Human Reconstruction

Jaesik Park
jaesik.park@postech.ac.kr
Assistant Professor
Computer Science & Engineering
This course

• 3D reconstruction from single image?
  • Possible for some examples
  • Known class, sufficiently large amount of training set

• Understanding key ideas for
  • Face reconstruction
  • Human reconstruction

• Thinking about what is the limit of the state-of-the-arts
This Course

Color image → Parametric model space → shape or appearance reconstruction
MoFA: Model-based Deep Convolutional Face Autoencoder for Unsupervised Monocular Reconstruction

Ayush Tewari\textsuperscript{1}  Michael Zollhöfer\textsuperscript{1}  Hyeongwoo Kim\textsuperscript{1}  Pablo Garrido\textsuperscript{1}  Florian Bernard\textsuperscript{1,2}  Patrick Pérez\textsuperscript{3}  Christian Theobalt\textsuperscript{1}

\textsuperscript{1}Max-Planck-Institute for Informatics  \textsuperscript{2}LCSB, University of Luxembourg  \textsuperscript{3}Technicolor
Goal

Ill-posed problem
Related Works

Optimization-based methods:
- High quality reconstructions
- Costly to optimize
- Non-convex optimization

[Thies et al. 2016]

[Garrido et al. 2016]

[Romdhani and Vetter 2016]
Related Works

Learning-based methods:
- Faster
- Lack of training data

[Richardson et al. 2016, 2017]

[Tran et al. 2017]

[Dou et al. 2017]
Model-based Face Autoencoder

- Integrate both in a single framework
- Train unsupervised on real images
Model-based Face Autoencoder

• Integrate both in a single framework
• Train unsupervised on real images
Model-based Face Autoencoder

- Integrate both in a single framework
- Train unsupervised on real images

- Define the reconstruction
- Semantic code vector
Model-based Face Autoencoder

- Integrate both in a single framework
- Train unsupervised on real images

- Known image formation model
- Fixed, does not require any training
Model-based Face Autoencoder

- Integrate both in a single framework
- Train unsupervised on real images
Model-based Face Autoencoder

Input

Deep Convolutional Encoder

Model Parameters

Image Formation Layer (expert-designed Decoder)

Output

$E_{loss}$

$E_{loss}$

Figure 1: VDD-Face CNN Architecture (Parkhi et al. [1]).
Model-based Face Autoencoder

\[ \text{Input} \rightarrow \text{Deep Convolutional Encoder} \rightarrow \text{Model Parameters} \rightarrow \text{Image Formation Layer (expert-designed Decoder)} \rightarrow \text{Output} \]

\[ E_{\text{loss}} \]
Parametric Face Model – Forward Pass

\[ P = \begin{pmatrix} \Phi \end{pmatrix} M \]

\[ |P| = 6 \]
Parametric Face Model – Forward Pass

\[ P = \begin{pmatrix} \Phi \\ \alpha \end{pmatrix} \]

\[ |P| = 6 + 80 \]

[Blanz et al. 1999]
Parametric Face Model – Forward Pass

\[ P = \begin{pmatrix} \Phi \\ \alpha \\ \beta \end{pmatrix} \]

\[ |P| = 6 + 80 + 80 \]

[Blanz et al. 1999]
Parametric Face Model – Forward Pass

\[ P = \begin{pmatrix} \Phi \\ \alpha \\ \beta \\ \delta \end{pmatrix} \]

\[ |P| = 6 + 80 + 80 + 64 \]

[Alexander et al. 2009]
Parametric Face Model – Forward Pass

\[ \begin{pmatrix} \Phi \\ \alpha \\ \beta \\ \delta \\ \gamma \end{pmatrix} = M \]

\[ |P| = 6 + 80 + 80 + 64 + 27 = 257 \]
Model-based Face Autoencoder

Model-based Face Autoencoder

Deep Convolutional Encoder

Model Parameters

Image Formation Layer (expert-designed Decoder)

Output

Perspective Transformation

3D Model

Synthetic Image
Model-based Face Autoencoder (Forward Pass)

Input → Deep Convolutional Encoder → Model Parameters → Image Formation Layer (expert-designed Decoder) → Output

