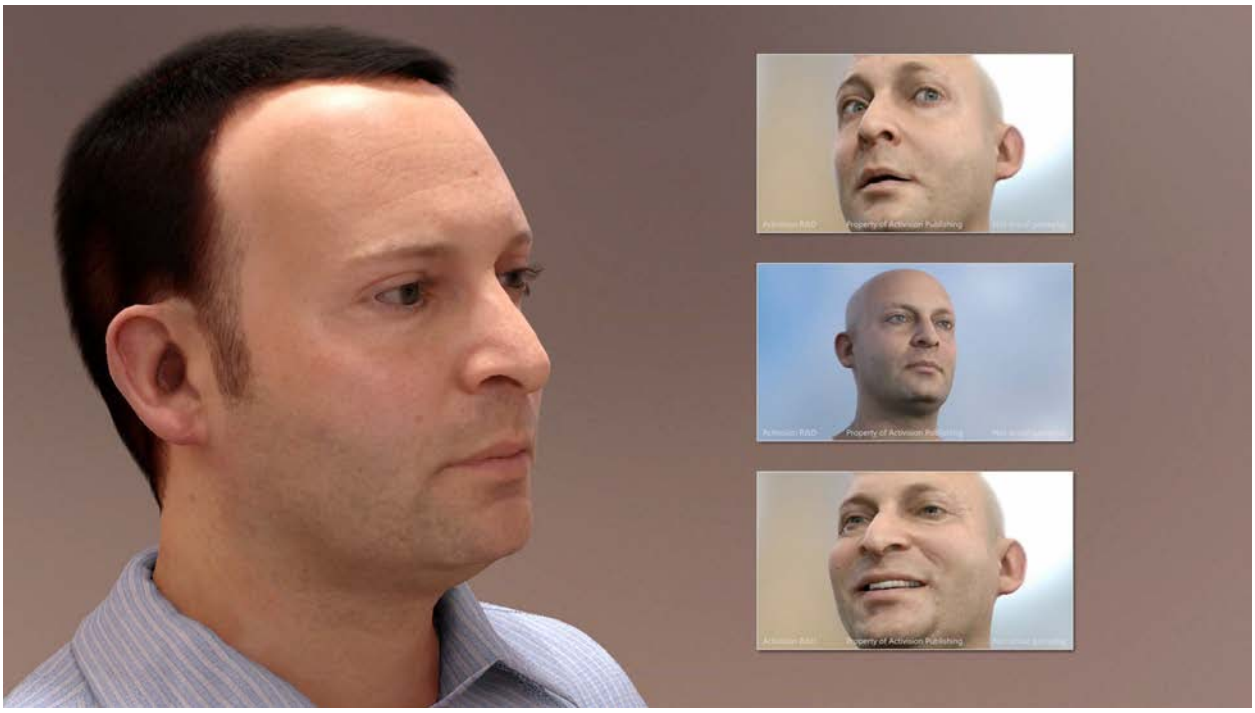


Digital Ira and Beyond:
Creating Photoreal Real-Time Digital Characters

Summary Statement

This course explains a complete process for creating next-generation realtime digital human characters, using the Digital Ira collaboration between USC ICT and Activision as an example, covering highres facial scanning, blendshape rigging, video-based performance capture, animation compression, realtime skin and eye shading, hair, latest results, and future directions.

Short Overview

This course will present the process of creating "Digital Ira" seen at the SIGGRAPH 2013 Real-Time live venue, covering the complete set of technologies from high resolution facial scanning, blendshape rigging, video-based performance capture, animation compression, realtime skin and eye shading, and hair rendering. The course will also present and explain late-breaking results and refinements and point the way along future directions which may increase the quality and efficiency of this kind of digital

SIGGRAPH 2014, August 10 – 14, 2014, Vancouver, British Columbia, Canada.

ACM 978-1-4503-2962-0/14/08

character pipeline. The actor from this project was scanned in 30 high-resolution expressions from which eight were chosen for real-time performance rendering. Performance clips were captured using multi-view video. Expression UVs were interactively corresponded to the neutral expression, retopologized to an artist mesh. An animation solver creates a performance graph representing dense GPU optical flow between video frames and the eight expressions; dense optical flow and 3D triangulation are computed, yielding per-frame spatially varying blendshape weights approximating the performance. The performance is converted to standard bone animation on a 4k mesh using a bone-weight and transform solver. Surface stress values are used to blend albedo, specular, normal, and displacement maps from the high-resolution scans per-vertex at run time. DX11 rendering includes SSS, translucency, eye refraction and caustics, physically based two-lobe specular reflection with microstructure, DOF, antialiasing, and grain. The course will explain each of processes, mentioning why each design choice was made and pointing to alternative components which may have been employed in place of any of the steps. We will also cover emerging technologies in performance capture and facial rendering. Attendees will receive a solid understanding of the techniques used to create photoreal digital characters in video games and other applications, and the confidence to incorporate some of the techniques into their own pipelines.

Project URL

http://gl.ict.usc.edu/Research/DigitalIra/

Intended Audience

Digital Character Artists, Game Developers, Texture Painters, and Researchers working on Performance Capture, Facial Modeling, and Real-Time Shading research

Prerequisites

Some experience with video game pipelines, facial animation, and shading models. The course is designed so that attendees with a wide range of experience levels will take away useful information and lessons from the course.

Course Schedule

1. Introduction/Overview - von der Pahlen

2. Facial Scanning and Microgeometry Capture - Debevec

3. Facial scan correspondence with Vuvuzela (live demo) - Alexander

4. Performance capture and animation solving - Fyffe

5. Compressing animation to a bone rig - Danvoye

6. Skin shading - Jimenez

7. Driving Expression Blending - Danvoye

8. Rendering Eyes - Jimenez

9. Rendering Hair - Jimenez

10. Latest Results and Future Work - von der Pahlen

12. Q&A - All

Instructor Bios:

JAVIER VON DER PAHLEN is Director or R&D at Activision Central Studios, leading a photreal character program since 2009. Javier started working on computer graphics in the Architecture program at Cornell University in the late 80s. Before joining Activision he co-created Softimage Face Robot in 2005, the first face commercially available facial animation software.

JORGE JIMENEZ is a real-time graphics researcher at Activision Blizzard. He received his PhD degree in Real-Time Graphics from Universidad de Zaragoza (Spain) in 2012. His interests include real-time photorealistic rendering, special effects, and squeezing rendering algorithms to be practical in game environments. He has contributions in conferences, books, and journals, including SIGGRAPH and GDC, the GPU Pro series, the Game Developer magazine, and the journal Transaction on Graphics. He co-organized the course "Filtering Approaches for Real-Time Anti-Aliasing at SIGGRAPH 2011. Some of his key achievements include Jimenez's MLAA, SMAA, and the separable subsurface scattering technique.

ETIENNE DANVOYE joined Activision Central Studio's R&D team in 2009. He has been involved in improving every step of the pipeline for realistic characters, from the high resolution scanning hardware to the tools to process the animation and texture data into a runtime-ready form. Before that, he spent seven years at Artificial Mind&Movement (now Behavior Interactive) as Lead Engine Programmer, with focus on animation, particles and physics. Areas of expertise include animation engines, and efficient game engine pipelines.

PAUL DEBEVEC is a Research Professor in the University of Southern California's Viterbi School of Engineering. He has worked on facial capture and rendering research beginning with his SIGGRAPH 2000 paper "Acquiring the Reflectance Field of the Human Face" which gave rise to the Light Stage systems recognized with an Academy Scientific and Engineering Award in 2010.

GRAHAM FYFFE is a computer scientist in the Graphics Lab of the USC Institute for Creative Technologies. He previously worked at Sway Studio in Los Angeles, CA, during which time he received a Visual Effects Society award in 2007 for Outstanding Visual Effects in a Music Video. He received his masters in computer science at the University of New Brunswick, Canada, which gave him a background in computer graphics and artificial intelligence. His research interests include computer graphics, computer vision, and physics simulation, especially as applied towards visual effects. His recent work focuses on facial geometry scanning and performance capture.

OLEG ALEXANDER is a technical artist specializing in facial rigging and animation. He received his MFA in Computer Arts from Florida Atlantic University. From 2006 to 2009 he was lead technical artist at Image Metrics. During this time, Oleg created hundreds of facial rigs for film, game, and TV projects. He became an expert in the Facial Action Coding System, facial rigging, and facial animation. In 2008, he directed and rigged the Digital Emily project, a demo featuring a photorealistic CG facial performance. Currently, Oleg is a technical artist at USC Institute for Creative Technologies.

Digital Ira and Beyond: Creating Real-Time Photoreal Digital Actors

Oleg Alexander Graham Fyffe Jay Busch Xueming Yu Jorge Jimenez Etienne Danvoye Bernardo Antionazzi
Ryosuke Ichikari Andrew Jones Paul Debevec*
USC Institute for Creative Technologies
Mike Eheler Zybnek Kysela Javier von der Pahlen†
Activision, Inc.

Figure 1: (Left) Three of eight high-res (0.1mm) light stage scans of the actor in static expressions. (Middle) Seven-camera HD performance recording. (Right) 180Hz video-driven blendshape model with screen-space subsurface scattering and advanced eye shading effects.

Overview In 2008, the "Digital Emily" project [Alexander et al. 2009] showed how a set of high-resolution facial expressions scanned in a light stage could be rigged into a real-time photoreal digital character and driven with video-based facial animation techniques. However, Digital Emily was rendered offline, involved just the front of the face, and was never seen in a tight closeup. This SIGGRAPH 2014 Course will describe in detail the processes used by USC ICT and Activision to create the "Digital Ira" character shown at SIGGRAPH 2013's Real-Time Live venue, which achieved a real-time, largely photoreal digital human character which could be seen from any viewpoint, in any lighting, and could perform realistically from video performance capture even in a tight closeup. In addition, the character ran in a real-time game-ready production pipeline, ultimately achieving 180 frames per second for a full-screen character on a two-year old graphics card. For 2014, the course will show additional character examples, discuss lessons learned, and suggest directions for future work.

3D Scanning We began by scanning accomodating researcher Ari Shapiro in thirty high-resolution expressions using the USC ICT's Light Stage X system [Ghosh et al. 2011], producing 0.1mm resoution geometry and 4K diffuse and specular reflectance maps per expression. We chose eight expressions for the real-time performance rendering, maximizing the variety of fine-scale skin deformation observed in the scans. The expressions were merged onto an artistically built back-of-the head model. To record performances for the character, we shot seven views of 30fps video of the actor improvising lines using the same seven Canon 1Dx cameras used for the scans. We used a new tool called Vuvuzela to interactively and precisely correspond all expression texture (u,v) coordinates to the neutral expression, which was retopologized to a low-polygon clean artist mesh.

Performance Animation Our offline animation solver creates a performance graph from dense GPU optical flow between the video frames and the eight expressions. This graph gets pruned by analyzing the correlation between the video frames and the expression scans over twelve facial regions. The algorithm then computes dense optical flow and 3D triangulation yielding per-frame spatially varying blendshape weights approximating the performance.

The Game Rig To create the game-ready facial rig, we transferred the mesh animation to standard bone animation on a 4K polygon mesh using a bone weight and transform solver. The solver optimizes the smooth skinning weights and the bone animated transforms to maximize the correspondence between the game mesh and the reference animated mesh.

Real-Time Rendering The rendering technique uses surface stress values to blend diffuse texture, specular, normal, and displacement maps from the different high-resolution expression scans per-vertex at run time. As a result, realistic wrinkles appear around the actor's eyes when he squints and on his foreheard when he raises his eyebrows; the color of the skin also changes with expression due to shifting blood content. The DirectX11 rendering takes into account light transport phenomena happening in the skin and eyes, from large scale events like the reflection of light of the own face into the eyes, to the shadowing and occlusion happening in the skin pores. In particular, it includes separable subsurface scattering [Jimenez et al. 2012] in screen-space, translucency, eye refraction and caustics, advanced shadow mapping and ambient occlusion, a physically-based two-lobe specular reflection with microstructure, depth of field, post effects, temporal antialiasing (SMAA T2x), and film grain.

Acknowledgements (Omitted for review)

References

ALEXANDER, O., ROGERS, M., LAMBETH, W., CHIANG, M., AND DEBEVEC, P. 2009. Thedigital emily project: photoreal facial modeling and animation. In ACM SIGGRAPH 2009 Courses, ACM, New York, NY, USA, SIGGRAPH '09, 12:1–12:15.

GHOSH, A., FYFFE, G., TUNWATTANAPONG, B., BUSCH, J., YU, X., AND DEBEVEC, P. 2011. Multiview face capture using polarized spherical gradient illumination. ACM Trans. Graph. 30, 6 (Dec.), 129:1–129:10.

JIMENEZ, J., JARABO, A., GUTIERREZ, D., DANVOYE, E., AND VON DER PAHLEN, J. 2012. Separable subsurface scattering and photorealistic eyes rendering. In ACM SIGGRAPH 2012 Courses, ACM, New York, NY, USA, SIGGRAPH 2012.

*debevec@ict.usc.edu †Javier.Pahlen@activision.com

SIGGRAPH 2013

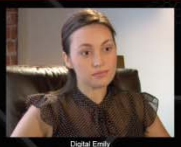
USC Institute for Creative Technologies

Digital Ira: Creating a Real-Time Photoreal Digital Actor

USC Institute for Creative Technologies

Overview

In 2008, the "Digital Emily" project [Alexander et al. 2009] showed how a set of high-resolution facial expressions scanned in a light stage could be rigged into a real-time photoreal digital character and driven with video-based facial animation techniques. However, Digital Emily was rendered offline, involved just the front of the face, and was never seen in a tight closeup. In this collaboration between Activision and USC ICT shown at SIGGRAPH 2013's Real-Time Live venue, we endeavoured to create a real-time, photoreal digital human character which could be seen from any viewpoint, in any lighting, and could perform realistically from video performance capture even in a tight closeup. In addition, we wanted this to run in a real-time game-ready production pipeline, ultimately achieving 180 frames per second for a full-screen character on a two-year old graphics card.



Digital Emily
[Alexander et al. 2009]

3D Scanning

Three of eight high-res (0.1mm) light stage scans of the actor in static expressions.

We began by scanning accomodating researcher Ari Shapiro in thirty high-resolution expressions using the USC ICT's Light Stage X system [Ghosh et al. 2011], producing 0.1mm resoution geometry and 4K diffuse and specular reflectance maps per expression. We chose eight expressions for the real-time performance rendering, maximizing the variety of fine-scale skin deformation observed in the scans. The expressions were merged onto an artistically built back-of-the head model. To record performances for the character, we shot seven views of 30fps video of the actor improvising lines using the same seven Canon 1Dx cameras used for the scans. We used a new tool called Vuvuzela to interactively and precisely correspond all expression texture (u,v) coordinates to the neutral expression, which was retopologized to a low-polygon clean artist mesh.

Performance Animation

Our offline animation solver creates a performance graph from dense GPU optical flow between the video frames and the eight expressions. This graph gets pruned by analyzing the correlation between the video frames and the expression scans over twelve facial regions. The algorithm then computes dense optical flow and 3D triangulation yielding per-frame spatially varying blendshape weights approximating the performance.

Seven-camera HD performance recording.

Game Rig

To create the game-ready facial rig, we transferred the mesh animation to standard bone animation on a 4K polygon mesh using a bone weight and transform solver. The solver optimizes the smooth skinning weights and the bone animated transforms to maximize the correspondence between the game mesh and the reference animated mesh, giving the most natural movement available.

Real-Time Rendering

Realtime DirectX11 rendering

The rendering technique uses surface stress values to blend diffuse texture, specular, normal, and displacement maps from the different high-resolution expression scans per-vertex at run time. As a result, realistic wrinkles appear around the actor's eyes when he squints and on his foreheard when he raises his eyebrows; the color of the skin also changes with expression due to shifting blood content.

Dynamic skin wrinkling

The DirectX11 rendering takes into account light transport phenomena happening in the skin and eyes, from large scale events like the reflection of light of the own face into the eyes, to the shadowing and occlusion happening in the skin pores. In particular, it includes separable subsurface scattering [Jimenez et al. 2012] in screen-space, translucency, eye refraction and caustics, advanced shadow mapping and ambient occlusion, a physically-based two-lobe specular reflection with microstructure, depth of field, post effects, temporal antialiasing (SMAA T2x), and film grain.

Caustics within eyes responding to enviornment

Acknowledgements

We thank Borom Tunwattanapong, Koki Nagano, Domi Piturro, Alejo von der Pahlen, Joe Alter, Curtis Beeson, Mark Daly, Mark Swain, Jen-Hsun Huang, Ari Shapiro, Valerie Dauphin, and Kathleen Haase for their important assistance and contributions to this work. This work was supported by USA RDECOM, USC, and Activision, Inc; no endorsement is implied.

