Face Image Editing

Tianhao Wei   Kangcheng Hou   Shuqi Wang

June 28, 2018

Introduction

- A large proportion of the images people upload is face image.
- We want to develop a software that ease the process of editing facial image.

Applications I

Applications II

Agenda

1. Generate a face
2. Fit a face model
3. Application: Expression Flow
4. Laplacian Surface Editing
5. Demo: Laplacian Surface Editing

Generate a face I

Bilinear face model[**?**]

$$\mathbf{f} = \mathcal{M} \times \mathbf{w}_{\text{id}}^{\top} \times \mathbf{w}_{\text{expr}}^{\top}$$

Generate a face II

Basel face model[**?**]

$$\mathbf{f} = \bar{\mathbf{f}} + \mathcal{M}_{\mathsf{id}} \times \mathbf{w}_{\mathsf{id}}^{\top} + \mathcal{M}_{\mathsf{expr}} \times \mathbf{w}_{\mathsf{expr}}^{\top}$$
$$\mathbf{w}_{\mathsf{id}} \sim \mathcal{N}(0, \mathsf{diag}(\sigma_{\mathsf{id}}^{(1)}, \ldots, \sigma_{\mathsf{id}}^{(N_{\mathsf{id}})})$$
$$\mathbf{w}_{\mathsf{expr}} \sim \mathcal{N}(0, \mathsf{diag}(\sigma_{\mathsf{expr}}^{(1)}, \ldots, \sigma_{\mathsf{expr}}^{(N_{\mathsf{expr}})})$$

Under the assumption of weak perspective camera model, with scale s, rotation $\mathbf{R}$ and translation $\mathbf{t}$, we have

$$\hat{\mathbf{f}} = s(\mathbf{R}\mathbf{f} + \mathbf{t})$$

Fit a face model

Given an image and 2D face landmarks $\{l_1, \ldots, l_{N_{\mathsf{lm}}}\}$ from a off-the-shelf tracker[1]. Solve the following optimization problem:

$$\min_{\mathbf{w}_{\mathsf{id}}, \mathbf{w}_{\mathsf{expr}}, s, \mathbf{R}, \mathbf{t}} \sum_{i=1}^{N_{\mathsf{lm}}} ||l_i - h_i||_2^2$$

where $\{h_1, \ldots, h_{N_{\mathsf{lm}}}\}$ is the corresponding points on 3d model.

[1] https://www.faceplusplus.com.cn/

Coordinate Descent

We alternate the optimization of $s, \mathbf{R}, \mathbf{t}$ and $\mathbf{w}_{\mathsf{id}}, \mathbf{w}_{\mathsf{expr}}$.

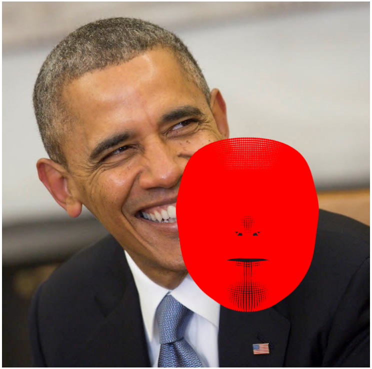
$$\min_{\mathbf{w}_{\mathsf{id}}, \mathbf{w}_{\mathsf{expr}}, s, \mathbf{R}, \mathbf{t}} \sum_{i=1}^{N_{\mathsf{lm}}} ||l_i - s(\mathbf{R}(\mathcal{M} \times [\mathbf{w}_{\mathsf{id}}, \mathbf{w}_{\mathsf{expr}}]^{\top})_i + \mathbf{t})||_2^2$$

