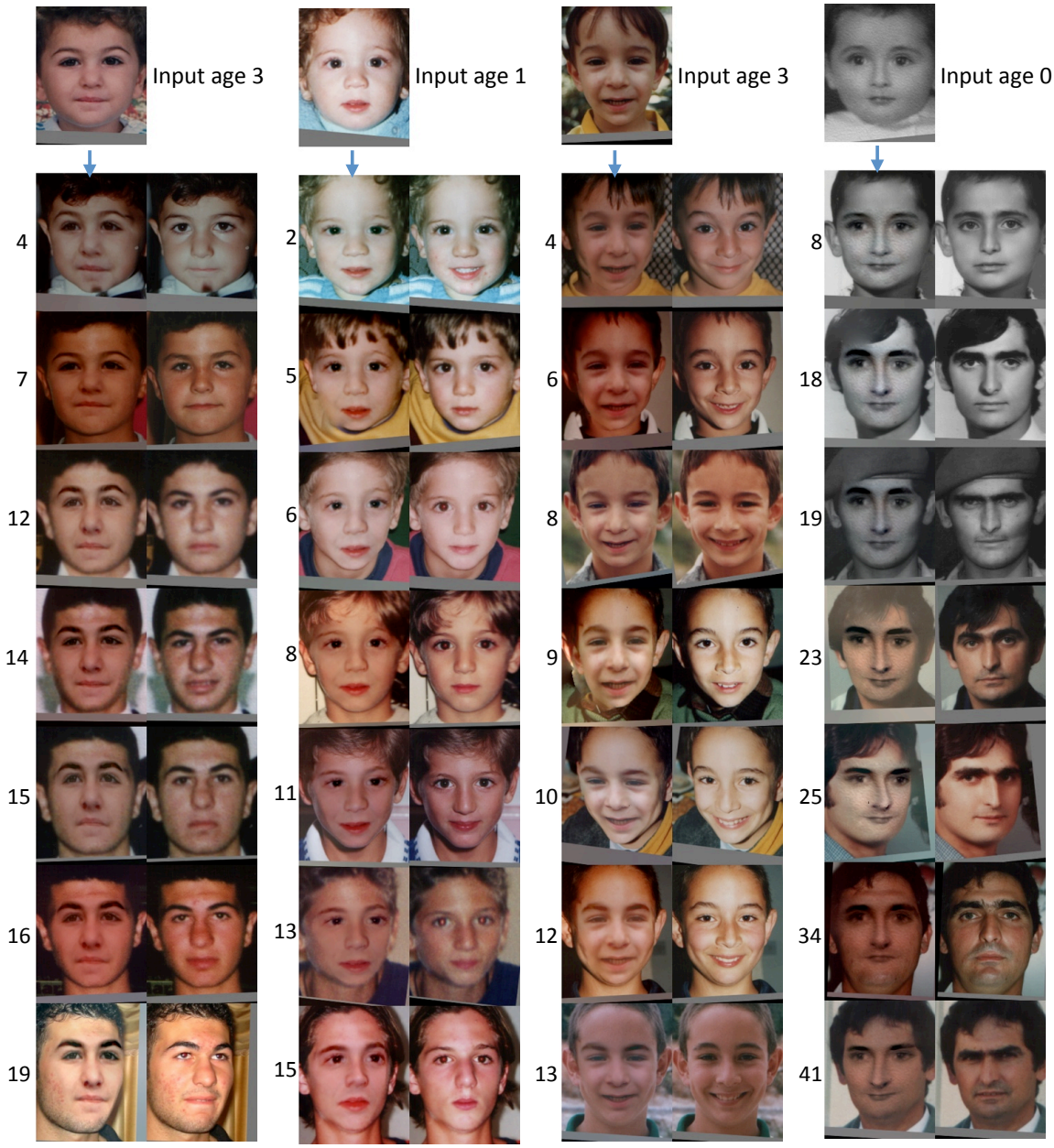


Figure 4.16: Additional age progressions and comparison to ground-truth images. We show the input image (top), our result for each age (left) and ground-truth (right). For each example the age label is on the left.

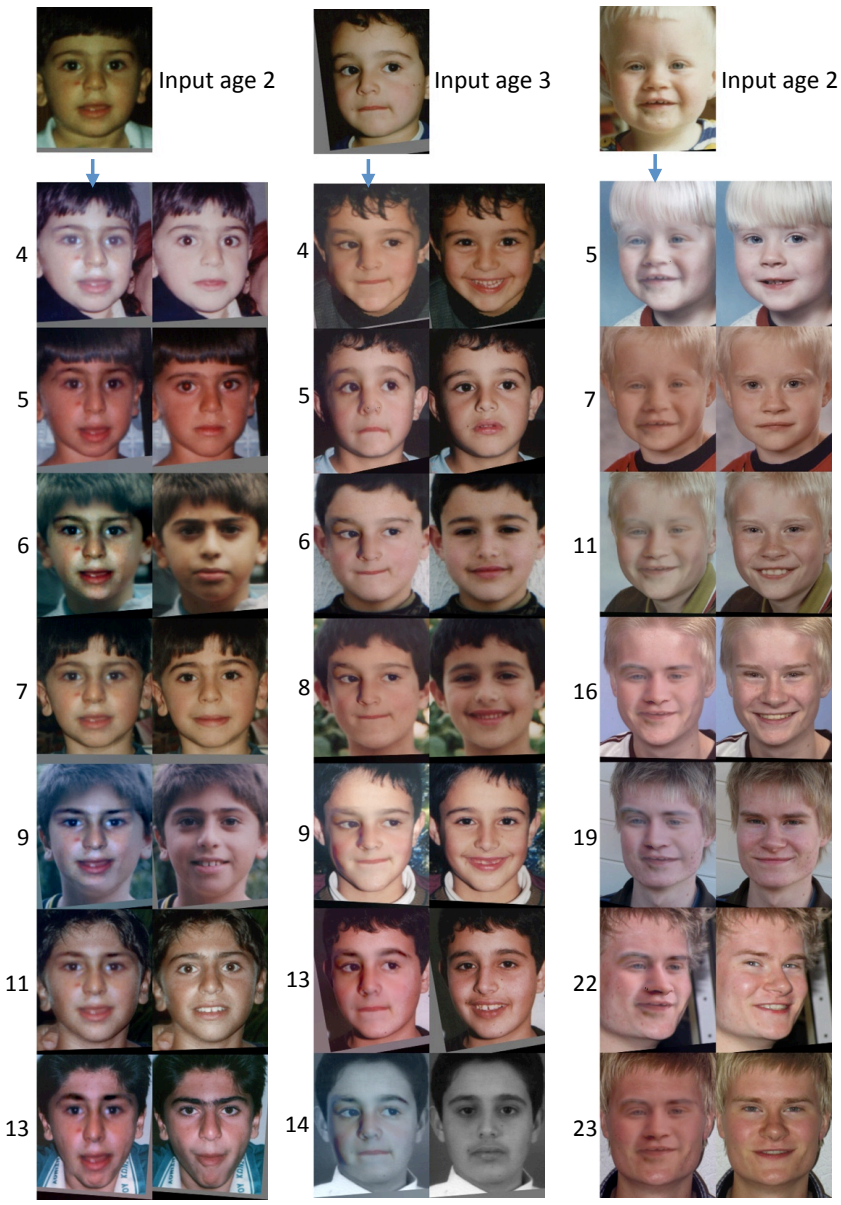


Figure 4.17: Additional age progressions and comparison to ground-truth images. We show the input image (top), our result for each age (left) and ground-truth (right). For each example the age label is on the left.

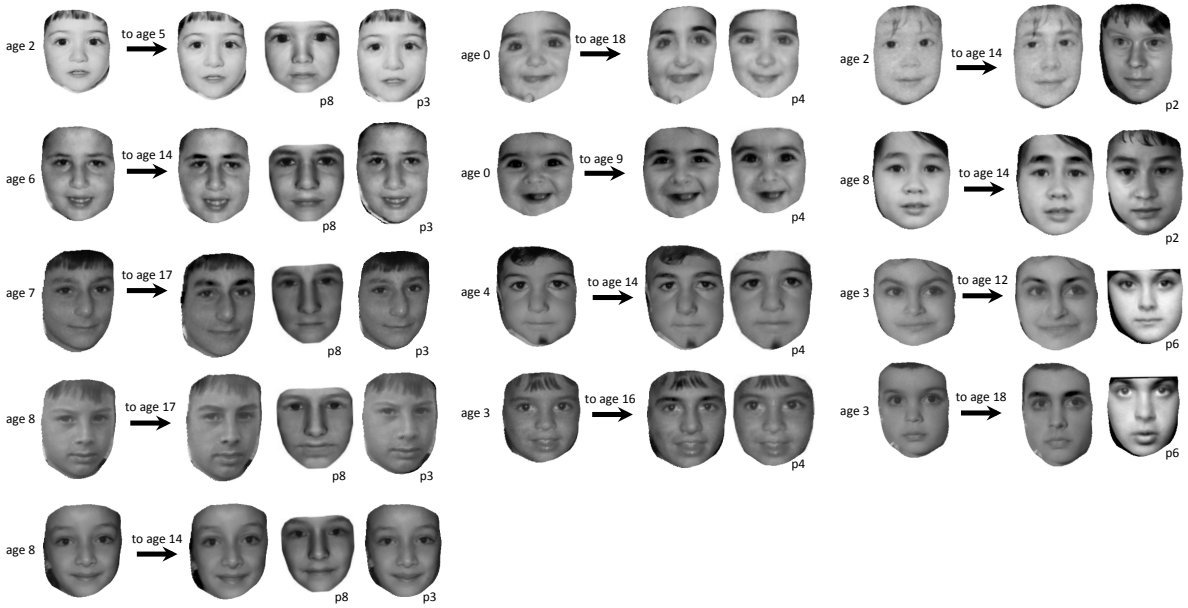


Figure 4.18: Comparison to related works. These are **all** the results of young children (under 9 years old) found in related works (p1-p8). In each case a single photo of a child is age progressed using our method and compared to age progression result of a related work on the same input (paper number is labeled at bottom right of each result, our result is not labeled).

Chapter 5

VISUAL SPEECH SYNTHESIS

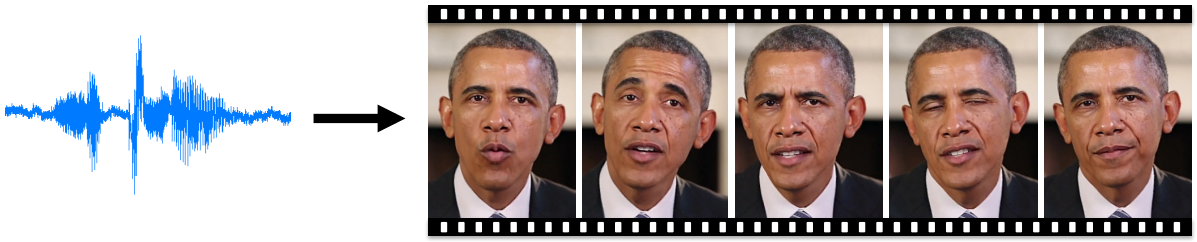


Figure 5.1: Given input audio of Obama speaking, we synthesize photorealistic, lip-synced Obama video.

This chapter is based on "Synthesizing Obama: Learning Lip Sync from Audio" [159] published in the proceedings of ACM SIGGRAPH 2017. The project webpage can be found at `http://grail.cs.washington.edu/projects/AudioToObama/` with video results `https://www.youtube.com/watch?v=9Yq67CjDqvw`.

How much can you infer about someone's persona from their video footage? Imagine learning how to replicate the sound and cadence of a person's voice, how they speak, what they say, how they converse and interact, and how they appear; their expressions and head motions.

In this chapter, we focus on the specific task of learning to generate video of a person from his or her voice. In particular, given just an input audio of someone speaking, we want to generate a corresponding lip-synced video that captures the way they talk and how they deliver that speech. For this task, we do a case study on former President Barack Obama who is ideally suited as an initial test subject for a number of reasons. First, there exists

an abundance of video footage from his weekly presidential addresses—17 hours, and nearly two million frames, spanning a period of eight years. Importantly, the video is online and public domain, and hence well suited for academic research and publication. Furthermore, the quality is high (HD), with the face region occupying a relatively large part of the frame. And, while lighting and composition varies a bit from week to week, and his head pose changes significantly, the shots are *relatively* controlled with the subject in the center and facing the camera. Finally, Obama's persona in this footage is consistent—it is the President addressing the nation directly, and adopting a serious and direct tone.

Despite the availability of such promising data, the problem of generating mouth video from audio is extremely difficult, due in part to the technical challenge of mapping from a one-dimensional signal to a (3D) time-varying image, but also due to the fact that humans are extremely attuned to subtle details in the mouth region; many previous attempts at simulating talking heads have produced results that look *uncanny*. In addition to generating realistic results, this chapter represents the first attempt to solve the audio speech to video speech problem by analyzing a large corpus of existing video data of a single person. As such, it opens to the door to modeling other public figures, or ourselves (through analyzing Skype footage, e.g.,).

Audio to video, aside from being interesting purely from a scientific standpoint, has a range of important practical applications. The ability to generate high quality video from audio could significantly reduce the amount of bandwidth needed in video coding/transmission (which makes up a large percentage of current internet bandwidth). For hearing-impaired people, video synthesis could enable lip-reading from over-the-phone audio. And digital humans are central to entertainment applications like film special effects and games.