$\mathbf{E}_{loss}$
Model-based Face Autoencoder (Backward Pass)

- **Input**
- **Deep Convolutional Encoder**
- **Model Parameters**
- **Image Formation Layer (expert-designed Decoder)**
- **Output**
- $E_{\text{loss}}$

- Differentiable decoder
Model-based Face Autoencoder

Input

Deep Convolutional Encoder

Model Parameters

Image Formation Layer (expert-designed Decoder)

Output

$E_{loss}$
Loss Function

\[ E(P) = E_{dense}(P) + E_{reg}(P) \]
Training Data

- CelebA
- FaceWarehouse
- 300VW
- LFW
Results

Input
Results

Input  Reconstruction
Results

Input → Reconstruction → Geometry
Results

Input  Reconstruction  Geometry  Reflectance
Results
Results (comparison to learning-based method)
Results (comparison to optimization-based method)

- Input
- Ours
- Garrido16 (w/ landmarks)
- Garrido16 (w/o landmarks)

~4 ms (250 fps)  
~1 minute (0.02 fps)
Results (comparison to optimization-based method)

<table>
<thead>
<tr>
<th>Input</th>
<th>Ours</th>
<th>Garrido16 (w/ landmarks)</th>
<th>Garrido16 (w/o landmarks)</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Input Image" /></td>
<td><img src="image2" alt="Ours Image" /></td>
<td><img src="image3" alt="Garrido16 (w/ landmarks) Image" /></td>
<td><img src="image4" alt="Garrido16 (w/o landmarks) Image" /></td>
</tr>
</tbody>
</table>

- **Ours**:
  - ~4 ms (25 0 fps)
- **Garrido16 (w/ landmarks)**:
  - ~1 minute (0.02 fps)
Take-home Message

• Unsupervised training for 3D face reconstruction

• Deep integration of computer graphics model in the network

• Applicable to other reconstruction problems
Input
Deep Video Portraits

Hyeongwoo Kim¹  Pablo Garrido²  Ayush Tewari¹  Weipeng Xu¹
Justus Thies³  Matthias Nießner³  Patrick Pérez²
Christian Richardt⁴  Michael Zollhöfer⁵  Christian Theobalt¹

¹ MPI Informatics  ² Technicolor  ³ TU Munich  ⁴ University of Bath  ⁵ Stanford University
CGI Face Editing

Professional video
CGI Face Editing

Personal video

Video: © https://www.youtube.com/watch?v=7Flvk91quLY
Contribution

• Editing of head pose, rotation, face expression and eye gaze
• Combination of model-based face capture and CNN

Video: courtesy of UK government (Open Government Licence)
Related Work

Model-based face capture and reenactment

Garrido et al., ToG 2016
Kemelmacher-Shlizerman et al., ECCV 2010
Shi et al., ToG 2014
Suwajanakorn et al., ICCV 2015
Thies et al., CVPR 2016
Averbuch-Elor et al., ToG 2017
Thies et al., SIGGRAPH 2018

CNN-based methods

Karras et al., ICLR 2018
Goodfellow et al., NIPS 2014
Isola et al., CVPR 2017
Chen and Koltun, ICCV 2017
Tewari et al., ICCV 2017
Olszewski et al., ICCV 2018
Wang et al., CVPR 2018
Overview

Monocular face reconstruction

Rendering-to-video translation network

Training video
Overview

Face reenactment

User interaction

Modified face parameters

Pose
Expression
Eyes
Identity...

Modified rendering

Rendering-to-video translation network
Result: Facial Reenactment

Full reenactment of head pose, head rotation, face expression and eye gaze

Source  
Result  
Face2Face  
(Thies et al., 2016)
Summary

Future work:
  • Pushing toward higher quality and resolution
  • Video authentication and forensics
Take-home Message

• Human Face Reconstruction
  • Model based approach vs. direct generative models
    • That has essential parameters to manipulate face models
    • Simple auto-encoder to estimate parameters
  • Approaches
    • Energy minimization to update parameters
    • End-to-end approach to generate realistic image
  • Limitations?
    • Face image in the wild – view point, self occlusions, expressions
    • Easily confused by occlusions