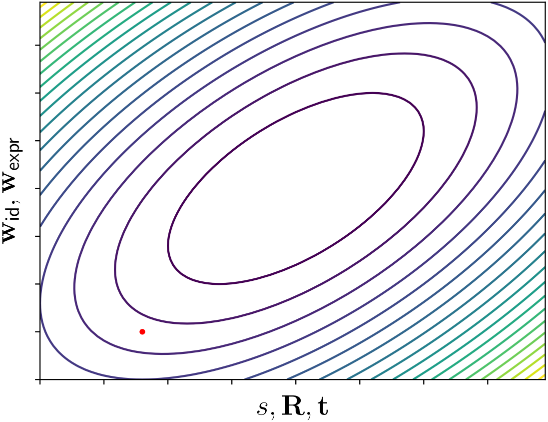
Algorithm: Fit a face model

Algorithm 1: Fit face model to a single image

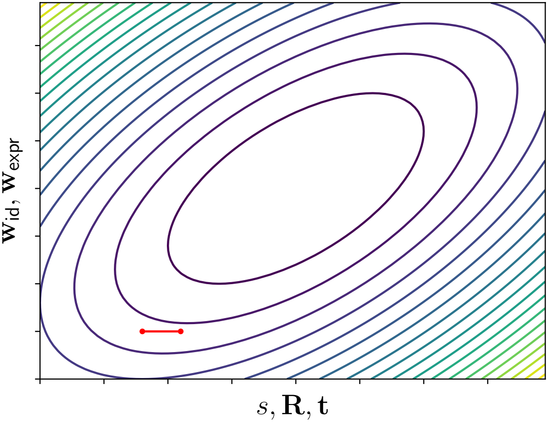
Input: facial landmarks $\{l_1, \ldots, l_{N_{\text{lm}}}\}$ and PCA model
Output: shape coefficients $\mathbf{w}$ and camera parameters $s, \mathbf{R}, \mathbf{t}$
Set $\mathbf{w} = \mathbf{0}$;
repeat
    Set $\mathbf{f} = \bar{\mathbf{f}} + \mathcal{M} \times \mathbf{w}^\top$;
    Find the camera parameters $s, \mathbf{R}, \mathbf{t}$ using $\mathbf{f}$ and $\{l_1, \ldots, l_{N_{\text{lm}}}\}$;
    Project all vertices of $\mathbf{f}$ onto the image plane: $\hat{\mathbf{f}} = s(\mathbf{R}\mathbf{f} + \mathbf{t})$;
    Find the convex hull of $\hat{\mathbf{f}}$ as hull($\hat{\mathbf{f}}$);
    For contour landmarks l_i, find the correspondence;
    Solve $\mathbf{w}$;
until $\mathbf{w}$ *converges*;