Oleg Alexander Graham Fyffe Jay Busch Xueming Yu Ryosuke Ichikari
Andrew Jones Paul Debevec
USC Institute for Creative Technologies

Jorge Jimenez Etienne Danvoye Bernardo Antionazzi Mike Eheler
Zbynek Kysela Xian-Chun Wu Javier von der Pahlen
Activision Inc.

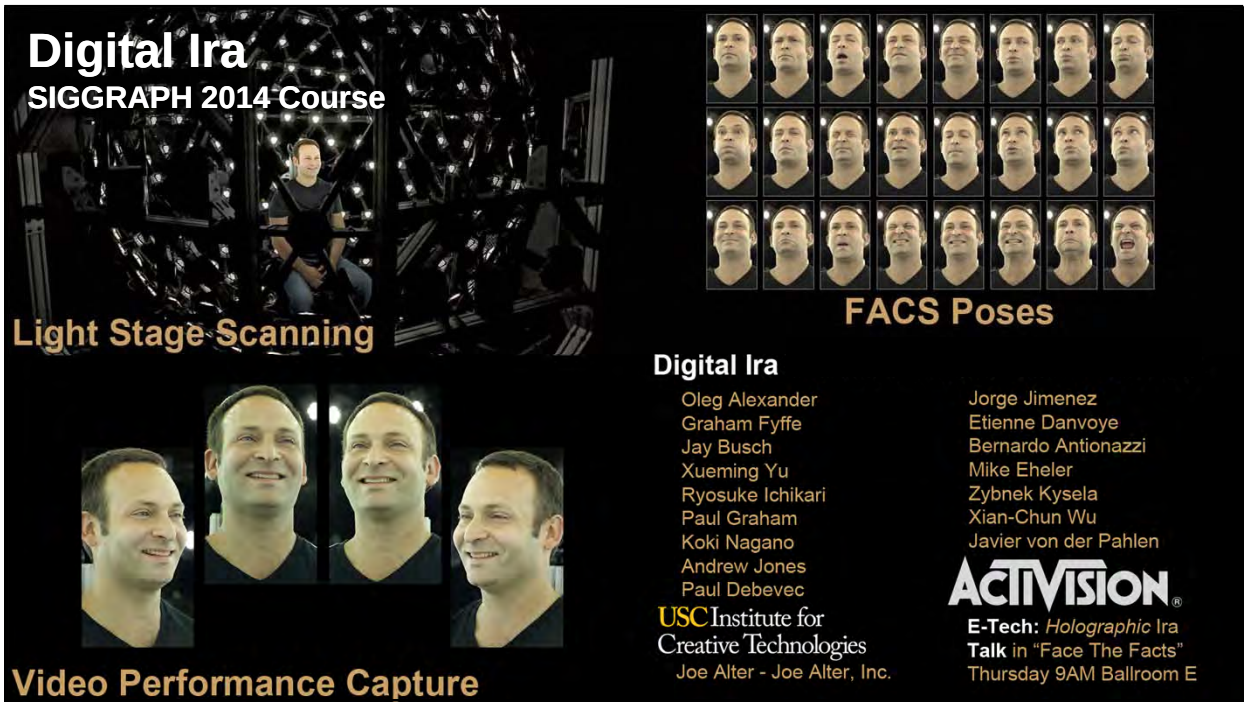
http://www.ict.usc.edu/

References:

ALEXANDER, O., ROGERS, M., LAMBETH, W., CHIANG, M., AND DEBEVEC, P. 2009. The digital emily project: photoreal facial modeling and animation. In ACM SIGGRAPH 2009 Courses, ACM, New York, NY, USA, SIGGRAPH '09, 12:1–12:15.

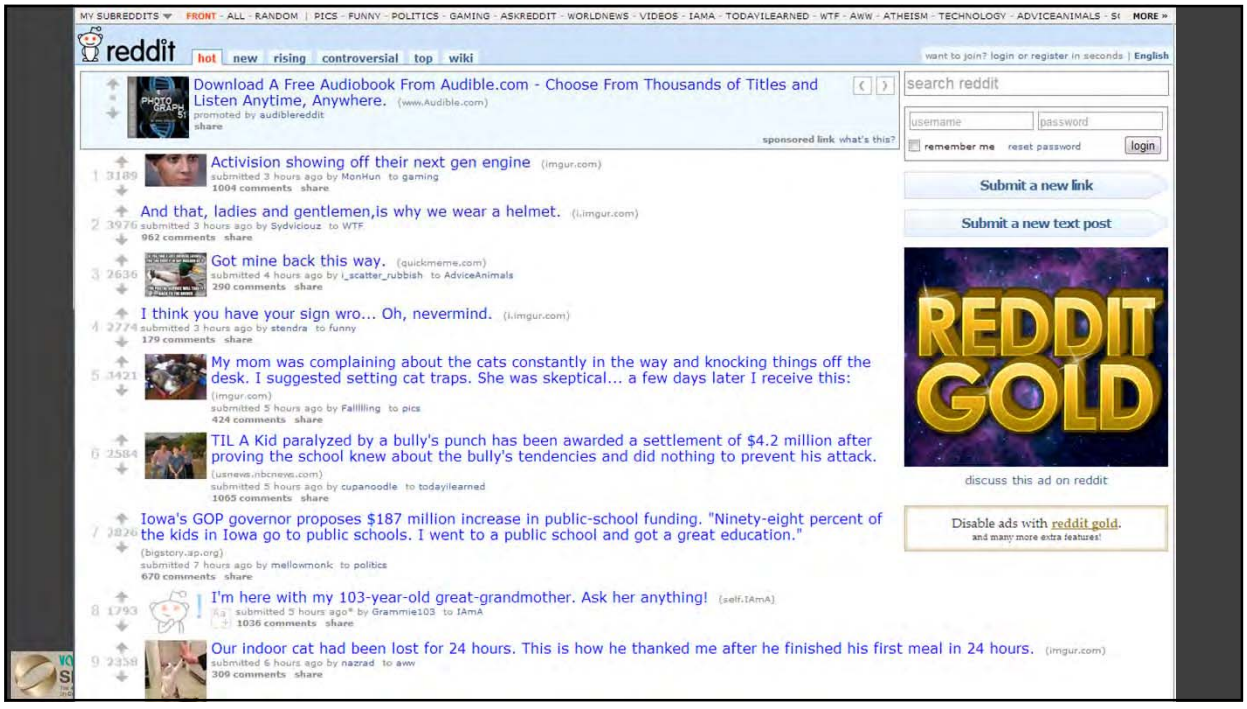
GHOSH, A., FYFFE, G., TUNWATTANAPONG, B., BUSCH, J., YU, X., AND DEBEVEC, P. 2011. Multiview face capture using polarized spherical gradient illumination. ACM Trans. Graph. 30, 6 (Dec.), 129:1–129:10.

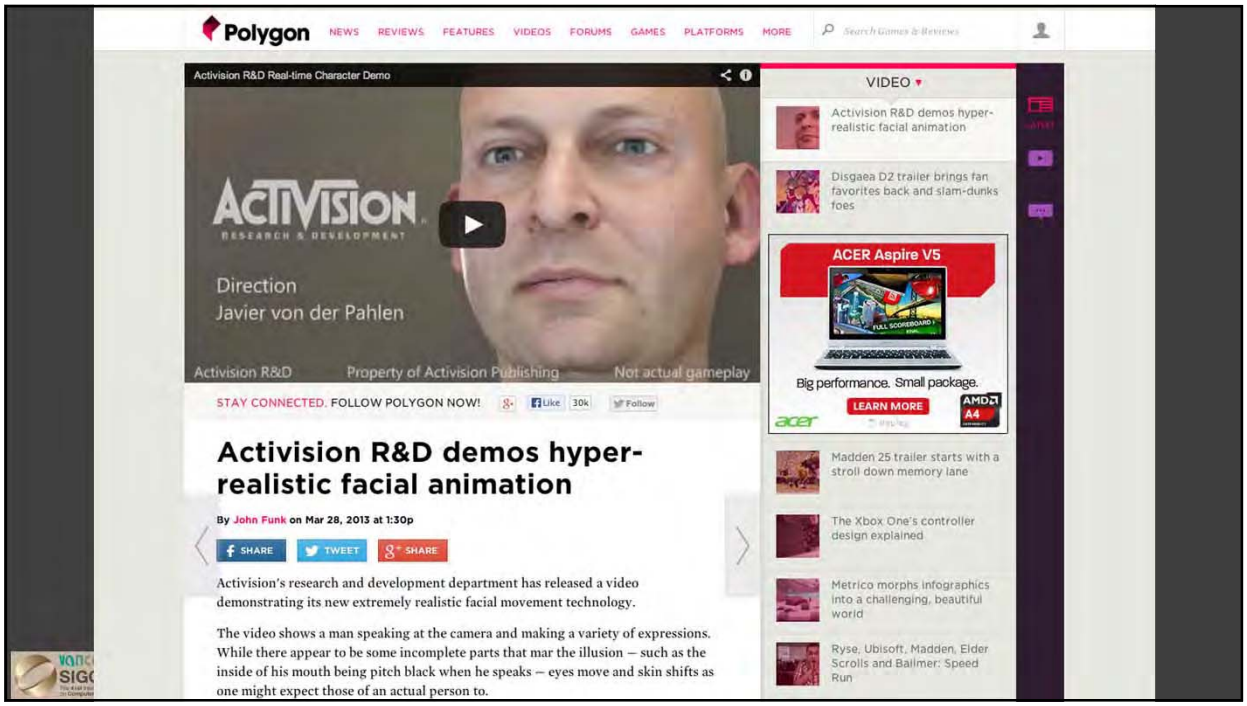
JIMENEZ, J., JARABO, A., GUTIERREZ, D., DANVOYE, E., AND VON DER PAHLEN, J. 2012. Separable subsurface scattering and photorealistic eyes rendering. In ACM SIGGRAPH 2012 Courses, ACM, New York, NY, USA, SIGGRAPH '12.

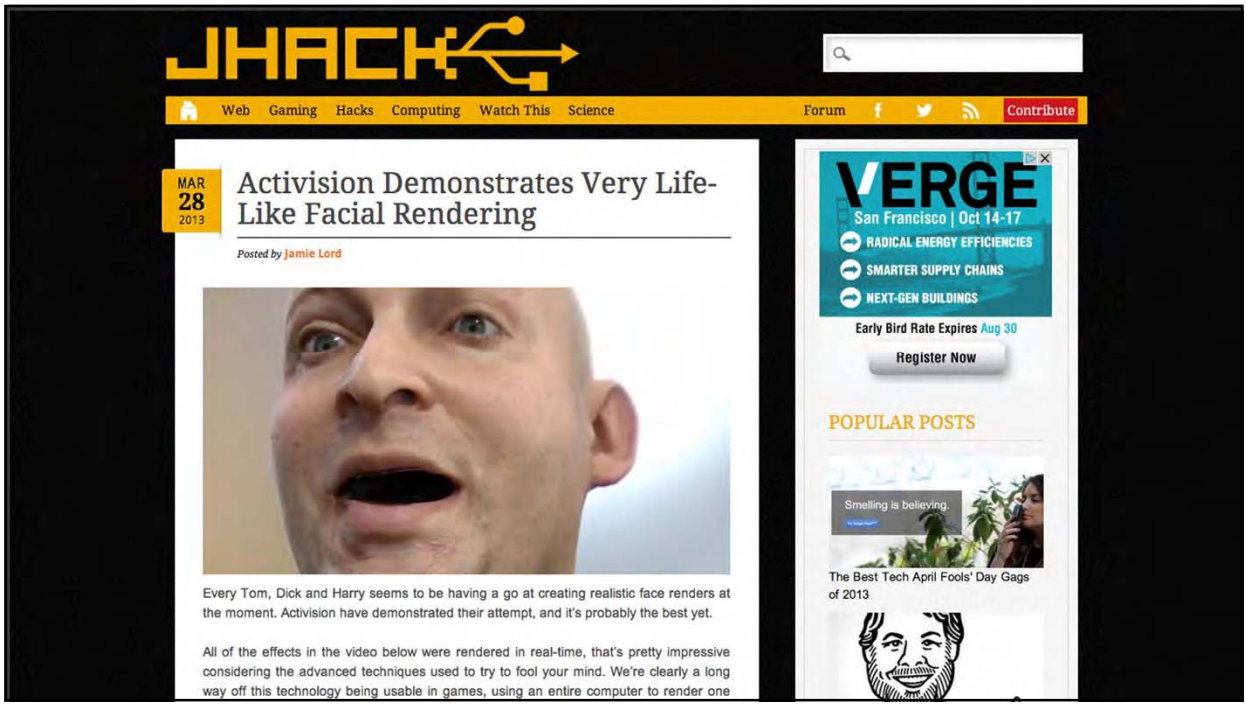

Digital Ira
SIGGRAPH 2014 Course
Light Stage Scanning
FACS Poses
Digital Ira
Oleg Alexander
Graham Fyffe
Jay Busch
Xueming Yu
Ryosuke Ichikari
Paul Graham
Koki Nagano
Andrew Jones
Paul Debevec
Jorge Jimenez
Etienne Danvoye
Bernardo Antionazzi
Mike Eheler
Zybnek Kysela
Xian-Chun Wu
Javier von der Pahlen
USC Institute for Creative Technologies
Joe Alter - Joe Alter, Inc.
ACTIVISION
E-Tech: Holographic Ira
Talk in "Face The Facts"
Thursday 9AM Ballroom E
Video Performance Capture


GDC 14
GAME DEVELOPERS CONFERENCE
SAN FRANCISCO, CA
MARCH 17-21, 2014
EXPO DATES: MARCH 19-21
2014
ABOUT
ATTEND
CONFERENCE
EXHIBIT
EVENTS
Session Scheduler
Next Generation Character Rendering
PASS TYPE
TRACK
FORMAT
DAY
Speakers
VANCOUVER
SIGGRAPH2014

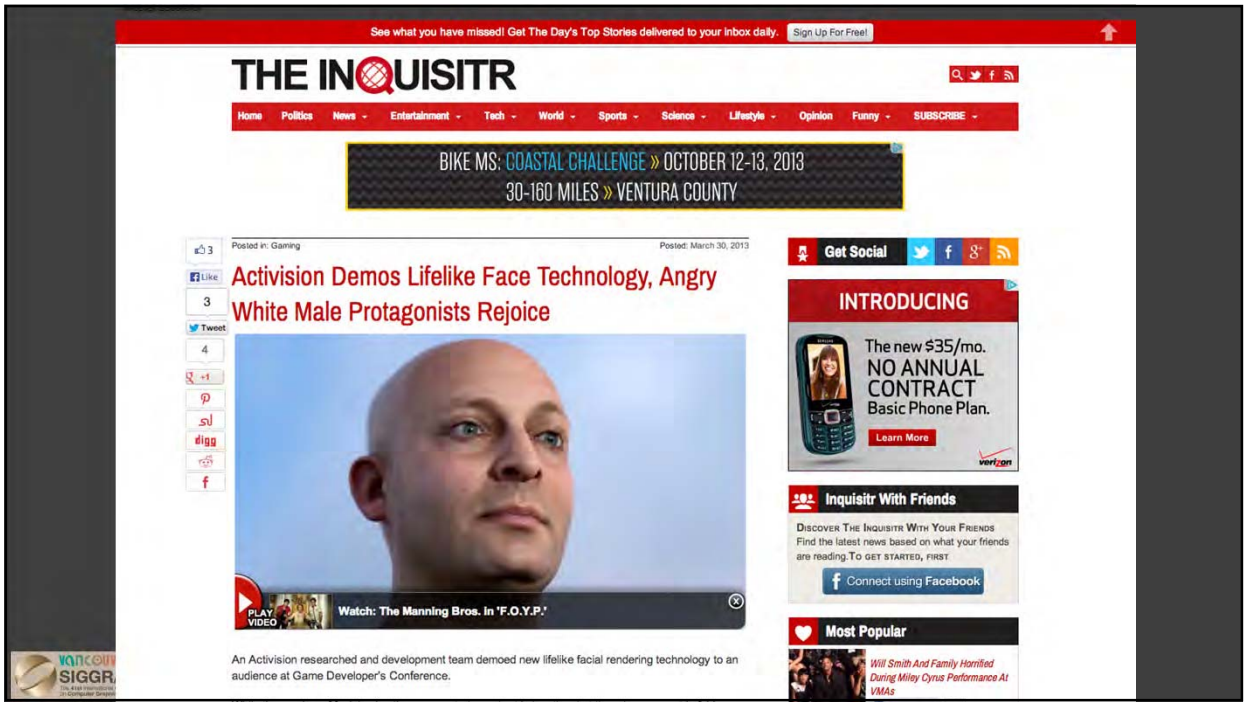
- http://www.youtube.com/watch?v=I6R6N4Vy0nE