Our approach is based on synthesizing video from audio in the region around the mouth, and using compositing techniques to borrow the rest of the head and torso from other stock footage (Fig. 5.2). Our compositing approach builds on similar talking head techniques like Face2Face [165], although Face2Face *transfers* the mouth from another video sequence whereas we synthesize the mouth shape directly from audio. A main contribution is our

recurrent neural network technique for synthesizing mouth shape from audio, trained on millions of video frames, that is significantly simpler than prior methods, yet produces very convincing results. We evaluated many different network architectures to arrive at our solution, but found that a surprisingly simple approach based on standard LSTM techniques produces excellent results. In addition, our approach for generating photorealistic *mouth texture* preserves fine detail in the lips and teeth, and reproduces time-varying wrinkles and dimples around the mouth and chin.

5.1 Related Work

Creating a photorealistic talking head model – a virtual character that sounds and appears real, has long been a goal both in digital special effects and in the computer graphics research community.

In their seminal paper, Bregler et al. bregler1997video demonstrated how to "rewrite" a person's lip movement in a video to match a new audio track represented as a phoneme sequence. Their approach was notable in automating all of the key components; face tracking, phoneme detection, mouth synthesis, and compositing, and produced compelling results for a few short sequences. However, the generality of the method in practice was limited due to insufficient phoneme and viseme reference data; as noted by the authors, correct triphones could be found only 6% of the time, and visemes had to be present for each desired pose. Nevertheless, Video Rewrite remains important as one of the very few techniques in the literature that operate on *existing video footage*, e.g., President John F. Kennedy, rather than training on laboratory-captured footage.

Almost all subsequent work that aims to produce photo-realistic speech from audio has required subjects captured in a controlled lab environment, e.g., [119, 60, 14, 152, 176]. The advantage of the lab environment is that the pose of the subject, their lighting, and the words they utter can all be controlled. Typically, the subject is instructed to say pre-determined, phonetically-rich sentences in a neutral expression (or repeat with up to six different emotions [14]). In contrast to these lab-based approaches, our goal is to develop methods that can

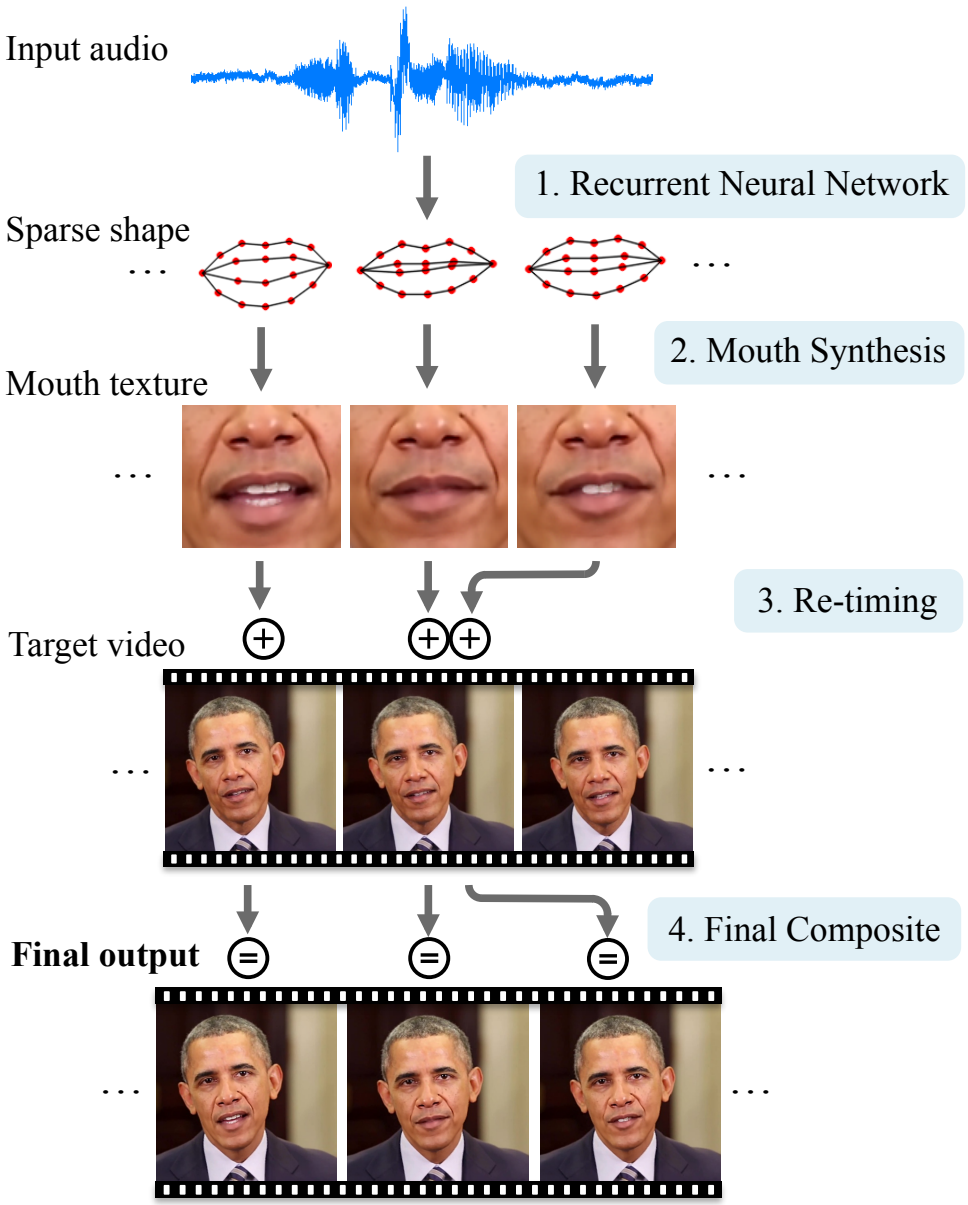


Figure 5.2: Our system first converts audio input to a time-varying sparse mouth shape. Based on this mouth shape, we generate photo-realistic mouth texture, that is composited into the mouth region of a target video. Before the final composite, the mouth texture sequence and the target video are matched and re-timed so that the head motion appears natural and fits the input speech.

eventually be applied to online video footage, i.e., from interviews, speeches, or skype feeds.

A requirement of most prior work in this area is the need for phoneme labels with millisecond-accurate timestamps. These labels are either provided manually, from a speech recognition system, or from a text-to-speech module [60, 119]. Automatic phoneme labeling tends to be error-prone, and thus limits the quality of lipsync. The input is often converted to a sequence of phonemes, or diphones and triphones that encode the surrounding phonemes [34, 60]. Additional known contexual information such as stress or position in the sentence may also be provided [60]. Two different schemes have been used to solve this regression problem that takes phonemes as input and predicts the visual speech. One is based on Hidden-Markov Models (HMM) [142, 64, 186, 185] and the other constructs the final visual speech by concatenating visemes generated from a learned phoneme-to-viseme mapping [161]. Both approaches require a significant amount of linguistic modeling and are quite complex. Voice Puppetry [32] is notable as an early (HMM-based) approach that does not require phoneme labels. Recently, a regression technique based on deep bidirectional long short-term memory [60] has been shown to outperform HMM-based approaches, although this still relies on phoneme labels. Similar to our approach, [152] use an LSTM neural net that does not require phoneme labels and works directly on audio features. However, their network lacks a time-delay, which we have found crucial to producing good results. They unfortunately have no video results available online, and the images in the paper are limited to low-res face images with mocap dots that have been manually annotated. In this work, we show that a simple time-shift recurrent network trained on an uncontrolled, unlabelled visual speech dataset is able to produce convincing results without any dependencies on a speech recognition system.

A key aspect of rendering realistic talking heads is synthesizing *visual speech*, i.e., the motion and appearance of the mouth and surrounding areas. While simple effects can be produced based on motion alone, i.e., using morphing techniques to warp the mouth into new poses [32], the results often look *cartoony*, as they fail to capture important geometric and shading changes, e.g., creases, dimples, that occur as you move your mouth. Most modern

methods therefore attempt to *synthesize* at least the mouth region of the face.

Face synthesis algorithms can be categorized into methods that use 3D face models [165, 14, 46, 45] and others that operate on 2D images. Even in most 3D-based methods, the mouth and teeth are not explicitly modeled in 3D but are represented with 2D texture, e.g. in [14]. One of the most common 2D face texture synthesis techniques is Active Appearance Models (AAM) [52] where a face is jointly modeled in a PCA-based representation for both the sparse shape and texture. Due to the low-rank PCA approximation, however, details such as mouth and teeth often appear blurry [60].

Alternatively, several authors have chosen to use a teeth proxy [164, 15, 71] or to copy teeth texture from original source frames [176, 165, 172]. Neither approach is full-proof, however, often appearing unnatural with artifacts near the lip boundary (see our overview of related work in the accompanying video).

A third source of artifacts in prior art is temporal flickering. To produce smoother results, triphones are sometimes used to find longer subsequences [34], or a visually-smooth path through the original frames is optimized with respect to some similarity cost function [34, 176]. Another approach is to use optical flow to interpolate in-between frames [109, 165]. Note that some of the similiarity metrics in these facial reenactment methods require a driving reference face to compare to, which is not available when only audio is given as input. Unfortunately, none of these techniques are full-proof, and some amount of flickering or unnatural warping often remains. In contrast, our mouth synthesis approach is simple, highly realistic, and naturally exhibits temporal continuity without the need for explicit temporal smoothing or interpolation.

We close our related work section by mentioning perhaps the simplest rewrite approach is to take raw video clips of a person talking, chop them up into word-long segments, and reorder the clips to fit the words of any desired new sentence. The popular website `talkobamato.me` does just that; the results can be fun, but also distracting, as the the background, pose, and tone changes rapidly and discontinuously.

5.2 Overview

Given a source audio track of President Barak Obama speaking, we seek to synthesize a corresponding video track. To achieve this capability, we propose to train on many hours of stock video footage of the President (from his weekly addresses) to learn how to map audio input to video output.

This problem may be thought of as learning a sequence to sequence mapping, from audio to video, that is tailored for one specific individual. This problem is challenging both due both to the fact that mapping goes from a lower dimensional (audio) to a higher dimensional (video) signal, but also the need to avoid the uncanny valley, as humans are highly attuned lip motion.

To make the problem easier, we focus on synthesizing the parts of the face that are *most correlated* to speech. At least for the Presidential address footage, we have found that the content of Obama's speech correlates most strongly to the region around the mouth (lips, cheeks, and chin), and also aspects of *head motion* – his head stops moving when he pauses his speech (which we model through a retiming technique). We therefore focus on synthesizing the region around his mouth, and borrow the rest of Obama (eyes, head, upper torso, background) from stock footage.

We use the following terms throughout the chapter: the many hours of online weekly address video is referred to as *stock* video footage. Stock footage will be used to train our audio-to-shape neural net. The input audio track is the *source*, and the *target video* is a stock video clip into which we composite the synthesized mouth region.

5.3 Audio to Video

The overall pipeline works as follows (Fig. 5.2): Given an audio of Obama, we first extract audio features to use as input to a recurrent neural network that outputs, for every output video frame, a sparse mouth shape (Section 5.3.1). From the sparse mouth shape, we synthesize texture for the mouth and lower region of the face (Section 5.3.2). The mouth texture

is then blended onto a stock video that is modified so that the final head motion appears natural and matches with the given input speech (Section 5.3.3). During blending, the jaw line is warped to match the chin of the new speech, and the face is composed to a target frame in the original pose. (Section 5.3.4).

5.3.1 Audio to Sparse Mouth Shape

Rather than synthesize video directly from audio, we decompose the problem into two steps: 1) map from audio features to sparse shape coefficients, and 2) map from shape to mouth texture. Like [152], we skip the error-prone process of phoneme extraction, and map directly from audio to sparse shape features.

In this step, we represent the audio using standard MFCC coefficients, and the mouth shape by 18 lip fiducials, ranked reduced by a PCA basis, as described next.

Audio Features For audio features, we use Mel-frequency cepstral coefficients (MFCC) which are computed as follows:

1. Given a 16KHz mono audio, we normalize the volume using RMS-based normalization in ffmpeg [23, 137].

2. Take the Discrete Fourier Transform on every 25ms-length sliding window over the audio with 10ms sampling interval.

3. Apply 40 triangular Mel-scale filters onto the Fourier power spectrum, apply logarithm to the outputs.

4. Apply the Discrete Cosine Transform to reduce dimensionality to a 13-D vector.

The final 28-D output feature vector consists of the 13-D vector plus the log mean energy to account for volume, and their first temporal derivatives.

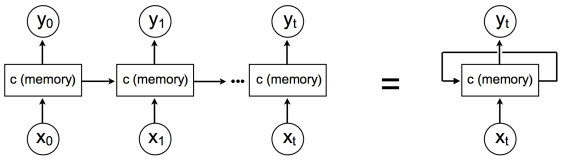
Mouth Shape Features To compute the mouth shape representation, we first detect and frontalize Obama's face in each video frame using the approach in [157]. For each frontalized face, we detect mouth landmarks using [188] which gives 18 points along the outer and inner contours of the lip. We reshape each 18-point mouth shape into a 36-D vector, apply PCA over all frames, and represent each mouth shape by the coefficients of the first 20 PCA coefficients; this step both reduces dimensionality and decorrelates the resulting feature set. Finally, we temporally upsample the mouth shape from 30Hz to 100Hz by linearly interpolating PCA coefficients, to match the audio sampling rate. Note that this upsampling is only used for training; we generate the final video at 30Hz.

Recurrent Neural Network

We seek to learn a mapping from MFCC audio coefficients to PCA mouth shape coefficients. Let's model this mapping using a Neural Network.

Consider Obama saying the word "America". He begins by making the sound *Uhhh*, which is a cue to the mouth synthesizer that he should start opening his mouth. Clearly our network needs the latest audio features as input to determine the mouth shape. But note also that the current mouth shape also depends on the *previous* shape; he will continue to say *Uhhh* for several milliseconds during which time the mouth will open wider, rather than reopening from a closed state.

