3D Shape Regression for Real-time Facial Animation
Chen Cao, Yanlin Weng, Stephen Lin, Kun Zhou

FaceWarehouse: a 3D facial Expression Database for Visual Computing
Chen Cao, Yanlin Weng, Shun Zhou, Yiyong Tong, Kun Zhou

Zhejiang University    MSRA

Presented by Shu Liang
(Black-on-white slides are Shu’s)
Facial Animation

• Facial animation is widely used in films & games
• Performance-based facial animation

Avatar 2009 © 21st Century Fox
L.A. Noire 2011 © Team Bondi
Related Work

• Performance-based Facial Animation

Quality

Special Equipment

Facial Markers

Camera arrays

Structured light

Device complexity

[Huang et al. 2011]  [Beeler et al. 2011]  [Weise et al. 2009]
Related Work

• Performance-based Facial Animation

Quality

Device complexity

Single Camera

Optical Flow

ASM & AAM

Regression-based alignment

[Vlasic et al. 2005]

[Cootes et al. 1992-2001]

[Cao et al. 2012]
Related Work

• Performance-based Facial Animation

![Graph showing quality vs. device complexity]

- Consumer RGBD Camera
  - Bouaziz et al. 2013
  - Li et al. 2013
  - Weise et al. 2011
Our Goal

- Real-time facial animation for average users
Our Goal

• **Real-time** facial animation for **ordinary users**
  – Single web camera
  – Robust
    • Fast motions
    • Large rotations
    • Exaggerated expressions
  – General environments
    • Indoors and **outdoors**
  – High performance
    • Mobile devices
Our Pipeline

Preprocess
- Captured images with labeled 2D landmarks
- 3D Facial Expression Database
- Training Data Construction
- Training Images and Shapes
- 3D Shape Regression Training
- User-specific 3D Shape Regressor
- User-specific Blendshapes

Runtime
- Video Stream
- User-specific 3D Shape Regressor
- 3D Shape Regression
- 3D Facial Shape
- Tracking
- Tracked Mesh
- Animation
- Digital Avatar
Our Pipeline

• One-time Preprocess

- Captured images with labeled 2D landmarks
- Training Data Construction
- 3D Facial Expression Database
- Training Images and Shapes
- User-specific Blendshapes
- 3D Shape Regression Training
- User-specific 3D Shape Regressor
Our Pipeline

- Runtime computation

Video Stream → 3D Shape Regression → 3D Facial Shape → User-specific Blendshapes → Tracking → Tracked Mesh → Animation → Digital Avatar
3D Face Shape Regression: Preprocess

• Data Collection
  – Image capturing & labeling
  – Blendshapes generation
  – Shape reconstruction
  – Training data generation

• Training
Our Pipeline

Preprocess
- Captured images with labeled 2D landmarks
  - 3D Facial Expression Database
  - Training Data Construction
  - Training Images and Shapes
  - 3D Shape Regression Training
  - User-specific 3D Shape Regressor

Runtime
- Video Stream
  - User-specific 3D Shape Regressor
    - 3D Shape Regression
    - 3D Facial Shape
    - Tracking
    - Tracked Mesh
    - Animation
    - Digital Avatar
Preprocess: Image Capturing & Labeling

Captured Images

Labeled 2D Feature Points
Preprocess: Blendshapes Generation

FaceWarehouse
[Cao et al. 2013]
150 identities × 47 expressions

Labeled 2D Feature Points

Fitting

User-specific Blendshapes
Our Pipeline

Preprocess

Captured images with labeled 2D landmarks → Training Data Construction → 3D Facial Expression Database → Training Images and Shapes → 3D Shape Regression Training → User-specific 3D Shape Regressor

User-specific Blendshapes

Runtime

Video Stream → 3D Shape Regression → 3D Facial Shape → Tracking → Tracked Mesh → Animation → Digital Avatar

FaceWarehouse
FaceWarehouse

- RGBD images of 150 individuals captured by Kinect
- Aged 7-80 from various ethnic backgrounds
- Different expressions, one neutral and 19 other expressions.
Mesh deformation

Neutral expression

Other expressions

$S_0, S_1, S_2 \ldots S_{19}$ for 20 expressions.