Coordinate Descent I

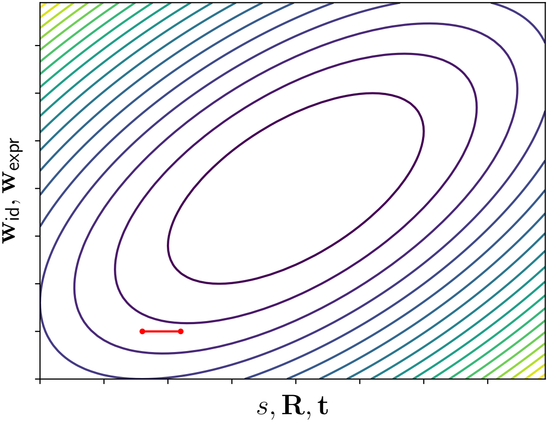




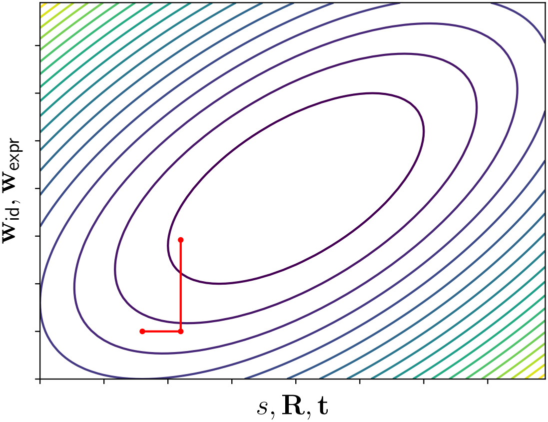
Coordinate Descent II



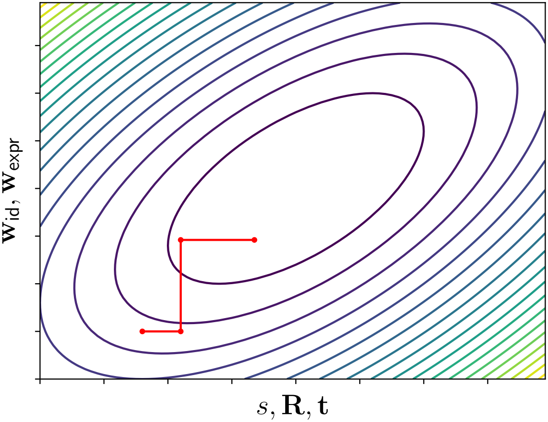
Coordinate Descent III



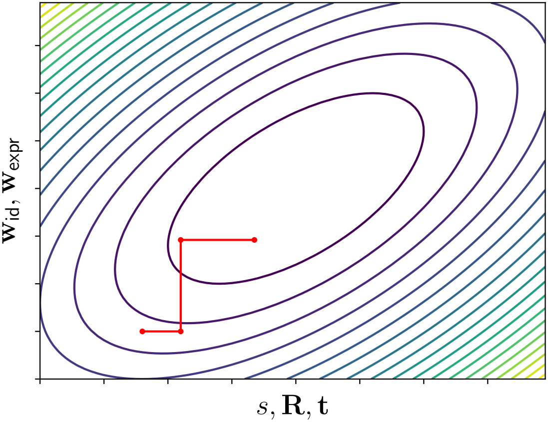
Coordinate Descent IV



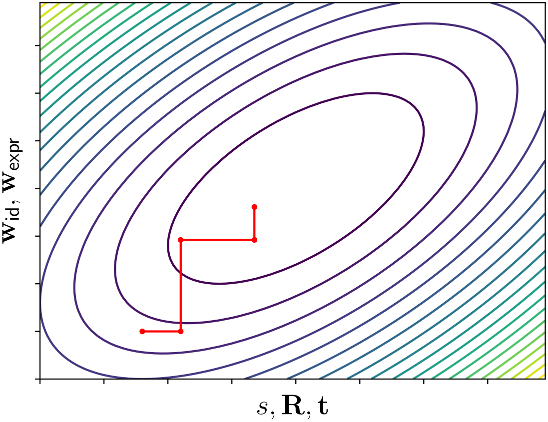
Coordinate Descent V



Coordinate Descent VI



Coordinate Descent VII

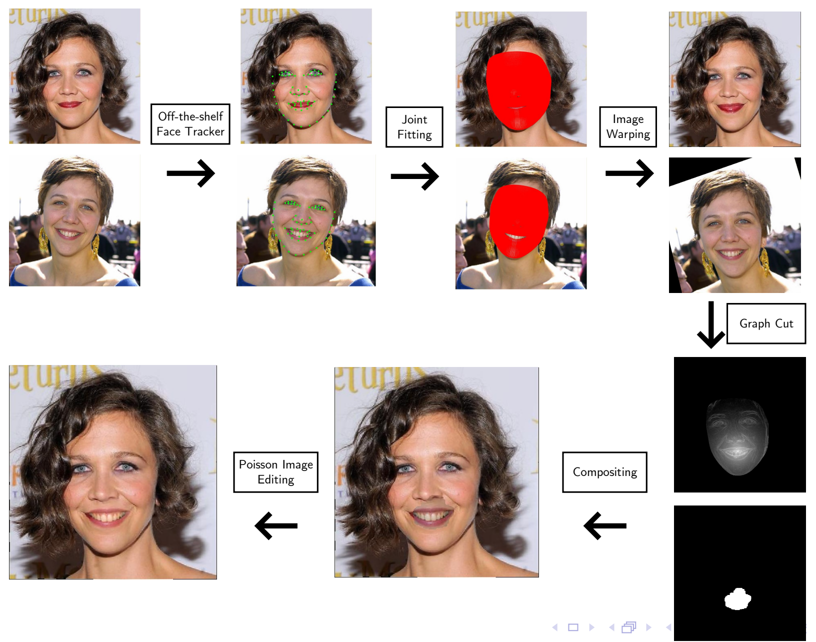


Estimation of Parameters

- Estimation of $\mathbf{w}_{\text{id}}, \mathbf{w}_{\text{expr}}$ is a regularized least square.
- Estimation of $s, \mathbf{R}, \mathbf{t}$ is achieved using POSIT algorithm.
- See more in our report!

Application: Expression Flow for 3D-Aware Face Component Transfer[?]

Expression Flow Pipeline



More Results

Reference

Target

Result

More Results

Reference

Target

Result

Laplacian Surface Editing I

Laplacian operator measures the flatness of the mesh:

$$\Delta f(\mathbf{x}) = \lim_{|B(\mathbf{x})| \to 0} \frac{1}{|B(\mathbf{x}))|} \int_{B(\mathbf{x})} f(\mathbf{z})\, d\mathbf{z} - f(\mathbf{x})$$

Where $B(\mathbf{x})$ is an infinitesimal region around $\mathbf{x}$. We want to the difference of mesh before and after deformation be small.

$$\begin{aligned}\int_{\Omega} ||\Delta(\mathbf{x} - \hat{\mathbf{x}})||^2 \, d\mathbf{A} &\approx \mathrm{tr}(\mathbf{D}^\top \mathbf{L}^\top \mathbf{M}^{-\top} \mathbf{M} \mathbf{M}^{-1} \mathbf{L} \mathbf{D}) \\ &= \mathrm{tr}(\mathbf{D}^\top \underbrace{\mathbf{L}^\top \mathbf{M}^{-1} \mathbf{L}}_{\mathbf{Q}} \mathbf{D})\end{aligned}$$

where $\mathbf{D}, \mathbf{L}, \mathbf{M}$ is the difference of mesh, laplacian and the mass matrix respectively.