These considerations motivate a *recurrent neural network* (RNN): at each moment in



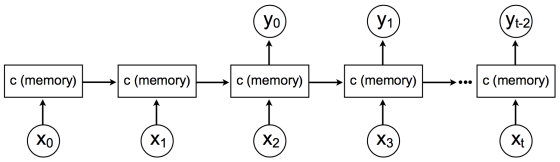
time, this network takes the latest audio input x_t, uses it to modify its hidden state, aka *memory* c, and outputs a new mouth shape vector y_t for that time instant, as well as passing its memory forward in time. In this way, the memory vector (which can be as large as you

want), can represent events potentially far into the past. This RNN technique is very popular for learning time-series problems. RNNs are similar to Hidden Markov Models (HMMs), which have been the basis for most prior talking face work [142, 64, 186, 185], but RNNs provide a more general memory mechanism, nonlinear transitions, and better performance on many problems. Many variants of RNNs have been proposed, and we use Long Short Term Memory (LSTM) models which provide a more efficient mechanism for modeling long term dependencies. LSTMs work by replacing each hidden unit with a series of gates that are specifically designed to facilitate remembering and forgetting (when useful) information (see [1] for a nice tutorial).

Sometimes, your mouth moves *before* you say something. I.e., by the time Obama says *Uhhh*, his mouth is already open. Hence, it's not enough to condition your mouth shape on past audio input – the network needs to look into the future. Shimba et al. [152] point this out as a limitation of their LSTM method. One possible solution is to make the network *bidirectional*. Indeed [61] uses a bidirectional LSTM to exploit future context. However, bidirectional networks require much more compute power and memory to train, as they must be unfolded in the backpropagation process, which usually limits not only the length of training examples, but also the length of the output. Instead, a much simpler way to introduce a short future context to a unidirectional network is to add a time delay to the output by shifting the network output forward in time as explored in [78] as "target delay." While bidirectional LSTMs are popular for speech recognition problems [78, 77], we find that the simpler time delay mechanism is sufficient for our task, likely due to the need to look less far in the future for audio to video, compared with speech recognition which may require looking multiple words ahead. We find that introducing this time delay dramatically improves the quality of results (Section 5.4.1), compared to prior architectures like [152] which omit it. A time-delayed RNN (for a delay of $d = 2$) looks like this:

We opt for a simple single-layer unidirectional LSTM [84]. In Section 5.4.1, we show a

[1] `http://colah.github.io/posts/2015-08-Understanding-LSTMs/`



comparison with other architectures such as multi-layer LSTMs, but we did not find significant improvements to merit the additional complexity. Given $[x_1, \ldots, x_n], [y_1, \ldots, y_n]$ as input and output vector sequences, our standard LSTM network is defined by the following functions:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{5.1}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{5.2}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{5.3}$$

$$c_t = c_{t-1} f_t + i_t \tanh(W_j \cdot [h_{t-1}, x_t] + b_j) \tag{5.4}$$

$$h_t = \tanh(c_t) o_t \tag{5.5}$$

$$\hat{y}_{t-d} = W_y h_t + b_y \tag{5.6}$$

where f, i, o, c, h are forget gate, input gate, output gate, cell state, cell output as proposed in [84]. σ is the sigmoid activation. Note that the cell state and cell output are transformed with tanh. $\hat{y}_{t-d}$ represents the predicted PCA coefficients at time $t - d$ where d is the time-delay parameter. Learned parameters are weight matrices W and bias vectors b. We use a 60 dimensional cell state c and a time delay d of 20 steps (200ms). The network is minimized using L2-loss on the coefficients and trained using Adam optimizer [102] implemented in TensorFlow [9], on many hours of stock Obama weekly address footage. More details can be found in Section 5.4.1.

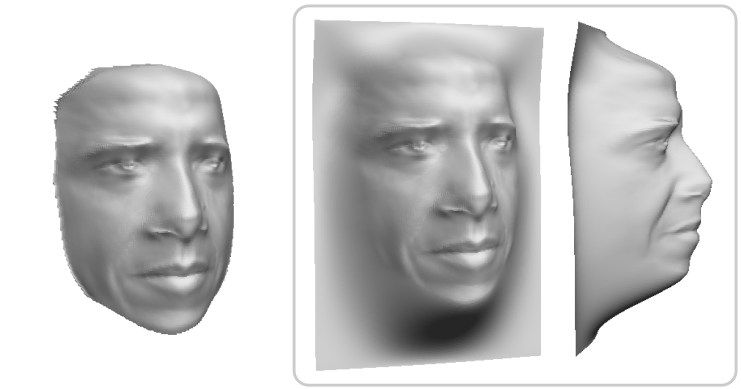
5.3.2 Facial Texture Synthesis

In this section, we describe our approach for synthesizing high detailed face textures from sparse mouth shapes. We focus on synthesizing the lower face area, i.e., mouth, chin, cheeks, and area surrounding the nose and mouth. Figure 5.8 shows the mask. The rest of Obama's appearance (eyes, head, torso, background) comes from stock footage of Obama's weekly addresses. Our texture synthesis algorithm is designed to satisfy two key requirements: 1) sharp and realistic appearance per video frame, 2) temporally smooth texture changes across frames.

We explored several approaches for synthesizing the mouth area based on prior art (see Section 5.1) but found that results were either too blurry (in the case of Active Appearance Models), the teeth were too non-rigid (with warping/flow techniques), or the illumination was mismatched. We compare these different techniques in Figure 5.10 and the supplementary video. Instead, we propose an approach that combines weighted median and high frequencies from a teeth proxy.

Given a sparse mouth shape sequence and a target video, we process each mouth shape independently. The algorithm overview is as follows: per mouth PCA shape, select a fixed number of target video frames that best match the given shape; apply weighted median on the candidates to synthesize a median texture; select teeth proxy frames from the target video, and transfer high-frequency teeth details from the proxy into the teeth region of the media texture. We describe those steps in detail below.

Candidate frame selection: Given a generated mouth shape, we seek a set of best matching frames from the target video. Candidate frames are selected as follows: we run a landmark detector [188] on the target video, estimate 3D pose, and frontalize every frame using a 3D model of Obama (Figure 5.3). We compute the 3D face model using [157], and augment it with rough approximations of chin and background shape. We found that the latter step significantly improved frontalization results. The 3D face model is extended to include the chin by assuming a planar background and solving for a smooth surface that



a) Reconstruction from [Suwajanakorn et al. 2014]

b) **Our extended model with a simple plane**

Figure 5.3: We augment an Obama model (a) reconstructed from [157] with a simple, near-planar background. This extension is used for frontalizing the chin and neck in addition to the face region.

connects the face to the background. Specifically, we minimize the surface's second partial derivatives: suppose the initial surface parametrized on a 2D depth map $f(x, y) : \Omega \to \mathbb{R}$ is only given on the face region $\Omega' \subset \Omega$. We solve for a new f^* on the entire domain Ω by:

$$\min_{f^*} \iint_{\Omega} \left(\frac{\partial^2 f^*}{\partial x^2} \right)^2 + \left(\frac{\partial^2 f^*}{\partial y^2} \right)^2 \mathrm{d}x \, \mathrm{d}y \tag{5.7}$$

$$\text{subject to } f^*|_{\Omega'} = f|_{\Omega'} \text{ and } \nabla f^*|_{\partial\Omega} = 0 \tag{5.8}$$

where $\partial\Omega$ denotes the boundary of Ω. The objective and constraints are turned into a linear least squares problem (with soft constraints) on a discrete pixel grid by finite differences. The extended 3D model is shown in Figure 5.3b. Next, we estimate drift-free 3D pose for each frame using [157], place the model onto each frame, and back project the head to the frontal view. Even though this extended geometry is inaccurate away from the face area, it suffices as a frontal-warping proxy since the final synthesized texture will be warped back to the original pose using the same geometry.

Texture is synthesized in an area defined by a manually drawn mask that includes the

lower face and neck areas in frontal pose. The mask is drawn only once. Additionally, since in some poses the neck is occluded, we automatically mask out the clothing in every video frame (by means of simple thresholding in HSV space; the same threshold is used in all results) and in-paint the masked region using [162], and the OpenCV [31] implementation.

Once all frames are frontalized and pose is computed, n frames that have the smallest L^2 distance between frame's mouth shape and target mouth shape are selected.

Weighted median texture synthesis: Given a set of frontal mouth candidate images $\{I_1, \ldots, I_n\}$ with associated mouth shapes $S_1, \ldots, S_n$ where $S_i \in \mathbb{R}^{2\times 18}$ and a target mouth shape S_{target}, we first compute the weighted median per pixel (u, v):

$$\text{median}(u, v) =_c \sum_{i=1}^{n} w_i |I_i(u, v) - c| \tag{5.9}$$

$$\text{subject to } \exists k, c = I_k(u, v) \tag{5.10}$$

This is computed independently for each of the R,G,B channels. c is the output pixel intensity and w_i represents how similar S_i is to S_{target} and is computed by:

$$w_i = e^{\frac{-\|S_i - S_{\text{target}}\|_2^2}{2\sigma^2}} \tag{5.11}$$

Choosing the right σ is critical. Small σ will create a peak distribution on a few images which can cause temporal flickering, similar to taking a single original frame, and large σ can produce a blurry result. Moreover, the optimal σ for one target shape can be suboptimal for another target shape depending on the number of good candidates, i.e., ones with small $\|S_i - S_{\text{target}}\|$. Because the optimal σ is tied to the number of good candidates, we adaptively select σ such that the weight contribution of n candidates is α-fraction of the weight of all available frames. In other words, we solve for σ for each target shape such that

$$\sum_{i=1}^{n} w_i(\sigma) = \alpha \sum_{i=1}^{N} w_i(\sigma) \tag{5.12}$$

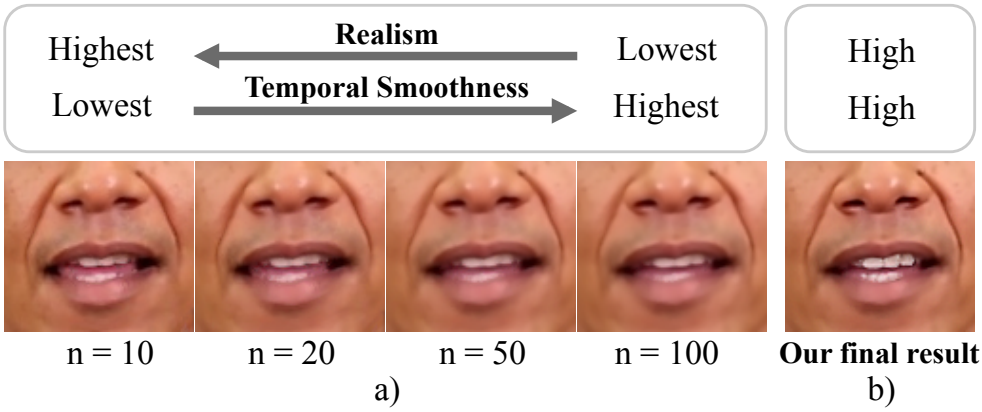


Figure 5.4: a) shows the visual quality with respect to the number of candidates (n). Even though averaging (through median) lower numbers produce sharper results, they are not temporally smooth when used in an animation. On the other hand, our final result shown in b) both minimizes blur and is temporally smooth.

where N is the total number of video frames. This can be efficiently solved with a binary search on σ. We fix α to 0.9 and tune n for the best balance between the visual quality and temporal smoothness (Figure 5.4). Once n is selected, only σ will vary for each output frame. With all w_i computed, Equation 5.9 is solved efficiently by sorting pixel intensities and picking the intensity situated at the half of the total weight.

Teeth Proxy: Synthesizing realistic teeth is surprisingly challenging. The teeth must appear rigid, sharp, pose aligned, lit correctly, and properly occluded by the lip. Our evaluation of prior methods (see Section 5.1 and submission video) all exhibited problems in one or more of these aspects. In particular, AAM-like models yielded blurry results, while teeth proxy approaches produced compositing artifacts.

We achieved our best results with a new, hybrid technique that combines low-frequencies from the weighted median texture, and high frequency detail from a teeth proxy image. This idea of combining frequencies from different sources is based on [127]. The key insight is that the median texture provides a good (but blurry) mask for the teeth region, whereas the teeth proxy does a good job of capturing sharp details. Hence, we apply the high frequencies

of the teeth proxy only in the (automatically detected) teeth region of the median image.

The teeth proxy reference frame is manually chosen to be one of the target frames where teeth are frontal-facing and highly visible. We need one proxy for the upper and another for the lower set of teeth; these two proxies may be chosen from different frames. This is a step in approach that is manual, and must be repeated for each target sequence.

The teeth region in the median texture, to which we will transfer proxy details, is detected by applying a threshold (low-saturation, high-value) in HSV space within the mouth region given by the landmarks.