FaceWarehouse

[Blanz et. al 2004]

[Huang et. al 2006]

[Sumner et. al 2006]

[Huang et. al 2006]
FaceWarehouse

- Individual-specific expression blendshapes
  - Example-based facial rigging algorithm:

  An expression \( H \) of the person can be:

  \[
  H = B_0 + \sum_{i=1}^{46} \alpha_i (B_i - B_0)
  \]

  \{B_1,B_2,...,B_{46}\} 46 FACS blendshapes

  - Begins with a generic blendshape model \( A = \{A_0,A_1,...,A_{46}\} \)
  - Optimized by minimizing the difference between \( S_j \) and linear combination of \( B_i \) with known weight for expression \( j \), the difference between the deformation from \( B_0 \) to \( B_i \) and that from \( A_0 \) to \( A_i \).
Bilinear face model
A rank-three data tensor $T$. 
(11K vertices \times 150 identities \times 47 expressions)
Used N-mode SVD to decompose the tensor.

$$V = C_r \times_2 w_{id}^T \times_3 w_{exp}^T,$$
Our Pipeline

Preprocess
- Captured images with labeled 2D landmarks
- 3D Facial Expression Database
- Training Data Construction
- Training Images and Shapes
- 3D Shape Regression Training
- User-specific 3D Shape Regressor
- User-specific Blendshapes

Runtime
- Video Stream
- User-specific 3D Shape Regressor
- 3D Shape Regression
- 3D Facial Shape
- User-specific Blendshapes
- Tracking
- Tracked Mesh
- Animation
- Digital Avatar
Preprocess: 3D Shape Reconstruction
Preprocess: Training Data Generation

- Data Augmentation

\[ I_i, S_{io} \]

Translations
\[ \{M_{ja}, 1 \leq j \leq m \} \]

3D Shape Space

\[ (I_i, M_{1a}, S_{i1}) \]

\[ (I_i, M_{2a}, S_{i2}) \]

... 

\[ (I_i, M_{m}, S_{im}) \]
Preprocess: Training Data Generation

- Training Set Construction

3D Shape Space

\[(I_i, M_{ja}, S_{ij})\]

Find \( G \cdot H \) guessed shapes

Training Data

\[(I_i, M_{ja}, S_{ij}, S_{ij1})\]
\[(I_i, M_{ja}, S_{ij}, S_{ij2})\]
\[\ldots\]
\[(I_i, M_{ja}, S_{ij}, S_{ijH})\]
Our Pipeline

Preprocess
- Captured images with labeled 2D landmarks
- 3D Facial Expression Database
- Training Data Construction
- Training Images and Shapes
- User-specific Blendshapes
- 3D Shape Regression Training
- User-specific 3D Shape Regressor

Runtime
- Video Stream
- User-specific 3D Shape Regressor
- 3D Shape Regression
- 3D Facial Shape
- Tracking
- Tracked Mesh
- Animation
- Digital Avatar
Preprocess: Training

- Appearance vector
- Primitive regressor: fern

Summary: two-level boosted regressor
Preprocess: Training

- Appearance vector
Preprocess: Training

- Primitive regressor

\[ \delta S_b = \frac{1}{1 + \beta/|\Omega_b|} \sum_{i \in \Omega_b} (S_i - S_{ic}) \]

\[ S_{ic} = S_{ic} + \delta S_{b,i} \in \Omega_b \]
Preprocess: Training

- Summary: two-level boosted regression

Level One

- Calculate Appearance Vector

Level Two

- Fern
  - All training data
  - Randomly select feature vectors
  - Clustering according to the feature vectors

• Summary: two-level boosted regression
Our Pipeline

Preprocess
- Captured images with labeled 2D landmarks
- 3D Facial Expression Database
- Training Data Construction
- Training Images and Shapes
- 3D Shape Regression Training
- User-specific 3D Shape Regressor