• Visual Turing test
  • A new benchmark dataset to classify real or fake video.
  • FaceForensics++: Learning to Detect Manipulated Facial Images, ICCV 2019
Real-time Human Pose Estimation using a RGB-D Image
Real-time Human Pose Estimation (Microsoft Kinect)
Real-Time Human Pose Recognition in Parts from Single Depth Images

CVPR 2011

Jamie Shotton, Andrew Fitzgibbon, Mat Cook, Toby Sharp, Mark Finocchio, Richard Moore, Alex Kipman, Andrew Blake
Real-time Human Pose Recognition

- From single depth image to body joint
- intermediate body part representation
- Regard per-pixel classification problem
- No temporal information
- 200 frames per second
- CVPR 2011 Best Paper Award

Shotton et al., Real-Time Human Pose Recognition in Parts from Single Depth Images, CVPR 2011
Depth Comparison Features

\[ f_{\theta}(I, \mathbf{x}) = d_I \left( \mathbf{x} + \frac{\mathbf{u}}{d_I(\mathbf{x})} \right) - d_I \left( \mathbf{x} + \frac{\mathbf{v}}{d_I(\mathbf{x})} \right) \]

Offset = (u, v)

(a) a large depth difference response
(b) smaller response

Shotton et al., Real-Time Human Pose Recognition in Parts from Single Depth Images, CVPR 2011
Decision Forests

- Each tree contains offsets and their decision rules
- The leaf node votes weak evidence of the shape
- The final histogram indicates the final part

Training: Labeled body parts obtained from synthetic dataset + motion capture dataset

Shotton et al., Real-Time Human Pose Recognition in Parts from Single Depth Images, CVPR 2011
Human Pose Estimation using a RGB Image
Learning 3D Human Dynamics from Video
Angjoo Kanazawa et al., CVPR 2019
Basic Body Shape Model
SMPL: Skinned Multi-Person Linear model

Matthew Loper  Naureen Mahmood  Javier Romero
Gerard Pons-Moll  Michael J. Black
SMPL

Key Idea: Obtain template body shape and controllable parameters for the deformation

Gray: ground truth, Orange: SMPL model

Loper et al., SMPL: A Skinned Multi-Person Linear Model
SMPL Model

Key Idea: Obtain template body shape and controllable parameters for the deformation

(a) $\bm{T}, \mathcal{W}$  
Template mesh with blend weights

(b) $\bm{T} + B_S(\beta), J(\beta)$  
Blend shape contribution

(c) $\bm{T}_P(\beta, \theta) = \bm{T} + B_S(\beta) + B_P(\theta)$  
Additional pose blend shape

(d) $W(\bm{T}_P(\beta, \theta), J(\beta), \theta, \mathcal{W})$  
Deformed vertices
Learning to Find Human Shape Parameters
Motivation

• Why it is called end-to-end approach?
  • From single color image, infer shape parameter, and produce models

• Replace energy minimization approach with
  • Regression module that infers the latent 3D representation
  • Use a discriminator to tell if these parameters come from a real human shape

• Implicit approach to penalize unrealistic human pose

• Avoids complicated optimization
Overview

\[ L_{\text{reproj}} = \|x - \hat{x}\|_2^2 \]

Projection \( \hat{x} \)
Simplified Model

Parameters need to be estimated

\[ \Theta = \{ \theta, \beta, R, t, s \} \]

\[ M(\theta, \beta) \]

\[ R \in \mathbb{R}^{3 \times 3} \quad t \in \mathbb{R}^{2} \quad s \in \mathbb{R} \]
Loss Functions – Joint Reprojection Error

\[ L = \lambda (L_{\text{reproj}} + \mathbb{1}L_{3D}) + L_{\text{adv}} \]

Optional: if 3D marker is given

\[ L_{\text{reproj}} = \sum_i ||v_i(x_i - \hat{x}_i)||_1 \]

Visibility

\[ \hat{x} = s\Pi(RX(\theta, \beta)) + t \]

3-dimensional 6980 vertices

orthographic projection
Loss Functions – Joint Reprojection Error

\[ L_{\text{reproj}} = \sum_i \| v_i (x_i - \hat{x}_i) \|_1 \]

Issue: directly regressing in one go is a challenging task, particularly because includes rotation parameters

**Key idea:** iterative error feedback (IEF)
Regressor takes image feature $\phi$ and current parameter $\Theta_t$ and produces residual $\Delta \Theta_t \Rightarrow$ The task gets easier!