Given a teeth proxy reference frame T whose pixel values are converted to be within $[0,1]^3$, we apply a high-pass filter to T:

$$H_{\sigma,s}(T) = (T - G_\sigma * T) \times s + 0.5 \tag{5.13}$$

where G_σ is a Gaussian kernel with standard deviation σ, $*$ is the convolution operator, and s is the adjustable strength. We also truncate the value of $H(T)$ to be within $[0,1]$. Then given a median texture I, we compute the final texture I' for pixel (u,v) in the mouth region as:

$$I'(u,v) = \begin{cases} 2I(u,v)H(u,v) & \text{if } H(u,v) < 0.5 \\ 1 - 2(1 - I(u,v))(1 - H(u,v)) & \text{otherwise} \end{cases} \tag{5.14}$$

Additionally, we enhance I with multiple $H_{\sigma,s}$ of different σ's to handle various frequency scales. This high-frequency addition, however, only works for the upper teeth, since they are stationary with respect to the face. For the lower teeth, we shift $H(u,v) \leftarrow H(u+\Delta u, v+\Delta v)$ by $\Delta u, \Delta v$ which represent the jaw difference between I and T estimated from the lower lip landmarks. Without accurate landmarks, this can cause flickering. So, instead of using the landmark output from our network or running a landmark detection on I which can be noisy, we compute a weighted average of the lip landmarks of all image candidates using the weights

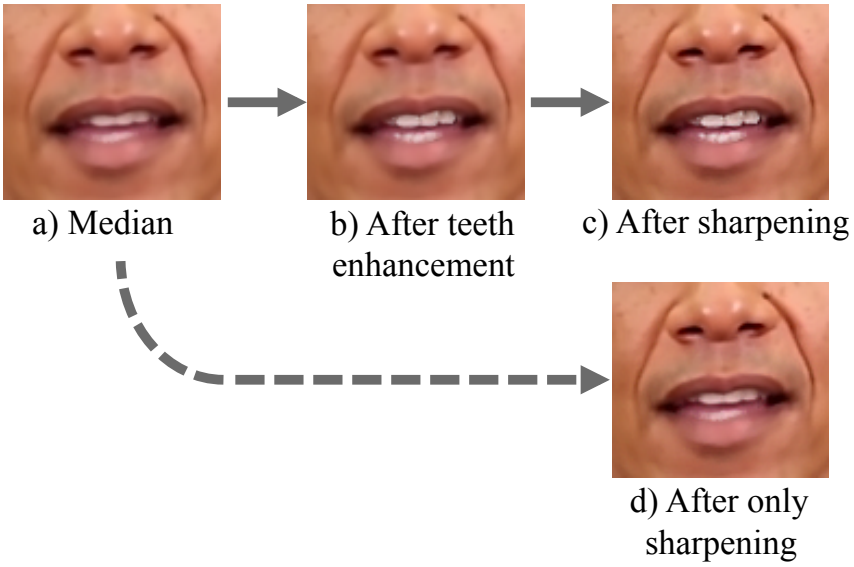


Figure 5.5: The effects of proxy-based teeth enhancement. a) shows the weighted median texture computed in Section 5.3.2. b) is after applying proxy-based teeth enhancement in Section 5.3.2. c) is after an additional high-pass filter. d) shows the result of a high-pass filter on the median texture, without the teeth proxy.

from Equation 5.11 to obtain an accurate, temporally smooth jaw location. The enhanced teeth texture after an additional spatial sharpening (unsharp masking) is illustrated in Figure 5.5.

5.3.3 Video Re-timing for Natural Head Motion

We assume the availability of a target video, into which our synthesized mouth region will be composited. Any of Obama's weekly presidential address videos work well as targets, for example. Since the speech in the target video is different from the source (input) speech, a naive composite can appear awkward. In particular, we've observed that it's important to align audio and visual pauses; if Obama pauses his speech, but his head or eyebrows keeps moving, it looks unnatural. To solve this problem, we use dynamic programming to re-time the target video. We look for the optimal monotonic mapping between N synthesized mouth animation frames and M target video frames such that:

- it prefers more motion during utterance and minimal motion during silence

- any target video frame may be repeated at most once but never skipped. This limits slow downs to at most 50% and the video cannot be sped up; otherwise a noticeable jump or freeze can occur.

- it prefers sections of the target video where slowing down would be least noticeable, i.e., not during blinking or quick expression changes.

Note that this technique is loosely related to "Video Textures" [146] which aims to generate a new sequence from small sections of an input video clip by minimizing noticeable discontinuities. In our task, we wish to avoid repeated head motion or motion loops, and we can directly use the target sequence, although re-timed, to avoid any visual discontinuities.

To formulate the dynamic programming objective, we first compute the motion speed for each frame j in the source video, denoted by $V(j)$, using the first derivative of the facial landmark positions as well as a binary flag indicating a blink, denoted by $B(j)$, by applying a threshold on the size of the eye landmarks. Then we assign $V(j) \leftarrow V(j) + \alpha_B B(j)$ where α_B is a balancing weight. For the source speech, we compute a binary flag, denoted by $A(i)$, indicating non-silence by applying a threshold on the audio volume. These binary flag sequences typically contain salt and pepper noise (random 0's or 1's), which we filter out by applying dilation followed by erosion with the same kernel size to remove small gaps of 0's. We additionally filter out very short consecutive sequences of 0's or 1's by a second

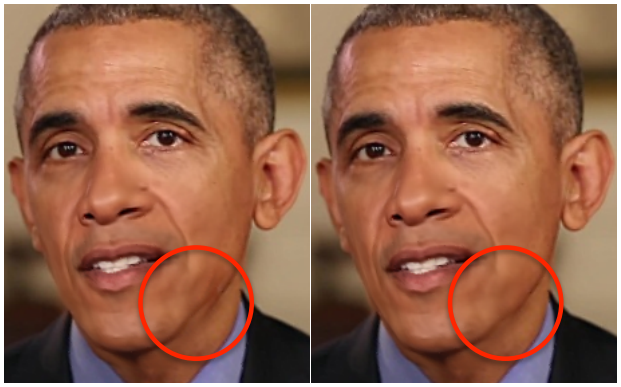
threshold. The recurrence relation is defined as follows:

$$F(i,j,0) =$$
$$\min(F(i-1,j-1,0), F(i-1,j-1,1)) + G(i,j) \tag{5.15}$$
$$F(i,j,1) = F(i-1,j,0) - \alpha_s V(j) + G(i,j) \tag{5.16}$$
$$G(i,j) = \begin{cases} V(j) & \text{if } A(i) = 0 \\ 0 & \text{otherwise} \end{cases} \tag{5.17}$$
$$+ \begin{cases} \alpha_u V(j) & \text{if } A(i-2) = 1 \text{ and } A(i-3) = 0 \\ 0 & \text{otherwise} \end{cases} \tag{5.18}$$

where $F(i,j,k)$ stores the score when the i^{th} mouth shape frame is matched with the j^{th} video frame and k is the number of times this mouth frame has been repeated. α_s and α_u are free parameters associated with penalizing large motion during silence and small motion during utterance, respectively. We initialize the base cases as follows:

$$F(0,j,0) = \begin{cases} V(j) & \text{if } A(i) = 0 \\ 0 & \text{otherwise} \end{cases} \tag{5.19}$$
$$F(i,0,0) = \begin{cases} \infty & \text{if } i > 0 \\ 0 & \text{otherwise} \end{cases} \tag{5.20}$$
$$F(i,j,1) = \infty \text{ if } i = 0 \text{ or } j = 0 \tag{5.21}$$

The optimal score is $\min_j \left\{ \min(F(N-1,j,0), F(N-1,j,1)) \right\}$ and the optimal mapping is found by back-tracing the minimal path through the 3-dimensional F array with an overall $O(MN)$ time complexity. We set $\alpha_B = 1$ and $\alpha_s = \alpha_u = 2$ in our implementation. Finally, to avoid having a completely static motion for the final composite when a frame is repeated, we warp the repeated frame half way between the previous and next frame. Specifically, suppose



a) Without jaw correction b) **With jaw correction**

Figure 5.6: a) shows a jawline discrepancy when the mouth texture of a different speech is blended onto a target video frame. b) shows our corrected result where two jawlines are connected.

frame i is a copy of frame $i-1$, we compute optical flows $F_{(i-1)\to(i+1)}$ and $F_{(i+1)\to(i-1)}$, and define the final frame i as the average of frame $i-1$ warped by $0.5F_{(i-1)\to(i+1)}$ and frame $i+1$ warped by $0.5F_{(i+1)\to(i-1)}$.

5.3.4 Composite into Target Video

Compositing into the target video is the final step of our algorithm. By this point, we have created a lower face texture for each mouth shape corresponding to the source audio. We have also re-timed the target video to naturally fit silence or talking moments in the source audio. The key part of the composition step is to create a natural, artifact-free chin motion and jawline blending of the lower face texture into the target head. Figure 5.6 illustrates how blending may look if jawlines are not explicitly corrected. The artifacts are especially visible when watching a video. Therefore, we created a jaw correction approach that operates per frame.

Jaw correction: The algorithm is illustrated in Figure 5.7. Optical flow (d) is computed between the lower face texture frame (b) and target video frame (a). Next, an alpha map is created based on fiducials to focus only on the area of the jawline (e). The flow is masked

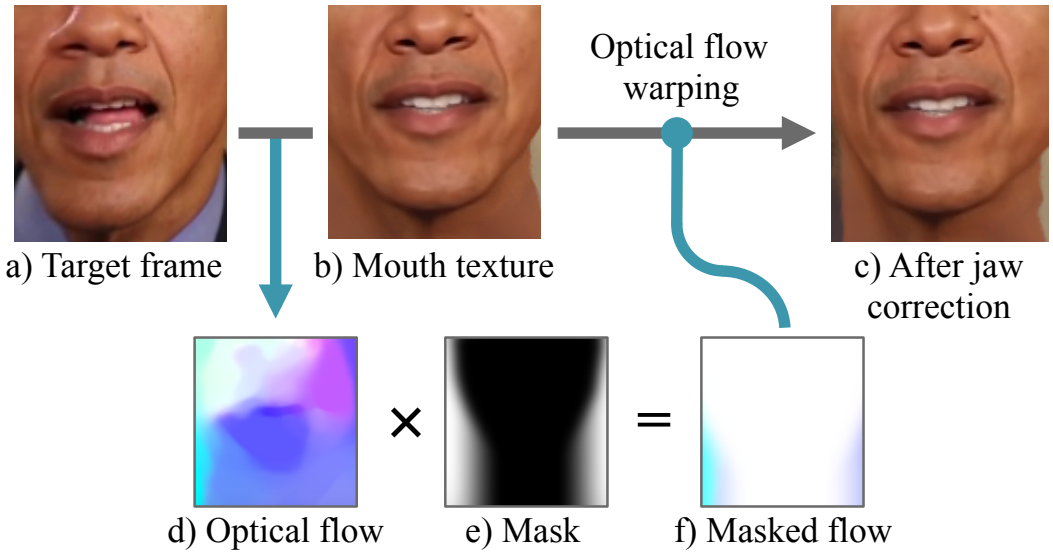


Figure 5.7: To prepare the mouth texture so that the final jawline appears seamless in Figure 5.6, we first compute optical flow between a target video frame (a) and our mouth texture (b). This resulting flow (d) is masked by (e) to produce (f) which is used to warp our mouth texture and produce the final texture in (c).

by the alpha map and then used to warp the lower face texture to fit the target frame.

Final compositing: Figure 5.8 illustrates the final masking and blending steps. The blending is done using Laplacian pyramids [39] in a layer based fashion. There are four layers that are blended in the following order from front to back: 1) Lower face texture (excluding the neck), 2) torso (shirt and jacket), 3) Neck, and 4) the rest. Parts 1 and 3 come from the synthesized texture, while parts 2 and 4 come from the target frame. The neck mask is the region under the chin in our synthesized mouth texture and the mouth mask is the region above. The chin is determined by splining face contour landmarks estimated from DLIB library [101]. In some target videos where the background is easy to segment, e.g. when it is a solid black, we create an additional mask for the background (via a color detector) and add it to the shirt mask to have the second layer include both the shirt and background. Although this is optional, it helps prevent the artifact shown in Figure 5.14b. The final texture is rendered back to the target frame pose using the 3D shape estimated in

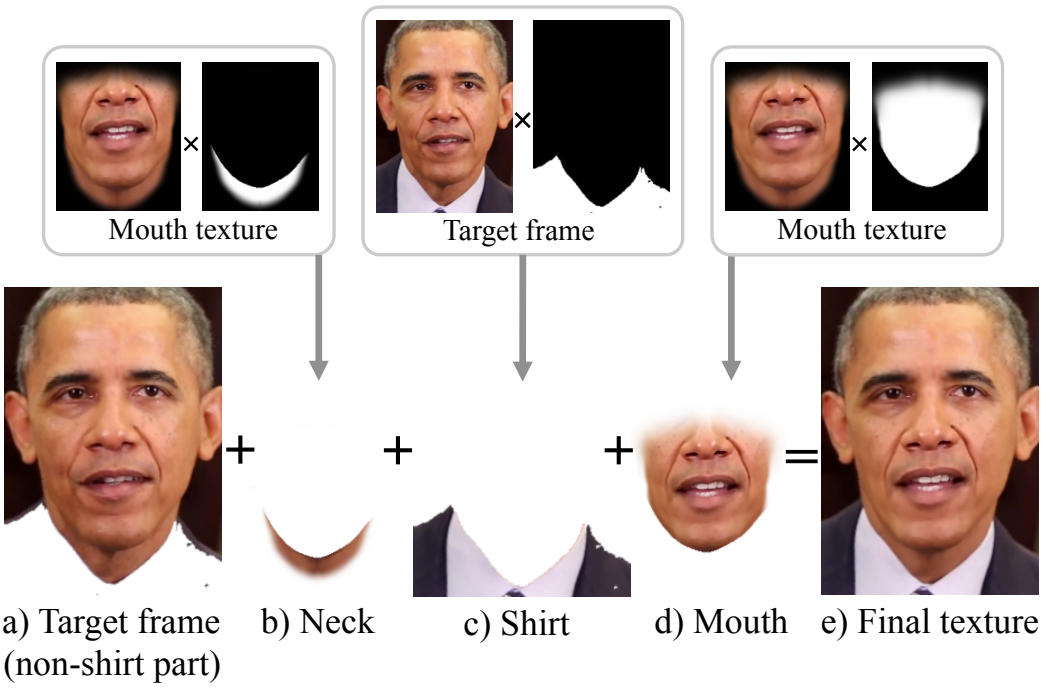


Figure 5.8: The final composite is produced by pyramid blending of the following layers from back to front: a) the target frame, b) the neck region under the chin in the mouth texture, c) Obama's shirt from the target frame, d) the mouth.

the synthesis part (Section 5.3.2).