$$\min_{\mathbf{D}_\mathsf{u}} \mathrm{tr}\left((\mathbf{D}_\mathsf{u}^\top \ \mathbf{D}_\mathsf{h}^\top) \begin{pmatrix} \mathbf{Q}_{\mathsf{u,u}} & \mathbf{Q}_{\mathsf{u,h}} \\ \mathbf{Q}_{\mathsf{h,u}} & \mathbf{Q}_{\mathsf{h,h}} \end{pmatrix} \begin{pmatrix} \mathbf{D}_\mathsf{u} \\ \mathbf{D}_\mathsf{h} \end{pmatrix} \right)$$

Laplacian Surface Editing II

$$\min_{\mathbf{D}_{\mathsf{u}}} \operatorname{tr} \left(\mathbf{D}_{\mathsf{u}}^{\top} \mathbf{Q}_{\mathsf{u},\mathsf{u}} \mathbf{D}_{\mathsf{u}} + 2\mathbf{D}_{\mathsf{u}}^{\top} \mathbf{Q}_{\mathsf{u},\mathsf{h}} \mathbf{D}_{\mathsf{h}} + \underbrace{\mathbf{D}_{\mathsf{h}}^{\top} \mathbf{Q}_{\mathsf{h},\mathsf{h}} \mathbf{D}_{\mathsf{h}}}_{\text{constant}} \right)$$

$$\min_{\mathbf{D}_{\mathsf{u}}} \operatorname{tr} \left(\mathbf{D}_{\mathsf{u}}^{\top} \mathbf{Q}_{\mathsf{u},\mathsf{u}} \mathbf{D}_{\mathsf{u}} + 2\mathbf{D}_{\mathsf{u}}^{\top} \mathbf{Q}_{\mathsf{u},\mathsf{h}} \mathbf{D}_{\mathsf{h}}) \right)$$

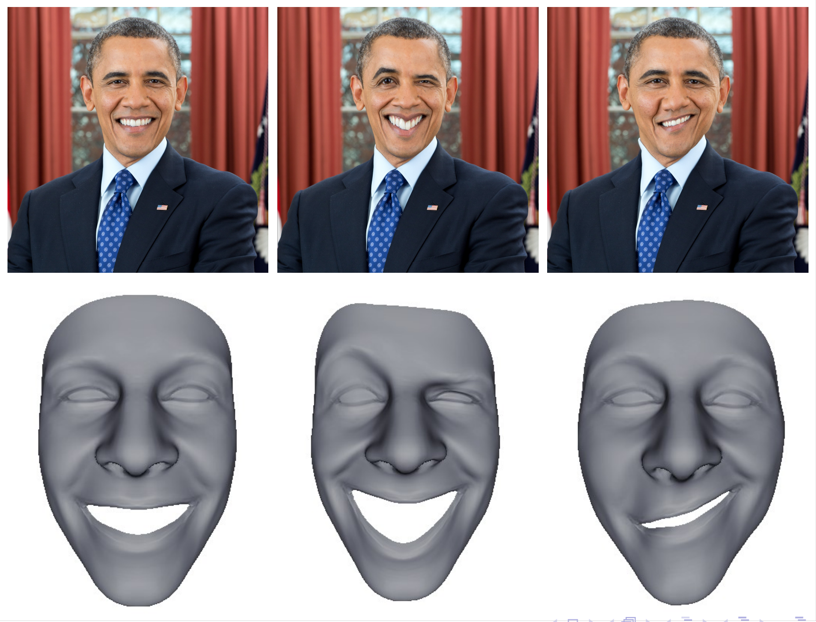
Set the gradient to zero

$$2\mathbf{Q}_{\mathsf{u},\mathsf{u}} \mathbf{D}_{\mathsf{u}} + 2\mathbf{Q}_{\mathsf{u},\mathsf{h}} \mathbf{D}_{\mathsf{h}} = 0 \rightarrow \mathbf{D}_{\mathsf{u}} = \mathbf{Q}_{\mathsf{u},\mathsf{u}}^{-1} \mathbf{Q}_{\mathsf{u},\mathsf{h}} \mathbf{D}_{\mathsf{h}}$$

Minimization w.r.t to the unconstrained points gives us the solution.

Demo:
Laplacian Surface Editing

Results



Summary

What have we implemented?

- Single/Joint Face fitting [**?**]
- Laplacian Surface Editing [**?**]
- Poisson Image Editing [**?**]
- Formulate our problem into graph cut[**?**] and solve it using GCO library [**?**]
- ARAP shape manipulation [**?**]
- **Combine the above all together**

References I

Thanks!