Polygon
NEWS
REVIEWS
FEATURES
VIDEOS
FORUMS
GAMES
PLATFORMS
MORE
Activision R&D Real-time Character Demo
ACTIVISION
RESEARCH & DEVELOPMENT
Direction
Javier von der Pahlen
Activision R&D
Property of Activision Publishing
Not actual gameplay
VIDEO
Activision R&D demos hyper-realistic facial animation
Disgaea D2 trailer brings fan favorites back and slam-dunks foes
ACER Aspire V5
Big performance. Small package.
LEARN MORE
STAY CONNECTED. FOLLOW POLYGON NOW!
Activision R&D demos hyper-realistic facial animation
By John Funk on Mar 28, 2013 at 1:30p
SHARE
TWEET
SHARE
Activision's research and development department has released a video demonstrating its new extremely realistic facial movement technology.
The video shows a man speaking at the camera and making a variety of expressions. While there appear to be some incomplete parts that mar the illusion — such as the inside of his mouth being pitch black when he speaks — eyes move and skin shifts as one might expect those of an actual person to.
Madden 25 trailer starts with a stroll down memory lane
The Xbox One's controller design explained
Metrico morphs infographics into a challenging, beautiful world
Ryse, Ubisoft, Madden, Elder Scrolls and Ballmer: Speed Run


JHACK
Web
Gaming
Hacks
Computing
Watch This
Science
Forum
Contribute
MAR
28
2013
Activision Demonstrates Very Life-Like Facial Rendering
Posted by Jamie Lord
Every Tom, Dick and Harry seems to be having a go at creating realistic face renders at the moment. Activision have demonstrated their attempt, and it's probably the best yet.
All of the effects in the video below were rendered in real-time, that's pretty impressive considering the advanced techniques used to try to fool your mind. We're clearly a long way off this technology being usable in games, using an entire computer to render one
VERGE
San Francisco | Oct 14-17
RADICAL ENERGY EFFICIENCIES
SMARTER SUPPLY CHAINS
NEXT-GEN BUILDINGS
Early Bird Rate Expires Aug 30
Register Now
POPULAR POSTS
Smelling is believing.
The Best Tech April Fools' Day Gags of 2013


also @ TechSpot: Nvidia allegedly working on dual-GPU, GK110-based GTX 790
TECHSPOT
TECHNOLOGY NEWS AND ANALYSIS
Search
Home
Reviews
Guides
Product Finder
Forums
Downloads
Drivers
Extras
Sign in
About
Trending
Features
Hardware
Software
Mobile
Gaming
The Web
IT
Apple
Microsoft
Security
More
acer
You know efficiency. Acer knows business.
TravelMate P6
Work easy. Play hard.
Windows 8
HOME › NEWS › GAMING
Activision demos incredible lifelike facial rendering technology
By Shawn Knight
On March 28, 2013, 10:00 AM
27
Like 91
+1 9
Tweet 22
TechSpot on:
Like
+1
Follow
Activision's research and development team has been working hard as of late to produce what they are calling next generation character rendering. The company showed off the new technology at the Game Developers Conference and although the video clip on display is short, it shows a number of facial animations running in real time that are stunning.
As you can see in the clip below, the animations are extremely lifelike – perhaps more so than any others we have seen to date. Activision said they used source material from USC Institute for Creative Technologies and converted it into a "70 bones rig" using advanced techniques to deliver realism to the eyes and skin.
Activision R&D Real-time Character Demo
Sprint
INTRODUCING UNLIMITED TALK, TEXT AND DATA. GUARANTEED FOR LIFE.


See what you have missed! Get The Day's Top Stories delivered to your inbox daily.
Sign Up For Free!
THE INQUISITR
Home
Politics
News
Entertainment
Tech
World
Sports
Science
Lifestyle
Opinion
Funny
SUBSCRIBE
BIKE MS: COASTAL CHALLENGE » OCTOBER 12-13, 2013
30-160 MILES » VENTURA COUNTY
3
Like
3
Tweet
4
+1
digg
Posted in: Gaming
Posted: March 30, 2013
Activision Demos Lifelike Face Technology, Angry White Male Protagonists Rejoice
PLAY VIDEO
Watch: The Manning Bros. In 'F.O.Y.P.'
An Activision researched and development team demoed new lifelike facial rendering technology to an audience at Game Developer's Conference.
Get Social
INTRODUCING
The new $35/mo.
NO ANNUAL CONTRACT
Basic Phone Plan.
Learn More
verizon
Inquisitr With Friends
Discover The Inquisitr With Your Friends
Find the latest news based on what your friends are reading. To get started, first
Connect using Facebook
Most Popular
Will Smith And Family Horrified During Miley Cyrus Performance At VMAs

The hi-tech scanner that turns actors into aliens, warts and all

Kaya Burgess

In the cinemas and video games of the near future, computer-generated characters will be so lifelike that individual skin cells and hair follicles will be visible, because of a new high-definition form of digital animation technology.

The distinction between real-life actors and computer graphics has already been blurred by films such as *Beowulf* and *Avatar* — which used computer-generated imagery (CGI) to create animated versions of leading actors — but until now they risked seeming rather plastic-looking.

But thanks to new super-high resolution facial scanning you will now be able to see every blemish and crease in Angelina Jolie's virtual cheek or Zoe Saldana's digital forehead.

Researchers at the University of Southern California and Imperial College London have developed techniques to scan centimetre-square patches of skin from the cheek, forehead, nose, chin and temple in such high resolution that a single skin cell covers three pixels on the screen.

The team has also polarised the light source used during the scanning to pick up not only the light reflecting off the skin's surface but also light that penetrates below the epidermis and scatters back, providing greater depth and tone to the final image.

"The scanning, which uses high-resolution stills cameras in a laboratory, also captures how the skin behaves under different types of light and during different facial expressions. The scanned patches can then be mapped on to a 3-D image of the actor, created with motion-capture technology.

As a result, computer-generated characters will no longer be so "plastic-looking", according to Paul Debevec, the associate director of graphics research at USC, whose earlier techniques were used on James Cameron's *Avatar*. "The bumpiness of the surface of the skin, at the micron scale, actually affects how light reflects off the surface," Professor Debevec explained.

"That's what makes it look healthy or oily or pasty or chalky. It makes someone look like a human being made out of organic material and not like a computer-generated zombie."

To make *Avatar*, artists had to go back to the CGI imagery of the blue-skinned Na'vi characters and add blemishes, such as moles or creases, by hand. This vastly increased the man-hours and expense of the film, which was nearly 60 per cent computer-generated and cost more than £150 million.

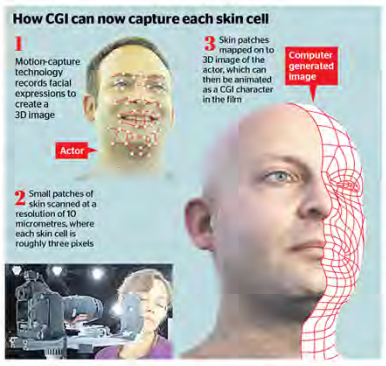
The process will now be much cheaper, Professor Debevec said, and video game developers at Activision have already created mathematical algorithms that can mimic many of the effects of the high-definition scanning, greatly reducing the time, expense and processing power needed.

This will allow hyper-realistic CGI characters to appear on video games consoles and could allow film directors to create CGI scenes in real time.

Professor Debevec said: "In the future it might be the less expensive movies that use CGI technology, while big budget movies will be the only ones who can still afford to go out on location and shoot in Paris or Bermuda and take up actors' time."

Abhijeet Ghosh, from the computing department at Imperial College London, helped to develop the "facial microgeometry scanning" process and was approached by the Avon cosmetics company to help it to analyse the effects of make-up on the skin.

He predicted that cosmetics customers may be able to use apps in future to see how their faces would look with different types of foundation. "When you start scanning skin at that scale, it could also have medical or dermatological applications," Dr Ghosh said.



- http://www.youtube.com/watch?v=SPeZNXmPzuI

Digital Ira at SIGGRAPH 2013 Real-Time Live

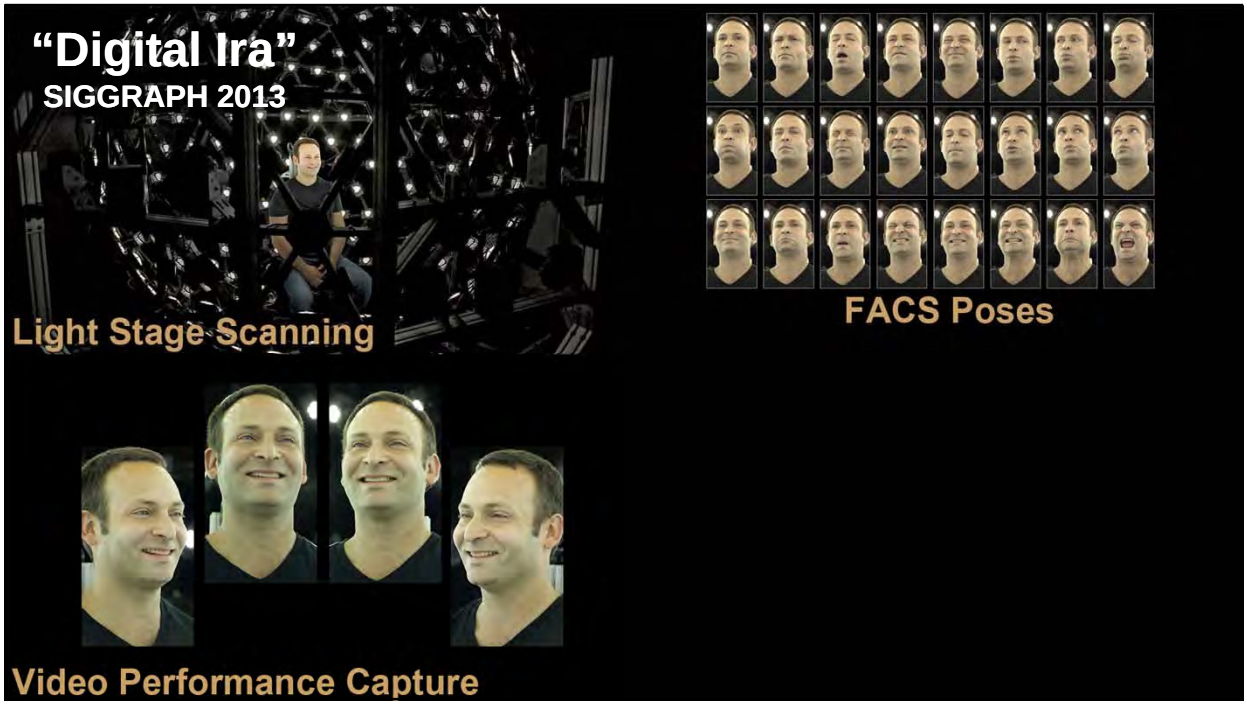
- World's first (reasonably) photoreal real-time digital character, collaboratively developed with **Activision**, debuts at ***Real-Time Live*** to 2000+ attendees
- Leverages Graham et al.'s ***Measurement-Based Synthesis of Facial Microgeometry*** Eurographics 2013 for skin detail synthesis
- First version with hair!
- **NVIDIA** shows improved Digital Ira in their SIGGRAPH booth on 4K monitor, including Digital Ira running on their "Project Logan" tablet prototype
- ► Come to **"Digital Ira" Project Overview** at ICT Monday 8/26 featuring Graham Fyffe, Oleg Alexander, and Javier von der Pahlen

vancouver SIGGRAPH2014
The 41st International Conference and Exhibition on Computer Graphics and Interactive Techniques

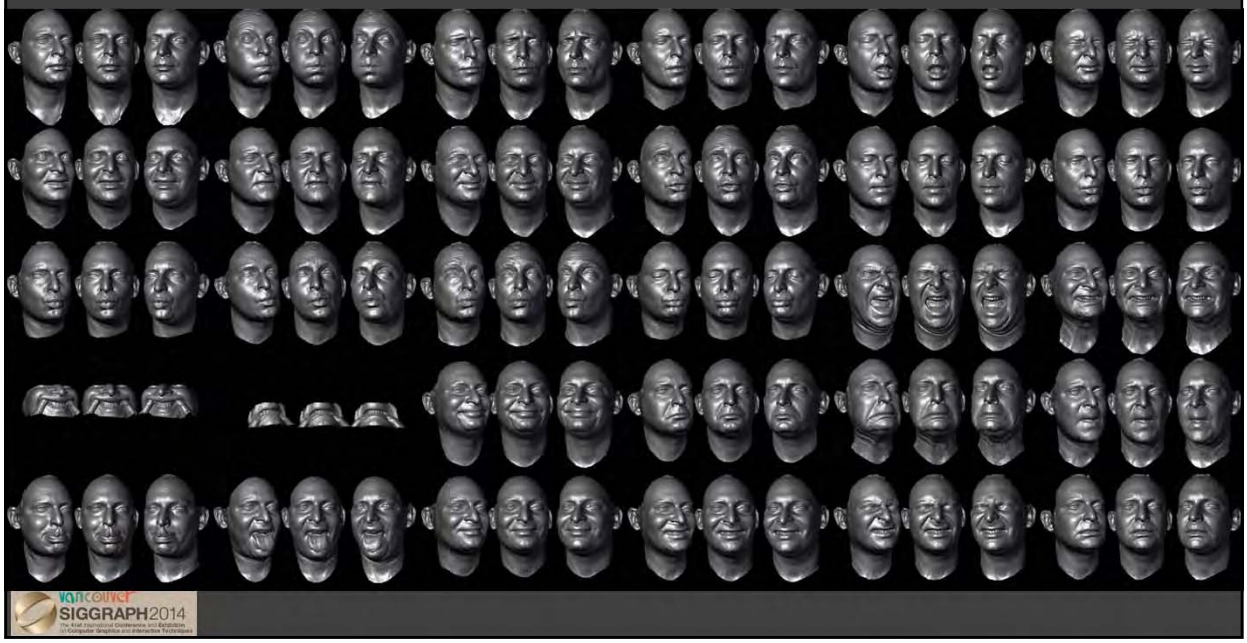
11

Digital Emily – SIGGRAPH 2008
USC ICT and Image Metrics

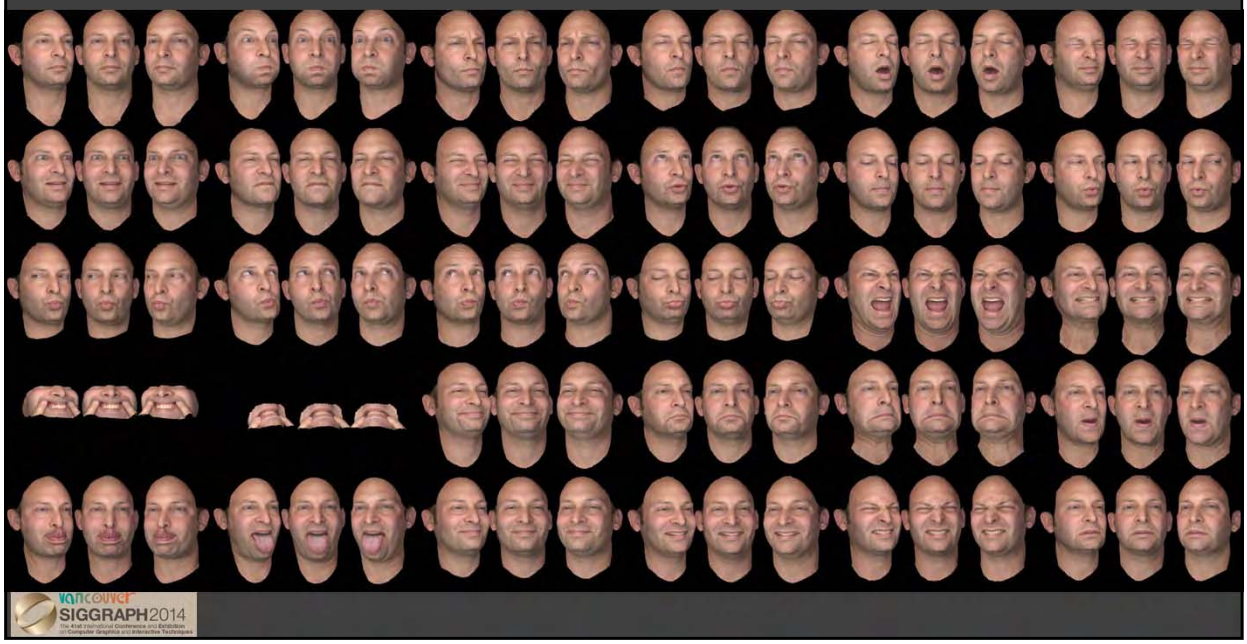

Light Stage X


"Digital Ira"
SIGGRAPH 2013
Light Stage Scanning
FACS Poses
Video Performance Capture