5.4 Experiments

In this section, we describe implementation details, evaluations, comparisons to prior work, limitations, and applications.

Running times and hardware: We report the following runtime based on NVIDIA TitanX for RNN inference, and Intel Core i7-5820K for other computations. For a source audio of 66 seconds in length, on a single CPU core, it took 45 minutes in total to produce a 30fps output video. The breakdown is as follows: 5 seconds to run RNN inference and generate 1980 mouth shapes (30fps for 66 seconds); mouth texture synthesis took 0.1s per frame (3.3 minutes total); and the final composite including chin correction, masking, and rendering took 0.35s per frame (11.5 minutes total). The retiming dynamic programming solution took 0.2s for the entire sequence, with an additional 4s per repeated frame for optical flow interpolation [114]. In practice, we parallelized most computations on a 24-core CPU and reduced the runtime from 45 to 3 minutes total (0.1s per frame).

5.4.1 LSTM Architecture and Data

For training we downloaded 300 weekly addresses available online[2] spanning 2009 to 2016. Each address lasts about 3 minutes on average, resulting in total of 17 hours of video. We extracted frames at 30fps and obtained around 1.9 million video frames. We randomly split out 20% of the addresses (3 hours) for validation and used 80% (14 hours) for training.

Our network consists of 60 LSTM nodes (dimension of c) and uses a 20 step time-delay d, corresponding to 200ms. We train the network with a batch size of 100 using truncated backpropagation through time with 100 time steps. We use the ADAM optimizer [102] with learning rate 0.001, implemented in TensorFlow [9]. Each dimension in the input vector is normalized by its mean and variance, but the output is unnormalized to keep the relative

[2] `https://www.whitehouse.gov/briefing-room/weekly-address`

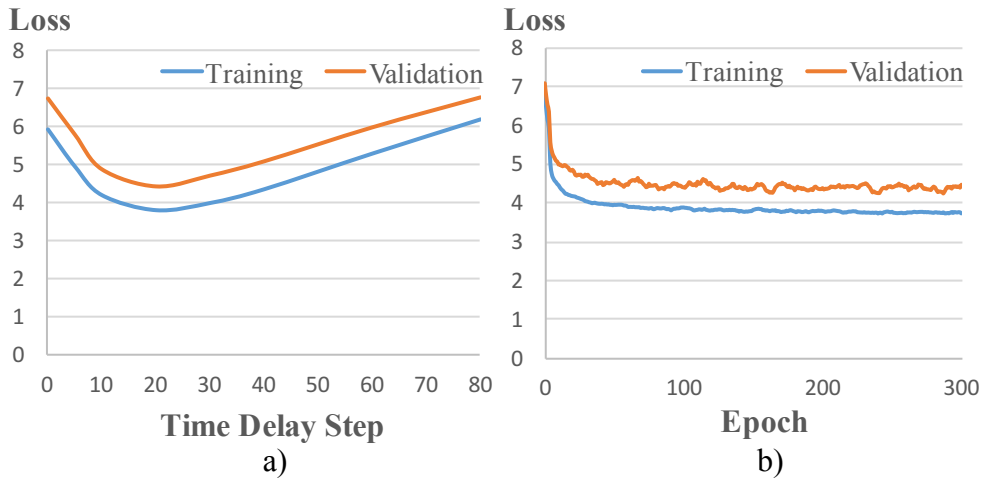


Figure 5.9: a) shows the losses at the end of the training of networks with varying time delay steps from 0 to 80 (800ms) trained with 300 unfold time steps. b) plots loss during training of our single-layer LSTM network with 20 time delay steps.

importances of the PCA coefficients. Training took 3 hours in total for 300 epochs on a NVIDIA TitanX.

We found that augmenting the LSTM with a time delay was critical for improving validation loss and visual quality. This modification effectively increases the receptive field beyond that of the MFCC window (25ms) to at least 200ms of future audio context. Figure 5.9 shows validation losses by varying the time delay steps for our single-layer 60-node LSTM network. Without the time delay, both the training and validation losses are high and the visual lip-sync quality is poor. Table 5.1 shows validations losses for varying time delays as well as the number of LSTM nodes. We found that for our architecture, the time delay of 200ms gives consistently lower validation losses across different numbers of LSTM nodes. Performance decreases beyond 200ms, likely due to the need to propagate information further across time.

Other network architectures are evaluated in Table 5.2 by keeping the time delay at 200ms and varying the number of stacked layers, LSTM nodes, and the dropout probability of the standard RNN regularization [192].

Time Delay:	50ms	100ms	200ms	400ms
L1-30	5.824	4.946	4.572	5.147
L1-60	5.782	4.877	4.400	5.089
L1-90	5.726	4.841	4.391	5.009
L1-120	5.732	4.946	4.572	5.147

Table 5.1: Validation losses for single-layer (L1) networks with varying (30, 60, 90, and 120) LSTM nodes and time delays.

Dropout probability:	0	0.1	0.3	0.5
L2-30	4.449	4.587	4.881	5.252
L2-60	4.389	4.420	4.621	4.923
L2-90	4.403	4.347	4.498	4.754
L3-30	4.409	4.548	4.850	5.237
L3-60	4.402	4.386	4.585	4.881
L3-90	4.439	4.310	4.487	4.718

Table 5.2: Validation losses for two (L2) and three (L3) layers networks with various LSTM nodes and dropout probability.

Additionally, we explored other regularization techniques, e.g. variational RNN dropout [67] on a few configurations but did not find a major improvement. These validation losses have high variance and do not necessarily translate to better visual lip-sync quality after reaching a certain value. One reason is that we do not have ground-truth mouth landmarks and instead rely on landmark estimates from imperfect algorithms. While we filtered out clear failure cases such as when the mouth shape is irregular or not detected, other error cases remain in both the training and validation sets. As such, zero loss does not equate to perfect visual results. We believe recently developed techniques such as recurrent batch normalization can further improve the loss, but larger accuracy gains may require improving the landmark data. For our purpose, we opt for the simplest network (L1-60, with 200ms delay) that achieves a similar loss and empirically similar visual quality to the more complex models.

Finally, we evaluated the effect of varying the amount of training data on the quality of the output video. We trained our network with: 0.35% of the data (3 minutes total), 10% (1 hour total), 50% (7 hours), and the full test dataset (14 hours). The supplementary video shows how quality improves with more training data. In particular, with more training data, lip-sync quality improves significantly and mouth jitter is reduced. There is significant improvement at each step, even from 7 hours to 14, indicating that having a large amount of training data is critical.

5.4.2 Lower Face Synthesis Evaluation

Figure 5.10 shows comparison of our synthesis algorithm to the classic AAM approach [52] that is prevalent in visual speech synthesis, and to a recent detail-enhancing Laplacian pyramid technique [158]. We observe that both AAM and [158] show significant blurriness. The blurriness appears due to use of data captured in uncontrolled and uncalibrated conditions, i.e., faces can be non-frontal, under different lighting, etc. The results in the figure are computed using the same set of frontalized face images as used in our algorithm, and even if lighting-invariant optical flow (i.e., collection flow [96] is performed as in [158]), the resulted synthesis is blurry in the teeth area due to fine misalignment across photos.

Figure 5.10 also compares weighted mean, median and mode, for generating facial texture from the same set of image candidates. Mean produces the blurriest result among all three, and mode appears noisy even when the sparse sampling (i.e., low number of candidates) is handled by using larger frequency bins or counting by merging nearby points in color space. This behavior can also be understood in an optimization framework where mean, median, and mode correspond to a minimization with L^2, L^1, L^0 norms, respectively. In the case of mode, the summation of the quasi-convex L^0 has multiple minima and is sensitive to image noise and misalignment, whereas L^2 produces an over-smooth average. Median strikes a good balance between these two, which we've seen in practice translates to better edge-preserving properties than mean and is less noisy than mode.

In Figure 5.11 we compare to the recent Face2face algorithm [165]. We provide the same

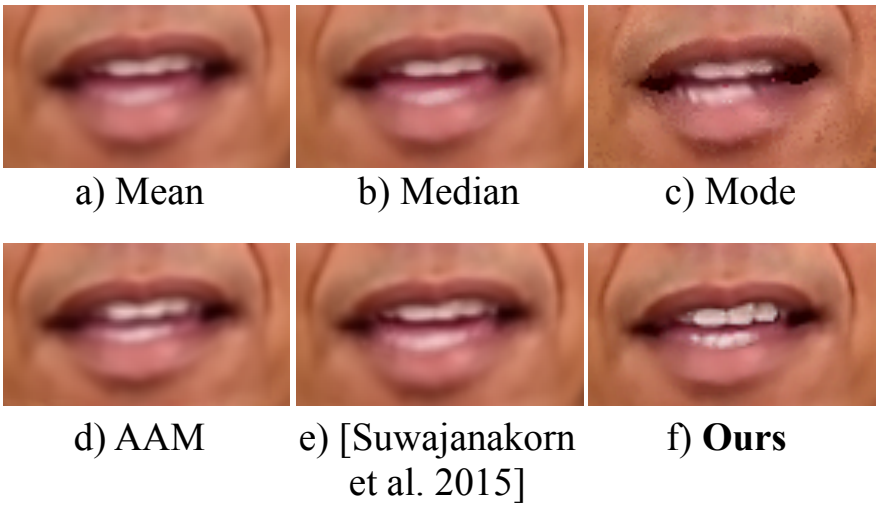


Figure 5.10: Mouth synthesis comparison to weighted-mean (a), weighted-median (b), weighted-mode (c), AAM-based techniques (d), and [158] (e). For all results shown here, we first frontalize all training images using the same frontalization technique as our result, and for (a,b,c,e), we use identical weights to ours computed from Equation 5.11. Notice how other techniques produce blurry results on our training dataset that contains mouth images from a real speech with natural head motion.

source video (a weekly presidential address) to both methods. Note that we use only the source audio as input, whereas their technique requires the source video–i.e., they effectively have access to the ground truth mouth appearance. In addition, we provide the same target video (a different weekly presidential address) to both methods. The Face2face were produced by the authors running their original system on our videos. The differences between the two methods are best viewed in the supplementary video. Some observations: our method tends to produce more realistic and *Obama-like* lip and head motion. Additionally, our method captures time-varying wrinkles and creases around the mouth whereas Face2face's texture for the surrounding skin appears more static. The teeth produced by Face2face sometimes appear non-rigid with occasional ghosting artifacts, whereas teeth produced by our method appear rigid and temporally smoother. The differences perhaps expected, since our system is trained on many videos of Obama and tuned to produce high quality realistic results while

a) Face2face [Thies et al. 2016]

b) **Our results**

Figure 5.11: Comparison to Face2face [165] for four different utterances in the same speech using the same source video. Note that [165] requires the video of the input speech to drive the animation and focuses on the real-time puppetry application whereas ours aims to synthesize a visual speech given only a recorded audio of Obama. Notice how our method can synthesize more realistic mouths with natural creases around the mouth. The differences between the two approaches are best viewed in the supplementary video.

Face2face is aimed at a real-time puppetry scenario that uses only two videos (source and target).

In addition to the above comparisons, we compiled a 4 minute video showing related face animation results of the last 20 years. The end of our supplementary video includes this compilation, as a helpful visual reference for our work.

5.4.3 Target Video Re-timing Evaluation

In the supplementary video (Video D) we show results generated with and without re-timing (Section 5.3.3). Since our dynamic programming approach also solves for the best

starting frame of the target video, for comparison we start both target videos at this same frame. Without retiming, there are occasional head motion and expression changes during vocal pauses which appears unnatural, as indicated by the red arrow. We also evaluate consistency of the re-timing approach across different target videos (Video E in sup. video), by generating output videos with four different targets. Most of the time, all four results start moving when Obama starts a new sentence and stop during long pauses. Occasionally, Obama moves slightly during small pauses, but fast motion during pauses is avoided.

5.4.4 Results & Applications

All of the result figures and supplementary videos are generated from input speeches that are not in the training or validation sets. In Figure 5.12, we run on the audio of Obama weekly address "Standing with Orlando" from YouTube with ID[3] nIxM8rL5GVE and use four different target videos also from weekly addresses (E3gfMumXCjI, 3vPdtajOJfw, 25GOnaY8ZCY, k4OZOTaf3lk). Fully generated videos for each of the targets are presented in supplementary Video E. Our results are realistic with convincing, natural head motion. When compared to prior techniques (Video I), our mouth texture looks sharper than those that rely on AAM and more temporally smooth than those that use frames from original footage. The upper teeth look sharp and realistic. We do note that the lower teeth and tongue appear blurrier and less plausible when compared to the ground-truth footage, e.g., in Video A. In Figure 5.13 and Video F, we compare our synthesized textures to the original video of the input speech. Our results show good agreement in terms of the mouth shapes, timing, and overall mouth animation.

To evaluate generalization to other types of input speech not from the weekly addresses on which we trained, we ran our system on the audio of Obama's interview with Steve Harvey (qMlLjFPCO4M), 60 Minutes (F8MxP9adPO8), and the View (Hdn1iX1a528). Our method can handle casual speech and generates plausible mouth animation even during a

[3]YouTube video can be accessed by `https://www.youtube.com/watch?v=ID` given video ID.

mumble or hesitation. We also test on the voice of a quarter century younger Obama from 1990 (7XGi3FGVmA0) and on a voice impressionist (vSAA5GH6OFg). The lipsync quality still looks reasonable although it degrades somewhat as the voice deviates from our training audio.

One useful application of our method is speech summarization, inspired by [25]. I.e., given a long address speech, create a short summary version by manually selecting the desired sections from transcribed text. Our method can then be used to generate a seamless video for the summarized speech shown in Video G. While our result looks similar to [25] for this particular example, an advantage of our system is that we can produce visually seamless cuts even when the head position, lighting, or background has changed between concatenated source footage clips.

