Lecture-11

Synthesizing Realistic Facial Expressions from Photographs

Pighin et al

SIGGRAPH'98

The Artist's Complete Guide to Facial Expression: Gary Faigin

- There is no landscape that we know as well as the human face. The twenty-five-odd square inches containing the features is the most intimately scrutinized piece of territory in existence, examined constantly, and carefully, with far more than an intellectual interest. Every detail of the nose, eyes, and mouth, every regularity in proportion, every variation from one individual to the next, are matters about which we are all authorities.

Main Points

- One view is not enough.
- Fitting of wire frame model to the image is a complex problem (pose estimation)
- Texture mapping is an important problem

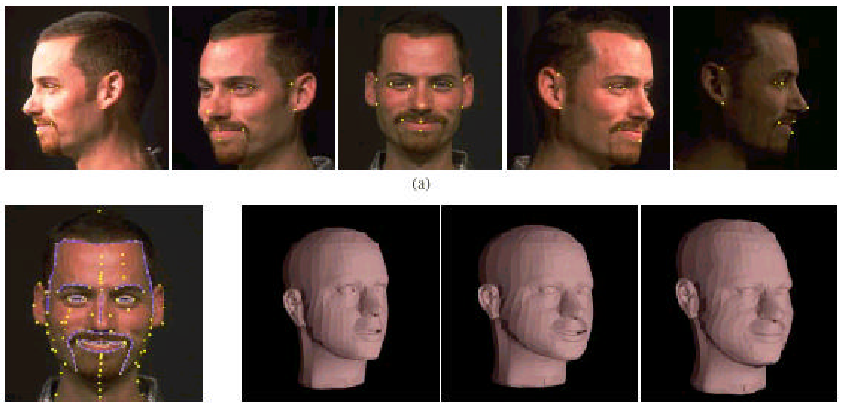
Synthesizing Realistic Facial Expressions

- Select 13 feature points manually in face image corresponding to points in face model created with Alias.
- Estimate camera poses and deformed 3d model points.
- Use these deformed values to deform the remaining points on the mesh using interpolation.

Synthesizing Realistic Facial Expressions

- Introduce more feature points (99) manually, and compute deformations as before by keeping the camera poses fixed.
- Use these deformed values to deform the remaining points on the mesh using interpolation as before.
- Extract texture.
- Create new expressions using morphing.

Synthesizing Realistic Facial Expressions



3D Rigid Transformation

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = R \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + T = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} T_X \\ T_Y \\ T_Z \end{bmatrix}$$

Camera coordinates

Wireframe coordinates

$$x_i'^k = f^k \frac{X_i'^k}{Z_i'^k}, \quad y_i'^k = f^k \frac{Y_i'^k}{Z_i'^k}$$ perspective

K=camera no.

3D Rigid Transformation

$$x_i'^k = f^k \frac{X_i'^k}{Z_i'^k}, \; y_i'^k = f^k \frac{Y_i'^k}{Z_i'^k}$$

$$x_i'^k = f_k \frac{r_{11}^k X_i + r_{12}^k Y_i + r_{13}^k Z_i + T_X^k}{r_{31}^k X_i + r_{32}^k Y_i + r_{33}^k Z_i + T_Z^k}$$

$$y_i'^k = f_k \frac{r_{21}^k X_i + r_{22}^k Y_i + r_{23}^k Z_i + T_Y^k}{r_{31}^k X_i + r_{32}^k Y_i + r_{33}^k Z_i + T_Z^k}$$

Model Fitting

$$x_i'^k = f_k \frac{\mathbf{r}_x^{\mathbf{k}} \mathbf{p}_{\mathbf{i}} + T_X^k}{\mathbf{r}_z^{\mathbf{k}} \mathbf{p}_{\mathbf{i}} + T_Z^k}$$

$$y_i'^k = f_k \frac{\mathbf{r}_y^{\mathbf{k}} \mathbf{p}_{\mathbf{i}} + T_Y^k}{\mathbf{r}_z^{\mathbf{k}} \mathbf{p}_{\mathbf{i}} + T_Z^k}$$