Runtime
- Video Stream
- User-specific 3D Shape Regressor
- 3D Shape Regression
- 3D Facial Shape
- Tracking
- Tracked Mesh
- Animation
- Digital Avatar
3D Face Shape Regression: Runtime

• Initialization: first frame
• Following frames
  – Find guessed shapes
  – Two-level boosted regression
Runtime: Initialization

- First frame

Face Detection
[Viola and Jones 2001]

2D Feature Alignment
[Cao et al. 2012]

3D Shape Recovery
Runtime: Following Frames

- Find guessed shape

3D Shape Space

Current Frame

Previous Frame

Transformed Shape

Find similar shapes

Guessed Shape
Runtime: Following Frames

- Two-level boosted regression
Tracking & Animation

Similar to [Weise et al. 2011]

User-specific Blendshapes

Tracking

Rigid
Non-Rigid

Tracked Mesh

Rigid Transformation
Blendshape Weight

Digital Avatar

Matched Shape
Evaluation: **Regressed shape vs. Kinect 3D vs. 2D vs. Optical Flow**

**Figure 8:** Comparison of depth from 3D shape regression and ground truth from Kinect.

<table>
<thead>
<tr>
<th>RMSE</th>
<th>&lt; 3 pixels</th>
<th>&lt; 4.5 pixels</th>
<th>&lt; 6 pixels</th>
</tr>
</thead>
<tbody>
<tr>
<td>3D Regression</td>
<td>73.3%</td>
<td>80.8%</td>
<td>100%</td>
</tr>
<tr>
<td>2D Regression</td>
<td>50.8%</td>
<td>64.2%</td>
<td>72.5%</td>
</tr>
<tr>
<td>Optical Flow</td>
<td>20.8%</td>
<td>24.2%</td>
<td>41.7%</td>
</tr>
</tbody>
</table>

**Table 1:** Percentages of frames with RMSE less than given thresholds for the tested video sequence.
Live Demo

• Demo
Evaluation: Regressed shape vs. Kinect
Evaluation: 3D vs. 2D vs. Optical Flow

Our 3D Regression    2D Regression    Optical Flow Based
More Results: Outdoor
Our System on Mobile Device
Timings

• Preprocess: 45 mins
  – Capture: 10 mins
  – User interaction: 25 mins
  – Training: 10 mins

• Runtime: less than 15 ms
  – Regression: 5 ms
  – Tracking & Animation: 8 ms
Limitations

• Much **training data**
  – **60** head poses and facial expressions
• **Dramatic lighting changes**
Summary

• 3D facial performance capture from **2D video**
  – **Regression-based** approach
  – **Robust**: fast motions, large rotations, exaggerated expressions
  – General environments: indoors and **outdoors**
  – High performance: **real-time**

• Future work
  – Handle lighting variations
  – Reduce training data
Acknowledgement

• Face capture: Marion Blatt, Steffen Toborg
• Anonymous reviewers
• Funding
  – NSFC (No.61003145 and No.61272305)
  – 973 program of China (No.2009CV320801)
• FaceWarehouse Data: http://gaps-zju.org/facewarehouse/

Thank you!
Preprocess: Camera Calibration

Blendshape Generations:

\[ E = \prod_{i=1}^{n} \prod_{k=1}^{75} \left\| \Pi Q \left( M_i \left( C_r \times 2 w_{id}^T \times 3 w_{exp,i}^T \right)^{(v_k)} - u_i^{(k)} \right) \right\|^2 \]

\[ \mathbf{Q} = \begin{pmatrix} f & 0 & u_0 \\ 0 & f & v_0 \\ 0 & 0 & 1 \end{pmatrix} \]
Why not directly use previous shape?

• Error accumulation
Why not directly regress parameters?

• Expression coefficients in [0:1]
• Animation prior
  – Temporal coherence
• Rigid transformation & expression coefficients
  – Different spaces