5.4.5 Failure Cases & Limitations

Below are some limitations of our method.

3D geometry errors: During the final composite, our mouth texture is composited over a target frame that may have a different mouth shape and chin location. Usually, our optical flow approach successfully aligns the two chins, but occasionally fails, e.g., when the chin occludes part of his shirt, and produces a double chin artifact shown in Figure 5.14a. Addressing this problem requires properly modeling the occluded shirt regions. Similarly, when the target pose is non-frontal, imperfect 3D face geometry can cause the mouth texture to be composited outside the face and onto the background (Figure 5.14b). Even though our method can cope with the reasonable range of head poses presented in the weekly address-style speech, our imperfect head geometry model prevents us from rendering onto a target video with extreme poses such as profiles.

Target video length: Our face texture synthesis approach relies on a full set of mouth shapes being available in the target video, enough to span the mouth shapes needed for the source audio. This restriction may limit the types and length of target video we can use.

Emotion modeling: Our method does not explicitly model emotions or predict the

sentiment of the input speech. Thus Obama's facial expressions in the final output video can appear too serious for a casual speech, or too happy for a serious speech. Some people feel, for example, that our synthesized rendition of Obama's *Standing with Orlando* speech at the beginning of the supplementary video looks too upbeat.

Tongue modeling: Our mouth texture synthesis assumes that the mouth texture can be fully determined by positions of lip fiducials. This may not be entirely true for some sounds such as 'th' that require the use of the tongue, which may be hard to distinguish based purely on lip fiducials.

5.5 Discussion

We show that by training on a large amount of video of the same person, and designing algorithms with the goal of photorealism in mind, we can create believable video from audio with convincing lipsync. This work opens up a number of interesting future directions, some of which we describe below.

Our pipeline includes one manual step that the user must perform for each target video: selecting and masking a teeth proxy. We believe this step could be automated by training a teeth detector (looking for a large, clear white teeth image)

Our method relies on MFCC audio features, which are not designed specifically for visual synthesis. This suggests that an even more end-to-end network may be able to achieve even better quality by going directly from raw audio waveforms to mouth shapes or textures. For example, [170] achieves better performance in natural audio generation by replacing MFCC and RNN with dilated convolution. It would be interesting to see how such a network could be applied to our audiovisual synthesis task. Similarly, it would be interesting to see if the network could learn to predict emotional state from audio to produce corresponding visuals (e.g., happy, sad, angry speech, etc.).

Training our system on another person, such as a non-celebrity, can be quite challenging due to the difficulty of obtaining hours of training data. However, the association between mouth shapes and utterances may be, to some extent, speaker-independent. Perhaps a

network trained on Obama could be retrained for another person with much less additional training data. Or perhaps a single universal network could be trained from videos of many different people, and then conditioned on individual speakers, e.g., by giving it a small video sample of the new person, to produce accurate mouth shapes for that person.

While we synthesize only the region around the mouth and borrow the rest of Obama from a target video, a more flexible system would synthesize more of Obama's face and body, and perhaps the background as well. Such a system could enable generating arbitrary length sequences, with much more control of how he moves and acts.

Figure 5.12: Results for the same input speech using four different target videos. Results along each column are generated from the same utterance.

Original Video for Input Audio

a) **Our result**

Original Video for Input Audio

b) **Our result**

Figure 5.13: Comparison of our mouth shapes to the ground-truth footage of the input audio (note good agreement–more results in supplemental video). a) is an interview from "60 minutes," and b) is a weekly address on health care.

a) Double chins b) Color bleeding

Figure 5.14: a) shows a double chin artifact. This happens when the chin of the target video is lower than our synthesized chin and occludes part of the shirt. b) shows a mouth texture bleeding onto the background.

Chapter 6

CONCLUSION

This thesis has presented methods for persona reconstruction from large unconstrained data of a single person with the goal of recreating a digital avatar that looks and acts like the person. The specific aspects of persona modelled in this thesis are facial shapes and appearances, motion and expressions, the aging process, and the lip synchronization and visual speech synthesis from audio. Below, I summarize each contribution.

- **Reconstruction of Time-Varying Facial Geometry.** I present an approach that takes a single photo or video of a person's face and reconstructs a highly detailed 3D model for each photo or video frame. The approach targets videos and photos taken under uncontrolled and uncalibrated imaging conditions, such as YouTube videos of celebrities. At the heart of this work is a new dense 3D flow estimation method, coupled with shape from shading. Unlike prior work, we do not assume the availability of a blend shape model, nor require the person to participate in a training/capturing process. Instead we leverage the large amounts of photos that are available in personal or Internet photo collections and demonstrate results for a variety of video sequences that include different lighting conditions, head poses, and facial expressions.

- **Synthesizing Expression-Dependent Textures.** I present a technique to synthesize facial textures for reconstructed geometry that are 1) sharp and detailed, 2) expression-dependent, and 3) consistent in the overall color by using a novel combination of a pixel-level dense alignment and frequency-based blending technique. Together with the time-varying geometric reconstruction, this technique represents the first system capable of building a complete, dynamic, textured model automatically from large

photo collections. Such a model can be rendered with any expression from any viewing angle with realistic appearance changes such as creases and wrinkles.

- **Facial Puppetry.** I propose a technique to realistically drive or puppeteer a reconstructed model of person A using any other video of a different actor B. Based on an illumination subspace matching technique, my algorithm finds vertex-pixel correspondence between the puppet model and the driving input video, to transfer expressions from an actor to a puppet while preserving the puppets appearance. This ability to reconstruct and drive a puppet of anyone given only their photo collections opens up many applications such as creating personal CG characters for movies and video games and driving the facial motion of an animated character in Virtual Reality applications. I have demonstrated convincing results on a large variety of celebrities derived from Internet imagery and video.

- **Age Progression.** I present an approach that predicts the facial changes due to aging. Specifically, the method takes a single photograph of a child as input and automatically produces a series of age-progressed outputs between 1 and 80 years of age, accounting for pose, expression, and illumination. Leveraging thousands of photos of children and adults at many ages from the Internet, I first show how to compute average image subspaces that are pixel-to-pixel aligned and model variable lighting. Key contributions include a new state-of-the-art for the most difficult aging case of babies to adults, relightable age subspaces, a novel technique for subspace-to-subspace alignment, and the most extensive evaluation of age progression techniques in the literature.

- **Visual Speech Synthesis.** I present a technique to synthesize a realistic high-quality video of a person speaking from input audio with accurate lip sync, demonstrated with a case study on President Barack Obama. Unlike prior work which requires a speech database consisting of video recordings of many subjects speaking predetermined sen-

tences, my solution solves the video speech problem by requiring only existing video footage of a single person. In particular, it combines computer graphics techniques and an LSTM-based recurrent neural network trained on 17-hour of Barack Obama's footage to synthesize a high-quality mouth video of him speaking. Our approach generates compelling and photorealistic videos that could enable a wide range of applications such as lip-reading for hearing-impaired people and video bandwidth reduction.

6.1 Future Work

In this section, I present some open problems and possible solutions to help improve performance or relax some of the assumptions in our proposed methods. Several interesting future directions toward the goal of recreating a digital human are also discussed.

6.1.1 Bas-Relief Ambiguity

One source of inaccuracy in our reconstructed shape in Chapter 2 comes from the bas-relief ambiguity [22], which is defined as the shape ambiguity of a reconstructed Lambertian object when the light directions are unknown. This is caused by the fact that we can render a transformed object to look identical to the original object from one viewpoint by simultaneously transforming the light directions and shape with the generalized bas-relief transformation $f'(x, y) = \lambda f(x, y) + ux + vy$ of a surface $f(x, y)$. However, this ambiguity can be resolved if multiple views of an object are seen. For faces, two of the three bas-relief parameters (λ, v) can be resolved by matching the model to a profile view. To completely resolve the ambiguity, one way is to reconstruct a sparse shape that includes highly distinctive and static features such as eye corners and ears using structure-from-motion algorithms and use the resulting shape to solve for the ambiguity parameters.

6.1.2 Head Reconstruction

Our proposed techniques focus on the reconstruction of faces and cannot render a complete head model from an arbitrary angle (In Chapter 5, we render a full video using a video

composite technique not with a full 3D model). Some digital human applications require a complete head model including hair and possibly the upper body. Recently, [110] proposes a head modeling technique from Internet photos based on combining shapes reconstructed using photometric stereo from multiple views. However, it cannot infer unseen regions like the top of the head and treats hair as a rigid part of the head geometry. Another promising technique in [45] generates a dynamic 3D avatar with a complete head and hair as well as fine scale details in real time, but it requires controlled captures of a user with predefined poses and expressions as input. Similarly, [87] can generate a fully rigged, personalized 3D facial avatars with a complete head from hand-held video input.

It would be interesting to combine the strengths of these techniques and still operate on unconstrained data. For example, to extend [110] and infer unseen regions, perhaps blendshapes could be used as a shape prior. Alternatively, a technique based on implicit volumetric shape prior such as Wulff shape [118] could be used to recover details that are usually lost using low-rank blendshape models and produce a useful segmentation of head into semantic parts such as skin, hair, and eyebrows.

6.1.3 Learning High Resolution Textures

The resolution of our synthesized texture directly depends on the resolutions of the collection of input photos. Can we synthesize a higher resolution texture with realistic and compelling details for any expression by relying on a smaller number of high-resolution photos or a large database of high-resolution face photos of other people? A new data-driven approach by [141] introduces an inference method that can generate high-resolution textures with details from a single unconstrained image. The method works by combining low-frequency albedo from the input image and high-frequency details inferred from a database of high-resolution face images. One open problem is how to create a single consistent set of high resolution textures of a person corresponding to various expressions with the ability to animate or interpolate between them. This ability is crucial in facial puppetry and digital human applications.

A key challenge is that high-frequency details produced independently for each expres-

sion can appear different at mesoscopic scale (e.g., freckles, pores) and cause flickering due to misalignment. Matching details across different textures for interpolation is extremely challenging even with state-of-the-art optical flow algorithms. Possible directions include incorporating low-frequency facial motion, e.g. obtained through optical flow, to guide the reconstruction of details through an interpolated correlation matrix (see [139]) or exploring a novel way to encode spatial information into the network architecture to facilitate motion interpolation.

6.1.4 Teeth & Tongue Modeling

An accurate rendering of teeth and tongue is crucial for compelling lip sync. Even though the current teeth proxy method performs well for the upper teeth which are stationary with respect to the head, it produces blurry results for the lower teeth which are in frequent motion. Our mouth texture synthesis approach also assumes that the mouth texture can be fully determined by positions of lip fiducials. This assumption may not be entirely true for some sounds such as 'th' that require the use of the tongue, which may be hard to distinguish based purely on lip fiducials. A possible solution is to also regress the position of the tongue in the recurrent neural network that produces the mouth shape. Alternatively, a joint representation of shape and appearance that does not suffer from a detail loss like AAM[52] does, is needed to preserve high resolution details in the output video. A new method for full 3D teeth reconstruction from a sparse set of photos has shown promising results [182] and could be incorporated in the rendering pipeline.

6.1.5 Learning the Space of Facial Shape & Appearance

With the accurate pose and shape estimation techniques introduced in Chapter 2, we can register and warp an entire personal photo collection to a frontal view (or some other canonical views) using the reconstructed 3D model. Then by applying principle component analysis to the set of aligned photos, we can explore interesting modes of appearances. For example, Figure 6.2 shows the 6th and 9th principle components of Tom Hanks photos which capture

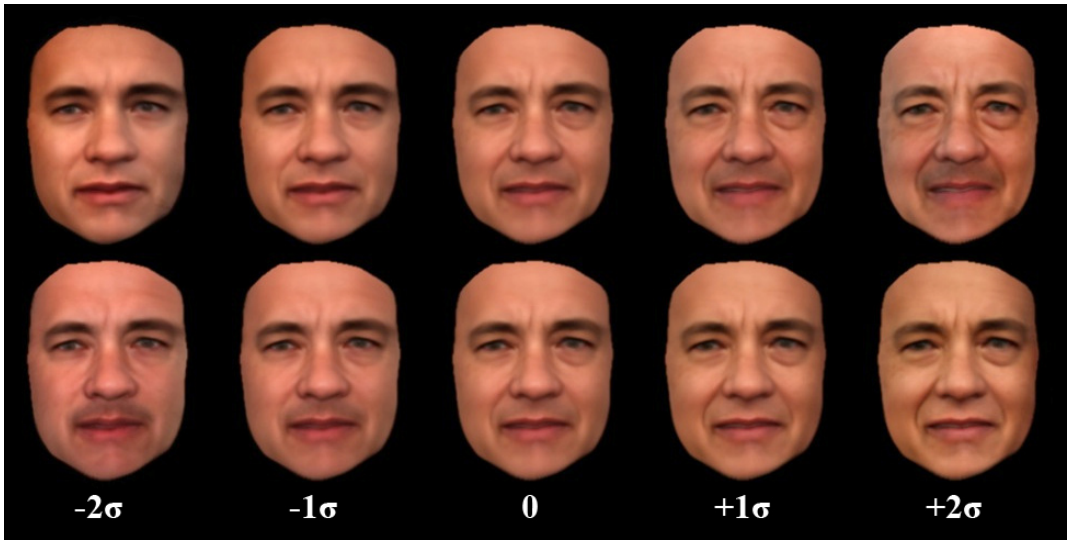


Figure 6.1: The first row shows the effect of adding $-2\sigma, -1\sigma, 0, 1\sigma, 2\sigma$ of 6th principle component vector to the average face where σ is the 6th singular value. The second row is generated similarly using the 9th principle component. The first row shows the aging effect while the second row shows an expression change.

the aging effect and expression change. Similarly, PCA can be applied to a set of 3D shapes reconstructed from a video sequence to capture expression changes.