Model Fitting

$$x_i'^k = f_k \frac{\mathbf{r}_x^{\mathbf{k}} \mathbf{p}_{\mathbf{i}} + T_X^k}{\mathbf{r}_z^{\mathbf{k}} \mathbf{p}_{\mathbf{i}} + T_Z^k}$$

$$y_i'^k = f_k \frac{\mathbf{r}_y^{\mathbf{k}} \mathbf{p}_{\mathbf{i}} + T_Y^k}{\mathbf{r}_z^{\mathbf{k}} \mathbf{p}_{\mathbf{i}} + T_Z^k} \qquad h^k = \frac{1}{T_Z^k}, s^k = f^k h^k$$

$$x_i'^k = s_k \frac{\mathbf{r}_x^{\mathbf{k}} \mathbf{p}_{\mathbf{i}} + T_X^k}{1 + h^k \mathbf{r}_z^{\mathbf{k}} \mathbf{p}_{\mathbf{i}}}$$

$$y_i'^k = s_k \frac{\mathbf{r}_y^{\mathbf{k}} \mathbf{p}_{\mathbf{i}} + T_Y^k}{1 + h^k \mathbf{r}_z^{\mathbf{k}} \mathbf{p}_{\mathbf{i}}}$$

Model Fitting

$$x_i'^k = s_k \frac{\mathbf{r}_x^{\mathbf{k}} \mathbf{p}_{\mathbf{i}} + T_X^k}{1 + h^k \mathbf{r}_z^{\mathbf{k}} \mathbf{p}_{\mathbf{i}}}$$

$$y_i'^k = s_k \frac{\mathbf{r}_y^{\mathbf{k}} \mathbf{p}_{\mathbf{i}} + T_Y^k}{1 + h^k \mathbf{r}_z^{\mathbf{k}} \mathbf{p}_{\mathbf{i}}} \qquad w_i^k = (1 + h^k (\mathbf{r}_z^k . \mathbf{p}_i))^{-1}$$

$$w_i^k (x_i'^k + x_i'^k h^k (\mathbf{r}_{\mathbf{z}}^{\mathbf{k}} . \mathbf{p}_{\mathbf{i}}) - s^k (\mathbf{r}_X^{\mathbf{k}} . \mathbf{p}_{\mathbf{i}} + T_X^k)) = 0$$

$$w_i^k (y_i'^k + y_i'^k h^k (\mathbf{r}_{\mathbf{z}}^{\mathbf{k}} . \mathbf{p}_{\mathbf{i}}) - s^k (\mathbf{r}_Y^{\mathbf{k}} . \mathbf{p}_{\mathbf{i}} + T_Y^k)) = 0$$

Model Fitting

- Solve for unknowns in five steps:

$$s^k; \mathbf{p}_i; \mathbf{R}^k; T_X^k, T_Y^k; \boldsymbol{h}^k$$

- Use linear least squares fit.

- When solving for an unknown, assume other parameters are known.

Least Squares Fit

$$a_j.x - b_j = 0$$

$$\sum_j (a_j.x - b_j)^2$$

$$\sum_j (a_j a_j^T) x = \sum_j b_j a_j$$

$$w_i^k (x_i'^k + x_i'^k \boldsymbol{h}^k (\mathbf{r_z^k}.\mathbf{p_i}) - s^k (\mathbf{r_X^k}.\mathbf{p_i} + T_X^k)) = 0$$

$$w_i^k (y_i'^k + y_i'^k \boldsymbol{h}^k (\mathbf{r_z^k}.\mathbf{p_i}) - s^k (\mathbf{r_Y^k}.\mathbf{p_i} + T_Y^k)) = 0$$