Latest Result: 30 High-Res Expressions Processed in One Week



Latest Result: 30 High-Res Expressions Processed in One Week



Measurement-Based Synthesis of Facial Microgeometry

Paul Graham, Borom Tunwattanapong, Jay Busch, Xueming Yu, Andrew Jones, Paul Debevec, Abhijeet Ghosh

USC Institute for Creative Technologies

Presented at Eurographics 2013 May 6-10, Girona (Spain)

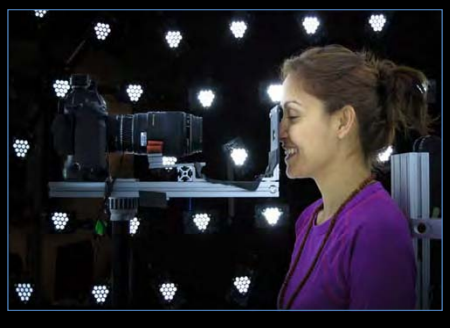
Rendering from Multi-view Scan

Photograph

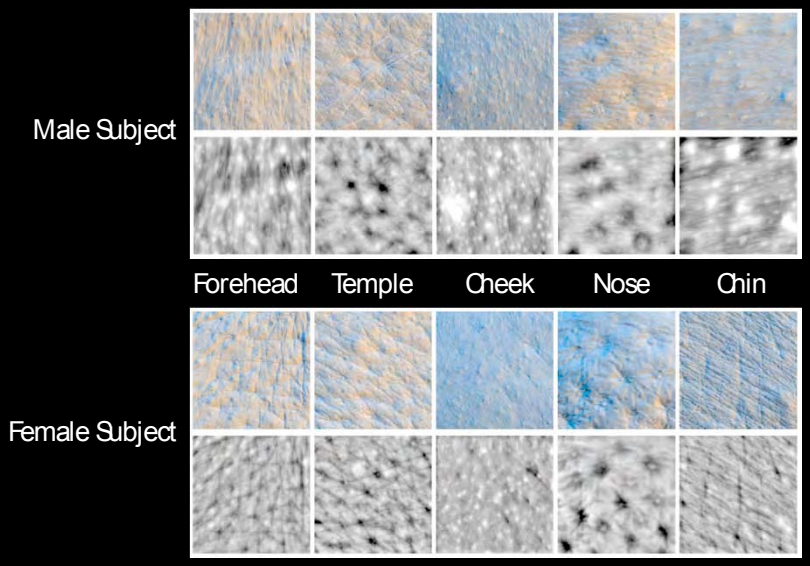
Rendering with Enhance Microstructure

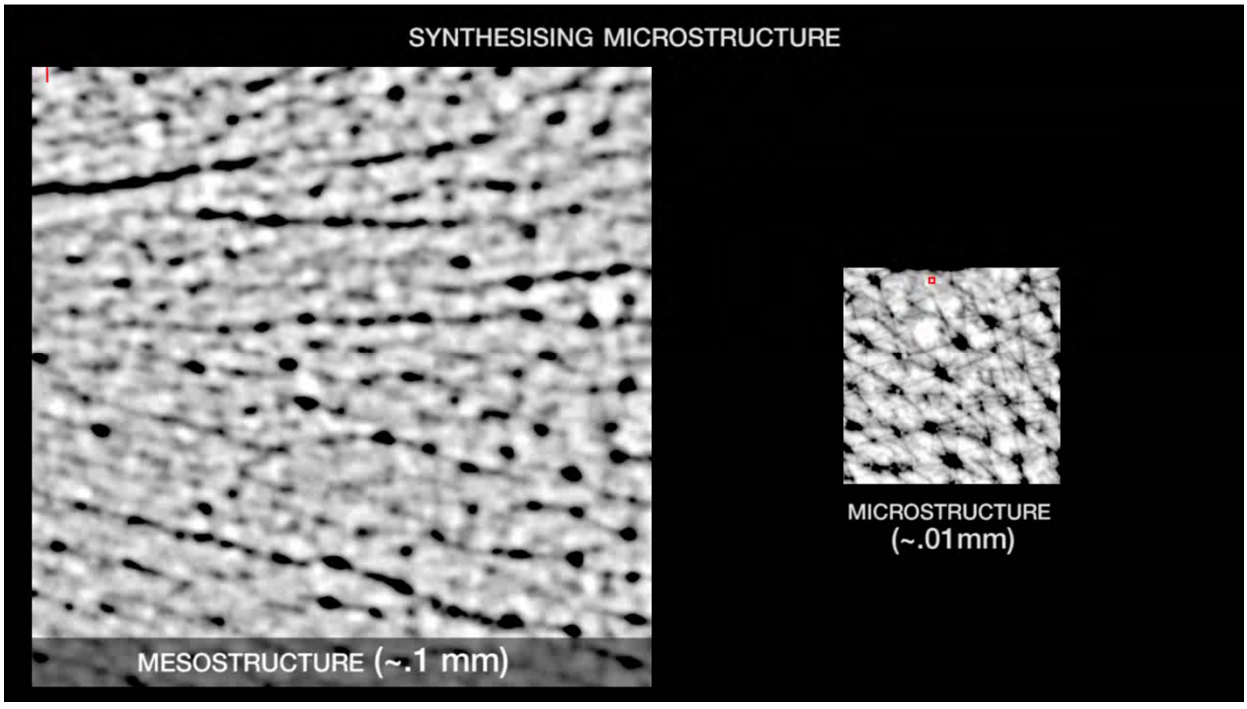
Recording skin microstructure

- 12-light hemispherical dome or
- Polarized LED sphere
  - higher lighting resolution for specular/oily skin
- Canon 1DMark III camera
  - Canon 100mm macro lens
- 24mm by 16mm aperture
  - 7 microns resolution



Specular Surface Normals and Displacement Maps



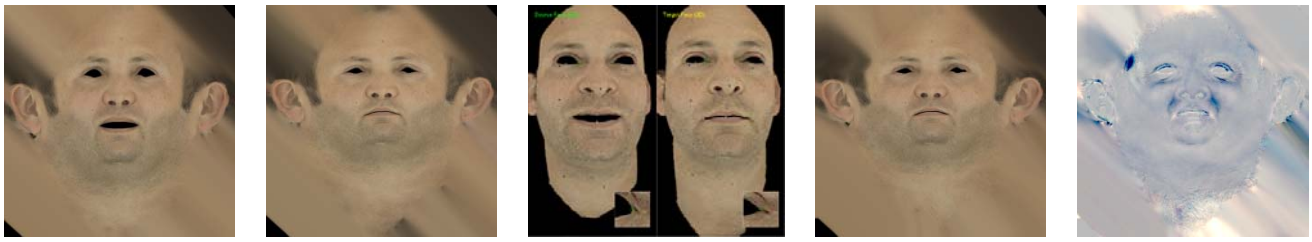


Vuvuzela Demo

Vuvuzela: A Facial Scan Correspondence Tool

Ryosuke Ichikari Oleg Alexander Paul Debevec

USC Institute for Creative Technologies oalexander@ict.usc.edu



(a) (b) (c) (d) (e)

Figure 1: (a) Source. (b) Target. (c) Vuvuzela workflow. (d) Warped source. (e) Difference between warped source and target.

1 Introduction

When scanning an actor's face in multiple static facial expressions, it is often desirable for the resulting scans to all have the same topology and for the textures to all be in the same UV space. Such "corresponded" scans would enable the straightforward creation of blendshape-based facial rigs. We present Vuvuzela, a semi-automated facial scan correspondence tool. Vuvuzela is currently being used in our facial rigging pipeline, and was one of the key tools in the Digital Ira project.

2 Our Approach

Our rig building process begins by scanning an actor's face in our Light Stage X device [Ghosh et al. 2011]. We capture a set of about 30 static facial expressions, roughly corresponding to Action Units from the Facial Action Coding System [Ekman and Friesen 1978]. We also capture a master "neutral" expression, which becomes the target scan in our correspondence pipeline.

Rather than storing our scans as geometry and textures, we choose instead to store our scans as images. Each one of our scans is stored as a set of 4K, 32 bit float EXR images, including diffuse, specular, specular normals, and a high resolution point cloud. The maps are in a cylindrically unwrapped UV space, representing our ear to ear data. However, the UV space differs slightly for each expression scan.

Vuvuzela exploits this image-based scan represantation by doing the scan correspondence in 2D rather than 3D. Vuvuzela takes as input two scans: one of the expressions as the source and the neutral expression as the target. Vuvuzela provides an OpenGL UI, allowing the user to interact with the scans in 3D. The scans are rendered with the diffuse textures only, and all of the correspondence processing uses only the diffuse textures.

The user clicks corresponding points in the source and target scans, such as corners of the eyes and lips, and other facial landmarks. We found that we don't need to put dots or markers on the face during scanning, because there is plenty of naturally occuring texture in the face, especially when over-sharpened. The placement of the correspondence points doesn't have to be exact—the points are used only as an initialization by our algorithm.

Once enough points have been placed, the user presses the Update button, which triggers our correspondence algorithm. The result is displayed to the user and the UI offers several modes to preview the quality of the correspondence, including a "blendshape" slider blending both geometry and/or texture. The user can then add, delete, or edit points, and repeat the process until a high quality correspondence is achieved.

Our algorithm has three steps and runs in 2D. First, we construct a Delaunay triangulation between the user supplied points and apply affine triangles to roughly pre-warp the source diffuse texture to the target. Second, we use GPU-accelerated optical flow to compute a dense warp field from the pre-warped source diffuse texture to the target. Finally, we apply the dense warp to each one of our source texture maps, including diffuse, specular, specular normals, and point cloud. The result is the source scan warped to the target UV space. The submillimeter correspondence is able to align individual pores across the majority of the face.

Some expressions are more challenging to correspond than others. Especially expressions with lots of occlusions, like mouth open to mouth closed. In such cases, optical flow will fail to get a good result. We assist optical flow in two ways. First, we paint black masks around occlusion regions in both source and target diffuse textures. Second, we mark some points as "pinned" and those points are rasterized into small black dots at runtime. Using both of these techniques in combination usually produces good results even in the toughest cases.

A useful byproduct of Vuvuzela is the ability to generate blendshapes directly from the corresponded scans. First, we remesh the neutral scan, creating an artist mesh with artist UVs. Then we load the artist mesh into Vuvuzela and export the blendshapes for all the scans by looking up vertex positions in the warped point clouds. All the texture maps are also warped into the artist UV space, which is simply an additional affine triangles 2D warp. The result is a set of blendshapes and texture maps ready to hand off to the facial rigger.

References

EKMAN, P., AND FRIESEN, W. 1978. Facial Action Coding System: A Technique for the Measurement of Facial Movement. Consulting Psychologists Press, Palo Alto.

GHOSH, A., FYFFE, G., TUNWATTANAPONG, B., BUSCH, J., YU, X., AND DEBEVEC, P. 2011. Multiview face capture using polarized spherical gradient illumination. ACM Trans. Graph. 30, 6 (Dec.), 129:1–129:10.

SIGGRAPH 2013

USC Institute for Creative Technologies

Vuvuzela: A Facial Scan Correspondence Tool

Ryosuke Ichikari Oleg Alexander Paul Debevec
USC Institute for Creative Technologies

USC Institute for Creative Technologies

Introduction

Digital Ira

When scanning an actor's face in multiple static facial expressions, it is often desirable for the resulting scans to all have the same topology and for the textures to all be in the same UV space. Such "corresponded" scans would enable the straightforward creation of blendshape-based facial rigs. We present Vuvuzela, a semiautomated facial scan correspondence tool. Vuvuzela is currently being used in our facial rigging pipeline, and was one of the key tools in the Digital Ira project.

(a) source map (b) target map (c) resulting warped source (d) difference between warped source and target

***Figure 1**: the original neutral expression (a) and the second expression needing to be placed into the same texture space (b) are imported into Vuvuzela. After processing, the original second expression is warped into the same texture space (c) as the neutral expression (a). The difference from the computed warp can be seen to the right (d).*

Our Approach

Data acquisition and preparation Our rig building process begins by scanning an actor's face in our Light Stage X device [Ghosh et al. 2011]. We capture a set of about 30 static facial expressions, roughly corresponding to Action Units from the Facial Action Coding System [Ekman and Friesen 1978]. We also capture a master "neutral" expression, which becomes the target scan in our correspondence pipeline.

Rather than storing our scans as geometry and textures, we choose instead to store our scans as images. Each one of our scans is stored as a set of 4K, 32 bit float EXR images, including diffuse, specular, specular normals, and a high resolution point cloud. The maps are in a cylindrically unwrapped UV space, representing our ear to ear data. However, the UV space differs slightly for each expression scan.

User interface Vuvuzela exploits this image-based scan represantation by doing the scan correspondence in 2D rather than 3D. Vuvuzela takes as input two scans: one of the expressions as the source and the neutral expression as the target. Vuvuzela provides an OpenGL UI, allowing the user to interact with the scans in 3D. The scans are rendered with the diffuse textures only, and all of the correspondence processing uses only the diffuse textures.

The user clicks corresponding points in the source and target scans, such as corners of the eyes and lips, and other facial landmarks. We found that we don't need to put dots or markers on the face during scanning, because there is plenty of naturally occuring texture in the face, especially when over-sharpened. The placement of the correspondence points doesn't have to be exact—the points are used only as an initialization by our algorithm.

References:

EKMAN, P., AND FRIESEN, W. 1978. Facial Action Coding System: A Technique for the Measurement of Facial Movement. Consulting Psychologists Press, Palo Alto.

GHOSH, A., FYFFE, G., TUNWATTANAPONG, B., BUSCH, J., YU, X., AND DEBEVEC, P. 2011. Multiview face capture using polarized spherical gradient illumination. ACM Trans. Graph. 30, 6 (Dec.), 129:1–129:10.

Once enough points have been placed, the user presses the Update button, which triggers our correspondence algorithm. The result is displayed to the user and the UI offers several modes to preview the quality of the correspondence, including a "blendshape" slider blending both geometry and/or texture. The user can then add, delete, or edit points, and repeat the process until a high quality correspondence is achieved.

posed mesh with corresponded maps

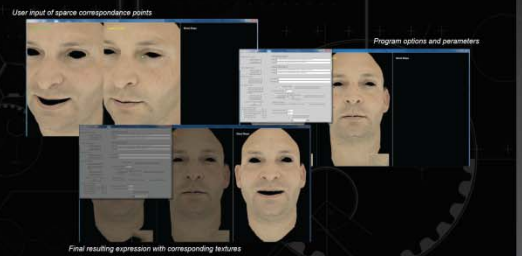
posed mesh with neutral expression maps

Correspondence algorithm Our algorithm has three steps and runs in 2D. First, we construct a Delaunay triangulation between the user supplied points and apply affine triangles to roughly pre-warp the source diffuse texture to the target. Second, we use GPU-accelerated optical flow to compute a dense warp field from the pre-warped source diffuse texture to the target. Finally, we apply the dense warp to each one of our source texture maps, including diffuse, specular, specular normals, and point cloud. The result is the source scan warped to the target UV space. The submillimeter correspondence is able to align individual pores across the majority of the face.

User intervention Some expressions are more challenging to correspond than others. Especially expressions with lots of occlusions, like mouth open to mouth closed. In such cases, optical flow will fail to get a good result. We assist optical flow in two ways. First, we paint black masks around occlusion regions in both source and target diffuse textures. Second, we mark some points as "pinned" and those points are rasterized into small black dots at runtime. Using both of these techniques in combination usually produces good results even in the toughest cases.

Blend shapes A useful byproduct of Vuvuzela is the ability to generate blendshapes directly from the corresponded scans. First, we remesh the neutral scan, creating an artist mesh with artist UVs. Then we load the artist mesh into Vuvuzela and export the blendshapes for all the scans by looking up vertex positions in the warped point clouds. All the texture maps are also warped into the artist UV space, which is simply an additional affine triangles 2D warp. The result is a set of blendshapes and texture maps ready to hand off to the facial rigger.

Software Interface and Workflow



To appear in ACM TOG ?(?).

Driving High-Resolution Facial Scans with Video Performance Capture

Graham Fyffe Andrew Jones Oleg Alexander Ryosuke Ichikari Paul Debevec
USC Institute for Creative Technologies

(a) (b) (c) (d)

Figure 1: (a) High resolution geometric and reflectance information from multiple static expression scans is automatically combined with (d) dynamic video frames to recover (b) matching animated high resolution performance geometry that can be (c) relit under novel illumination from a novel viewpoint. In this example, the performance is recovered using only the single camera viewpoint in (d).