Iteratively update

\[ \Theta_{t+1} = \Theta_t + \Delta \Theta_t \]
Loss Functions – Additional Direct Supervision

\[ L = \lambda (L_{\text{reproj}} + L_{\text{3D}}) + L_{\text{adv}} \]

Optional: if 3D marker is given

Direct regression of human shape.

Note: only penalize 3D position and parameters

\[
L_{\text{3D}} = L_{\text{3D joints}} + L_{\text{3D smpl}}
\]

\[
L_{\text{joints}} = \| (X_i - \hat{X}_i) \|_2^2
\]

\[
L_{\text{smpl}} = \| [\beta_i, \theta_i] - [\hat{\beta}_i, \hat{\theta}_i] \|_2^2
\]

Reference: full SMPL model
• Implausible 3D bodies or bodies with gross self-intersections may still minimize the reprojection loss.
• Introduce discriminator network $D$

$$\min L_{adv}(E) = \sum \mathbb{E}_{\Theta \sim p_E} [(D_i(E(I)) - 1)^2]$$

Encoder including image and 3D module

Discriminator

• and the objective for each discriminator is

$$\min L(D_i) = \mathbb{E}_{\Theta \sim p_{data}} [(D_i(\Theta) - 1)^2] + \mathbb{E}_{\Theta \sim p_E} [D_i(E(I))^2]$$
End-to-end Recovery of Human Shape and Pose

Angjoo Kanazawa, Michael J Black, David W. Jacobs, Jitendra Malik

Supplementary Materials
Temporal Extension
Motivations

• Instead of learn human shape parameters frame-by-frame, Learn a new temporal representation for human shape from video
• Key Idea: learn a hallucinator that can generate motions from temporal representation.
• Why we need hallucinator?
  • Learning hallucinator introduces a new a loss function.
  • Good hallucinator should influence to learn better temporal representation.
  • A by-product to predict motion from single image
  • Transfer the learned 3D dynamics knowledge to static images
Overview
Losses (Static)

\[ L_{2D} = \left\| v_t(x_t - \hat{x}_t) \right\|_2^2 \]

2D key points visibility

\[ L_{3D} = \left\| \Theta_t - \hat{\Theta}_t \right\|_2^2 \]

Predicted parameter

\[ \Theta = \{ \beta, \theta, \Pi \} \]

Shape Pose Camera Parameter

\[ L_{adv \ prior} = \sum_k (D_k(\Theta) - 1)^2 \]

Discriminator for each joint rotation of the body model

\[ L_t = L_{2D} + L_{3D} + L_{adv \ prior} + L_{\beta \ prior} \]
Losses (Temporal)

Within a window $\Delta t$, a losses are defined as

$$L_{\text{temporal}} = \sum_{t} L_{t} + \sum_{\Delta t} L_{t+\Delta t} + L_{\text{const shape}}$$

The shape in the next frame should be similar to the current frame

$$L_{\text{const shape}} = \sum_{t=1}^{T-1} ||\beta_t - \beta_{t+1}||$$
The hallucinator can be trained in a weakly-supervised manner, minimizing the difference between the hallucinated movie strip and the actual movie strip obtained from $f_{movie}$

$$L_{\text{hal}} = \|\Phi_t - \tilde{\Phi}_t\|_2$$

Overall loss term is

$$L = \sum_t L_t + \sum_{\Delta t} L_{t+\Delta t} + L_{\text{const shape}} + L_t(\tilde{\Phi}_t) + \sum_{\Delta t} L_{t+\Delta t}(\tilde{\Phi}_t)$$
Take-home Message

• Human Shape Estimation
  • Got benefit from prior model built by human scans
    • Parametric models that are easy to train
  • Approaches
    • Energy minimization to update parameters
    • End-to-end approach that infer shape parameters from image
    • Temporal extension by hallucination

• Limitations?
  • Ambiguous human pose
  • Not tightly bonded with the object boundary
  • Easily confused by occlusions