A limitation of PCA, however, is its reliance on the linear assumption. PCA may only factor out modes that are well represented linearly, but fails to discover other semantically meaningful modes that are non-linear. An interesting topic to explore is how to better model the shape and appearance spaces. Can we discover interesting and semantically meaningful modes using other means such as clustering algorithms or locally linear embedding? Another promising solution may lie in unsupervised learning techniques such as Generative Adversarial Networks [74] or Variational Auto-Encoder [103] whose goal is to learn compact, disengangled representation in an unsupervised manner.

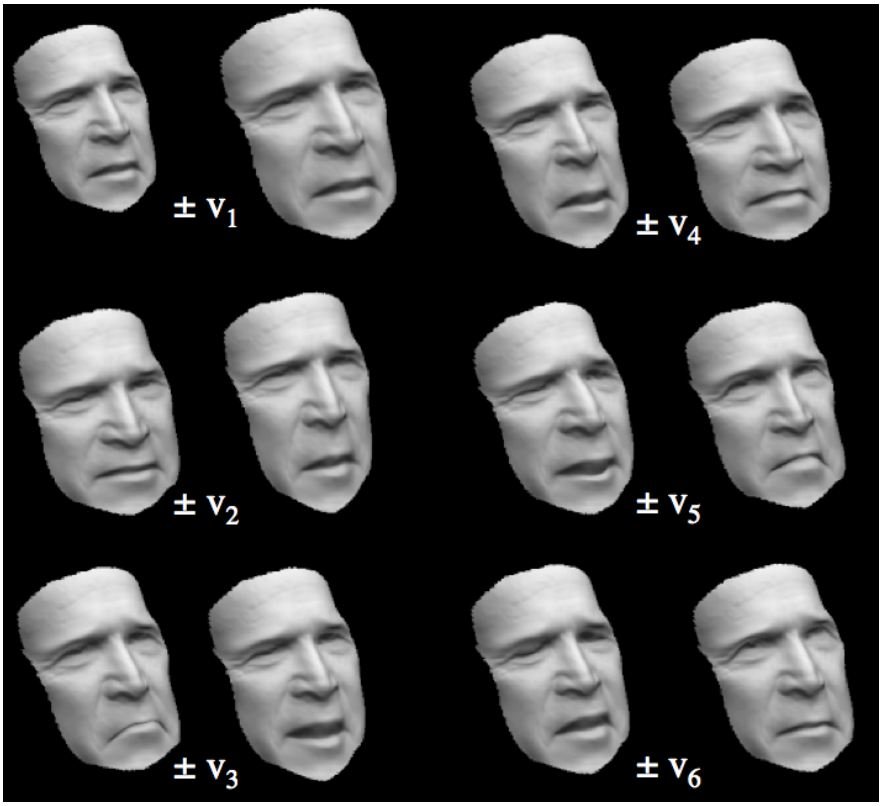


Figure 6.2: Each pair shows the effect of adding the first through sixth principle components to the mean shape of George W. Bush.

6.1.6 Age progression

In Chapter 4, I introduced a technique that generates a plausible age-progressed photo of a child. Age progression, however, is not a deterministic process and there could be many physically plausible outcomes. Instead of restricting the prediction to one single output, an interesting direction would be to predict a distribution of possible outcomes, e.g., through a mixture model. Furthermore, we can try to learn not just the average transformation between a pair of age groups but a transformation conditioned on the input image to account for identity. Traditionally, learning such a transformation would require a database of people that span many years for each person. The training photos gathered from the Internet in Chapter 4, however, are "unpaired" meaning that each individual may only appear once in

their age group.

Fortunately, this kind of unsupervised training set without input-output pairs fits into a recently developed framework of cycle-consistent adversarial networks [196]. [196] introduces additional cycle consistency losses in generative adversarial networks [74] which allows a transformation of an image in one class to another without explicit training pairs. Such a transformation may also handle other regions beside the face such as the hair automatically. Combining this with a mixture density network [26] to predict a set of plausible age-progressed outcomes could be a promising direction.

6.1.7 End-to-end Lip Sync Generation

Our visual speech synthesis method in Chapter 5 relies on MFCC audio features, which are not designed specifically for visual synthesis. This suggests that an even more end-to-end network may be able to achieve even better quality by going directly from raw audio waveforms to mouth shapes or textures. For example, WaveNet [170] achieves better performance in natural audio generation by replacing MFCC and RNN with dilated convolution. It would be interesting to see if such a network could be applied to our audiovisual synthesis task and if it could learn to predict emotional state from raw audio to produce corresponding visuals (e.g., happy, sad, angry speech, etc.). Speech animation work that incorporate emotions have been proposed [46, 163], although extending these techniques to unstructured imagery remains an open problem.

6.1.8 Generalizing Lip Sync from Audio

Training the visual speech synthesis system on another person, such as a non-celebrity, can be quite challenging due to the difficulty of obtaining hours of training data. Prior speech systems designed to generalize to multiple speakers relied on phonemes as an intermediate representation [34, 60, 144, 119, 142, 64, 186]. However, automatic phoneme labeling tends to be error-prone and requires a manually labelled dataset for every new language. Interestingly, the association between mouth shapes and utterances is, to some extent, speaker-

independent. One example is the sound "O" where universally everyone has to open the mouth. Perhaps a network trained on Obama could be retrained for another person with much less additional training data. Going a step further, it would be interesting to explore a single universal network that could be trained on videos of many different people, and then conditioned on individual speakers, e.g., by giving it a small video sample of the new person, to produce accurate mouth shapes for that person.

6.1.9 Modeling Interactions

One very exciting futuer challenge of persona reconstruction is interaction. How can we make the model talk and respond in a way that resembles the person? What should the model say when being asked a question? Solving this would require a novel integration of a natural language processing system that analyzes and conducts a conversation, a text-to-speech system that turns the text response from the NLP system into a personalized voice, and a computer graphics system that renders a convincing animation of the model saying those words. On the computer graphics side, further research is needed to generate arbitrary length animations that correspond to any input speech of indefinite length. For example, apart from the lip sync, we also want to capture reactions such as a slight nod of someone listening to a question or how they make certain gestures as they speak. A first step could be to explore how to model basic aspects such as the head motion during a conversation, e.g., by employing an auto-regressive recurrent neural network [75] to predict the head motion from the input audio. Additionally, it would be interesting to see if the network could learn to generate an expression animation that corresponds to the content or sentiment of the input speech.

BIBLIOGRAPHY

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

[2] Altspacevr. https://altvr.com/. Accessed 2017-06-03.

[3] The curious case of reverse-aging brad pitt. https://www.wired.com/2008/12/pl-screen-16/. Accessed 2017-06-06.

[4] Ibm watson speech to text. https://speech-to-text-demo.mybluemix.net/. Accessed: 2016-01-16.

[5] Oben, inc. creating your personalized and realistic virtual identity. https://oben.me/. Accessed 2017-06-05.

[6] President Barack Obama weekly address. https://www.whitehouse.gov/briefing-room/weekly-address. Accessed: 2016-01-16.

[7] Ted: The magic of benjamin button. https://www.wired.com/2009/02/ted-the-magic-o/. Accessed 2017-06-06.

[8] Call of duty advanced warfare trailer - kevin spacey. https://www.youtube.com/watch?v=VKYHuhg0l3I, 2014. Accessed 2017-06-03.

[9] Martın Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

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

[11] 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.

[12] Oleg Alexander, Mike Rogers, William Lambeth, Jen-Yuan Chiang, Wan-Chun Ma, Chuan-Chang Wang, and Paul Debevec. The digital emily project: Achieving a photorealistic digital actor. *Computer Graphics and Applications, IEEE*, 30(4):20–31, 2010.

[13] 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.

[14] Robert Anderson, Björn Stenger, Vincent Wan, and Roberto Cipolla. An expressive text-driven 3d talking head. In *ACM SIGGRAPH 2013 Posters*, page 80. ACM, 2013.

[15] Robert Anderson, Bjorn Stenger, Vincent Wan, and Roberto Cipolla. Expressive visual text-to-speech using active appearance models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3382–3389, 2013.

[16] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[17] 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.

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

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

[20] 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.

[21] 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.

[22] Peter N Belhumeur, David J Kriegman, and Alan L Yuille. The bas-relief ambiguity. *International journal of computer vision*, 35(1):33–44, 1999.

[23] Fabrice Bellard, M Niedermayer, et al. Ffmpeg. *Availabel from: http://ffmpeg. org*, 2012.

[24] P.J. Benson and D.I. Perrett. Extracting prototypical facial images from exemplars. *Perception*, 22:257–262, 1993.

[25] Floraine Berthouzoz, Wilmot Li, and Maneesh Agrawala. Tools for placing cuts and transitions in interview video. *ACM Trans. Graph.*, 31(4):67–1, 2012.

[26] Christopher M Bishop. Mixture density networks. 1994.

[27] 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.

[28] 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.

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

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

[31] G. Bradski. *Dr. Dobb's Journal of Software Tools*, 2000.

[32] Matthew Brand. Voice puppetry. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '99, pages 21–28, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co.

[33] 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.

[34] Christoph Bregler, Michele Covell, and Malcolm Slaney. Video rewrite: Driving visual speech with audio. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 353–360. ACM Press/Addison-Wesley Publishing Co., 1997.

[35] 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.

[36] 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.

[37] 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.

[38] D.M. Burt and D.I. Perrett. Perception of age in adult caucasian male faces–comp graphic manipulation of shape and color information. *Pr. Royal S. London*, 259:137–143, 1995.

[39] Peter J Burt and Edward H Adelson. A multiresolution spline with application to image mosaics. *ACM Transactions on Graphics (TOG)*, 2(4):217–236, 1983.

[40] Oleg Alexander Graham Fyffe Jay Busch, Xueming Yu, Ryosuke Ichikari Andrew Jones Paul Debevec, and Jorge Jimenez Etienne Danvoye Bernardo Antionazzi. Digital ira: Creating a real-time photoreal digital actor.

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

[42] 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.

[43] 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.

[44] 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.

[45] 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.

[46] 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.

[47] 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.

[48] Antonin Chambolle and Thomas Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *J. Mathematical Imaging and Vision*, 40(1):120–145, 2011.

[49] Yao-Jen Chang and Tony Ezzat. Transferable videorealistic speech animation. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 143–151. ACM, 2005.

[50] Jan K Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio. Attention-based models for speech recognition. In *Advances in Neural Information Processing Systems*, pages 577–585, 2015.

[51] T Cootes and A Lanitis. The fg-net aging database, 2008.

[52] Timothy F. Cootes, Gareth J. Edwards, and Christopher J. Taylor. Active appearance models. *IEEE Transactions on pattern analysis and machine intelligence*, 23(6):681–685, 2001.

[53] 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.

[54] 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.

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

[56] 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.

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

[58] Tony Ezzat, Gadi Geiger, and Tomaso Poggio. *Trainable videorealistic speech animation*, volume 21. ACM, 2002.

[59] 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.

[60] Bo Fan, Lijuan Wang, Frank K Soong, and Lei Xie. Photo-real talking head with deep bidirectional lstm. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4884–4888. IEEE, 2015.

[61] Bo Fan, Lei Xie, Shan Yang, Lijuan Wang, and Frank K Soong. A deep bidirectional lstm approach for video-realistic talking head. *Multimedia Tools and Applications*, pages 1–23, 2015.

[62] Leslie G. Farkas. *Anthropometry of the Head and Face*. 1994.

[63] FGNET. http://www.fgnet.rsunit.com/, 2009.

[64] Shengli Fu, Ricardo Gutierrez-Osuna, Anna Esposito, Praveen K Kakumanu, and Oscar N Garcia. Audio/visual mapping with cross-modal hidden markov models. *IEEE Transactions on Multimedia*, 7(2):243–252, 2005.

[65] Yun Fu, Guodong Guo, and T.S. Huang. Age synthesis and estimation via faces: A survey. *PAMI*, 32(11):1955–1976, 2010.

[66] Graham Fyffe, Andrew Jones, Oleg Alexander, Ryosuke Ichikari, and Paul Debevec. Driving high-resolution facial scans with video performance capture. *ACM Transactions on Graphics (TOG)*, 34(1):8, 2014.

[67] Yarin Gal. A theoretically grounded application of dropout in recurrent neural networks. *arXiv preprint arXiv:1512.05287*, 2015.

[68] F.J. Galton. Composite portraits. *Nature*, 18:97–100, 1878.

[69] 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.

[70] 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.

[71] Pablo Garrido, Levi Valgaerts, Hamid Sarmadi, Ingmar Steiner, Kiran Varanasi, Patrick Perez, and Christian Theobalt. Vdub: Modifying face video of actors for plausible visual alignment to a dubbed audio track. In *Computer Graphics Forum*, volume 34, pages 193–204. Wiley Online Library, 2015.

[72] 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.

[73] 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.

[74] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[75] Alex Graves. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*, 2013.

[76] Alex Graves, Navdeep Jaitly, and Abdel-rahman Mohamed. Hybrid speech recognition with deep bidirectional lstm. In *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*, pages 273–278. IEEE, 2013.

[77] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on*, pages 6645–6649. IEEE, 2013.

[78] Alex Graves and Jürgen Schmidhuber. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks*, 18(5):602–610, 2005.

[79] 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.

[80] Guodong Guo and Guowang Mu. Simultaneous dimensionality reduction and human age estimation via kernel partial least squares regression. In *CVPR*, 2011.

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

[82] 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.

[83] H. Heafner. Age-progression technology and its application to law enforcement. In *SPIE*, pages 49–55, 1996.

[84] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[85] 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.

[86] Xuedong Huang, Fileno Alleva, Hsiao-Wuen Hon, Mei-Yuh Hwang, Kai-Fu Lee, and Ronald Rosenfeld. The sphinx-ii speech recognition system: an overview. *Computer Speech & Language*, 7(2):137–148, 1993.