Abstract

We present a process for rendering a realistic facial performance with control of viewpoint and illumination. The performance is based on one or more high-quality geometry and reflectance scans of an actor in static poses, driven by one or more video streams of a performance. We compute optical flow correspondences between neighboring video frames, and a sparse set of correspondences between static scans and video frames. The latter are made possible by leveraging the relightability of the static 3D scans to match the viewpoint(s) and appearance of the actor in videos taken in arbitrary environments. As optical flow tends to compute proper correspondence for some areas but not others, we also compute a smoothed, per-pixel confidence map for every computed flow, based on normalized cross-correlation. These flows and their confidences yield a set of weighted triangulation constraints among the static poses and the frames of a performance. Given a single artist-prepared face mesh for one static pose, we optimally combine the weighted triangulation constraints, along with a shape regularization term, into a consistent 3D geometry solution over the entire performance that is drift-free by construction. In contrast to previous work, even partial correspondences contribute to drift minimization, for example where a successful match is found in the eye region but not the mouth. Our shape regularization employs a differential shape term based on a spatially varying blend of the differential shapes of the static poses and neighboring dynamic poses, weighted by the associated flow confidences. These weights also permit dynamic reflectance maps to be produced for the performance by blending the static scan maps. Finally, as the geometry and maps are represented on a consistent artist-friendly mesh, we render the resulting high-quality animated face geometry and animated reflectance maps using standard rendering tools.

e-mail: {fyffe,jones,oalexander,debevec}@ict.usc.edu

1 Introduction

Recent facial geometry scanning techniques can capture very high resolution geometry, including high-frequency details such as skin pores and wrinkles. When animating these highly detailed faces, highly accurate temporal correspondence is required. At present, the highest quality facial geometry is produced by static scanning techniques, where the subject holds a facial pose for several seconds. This permits the use of high-resolution cameras for accurate stereo reconstruction and active illumination to recover pore-level resolution surface details. Such techniques also capture high-quality surface reflectance maps, enabling realistic rendering of the captured faces. Alternatively, static facial poses may be captured using facial casts combined with detail acquired from surface imprints. Unfortuately, dynamic scanning techniques are unable to provide the same level of detail as static techniques, even when high-speed cameras and active illumination are employed.

The classic approach to capturing facial motion is to use markers or face paint to track points on the face. However, such techniques struggle to capture the motion of the eyes and mouth, and rely on a high-quality facial rig to provide high-frequency skin motion and wrinkling. The best results are achieved when the rig is based on high-resolution static scans of the same subject. A second approach is to capture a performance with one or more passive video cameras. Such setups are lightweight as they use environmental illumination and off-the-shelf video cameras. As the camera records the entire face, it should be possible to recover eye and mouth motion missed by sparse markers. Still, by itself, passive video cannot match the resolution of static scans. While it is possible to emboss some video texture on the face [Bradley et al. 2010][Beeler et al. 2011][Valgaerts et al. 2012], many facial details appear only in specular reflections and are not visible under arbitrary illumination.

We present a technique for creating realistic facial animation from a set of high-resolution scans of an actor's face, driven by passive video of the actor from one or more viewpoints. The videos can be shot under existing environmental illumination using off-the-shelf

HD video cameras. The static scans can come from a variety of sources including facial casts, passive stereo, or active illumination techniques. High-resolution detail and relightable reflectance properties in the static scans can be transferred to the performance using generated per-pixel weight maps. We operate our algorithm on a performance flow graph that represents dense correspondences between dynamic frames and multiple static scans, leveraging GPU-based optical flow to efficiently construct the graph. Besides a single artist remesh of a scan in neutral pose, our method requires no rigging, no training of appearance models, no facial feature detection, and no manual annotation of any kind. As a byproduct of our method we also obtain a non-rigid registration between the artist mesh and each static scan. Our principal contributions are:

- An efficient scheme for selecting a sparse subset of image pairs for optical flow computation for drift-free tracking.
- A fully coupled 3D tracking method with differential shape regularization using multiple locally weighted target shapes.
- A message-passing-based optimization scheme leveraging lazy evaluation of energy terms enabling fully-coupled optimization over an entire performance.

2 Related Work

As many systems have been built for capturing facial geometry and reflectance, we will restrict our discussion to those that establish some form of dense temporal correspondence over a performance.

Many existing algorithms compute temporal correspondence for a sequence of temporally inconsistent geometries generated by e.g. structured light scanners or stereo algorithms. These algorithms operate using only geometric constraints [Popa et al. 2010] or by deforming template geometry to match each geometric frame [Zhang et al. 2004]. The disadvantage of this approach is that the per-frame geometry often contains missing regions or erroneous geometry which must be filled or filtered out, and any details that are missed in the initial geometry solution are non-recoverable.

Other methods operate on video footage of facial performances. Methods employing frame-to-frame motion analysis are subject to the accumulation of error or "drift" in the tracked geometry, prompting many authors to seek remedies for this issue. We therefore limit our discussion to methods that make some effort to address drift. Li et al. [1993] compute animated facial blendshape weights and rigid motion parameters to match the texture of each video frame to a reference frame, within a local minimum determined by a motion prediction step. Drift is avoided whenever a solid match can be made back to the reference frame. [DeCarlo and Metaxas 1996] solves for facial rig control parameters to agree with sparse monocular optical flow constraints, applying forces to pull model edges towards image edges in order to combat drift. [Guenter et al. 1998] tracks motion capture dots in multiple views to deform a neutral facial scan, increasing the realism of the rendered performance by projecting video of the face (with the dots digitally removed) onto the deforming geometry. The "Universal Capture" system described in [Borshukov et al. 2003] dispenses with the dots and uses dense multi-view optical flow to propagate vertices from an initial neutral expression. User intervention is required to correct drift when it occurs. [Hawkins et al. 2004] uses performance tracking to automatically blend between multiple high-resolution facial scans per facial region, achieving realistic multi-scale facial deformation without the need for reprojecting per-frame video, but uses dots to avoid drift. Bradley et al. [2010] track motion using dense multi-view optical flow, with a final registration step between the neutral mesh and every subsequent frame to reduce drift. Beeler et al. [2011] explicitly identify anchor frames that are similar to a manually chosen reference pose using a simple image difference metric, and track the performance bidirectionally between anchor frames. Non-sequential surface tracking [Klaudiny and Hilton 2012] finds a minimum-cost spanning tree over the frames in a performance based on sparse feature positions, tracking facial geometry across edges in the tree with an additional temporal fusion step. Valgaerts et al. [2012] apply scene flow to track binocular passive video with a regularization term to reduce drift.

One drawback to all such optical flow tracking algorithms is that the face is tracked from one pose to another as a whole, and success of the tracking depends on accurate optical flow between images of the entire face. Clearly, the human face is capable of repeating different poses over different parts of the face asynchronously, which the holistic approaches fail to model. For example, if the subject is talking with eyebrows raised and later with eyebrows lowered, a holistic approach will fail to exploit similarities in mouth poses when eyebrow poses differ. In contrast, our approach constructs a graph considering similarities over multiple regions of the face across the performance frames and a set of static facial scans, removing the need for sparse feature tracking or anchor frame selection.

Blend-shape based animation rigs are also used to reconstruct dynamic poses based on multiple face scans. The company Image Metrics (now Faceware) has developed commercial software for driving a blend-shape rig with passive video based on active appearance models [Cootes et al. 1998]. Weise et al. [2011] automatically construct a personalized blend shape rig and drive it with Kinect depth data using a combination of as-rigid-as-possible constraints and optical flow. In both cases, the quality of the resulting tracked performance is directly related to the quality of the rig. Each tracked frame is a linear combination of the input blend-shapes, so any performance details that lie outside the domain spanned by the rig will not be reconstructed. Huang et al. [2011] automatically choose a minimal set of blend shapes to scan based on previously captured performance with motion capture markers. Recreating missing detail requires artistic effort to add corrective shapes and cleanup animation curves [Alexander et al. 2009]. There has been some research into other non-traditional rigs incorporating scan data. Ma et al. [2008] fit a polynomial displacement map to dynamic scan training data and generate detailed geometry from sparse motion capture markers. Bickel et al. [2008] locally interpolate a set of static poses using radial basis functions driven by motion capture markers. Our method combines the shape regularization advantages of blendshapes with the flexibility of optical flow based tracking. Our optimization algorithm leverages 3D information from static scans without constraining the result to lie only within the linear combinations of the scans. At the same time, we obtain per-pixel blend weights that can be used to produce per-frame reflectance maps.

3 Data Capture and Preparation

We capture high-resolution static geometry using multi-view stereo and gradient-based photometric stereo [Ghosh et al. 2011]. The scan set includes around 30 poses largely inspired by the Facial Action Coding System (FACS) [Ekman and Friesen 1978], selected to span nearly the entire range of possible shapes for each part of the face. For efficiency, we capture some poses with the subject combining FACS action units from the upper and lower half of the face. For example, combining eyebrows raise and cheeks puff into a single scan. Examples of the input scan geometry can be seen in Fig. 2. A base mesh is defined by an artist for the neutral pose scan. The artist mesh has an efficient layout with edge loops following the wrinkles of the face. The non-neutral poses are represented as raw scan geometry, requiring no artistic topology or remeshing.

We capture dynamic performances using up to six Canon 1DX DSLR cameras under constant illumination. In the simplest case, we use the same cameras that were used for the static scans and switch to 1920□1080 30p movie mode. We compute a sub-frame-accurate synchronization offset between cameras using a correlation analysis of the audio tracks. This could be omitted if cameras with hardware synchronization are employed. Following each performance, we capture a video frame of a calibration target to calibrate camera intrinsics and extrinsics. We relight (and when necessary, repose) the static scan data to resemble the illumination conditions observed in the performance video. In the simplest case, the illumination field resembles one of the photographs taken during the static scan process and no relighting is required.

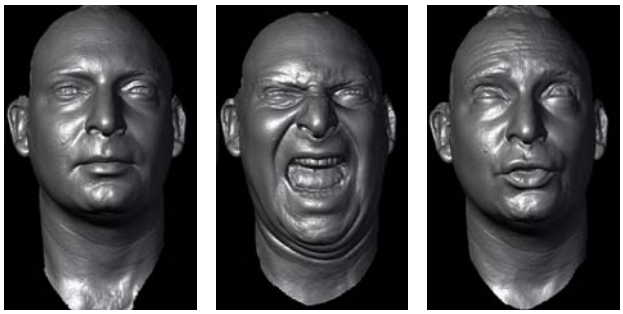


Figure 2: Sample static scans (showing geometry only).

4 The Performance Flow Graph

Optical-flow-based tracking algorithms such as [Bradley et al. 2010][Beeler et al. 2011][Klaudiny and Hilton 2012] relate frames of a performance to each other based on optical flow correspondences over a set of image pairs selected from the performance. These methods differ in part by the choice of the image pairs to be employed. We generalize this class of algorithms using a structure we call the performance flow graph, which is a complete graph with edges representing dense 2D correspondences between all pairs of images, with each edge having a weight, or confidence, of the associated estimated correspondence field. The graphs used in previous works, including anchor frames [Beeler et al. 2011] and non-sequential alignment with temporal fusion [Klaudiny and Hilton 2012], can be represented as a performance flow graph having unit weight for the edges employed by the respective methods, and zero weight for the unused edges. We further generalize the performance flow graph to include a dense confidence field associated with each correspondence field, allowing the confidence to vary spatially over the image. This enables our technique to exploit relationships between images where only a partial correspondence was able to be computed (for example, a pair of images where the mouth is similar but the eyes are very different). Thus our technique can be viewed as an extension of anchor frames or minimum spanning trees to minimize drift independently over different regions of the face.

A performance capture system that considers correspondences between all possible image pairs naturally minimizes drift. However, this would require an exorbitant number of graph edges, so we instead construct a graph with a reduced set of edges that approximates the complete graph, in the sense that the correspondences are representative of the full set with respect to confidence across the regions of the face. Our criterion for selecting the edges to include in the performance flow graph is that any two images having a high confidence correspondence between them in the complete graph of possible correspondences ought to have a path between them (a concatenation of one or more correspondences) in the constructed graph with nearly as high confidence (including the reduction in confidence from concatenation). We claim that correspondences between temporally neighboring dynamic frames are typically of high quality, and no concatenation of alternative correspondences can be as confident, therefore we always include a graph edge between each temporally neighboring pair of dynamic frames. Correspondences between frames with larger temporal gaps are well-approximated by concatenating neighbors, but decreasingly so over larger temporal gaps (due to drift). We further claim that whenever enough drift accumulates to warrant including a graph edge over the larger temporal gap, there exists a path with nearly as good confidence that passes through one of the predetermined static scans (possibly a different static scan for each region of the face). We justify this claim by noting the 30 static poses based on FACS ought to span the space of performances well enough that any region of any dynamic frame can be corresponded to some region in some static scan with good confidence. Therefore we do not include any edges between non-neighboring dynamic frames, and instead consider only edges between a static scan and a dynamic frame as candidates for inclusion (visualized in Fig. 3). Finally, as the drift accumulated from the concatenation described above warrants additional edges only sparsely over time, we devise a coarse-to-fine graph construction strategy using only a sparse subset of static-to-dynamic graph edges. We detail this strategy in Section 4.1.

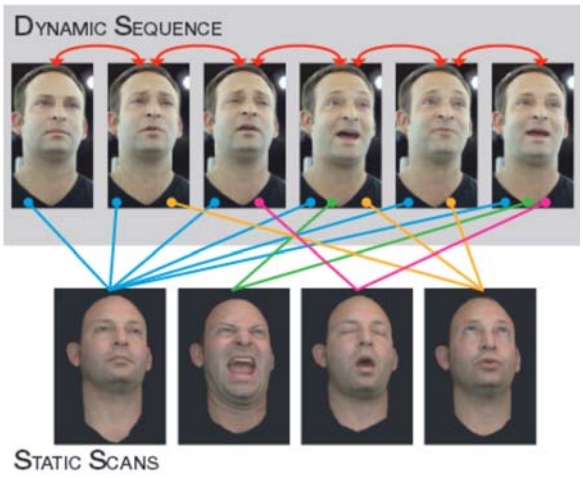


Figure 3: performance flow graph showing optical flow correspondences between static and dynamic images. Red lines represent optical flow between neighboring frames within a performance. Blue, green, and orange lines represent optical flow between dynamic and static images. Based on initial low-resolution optical flow, we construct a sparse graph requiring only a small subset of high resolution flows to be computed between static scans and dynamic frames.

4.1 Constructing the Performance Flow Graph

The images used in our system consist of one or more dynamic sequences of frames captured from one or more viewpoints, and roughly similar views of a set of high-resolution static scans. The nodes in our graph represent static poses (associated with static scans) and dynamic poses (associated with dynamic frames from one or more sequences). We construct the performance flow graph by computing a large set of static-to-dynamic optical flow correspondences at a reduced resolution for only a single viewpoint, and then omit redundant correspondences using a novel voting algorithm to select a sparse set of correspondences that is representative of the original set. We then compute high-quality optical flow correspondences at full resolution for the sparse set, and include all

viewpoints. The initial set of correspondences consists of quarter-resolution optical flows from each static scan to every n^{th} dynamic frame. For most static scans we use every 5^{th} dynamic frame, while for the eyes-closed scan we use every dynamic frame in order to catch rapid eye blinks. We then compute normalized cross correlation fields between the warped dynamic frames and each original static scan to evaluate the confidence of the correspondences. These correspondences may be computed in parallel over multiple computers, as there is no sequential dependency between them. We find that at quarter resolution, flow-based cross correlation correctly assigns low confidence to incorrectly matched facial features, for example when flowing disparate open and closed mouth shapes. To reduce noise and create a semantically meaningful metric, we average the resulting confidence over twelve facial regions (see Fig. 4). These facial regions are defined once on the neutral pose, and are warped to all other static poses using rough static-to-static optical flow. Precise registration of regions is not required, as they are only used in selecting the structure of the performance graph. In the subsequent tracking phase, per-pixel confidence is used.

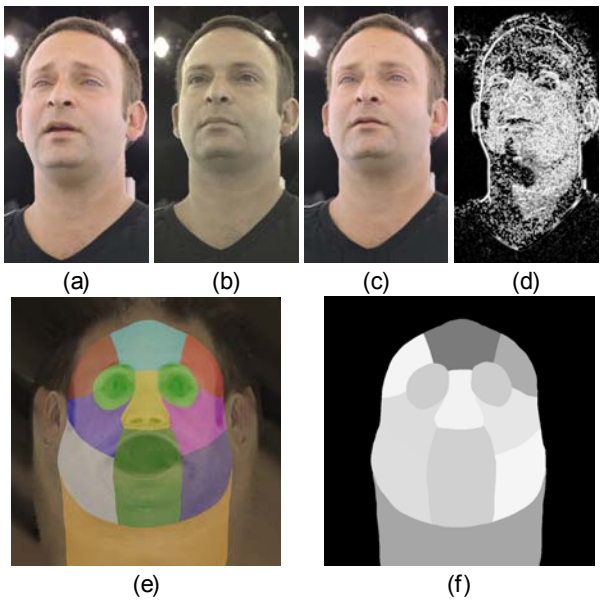


Figure 4: We compute an initial low-resolution optical flow between a dynamic image (a) and static image (b). We then compute normalized cross correlation between the static image (b) and the warped dynamic image (c) to produce the per-pixel confidence shown in (d). We average these values for 12 regions (e) to obtain a per-region confidence value (f). This example shows correlation between the neutral scan and a dynamic frame with the eyebrows raised and the mouth slightly open. The forehead and mouth regions are assigned appropriately lower confidences.