Update for p

$$a_{2k+0} = w_i^k (x_i^k \boldsymbol{h}^k r_z^k - s^k r_x^k) \qquad \mathrm{b}_{2k+0} = w_i^k (s^k T_x^k - x_i^k)$$

$$a_{2k+1} = w_i^k (y_i^k \boldsymbol{h}^k r_z^k - s^k r_y^k) \qquad \mathrm{b}_{2k+1} = w_i^k (s^k T_y^k - y_i^k)$$

$$a_j . x - b_j = 0$$

$$\sum_j (a_j . x - b_j)^2$$

$$\sum_j (a_j a_j^T) x = \sum_j b_j a_j$$

$$w_i^k (x_i'^k + x_i'^k h^k (\mathbf{r_z^k} . \mathbf{p_i}) - s^k (\mathbf{r_X^k} . \mathbf{p_i} + T_X^k)) = 0$$

$$w_i^k (y_i'^k + y_i'^k h^k (\mathbf{r_z^k} . \mathbf{p_i}) - s^k (\mathbf{r_Y^k} . \mathbf{p_i} + T_Y^k)) = 0$$

Update for s^k

$$a_{2k+0} = w_i^k (r_x^k . p_i + t_x^k) \qquad \mathrm{b}_{2k+0} = w_i^k (x_i^k + x_i^k \boldsymbol{h}^k (r_z^k . p_i))$$

$$a_{2k+1} = w_i^k (r_y^k . p_i + t_y^k) \qquad \mathrm{b}_{2k+1} = w_i^k (y_i^k + y_i^k \boldsymbol{h}^k (r_z^k . p_i))$$

Rotation Around an Arbitrary Axis (Rodriguez's Formula)

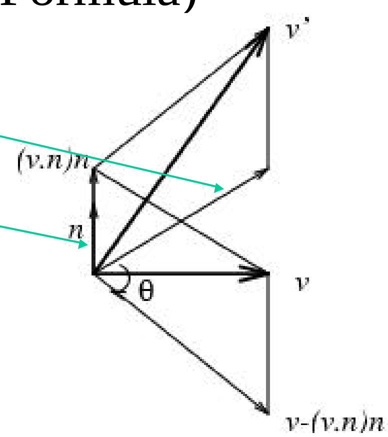
$$V = (V.n)n + (V - (V.n)n)$$

HW 4.1

$$V'_\perp = \cos \boldsymbol{q} (V - (V.n)n) + \sin \boldsymbol{q} (n \times (V - (V.n)n)$$

$$V'_{II} = (V.n)n$$

$$V' = V'_\perp + V'_{II}$$



$$V' = \cos q \; V + \sin q \; n \times V + (1 - \cos q)(V.n)n$$

$$V' = V + \sin q \; n \times V + (1 - \cos q) n \times (n \times V)$$

$$n \times (n \times V) = (V.n)n - V$$

$$n \times (n \times V) + V = (V.n)n$$

Rotation Around an Arbitrary Axis (Rodriguez's Formula)

$$V' = V + \sin q \; n \times V + (1 - \cos q) \; n \times (n \times V)$$

$$V' = R(n, q)V$$

$$R(n, q) = I + \sin q \; X(n) + (1 - \cos q) X^2(n)$$ HW 4.2

$$X(n) = \begin{bmatrix} 0 & -n_z & n_x \\ n_z & 0 & -n_x \\ -n_y & n_x & 0 \end{bmatrix}$$