[87] 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.

[88] Andrew Jones, Jonas Unger, Koki Nagano, Jay Busch, Xueming Yu, Hsuan-Yueh Peng, Oleg Alexander, and Paul Debevec. Creating a life-sized automultiscopic morgan spurlock for cnns inside man. In *ACM SIGGRAPH 2014 Talks*, page 2. ACM, 2014.

[89] Ben Jones and Lisa DeBruine. Face research online tool, http://www.faceresearch.org/demos/, 2013.

[90] Andrej Karpathy. Multi-layer recurrent neural networks (lstm, gru, rnn) for character-level language models in torch. `https://github.com/karpathy/char-rnn`. Accessed 2017-06-05.

[91] Masahide Kawai, Tomoyori Iwao, Daisuke Mima, Akinobu Maejima, and Shigeo Morishima. Data-driven speech animation synthesis focusing on realistic inside of the mouth. *Journal of information processing*, 22(2):401–409, 2014.

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

[93] 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.

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

[95] 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.

[96] 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.

[97] 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.

[98] 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.

[99] 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.

[100] Taehwan Kim, Yisong Yue, Sarah Taylor, and Iain Matthews. A decision tree framework for spatiotemporal sequence prediction. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 577–586. ACM, 2015.

[101] Davis E. King. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 10:1755–1758, 2009.

[102] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[103] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[104] Paul Lamere, Philip Kwok, Evandro Gouvea, Bhiksha Raj, Rita Singh, William Walker, Manfred Warmuth, and Peter Wolf. The cmu sphinx-4 speech recognition system. In *IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP 2003), Hong Kong*, volume 1, pages 2–5. Citeseer, 2003.

[105] A. Lanitis, C. J. Taylor, and T. F. Cootes. Toward automatic simulation of aging effects on face images. *PAMI*, 24, 2002.

[106] 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.

[107] Changsheng Li, Qingshan Liu, Jing Liu, and Hanqing Lu. Learning ordinal discriminative features for age estimation. In *CVPR*, 2012.

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

[109] Kai Li, Feng Xu, Jue Wang, Qionghai Dai, and Yebin Liu. A data-driven approach for facial expression synthesis in video. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 57–64. IEEE, 2012.

[110] 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.

[111] YiXiong Liang, Chengrong Li, Hongqiang Yue, and Yangyu Luo. Age simulation in young face images. In *Bioinf. and Biomed. Eng.*, 2007.

[112] Yixiong Liang, Ying Xu, Lingbo Liu, Shenghui Liao, and Beiji Zou. Transactions on edutainment vi. chapter A multi-layer model for face aging simulation, pages 182–192. 2011.

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

[114] Ce Liu, William T Freeman, Edward H Adelson, and Yair Weiss. Human-assisted motion annotation. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[115] 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.

[116] 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.

[117] Charles Malleson, Jean-Charles Bazin, Oliver Wang, Derek Bradley, Thabo Beeler, Adrian Hilton, and Alexander Sorkine-Hornung. Facedirector: Continuous control of facial performance in video. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3979–3987, 2015.

[118] 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.

[119] Wesley Mattheyses, Lukas Latacz, and Werner Verhelst. Comprehensive many-to-many phoneme-to-viseme mapping and its application for concatenative visual speech synthesis. *Speech Communication*, 55(7):857–876, 2013.

[120] Wesley Mattheyses and Werner Verhelst. Audiovisual speech synthesis: An overview of the state-of-the-art. *Speech Communication*, 66:182–217, 2015.

[121] Hongzhi Wu Hao Yang Kun Zhou Meng Zhang, Menglei Chai. A data-driven approach to four-view image-based hair modeling. In *SIGGRAPH*, 2017.

[122] MerrillLynch. http://faceretirement.merrilledge.com/, 2013.

[123] Umar Mohammed, Simon J. D. Prince, and Jan Kautz. Visio-lization: generating novel facial images. *ACM Trans. Graph.*, 28(3):57:1–57:8, July 2009.

[124] NCMEC. Age progression. Technical report, National Center for Missing and Exploited Children, 2010.

[125] 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.

[126] B. Ni, Z. Song, and S. Yan. Web image mining towards universal age estimator. *Proc. ACM Multimedia*, 2009.

[127] Aude Oliva, Antonio Torralba, and Philippe G. Schyns. Hybrid images. *ACM Trans. Graph.*, 25(3):527–532, July 2006.

[128] Unsang Park, Yiying Tong, and Anil K. Jain. Face recognition with temporal invariance: A 3d aging model. In *FG*, 2008.

[129] Eric Patterson, Amrutha Sethuram, Midori Albert, and Karl Ricanek. Comparison of synthetic face aging to age progression by forensic sketch artist. In *Vis Img. Proc.*, pages 247–252, 2007.

[130] Pascal Paysan. *Statistical modeling of facial aging based on 3D scans.* PhD thesis, University of Basel, 2010.

[131] Dave Perrett. Face transformer online tool, http://morph.cs.st-andrews.ac.uk/transformer/, 2013.

[132] Auriole Prince. Age progression, forensic and medical artist, http://aurioleprince.wordpress.com/, 2013.

[133] Ravi Ramamoorthi and Pat Hanrahan. A signal-processing framework for inverse rendering. In *SIGGRAPH*, 2001.

[134] N. Ramanathan and R. Chellappa. Modeling age progression in young faces. In *CVPR*, volume 1, pages 387–394, 2006.

[135] N. Ramanathan, R. Chellappa, and S. Biswas. Age progression in human faces : A survey. *J. of Vis. Lang. Comp.*, 2009.

[136] Karl Ricanek and Tamirat Tesafaye. Morph: A longitudinal image database of normal adult age-progression. In *Proc. Int. Conf. on Aut. Face and Gesture Recog.*, FGR '06, pages 341–345, 2006.

[137] Wener Robitza. ffmpeg-normalize. `https://github.com/slhck/ffmpeg-normalize`, 2016.

[138] Vladimir Rokhlin, Arthur Szlam, and Mark Tygert. A randomized algorithm for pca. *SIAM J. Mat. Anal.*, 31(3):1100–1124, 2009.

[139] Manuel Ruder, Alexey Dosovitskiy, and Thomas Brox. Artistic style transfer for videos. In *German Conference on Pattern Recognition*, pages 26–36. Springer, 2016.

[140] Shunsuke Saito, Tianye Li, and Hao Li. Real-time facial segmentation and performance capture from rgb input. *arXiv preprint arXiv:1604.02647*, 2016.

[141] 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.

[142] Shinji Sako, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi, and Tadashi Kitamura. Hmm-based text-to-audio-visual speech synthesis. In *INTERSPEECH*, pages 25–28, 2000.

[143] 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.

[144] Yisong Yue Moshe Mahler James Krahe Anastasio Garcia Rodriguez Jessica Hodgins Iain Matthews Sarah Taylor, Taehwan Kim. A deep learning approach for generalized speech animation. In *SIGGRAPH*, 2017.

[145] Kristina Scherbaum, Martin Sunkel, Hans-Peter Seidel, and Volker Blanz. Prediction of individual non-linear aging trajectories of faces. *EUROGRAPHICS*, (3):285–294, 2007.

[146] Arno Schödl, Richard Szeliski, David H Salesin, and Irfan Essa. Video textures. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 489–498. ACM Press/Addison-Wesley Publishing Co., 2000.

[147] Steven M Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Computer vision and pattern recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 519–528. IEEE, 2006.

[148] Amnon Shashua and Tammy Riklin-Raviv. The quotient image: Class-based rerendering and recognition with varying illuminations. *PAMI*, 23(2):129–139, 2001.

[149] Cheng-Ta Shen, Wan-Hua Lu, Sheng-Wen Shih, and H.-Y.M. Liao. Exemplar-based age progression prediction in children faces. In *IEEE Int. Symp. on Multimedia*, pages 123 –128, 2011.

[150] 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.

[151] YiChang Shih, Sylvain Paris, Connelly Barnes, William T Freeman, and Frédo Durand. Style transfer for headshot portraits. 2014.

[152] Taiki Shimba, Ryuhei Sakurai, Hirotake Yamazoe, and Joo-Ho Lee. Talking heads synthesis from audio with deep neural networks. In *2015 IEEE/SICE International Symposium on System Integration (SII)*, pages 100–105. IEEE, 2015.

[153] Noah Snavely, Steven M Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM transactions on graphics (TOG)*, volume 25, pages 835–846. ACM, 2006.

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

[155] Jinli Suo, Song-Chun Zhu, Shiguang Shan, and Xilin Chen. A compositional and dynamic model for face aging. *PAMI*, 2010.

[156] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

[157] Supasorn Suwajanakorn, Ira Kemelmacher-Shlizerman, and Steven M Seitz. Total moving face reconstruction. In *European Conference on Computer Vision*, pages 796–812. 2014.

[158] 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.

[159] Supasorn Suwajanakorn, Steven M Seitz, and Ira Kemelmacher-Shlizerman. Synthesizing Obama: learning lip sync from audio. *ACM Transactions on Graphics (TOG)*, 36(4):95, 2017.

[160] Sarah Taylor, Akihiro Kato, Ben Milner, and Iain Matthews. Audio-to-visual speech conversion using deep neural networks. 2016.

[161] Sarah L Taylor, Moshe Mahler, Barry-John Theobald, and Iain Matthews. Dynamic units of visual speech. In *Proceedings of the 11th ACM SIGGRAPH/Eurographics conference on Computer Animation*, pages 275–284. Eurographics Association, 2012.

[162] Alexandru Telea. An image inpainting technique based on the fast marching method. *Journal of graphics tools*, 9(1):23–34, 2004.

[163] Samuli Laine Antti Herva Jaakko Lehtinen Tero Kerras, Timo Aila. Audio-driven facial animation by joint end-to-end learning of pose and emotion. In *SIGGRAPH*, 2017.

[164] Justus Thies, Michael Zollhöfer, Matthias Nießner, Levi Valgaerts, Marc Stamminger, and Christian Theobalt. Real-time expression transfer for facial reenactment. *ACM Transactions on Graphics (TOG)*, 34(6):183, 2015.

[165] Justus Thies, Michael Zollhöfer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Face2face: Real-time face capture and reenactment of rgb videos. *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 1, 2016.

[166] B. Tiddeman, M. Stirrat, and D. Perrett. Towards realism in facial transformation: results of a wavelet mrf method. *Computer Graphics Forum, Eurographics*, 24, 2005.

[167] David Traum, Andrew Jones, Kia Hays, Heather Maio, Oleg Alexander, Ron Artstein, Paul Debevec, Alesia Gainer, Kallirroi Georgila, Kathleen Haase, et al. New dimensions in testimony: Digitally preserving a holocaust survivors interactive storytelling. In *Interactive Storytelling*, pages 269–281. Springer, 2015.

[168] 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.

[169] 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.

[170] Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.

[171] 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.

[172] 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.

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

[174] Lijuan Wang, Wei Han, and Frank K Soong. High quality lip-sync animation for 3d photo-realistic talking head. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4529–4532. IEEE, 2012.

[175] Lijuan Wang, Xiaojun Qian, Wei Han, and Frank K Soong. Photo-real lips synthesis with trajectory-guided sample selection. In *SSW*, pages 217–222, 2010.

[176] Lijuan Wang, Xiaojun Qian, Wei Han, and Frank K Soong. Synthesizing photo-real talking head via trajectory-guided sample selection. In *INTERSPEECH*, volume 10, pages 446–449, 2010.

[177] 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.

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

[179] 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.

[180] Yair Weiss. Deriving intrinsic images from image sequences. In *ICCV*, pages 68–75, 2001.

[181] Irina Werning. http://irinawerning.com/back-to-the-future/, 2013.

[182] Chenglei Wu, Derek Bradley, Pablo Garrido, Michael Zollhöfer, Christian Theobalt, Markus Gross, and Thabo Beeler. Model-based teeth reconstruction. *ACM Transactions on Graphics (TOG)*, 35(6):220, 2016.

[183] Chenglei Wu, Carsten Stoll, Levi Valgaerts, and Christian Theobalt. On-set performance capture of multiple actors with a stereo camera. *ACM Transactions on Graphics (TOG)*, 32(6), 2013.

[184] Chenglei Wu, Kiran Varanasi, Yebin Liu, Hans-Peter Seidel, and Christian Theobalt. Shading-based dynamic shape refinement from multi-view video under general illumination. In *International Conference on Computer Vision (ICCV)*, 2011.

[185] Lei Xie and Zhi-Qiang Liu. A coupled hmm approach to video-realistic speech animation. *Pattern Recognition*, 40(8):2325–2340, 2007.

[186] Lei Xie and Zhi-Qiang Liu. Realistic mouth-synching for speech-driven talking face using articulatory modelling. *IEEE Transactions on Multimedia*, 9(3):500–510, 2007.

[187] Caiming Xiong, Victor Zhong, and Richard Socher. Dynamic coattention networks for question answering. *arXiv preprint arXiv:1611.01604*, 2016.

[188] 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.

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

[190] 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.

[191] Alan L Yuille, Daniel Snow, Russell Epstein, and Peter N Belhumeur. Determining generative models of objects under varying illumination: Shape and albedo from multiple images using svd and integrability. *International Journal of Computer Vision*, 35(3):203–222, 1999.

[192] Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. Recurrent neural network regularization. *arXiv preprint arXiv:1409.2329*, 2014.

[193] 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.

[194] 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.

[195] Xinjian Zhang, Lijuan Wang, Gang Li, Frank Seide, and Frank K Soong. A new language independent, photo-realistic talking head driven by voice only. In *INTERSPEECH*, pages 2743–2747, 2013.

[196] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *arXiv preprint arXiv:1703.10593*, 2017.

[197] 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.