Ideally we want the performance flow graph to be sparse. Besides temporally adjacent poses, dynamic poses should only connect to similar static poses and edges should be evenly distributed over time to avoid accumulation of drift. We propose an iterative greedy voting algorithm based on the per-region confidence measure to identify good edges. The confidence of correspondence between the dynamic frames and any region of any static facial scan can be viewed as a curve over time (depicted in Fig. 5). In each iteration we identify the maximum confidence value over all regions, all scans, and all frames. We add an edge between the identified dynamic pose and static pose to the graph. We then adjust the recorded confidence of the identified region by subtracting a hat function scaled by the maximum confidence and centered around the maximum frame, indicating that the selected edge has been accounted for, and temporal

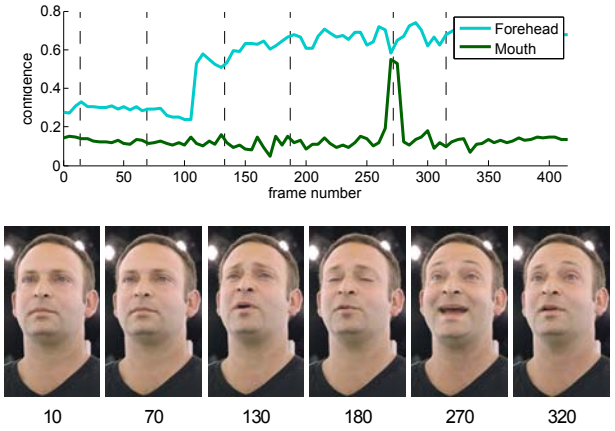


Figure 5: A plot of the per-region confidence metric over time. Higher numbers indicate greater correlation between the dynamic frames and a particular static scan. The cyan curve represents the center forehead region of a brows-raised static scan which is active throughout the later sequence. The green curve represents the mouth region for an extreme mouth-open scan which is active only when the mouth opens to its fullest extent. The dashed lines indicate the timing of the sampled frames shown on the bottom row.

neighbors partly so. All other regions are adjusted by subtracting similar hat functions, scaled by the (non-maximal) per-region confidence of the identified flow. This suppresses any other regions that are satisfied by the flow. The slope of the hat function represents a loss of confidence as this flow is combined with adjacent dynamic-to-dynamic flows. We then iterate and choose the new highest confidence value, until all confidence values fall below a threshold. The two parameters (the slope of the hat function and the final threshold value) provide intuitive control over the total number of graph edges. We found a reasonable hat function falloff to be a 4% reduction for every temporal flow and a threshold value that is 20% of the initial maximum confidence. After constructing the graph, a typical 10-20 second performance flow graph will contain 100-200 edges between dynamic and static poses. Again, as the change between sequential frames is small, we preserve all edges between neighboring dynamic poses.

After selecting the graph edges, final HD resolution optical flows are computed for all active cameras and for all retained graph edges. We directly load video frames using nVidia's h264 GPU decoder and feed them to the FlowLib implementation of GPU-optical flow [Werlberger 2012]. Running on a Nvidia GTX 680, computation of quarter resolution flows for graph construction take less than one second per flow. Full-resolution HD flows for dynamic-to-dynamic images take 8 seconds per flow, and full-resolution flows between static and dynamic images take around 23 seconds per flow due to a larger search window. More sophisticated correspondence estimation schemes could be employed within our framework, but our intention is that the framework be agnostic to this choice and robust to imperfections in the pairwise correspondences. After computing optical flows and confidences, we synchronize all the flow sequences to a primary camera by warping each flow frame forward or backward in time based on the sub-frame synchronization offsets between cameras.

We claim that an approximate performance flow graph constructed in this manner is more representative of the complete set of possible correspondences than previous methods that take an all-or-nothing approach to pair selection, while still employing a number of optical flow computations on the same order as previous methods (i.e. temporal neighbors plus additional sparse image pairs).

5 Fully Coupled Performance Tracking

The performance flow graph is representative of all the constraints we could glean from 2D correspondence analysis of the input images, and now we aim to put those constraints to work. We formulate an energy function in terms of the 3D vertex positions of the artist mesh as it deforms to fit all of the dynamic and static poses in the performance flow graph in a common head coordinate system, as well as the associated head-to-world rigid transforms. We collect the free variables into a vector $\Theta = (x_i^p; R_p; t_p | p \in D \cup S; i \in V)$, where x_i^p represents the 3D vertex position of vertex i at pose p in the common head coordinate system, R_p and t_p represent the rotation matrix and translation vector that rigidly transform pose p from the common head coordinate system to world coordinates, D is the set of dynamic poses, S is the set of static poses, and V is the set of mesh vertices. The energy function is then:

$$E(\Theta) = \sum_{(p,q)\in F} (E_{corr}^{pq} + E_{corr}^{qp}) + \alpha \sum_{p \in D \cup S} |F_p| E_{shape}^p + \beta \sum_{p \in S} |F_p| E_{wrap}^p + \gamma |F_g| E_{ground}; \quad (1)$$

where F is the set of performance flow graph edges, F_p is the subset of edges connecting to pose p, and g is the ground (neutral) static pose. This function includes:

- dense correspondence constraints E_{corr}^{pq} associated with the edges of the performance flow graph,
- shape regularization terms E_{shape}^p relating the differential shape of dynamic and static poses to their graph neighbors,
- "shrink wrap" terms E_{wrap}^p to conform the static poses to the surface of the static scan geometries,
- a final grounding term E_{ground} to prefer the vertex positions in a neutral pose to be close to the artist mesh vertex positions.

We detail these terms in sections 5.2 - 5.5. Note we do not employ a stereo matching term, allowing our technique to be robust to small synchronization errors between cameras. As the number of poses and correspondences may vary from one dataset to another, the summations in (1) contain balancing factors (to the immediate right of each $\sum$) in order to have comparable total magnitude (proportional to $|F|$). The terms are weighted by tunable term weights α, β and γ, which in all examples we set equal to 1.

5.1 Minimization by Lazy DDMS-TRWS

In contrast to previous work, we consider the three-dimensional coupling between all terms in our formulation, over all dynamic and static poses simultaneously, thereby obtaining a robust estimate that gracefully fills in missing or unreliable information. This presents two major challenges. First, the partial matches and loops in the performance flow graph preclude the use of straightforward mesh propagation schemes used in previous works. Such propagation would produce only partial solutions for many poses. Second (as a result of the first) we lack a complete initial estimate for traditional optimization schemes such as Levenberg-Marquadt.

To address these challenges, we employ an iterative scheme that admits partial intermediate solutions, with pseudocode in Algorithm 1. As some of the terms in (1) are data-dependent, we adapt the outer loop of Data Driven Mean-Shift Belief Propagation (DDMSBP) [Park et al. 2010], which models the objective function in each iteration as an increasingly-tight Gaussian (or quadratic) approximation of the true function. Within each DDMS loop, we use Gaussian Tree-Reweighted Sequential message passing (TRW-S) [Kolmogorov 2006], adapted to allow the terms in the model to be constructed lazily as the solution progresses over the variables. Hence we call our scheme Lazy DDMS-TRWS. We define the ordering of the variables to be pose-major (i.e. visiting all the vertices of one pose, then all the vertices of the next pose, etc.), with static poses followed by dynamic poses in temporal order. We decompose the Gaussian belief as a product of 3D Gaussians over vertices and poses, which admits a pairwise decomposition of (1) as a sum of quadratics. We denote the current belief of a vertex i for pose p as $\bar{x}_i^p$ with covariance Λ_i^p (stored as inverse covariance for convenience), omitting the i subscript to refer to all vertices collectively. We detail the modeling of the energy terms in sections 5.2 - 5.5, defining $\bar{y}_i^p = R_p \bar{x}_i^p + t_p$ as shorthand for world space vertex position estimates. We iterate the DDMS loop 6 times, and iterate TRW-S until 95% of the vertices converge to within 0.01mm.

Algorithm 1 Lazy DDMS-TRWS for (1)

```
∀p,i : (Λ_i^p)^{-1} ← 0.
for DDMS outer iterations do
    // Reset the model:
    ∀p,q : E_corr^pq, E_shape^p, E_wrap^p ← undefined (effectively 0).
    for TRW-S inner iterations do
        // Major TRW-S loop over poses:
        for each p ∈ D ∪ S in order of increasing o(p) do
            // Update model where possible:
            for each q | (p,q) ∈ F do
                if (Λ^p)^{-1} ≠ 0 and E_corr^pq undefined then
                    E_corr^pq ← model fit using (2) in section 5.2.
                if (Λ^q)^{-1} ≠ 0 and E_corr^qp undefined then
                    E_corr^qp ← model fit using (2) in section 5.2.
            if (Λ^p)^{-1} ≠ 0 and E_wrap^p undefined then
                E_wrap^p ← model fit using (8) in section 5.4.
            if ∃(p,q)∈F |(Λ^q)^{-1} ≠ 0 and E_shape^p undefined then
                E_shape^p ← model fit using (5) in section 5.3.
            // Minor TRW-S loop over vertices:
            Pass messages based on (1) to update x̄^p, (Λ^p)^{-1}.
            Update R_p, t_p as in section 5.6.
        // Reverse TRW-S ordering:
        o(s) ← k(D ∪ S) + 1 − o(s).
```

5.2 Modeling the Correspondence Term

The correspondence term in (1) penalizes disagreement between optical flow vectors and projected vertex locations. Suppose we have a 2D optical flow correspondence field between poses p and q in (roughly) the same view c. We may establish a 3D relationship between x_i^p and x_i^q implied by the 2D correspondence field, which we model as a quadratic penalty function:

$$E_{corr}^{pq} = \frac{1}{|C||V|} \sum_{c \in C} \sum_{i \in V} (x_i^q - x_i^p - f_{i\,pq}^c)^T F_{i\,pq}^c (x_i^q - x_i^p - f_{i\,pq}^c); \quad (2)$$

where C is the set of camera viewpoints, and $f_{i\,pq}^c; F_{i\,pq}^c$ are respectively the mean and precision matrix of the penalty, which we estimate from the current estimated positions as follows. We first project $\bar{y}_i^p$ into the image plane of view c of pose p. We then warp the 2D image position from view c of pose p to view c of pose q using the correspondence field. The warped 2D position defines a world-space view ray that the same vertex i ought to lie on in pose q. We transform this ray back into common head coordinates (via

$-t_q$, R_q^T) and penalize the squared distance from x_i^q to this ray. Letting r_{pq}^c represent the direction of this ray, this yields:

$$f_{pq}^c = (I - r_{pq}^c r_{pq}^{cT})(R_q^T(c_q^c - t_q) - \bar{x}_i^p); \quad (3)$$

where c_q^c is the nodal point of view c of pose q, and $r_{pq}^c = R_q^T d_{pq}^c$ with d_{pq}^c the world-space direction of the ray in view c of pose q through the 2D image plane point $f_{pq}^c[P_p^c(\bar{y}_i^p)]$ (where square brackets represent bilinearly interpolated sampling of a field or image), f_{pq}^c the optical flow field transforming an image-space point from view c of pose p to the corresponding point in view c of pose q, and $P_p^c(x)$ the projection of a point x into the image plane of view c of pose p (which may differ somewhat from pose to pose). If we were to use the squared-distance-to-ray penalty directly, F_{pq}^c would be $I - r_{pq}^c r_{pq}^{cT}$, which is singular. To prevent the problem from being ill-conditioned and also to enable the use of monocular performance data, we add a small regularization term to produce a non-singular penalty, and weight the penalty by the confidence of the optical flow estimate. We also assume the optical flow field is locally smooth, so a large covariance Σ_i^p inversely influences the precision of the model, whereas a small covariance Σ_i^p does not, and weight the model accordingly. Intuitively, this weighting causes information to propagate from the ground term outward via the correspondences in early iterations, and blends correspondences from all sources in later iterations. All together, this yields:

$$F_{pq}^c = \min(1; \det(\Sigma_i^p)^{-\frac{1}{3}}) v_{pq}^c \alpha_{pq}^c [P_p^c(\bar{y}_i^p)](I - r_{pq}^c r_{pq}^{cT} + \varepsilon I); \quad (4)$$

where v_{pq}^c is a soft visibility factor (obtained by blurring a binary vertex visibility map and modulated by the cosine of the angle between surface normal and view direction), α_{pq}^c is the confidence field associated with the correspondence field f_{pq}^c, and ε is a small regularization constant. We use $\det(\cdot)^{-1/3}$ as a scalar form of precision for 3D Gaussians.

5.3 Modeling the Differential Shape Term

The shape term in (1) constrains the differential shape of each pose to a spatially varying convex combination of the differential shapes of the neighboring poses in the performance flow graph:

$$E_{shape}^p = \sum_{(i;j) \in E} \left\| x_j^p - x_i^p - l_{ij}^p \right\|^2; \quad (5)$$

$$l_{ij}^p = \frac{\varepsilon(g_j - g_i) + \sum_{q|(p;q) \in F} w_{ij}^{pq} (\bar{x}_j^q - \bar{x}_i^q)}{\varepsilon + \sum_{q|(p;q) \in F} w_{ij}^{pq}}; \quad (6)$$

$$w_{ij}^{pq} = \frac{w_i^{pq} w_j^{pq}}{w_i^{pq} + w_j^{pq}}; \quad (7)$$

where E is the set of edges in the geometry mesh, $w_i^{pq} = \det(\frac{1}{|C|} \sum_{c \in C} F_{pq}^c + F_{qp}^c)^{1/3}$ (which is intuitively the strength of the relationship between poses p and q due to the correspondence term), g denotes the artist mesh vertex positions, and ε is a small regularization constant. The weights w_i^{pq} additionally enable trivial synthesis of high-resolution reflectance maps for each dynamic frame of the performance by blending the static pose data.

5.4 Modeling the Shrink Wrap Term

The shrink wrap term in (1) penalizes the distance between static pose vertices and the raw scan geometry of the same pose. We model this as a regularized distance-to-plane penalty:

$$E_{wrap}^p = \sum_{i \in V} (x_i^p - d_i^p)^T g_i^p (n_i^p n_i^{pT} + \varepsilon I)(x_i^p - d_i^p); \quad (8)$$

where $(n_i^p; d_i^p)$ are the normal and centroid of a plane fitted to the surface of the static scan for pose p close to the current estimate $\bar{x}_i^p$ in common head coordinates, and g_i^p is the confidence of the planar fit. We obtain the planar fit inexpensively by projecting $\bar{y}_i^p$ into each camera view, and sampling the raw scan surface via a set of precomputed rasterized views of the scan. (Alternatively, a 3D search could be employed to obtain the samples.) Each surface sample (excluding samples that are occluded or outside the rasterized scan) provides a plane equation based on the scan geometry and surface normal. We let n_i^p and d_i^p be the weighted average values of the plane equations over all surface samples:

$$n_i^p = \sum_{c \in C} \omega_i^c R_p^T \hat{n}_p^c [P_p^c(\bar{y}_i^p)] \text{ (normalized)}; \quad (9)$$

$$d_i^p = \left(\sum_{c \in C} \omega_i^c \right)^{-1} \sum_{c \in C} \omega_i^c R_p^T (\hat{d}_p^c [P_p^c(\bar{y}_i^p)] - t_p); \quad (10)$$

$$g_i^p = \min(1; \det(\Sigma_i^p)^{-\frac{1}{3}}) \sum_{c \in C} \omega_i^c; \quad (11)$$

where $(\hat{n}_p^c; \hat{d}_p^c)$ are the world-space surface normal and position images of the rasterized scans, and $\omega_i^c = 0$ if the vertex is occluded in view c or lands outside of the rasterized scan, otherwise $\omega_i^c = v_i^c \exp(-k \hat{d}_p^c [P_p^c(\bar{y}_i^p)] - \bar{y}_i^p \|^2)$.

5.5 Modeling the Ground Term

The ground term in (1) penalizes the distance between vertex positions in the ground (neutral) pose and the artist mesh geometry:

$$E_{ground} = \sum_{i \in V} \left\| x_i^g - R_g^T g_i \right\|^2; \quad (12)$$

where g_i is the position of the vertex in the artist mesh. This term is simpler than the shrink-wrap term since the pose vertices are in one-to-one correspondence with the artist mesh vertices.