Rotation Around an Arbitrary Axis (Rodriguez's Formula)

$$r = \| r \| \frac{r}{\| r |} = qn$$

$$R(r, q) = I + \sin q \frac{X(r)}{\| r \|} + (1 - \cos q) \frac{X^2(r)}{\| r \|^2}$$

$$X(r) = \begin{bmatrix} 0 & -r_z & r_x \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{bmatrix}$$

$$R^k \leftarrow \tilde{R} R^k$$

$$\tilde{R}(n, \boldsymbol{q}) = I + \sin \boldsymbol{q} X(n) + (1 - \cos \boldsymbol{q}) X^2(n)$$

$$\tilde{R} \approx I + qX(m)$$

$$m = \boldsymbol{q} n = (m_x, m_y, m_z)$$

$$w_i^k (x_i'^k + x_i'^k h^k (\mathbf{r_z^k} . \mathbf{p_i}) - s^k (\mathbf{r_X^k} . \mathbf{p_i} + T_X^k)) = 0$$
$$w_i^k (y_i'^k + y_i'^k h^k (\mathbf{r_z^k} . \mathbf{p_i}) - s^k (\mathbf{r_Y^k} . \mathbf{p_i} + T_Y^k)) = 0$$

$$R^k \leftarrow \tilde{R}(n^k, q^k) R^k$$

$$\tilde{r}_x^k = (1, -m_z, m_y)$$
$$\tilde{r}_y^k = (m_z, 1, -m_x)$$
$$\tilde{r}_z^k = (-m_y, m_x, 1)$$

$$w_i^k (x_i^k + x_i^k \boldsymbol{h} (\tilde{r}_z^k . q_i) - s^k (\tilde{r}_x^k . q_i + t_x^k)) = 0$$
$$w_i^k (y_i^k + y_i^k (\tilde{r}_z^k . q_i) - s^k (\tilde{r}_y^k . q_i + t_y^k)) = 0$$

$$q_i = R^k p_i$$

Interpolation

- Use initial set of coordinates for the feature points (13 points), to deform the remaining vertices using interpolation.

Interpolation

$$f(\mathbf{p}) = \sum_i c_i f(\|\mathbf{p} - \mathbf{p}_i\|) + \mathbf{M}\mathbf{p} + \mathbf{t}$$

Original point position: $\mathbf{p_i}$
Updated point position: $\mathbf{p_i^0}$

$$u_i = \mathbf{p}_i - \mathbf{p_i^0}, u_i = f(\mathbf{p}_i)$$

$$\sum_i c_i = 0, \sum_i c_i \mathbf{p}_i = 0$$

1.First estimate c_i's , M, and t using the known 13 points
2.Compute the displacement of Other points using *f(p)*

$$f(r) = e^{\frac{-r}{64}}$$

Homework 4.3
Describe how to
Solve for c_i's , M, and t

Texture Extraction

- Given a collection of photographs, the recovered viewing parameters, and the fitted face model, compute for each point p on the face model its texture color T(p).

Texture Extraction

$$T(\mathbf{p}) = \frac{\sum_k m^k(\mathbf{p}) I^k(x^k, y^k)}{\sum_k m^k(\mathbf{p})}$$

I^k is k-th image

m^k is weight

Weights

- Self-occlusion
  - m^k should be zero unless *P* is front facing with k-th image and is visible in it.
- Smoothness
  - The weight map should vary smoothly, in order to ensure a seamless blend between different images.
- Positional certainty
  - It is the dot product of surface normal at *P* and k-th direction of projection.
- View similarity
  - This depends on the angle between direction of projection of *P* onto j-th image and its direction of projection in the new image.

Texture Extraction

•Visibility map $F^k(u,v)$ is set to 1 if the corresponding point p is visible in k-th image, and zero otherwise.

• Positional certainty, $P^k(\mathbf{p})$ is define as a dot product of surface normal at p and the k-th direction of projection.

Texture Extraction

- View-independent texture mapping:

$$m^k(u,v) = F^k(u,v)P^k(\mathbf{p})$$

- View-dependent texture mapping:

$$m^k(u,v) = F^k(x^k, y^k)P^k(\mathbf{p})V^k(d)$$

$$V^k(\mathbf{d}) = \mathbf{d}.\mathbf{d}^k - \mathbf{d}^l.\mathbf{d}^{l+1}$$

Video Clip.