5.6 Updating the Rigid Transforms

We initialize our optimization scheme with all $(\Sigma_i^p)^{-1} = 0$ (and hence all $\bar{x}_i^p$ moot), fully relying on the lazy DDMS-TRWS scheme to propagate progressively tighter estimates of the vertex positions x_i^p throughout the solution. Unfortunately, in our formulation the rigid transforms $(R_p; t_p)$ enjoy no such treatment as they always occur together with x_i^p and would produce non-quadratic terms if they were included in the message passing domain. Therefore we must initialize the rigid transforms to some rough initial guess, and update them after each iteration. The neutral pose is an exception, where the transform is specified by the user (by rigidly posing the artist mesh to their whim) and hence not updated. In all our examples, the initial guess for all poses is simply the same as the user-specified rigid transform of the neutral pose. We update $(R_p; t_p)$ using a simple scheme that aligns the neutral artist mesh to the current result. Using singular value decomposition, we compute the closest rigid transform minimizing $\sum_{i \in V} r_i \left\| R_p g_i + t_p - \bar{R}_p \bar{x}_i^p - \bar{t}_p \right\|^2$, where r_i is a rigidity weight value (high weight around the eye sockets and temples, low weight elsewhere), g_i denotes the artist mesh vertex positions, and $(\bar{R}_p; \bar{t}_p)$ is the previous transform estimate.

5.7 Accelerating the Solution Using Keyframes

Minimizing the energy in (1) over the entire sequence requires multiple iterations of the TRW-S message passing algorithm, and multiple iterations of the DDMS outer loop. We note that the performance flow graph assigns static-to-dynamic flows to only a sparse subset of performance frames, which we call keyframes. Correspondences among the spans of frames in between keyframes are reliably represented using concatenation of temporal flows. Therefore to reduce computation time we first miminize the energy at only the keyframes and static poses, using concatenated temporal flows in between keyframes. Each iteration of this reduced problem is far cheaper than the full problem, so we may obtain a satisfactory solution of the performance keyframes and static poses more quickly. Next, we keep the static poses and keyframe poses fixed, and solve the spans of in-between frames, omitting the shrink-wrap and grounding terms as they affect only the static poses. This subsequent minimization requires only a few iterations to reach a satisfactory result, and each span of in-between frames may be solved independently (running on multiple computers, for example).

6 Handling Arbitrary Illumination and Motion

Up to now, we have assumed that lighting and overall head motion in the static scans closely matches that in the dynamic frames. For performances in uncontrolled environments, the subject may move or rotate their head to face different cameras, and lighting may be arbitrary. We handle such complex cases by taking advantage of the 3D geometry and relightable reflectance maps in the static scan data. For every 5th performance frame, we compute a relighted rendering of each static scan with roughly similar rigid head motion and lighting environment as the dynamic performance. These renderings are used as the static expression imagery in our pipeline. The rigid head motion estimate does not need to be exact as the optical flow computation is robust to a moderate degree of misalignment. In our results, we (roughly) rigidly posed the head by hand, though automated techniques could be employed [Zhu and Ramanan 2012]. We also assume that a HDR light probe measurement [Debevec 1998] exists for the new lighting environment, however, lighting could be estimated from the subject's face [Valgaerts et al. 2012] or eyes [Nishino and Nayar 2004].

The complex backgrounds in real-world uncontrolled environments pose a problem, as optical flow vectors computed on background pixels close to the silhouette of the face may confuse the correspondence term if the current estimate of the facial geometry slightly overlaps the background. This results in parts of the face "sticking" to the background as the subject's face turns from side to side (Fig. 6). To combat this, we weight the correspondence confidence field by a simple soft segmentation of head vs. background. Since head motion is largely rigid, we fit a 2D affine transform to the optical flow vectors in the region of the current head estimate. Then, we weight optical flow vectors by how well they agree with the fitted transform. We also assign high weight to the region deep inside the current head estimate using a simple image-space erosion algorithm, to prevent large jaw motions from being discarded. The resulting soft segmentation effectively cuts the head out of the background whenever the head is moving, thus preventing the optical flow vectors of the background from polluting the edges of the face. When the head is not moving against the background the segmentation is poor, but in this case the optical flow vectors of the face and background agree and pollution is not damaging.

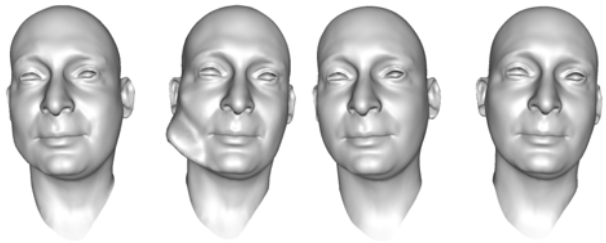


Figure 6: (a, b) Two frames of a reconstructed performance in front of a cluttered background, where the subject turns his head over the course of ten frames. The silhouette of the jaw "sticks" to the background because the optical flow vectors close to the jaw are stationary. (c, d) A simple segmentation of the optical flow field to exclude the background resolves the issue.

7 Results

We ran our technique on several performances from three different subjects. Each subject had 30 static facial geometry scans captured before the performance sessions, though the performance flow graph construction often employs only a fraction of the scans. An artist produced a single face mesh for each subject based on their neutral static facial scan.

7.1 Performances Following Static Scan Sessions

We captured performances of three subjects directly following their static scan sessions. The performances were recorded from six camera views in front of the subject with a baseline of approximately 15 degrees. Our method produced the performance animation results shown in Fig. 19 without any further user input.

7.2 Performances in Other Locations

We captured a performance of a subject using four consumer HD video cameras in an office environment. An animator rigidly posed a head model roughly aligned to every 5th frame of the performance, to produce the static images for our performance flow graph. Importantly, this rigid head motion does not need to be very accurate for our method to operate, and we intend that an automated technique could be employed. A selection of video frames from one of the views is shown in Fig. 7, along with renderings of the results of our method. Despite the noisy quality of the videos and the smaller size of the head in the frame, our method is able to capture stable facial motion including lip synching and brow wrinkles.

7.3 High-Resolution Detail Transfer

After tracking a performance, we transfer the high-resolution reflectance maps from the static scans onto the performance result. As all results are registered to the same UV parameterization by our method, the transfer is a simple weighted blend using the cross-correlation-based confidence weights w_i^{pq} of each vertex, interpolated bilinearly between vertices. We also compute values for w_i^{pq} for any dynamic-to-static edge pq that was not present in the performance flow graph, to produce weights for every frame of the performance. This yields detailed reflectance maps for every performance frame, suitable for realistic rendering and relighting. In addition to transferring reflectance, we also transfer geometric details in the form of a displacement map, allowing the performance tracking to operate on a medium-resolution mesh instead of the full scan resolution. Fig. 8 compares transferring geometric details

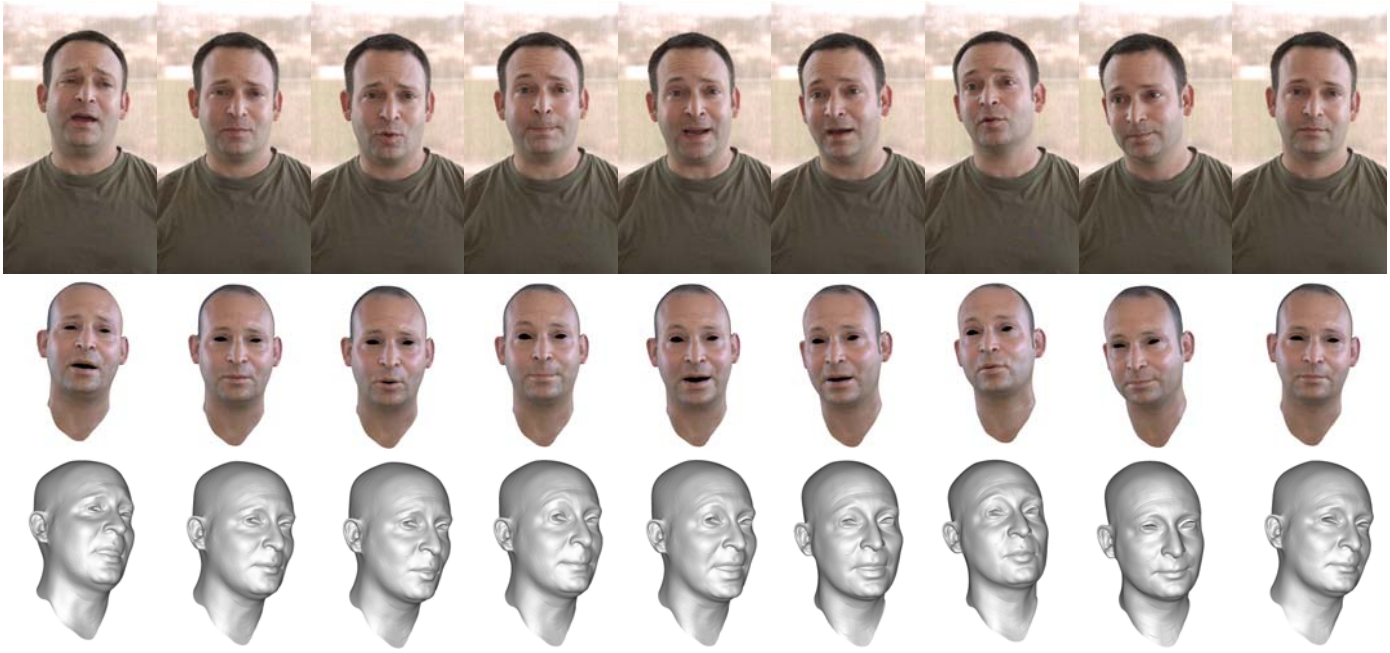
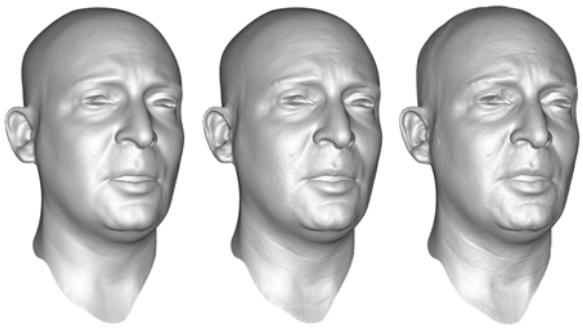


Figure 7: A performance captured in an office environment with uncontrolled illumination, using four HD consumer video cameras and seven static expression scans. Top row: a selection of frames from one of the camera views. Middle row: geometry tracked using the proposed method, with reflectance maps automatically assembled from static scan data, shaded using a high-dynamic-range light probe. The reflectance of the top and back of the head were supplemented with artist-generated static maps. The eyes and inner mouth are rendered as black as our method does not track these features. Bottom row: gray-shaded geometry for the same frames, from a novel viewpoint. Our method produces stable animation even with somewhat noisy video footage and significant head motion. Dynamic skin details such as brow wrinkles are transferred from the static scans in a manner faithful to the video footage.



(a) (b) (c)

Figure 8: High-resolution details may be transferred to a medium-resolution tracked model to save computation time. (a) medium-resolution tracked geometry using six views. (b) medium-resolution geometry with details automatically transferred from the high-resolution static scans. (c) high-resolution tracked geometry. The transferred details in (b) capture most of the dynamic facial details seen in (c) at a reduced computational cost.

from the static scans onto a medium-resolution reconstruction to directly tracking a high-resolution mesh. As the high-resolution solve is more expensive, we first perform the medium-resolution solve and use it to prime the DDMS-TRWS belief in the high-resolution solve, making convergence more rapid. In all other results, we show medium-resolution tracking with detail transfer, as the results are satisfactory and far cheaper to compute.

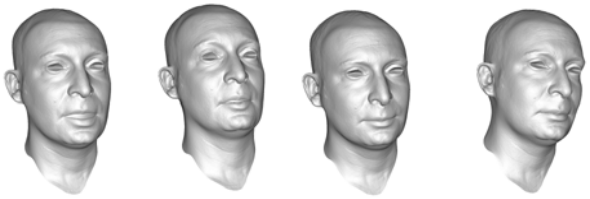


Figure 9: Results using only a single camera view, showing the last four frames from Fig. 7. Even under uncontrolled illumination and significant head motion, tracking is possible from a single view, at somewhat reduced fidelity.

7.4 Monocular vs. Binocular vs. Multi-View

Our method operates on any number of camera views, producing a result from even a single view. Fig. 9 shows results from a single view for the same uncontrolled-illumination sequence as Fig. 7. Fig. 10 shows the incremental improvement in facial detail for a controlled-illumination sequence using one, two, and six views. Our method is applicable to a wide variety of camera and lighting setups, with graceful degradation as less information is available.

7.5 Influence of Each Energy Term

The core operation of our method is to propagate a known facial pose (the artist mesh) to a set of unknown poses (the dynamic frames and other static scans) via the ground term and correspondence terms in our energy formulation. The differential shape term and shrink wrap term serve to regularize the shape of the solution. We next explore the influence of these terms on the solution.

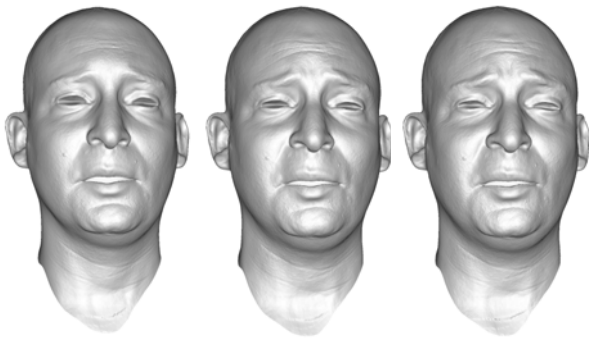


Figure 10: Example dynamic performance frame reconstructed from (a) one view, (b) two views and (c) six views. Our method gracefully degrades as less information is available.

Figure 11: The artist mesh is non-rigidly registered to each of the other static expression scans as a byproduct of our method. The registered artist mesh is shown for a selection of scans from two different subjects. Note the variety of mouth shapes, all of which are well-registered by our method without any user input.

Correspondence Term The correspondence term produces a consistent parameterization of the geometry suitable for texturing and other editing tasks. As our method computes a coupled solution of performance frames using static poses to bridge larger temporal gaps, the artist mesh is non-rigidly registered to each of the static scans as a byproduct of the optimization. (See Fig. 11 for examples.) Note especially that our method automatically produces a complete head for each expression, despite only having static facial scan geometry for the frontal face surface. As shown in Fig. 12, this consistency is maintained even when the solution is obtained from a different performance. Fig. 13 illustrates that the use of multiple static expression scans in the performance flow graph produces a more expressive performance, with more accentuated facial expression features, as there are more successful optical flow regions in the face throughout the performance.

Differential Shape Term In our formulation, the differential shape of a performance frame or pose is tied to a blend of its neighbors on the performance flow graph. This allows details from multiple static poses to propagate to related poses. Even when only one

Figure 12: Top row: neutral mesh with checker visualization of texture coordinates, followed by three non-rigid registrations to other facial scans as a byproduct of tracking a speaking performance. Bottom row: the same, except the performance used was a series of facial expressions with no speaking. The non-rigid registration obtained from the performance-graph-based tracking is both consistent across expressions and across performances. Note, e.g. the consistent locations of the checkers around the contours of the lips.

static pose is used (i.e. neutral), allowing temporal neighbors to influence the differential shape provides temporal smoothing without overly restricting the shape of each frame. Fig. 13 (c, d) illustrates the loss of detail when temporal neighbors are excluded from the differential shape term (compare to a, b).

Shrink Wrap Term The shrink wrap term conforms the static poses to the raw geometry scans (Fig. 14). Without this term, subtle details in the static scans cannot be propagated to the performance result, and the recovered static poses have less fidelity to the scans.

7.6 Comparison to Previous Work

We ran our method on the data from [Beeler et al. 2011], using their recovered geometry from the first frame (frame 48) as the "artist" mesh in our method. For expression scans, we used the geometry from frames 285 (frown) and 333 (brow raise). As our method makes use of the expression scans only via image-space operations on camera footage or rasterized geometry, any point order information present in the scans is entirely ignored. Therefore in this test, it is as if the static scans were produced individually by the method of [Beeler et al. 2010]. We constructed a simple UV projection on the artist mesh for texture visualization purposes, and projected the video frames onto each frame's geometry to produce a per-frame UV texture map. To measure the quality of texture alignment over the entire sequence, we computed the temporal variance of each pixel in the texture map (shown in Fig.15 (a, b)), using contrast normalization to disregard low-frequency shading variation. The proposed method produces substantially lower temporal texture variance, indicating a more consistent alignment throughout the sequence, especially around the mouth. Examining the geometry in Fig.15 (c-f), the proposed method has generally comparable quality as the previous work, with the mouth-closed shape recovered more faithfully (which is consistent with the variance analy-

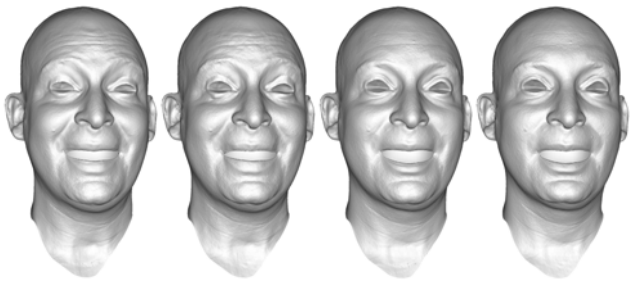


Figure 13: Using multiple static expressions in the performance flow graph produces more detail than using just a neutral static expression. Multiple static expressions are included in the performance flow graph in (a, c), whereas only the neutral expression is included in (b, d). By including temporal neighbors and static scans in determining the differential shape, details from the various static scans can be propagated throughout the performance. Differential shape is determined by the static expression(s) and temporal neighbors in (a, b), whereas temporal neighbors are excluded from the differential shape term in (c, d). Note the progressive loss of detail in e.g. the brow region from (a) to (d).

sis). We also compared to [Klaudiny and Hilton 2012] in a similar manner, using frame 0 as the artist mesh, and frames 25, 40, 70, 110, 155, 190, 225, 255 and 280 as static expressions. Again, no point order information is used. Fig. 16 again shows an overall lower temporal texture variance from the proposed method.

7.7 Performance Timings

We report performance timings in Fig. 17 for various sequences, running on a 16-core 2.4 GHz Xeon E5620 workstation (some operations are multithreaded across the cores). All tracked meshes have 65 thousand vertices, except Fig. 8(c) and Fig. 15 which have one million vertices. We report each stage of the process: "Graph" for the performance graph construction, "Flow" for the high-resolution optical flow calculations, "Key" for the performance tracking solve on key frames, and "Tween" for the performance tracking solve in between key frames. We mark stages that could be parallelized over multiple machines with an asterisk (*). High-resolution solves (Fig. 8(c) and Fig. 15) take longer than medium-resolution solves. Sequences with uncontrolled illumination (Fig. 7 and Fig. 9) take longer for the key frames to converge since the correspondence tying the solution to the static scans has lower confidence.

7.8 Discussion

Our method produces a consistent geometry animation on an artist-created neutral mesh. The animation is expressive and lifelike, and the subject is free to make natural head movements within a certain degree. Fig. 18 shows renderings from such a facial performance rendered using advanced skin and eye shading techniques as described in [Jimenez et al. 2012]. One notable shortcoming of our performance flow graph construction algorithm is the neglect of eye blinks. This results in a poor representation of the blinks in the final animation. Our method requires one artist-generated mesh per subject to obtain results that are immediately usable in production pipelines. Automatic generation of this mesh could be future work, or use existing techniques for non-rigid registration. Omitting this step would still produce a result, but would require additional cleanup around the edges as in e.g. [Beeler et al. 2011][Klaudiny and Hilton 2012].

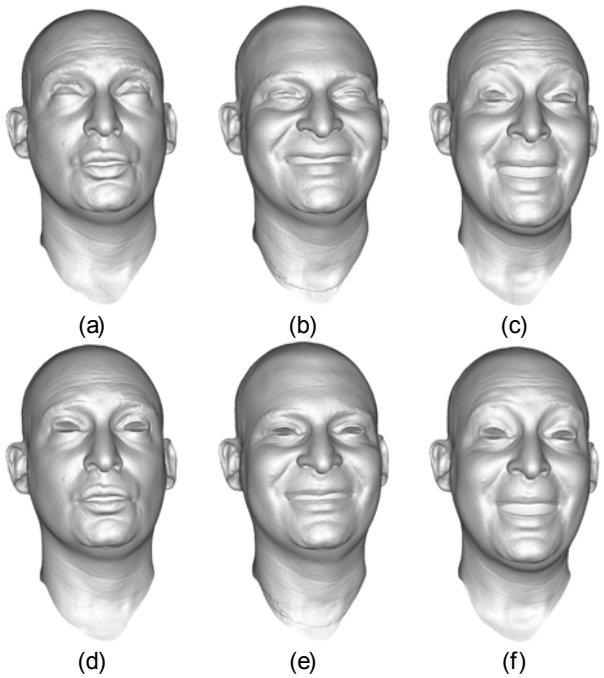


Figure 14: The shrink wrap term conforms the artist mesh to the static scan geometry, and also improves the transfer of expressive details to the dynamic performance. The registered artist mesh is shown for two static poses in (a) and (b), and a dynamic pose that borrows brow detail from (a) and mouth detail from (b) is shown in (c). Without the shrink wrap term, the registration to the static poses suffers (d, e) and the detail transfer to the dynamic performance is less sucessful (f). Fine-scale details are still transferred via displacement maps, but medium-scale expressive details are lost.

8 Future Work

One of the advantages of our technique is that it relates a dynamic performance back to facial shape scans using per-pixel weight maps. It would be desirable to further factor our results to create multiple localized blend shapes which are more semantically meaningful and artist friendly. Also, our algorithm does not explicitly track eye or mouth contours. Eye and mouth tracking could be further refined with additional constraints to capture eye blinks and more subtle mouth behavior such as "sticky lips" [Alexander et al. 2009]. Another useful direction would be to retarget performances from one subject to another. Given a set of static scans for both subjects, it should be possible to clone one subject's performance to the second subject as in [Seol et al. 2012]; providing more meaningful control over this transfer remains a subject for future research. Finally, as our framework is agnostic to the particular method employed for estimating 2D correspondences, we would like to try more recent optical flow algorithms such as the top performers on the Middlebury benchmark [Baker et al. 2011]. Usefully, the quality of our performance tracking can be improved any time that an improved optical flow library becomes available.

Acknowledgements

The authors thank the following people for their support and assistance: Ari Shapiro, Sin-Hwa Kang, Matt Trimmer, Koki Nagano, Xueming Yu, Jay Busch, Paul Graham, Kathleen Haase, Bill Swartout, Randall Hill and Randolph Hall. We thank the authors of [Beeler et al. 2010] and [Klaudiny et al. 2010] for graciously

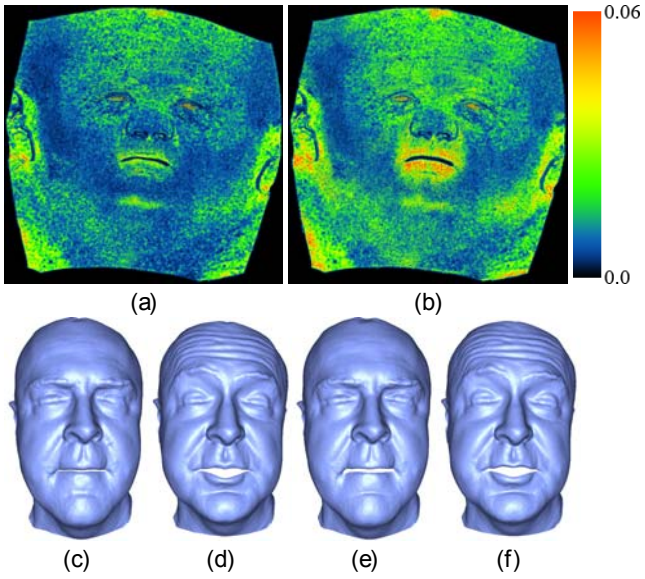


Figure 15: Top row: Temporal variance of contrast-normalized texture (false color, where blue is lowest and red is highest), with (a) the proposed method and (b) the method of [Beeler et al. 2011]. The variance of the proposed method is substantially lower, indicating a more consistent texture alignment throughout the sequence. Bottom row: Geometry for frames 120 and 330 of the sequence, with (c, d) the proposed method and (e, f) the prior work.

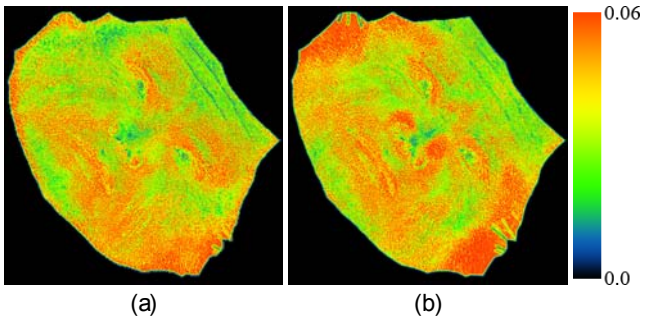


Figure 16: Temporal variance of contrast-normalized texture (false color, where blue is lowest and red is highest), with (a) the proposed method and (b) the method of [Klaudiny et al. 2010]. As in Fig.15, the variance of the proposed method is generally lower.

providing the data for the comparisons in Figs. 15 and 16, respectively. We thank Jorge Jimenez, Etienne Danvoye, and Javier von der Pahlen at Activision R&D for the renderings in Fig. 18. This work was sponsored by the University of Southern California Office of the Provost and the U.S. Army Research, Development, and Engineering Command (RDECOM). The content of the information does not necessarily reflect the position or the policy of the US Government, and no official endorsement should be inferred.

References

ALEXANDER, O., ROGERS, M., LAMBETH, W., CHIANG, M., AND DEBEVEC, P. 2009. Creating a photoreal digital actor: The digital emily project. In Visual Media Production, 2009. CVMP '09. Conference for, 176–187.

BAKER, S., SCHARSTEIN, D., LEWIS, J. P., ROTH, S., BLACK, M. J., AND SZELISKI, R. 2011. A database and evaluation methodology for optical flow. International Journal of Computer Vision 92, 1 (Mar.), 1–31.

BEELER, T., BICKEL, B., BEARDSLEY, P., SUMNER, B., AND GROSS, M. 2010. High-quality single-shot capture of facial geometry. ACM Trans. on Graphics (Proc. SIGGRAPH) 29, 3, 40:1–40:9.

BEELER, T., HAHN, F., BRADLEY, D., BICKEL, B., BEARDSLEY, P., GOTSMAN, C., SUMNER, R. W., AND GROSS, M. 2011. High-quality passive facial performance capture using anchor frames. In ACM SIGGRAPH 2011 papers, ACM, New York, NY, USA, SIGGRAPH '11, 75:1–75:10.

BICKEL, B., LANG, M., BOTSCH, M., OTADUY, M. A., AND GROSS, M. 2008. Pose-space animation and transfer of facial details. In Proceedings of the 2008 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, SCA '08, 57–66.

BORSHUKOV, G., PIPONI, D., LARSEN, O., LEWIS, J. P., AND TEMPELAAR-LIETZ, C. 2003. Universal capture: image-based facial animation for "the matrix reloaded". In SIGGRAPH, ACM, A. P. Rockwood, Ed.

BRADLEY, D., HEIDRICH, W., POPA, T., AND SHEFFER, A. 2010. High resolution passive facial performance capture. In ACM SIGGRAPH 2010 papers, ACM, New York, NY, USA, SIGGRAPH '10, 41:1–41:10.

COOTES, T. F., EDWARDS, G. J., AND TAYLOR, C. J. 1998. Active appearance models. In IEEE Transactions on Pattern Analysis and Machine Intelligence, Springer, 484–498.

DEBEVEC, P. 1998. Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques, ACM, New York, NY, USA, SIGGRAPH '98, 189–198.

DECARLO, D., AND METAXAS, D. 1996. The integration of optical flow and deformable models with applications to human face shape and motion estimation. In Proceedings of the 1996 Conference on Computer Vision and Pattern Recognition (CVPR '96), IEEE Computer Society, Washington, DC, USA, CVPR '96, 231–238.

EKMAN, P., AND FRIESEN, W. 1978. Facial Action Coding System: A Technique for the Measurement of Facial Movement. Consulting Psychologists Press, Palo Alto.

GHOSH, A., FYFFE, G., TUNWATTANAPONG, B., BUSCH, J., YU, X., AND DEBEVEC, P. 2011. Multiview face capture using

Sequence	Frames	Graph*	Flow*	Key	Tween*
Fig. 7	170	0.5 hr	8.0 hr	5.2 hr	1.2 hr
Fig. 8(b)	400	1.1 hr	24 hr	4.3 hr	4.3 hr
Fig. 8(c)	400	1.1 hr	24 hr	24 hr	26 hr
Fig. 9	170	0.5 hr	2.0 hr	3.6 hr	0.9 hr
Fig. 15	347	0.1 hr	15 hr	16 hr	17 hr
Fig. 16	300	0.2 hr	12 hr	3.0 hr	3.0 hr
Fig. 19 row 2	600	1.6 hr	36 hr	6.5 hr	7.0 hr
Fig. 19 row 4	305	0.8 hr	18 hr	3.3 hr	3.5 hr
Fig. 19 row 6	250	0.7 hr	15 hr	2.6 hr	2.8 hr

Figure 17: Timings for complete processing of the sequences used in various figues, using a single workstation. A * indicates an operation that could trivially be run in parallel across many machines.

Figure 18: Real-time renderings of tracked performances using advanced skin and eye shading [Jimenez et al. 2012].

polarized spherical gradient illumination. In Proceedings of the 2011 SIGGRAPH Asia Conference, ACM, New York, NY, USA, SA '11, 129:1–129:10.

GUENTER, B., GRIMM, C., WOOD, D., MALVAR, H., AND PIGHIN, F. 1998. Making faces. In Proceedings of the 25th annual conference on Computer graphics and interactive techniques, ACM, New York, NY, USA, SIGGRAPH '98, 55–66.

HAWKINS, T., WENGER, A., TCHOU, C., GARDNER, A., GÖRANSSON, F., AND DEBEVEC, P. 2004. Animatable facial reflectance fields. In Rendering Techniques 2004: 15th Eurographics Workshop on Rendering, 309–320.

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

JIMENEZ, J., JARABO, A., GUTIERREZ, D., DANVOYE, E., AND VON DER PAHLEN, J. 2012. Separable subsurface scattering and photorealistic eyes rendering. In ACM SIGGRAPH 2012 Courses, ACM, New York, NY, USA, SIGGRAPH 2012.

KLAUDINY, M., AND HILTON, A. 2012. High-detail 3d capture and non-sequential alignment of facial performance. In 3DIMPVT.

KLAUDINY, M., HILTON, A., AND EDGE, J. 2010. High-detail 3d capture of facial performance. In 3DPVT.

KOLMOGOROV, V. 2006. Convergent tree-reweighted message passing for energy minimization. IEEE Trans. Pattern Anal. Mach. Intell. 28, 10, 1568–1583.

LI, H., ROIVAINEN, P., AND FORCHEIMER, R. 1993. 3-d motion estimation in model-based facial image coding. IEEE Trans. Pattern Anal. Mach. Intell. 15, 6 (June), 545–555.

MA, W.-C., JONES, A., CHIANG, J.-Y., HAWKINS, T., FREDERIKSEN, S., PEERS, P., VUKOVIC, M., OUHYOUNG, M., AND DEBEVEC, P. 2008. Facial performance synthesis using deformation-driven polynomial displacement maps. ACM Trans. Graph. 27, 5 (Dec.), 121:1–121:10.

NISHINO, K., AND NAYAR, S. K. 2004. Eyes for relighting. ACM Trans. Graph. 23, 3, 704–711.

PARK, M., KASHYAP, S., COLLINS, R., AND LIU, Y. 2010. Data driven mean-shift belief propagation for non-gaussian mrfs. In Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, 3547 –3554.

POPA, T., SOUTH-DICKINSON, I., BRADLEY, D., SHEFFER, A., AND HEIDRICH, W. 2010. Globally consistent space-time reconstruction. Computer Graphics Forum (Proc. SGP).

SEOL, Y., LEWIS, J., SEO, J., CHOI, B., ANJYO, K., AND NOH, J. 2012. Spacetime expression cloning for blendshapes. ACM Trans. Graph. 31, 2 (Apr.), 14:1–14:12.

VALGAERTS, L., WU, C., BRUHN, A., SEIDEL, H.-P., AND THEOBALT, C. 2012. Lightweight binocular facial performance capture under uncontrolled lighting. ACM Trans. Graph. 31, 6 (Nov.), 187:1–187:11.

WEISE, T., BOUAZIZ, S., LI, H., AND PAULY, M. 2011. Realtime performance-based facial animation. In ACM SIGGRAPH 2011 papers, ACM, New York, NY, USA, SIGGRAPH '11, 77:1–77:10.

WERLBERGER, M. 2012. Convex Approaches for High Performance Video Processing. PhD thesis, Institute for Computer Graphics and Vision, Graz University of Technology, Graz, Austria.

ZHANG, L., SNAVELY, N., CURLESS, B., AND SEITZ, S. M. 2004. Spacetime faces: high resolution capture for modeling and animation. In SIGGRAPH '04: ACM SIGGRAPH 2004 Papers, ACM, New York, NY, USA, 548–558.

ZHU, X., AND RAMANAN, D. 2012. Face detection, pose estimation, and landmark localization in the wild. In CVPR, 2879–2886.