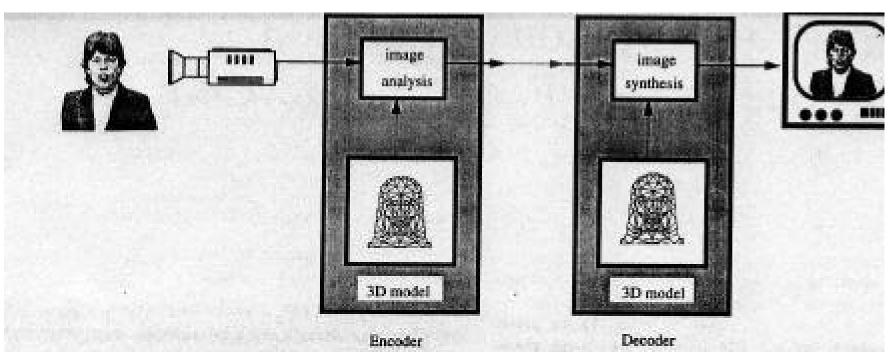
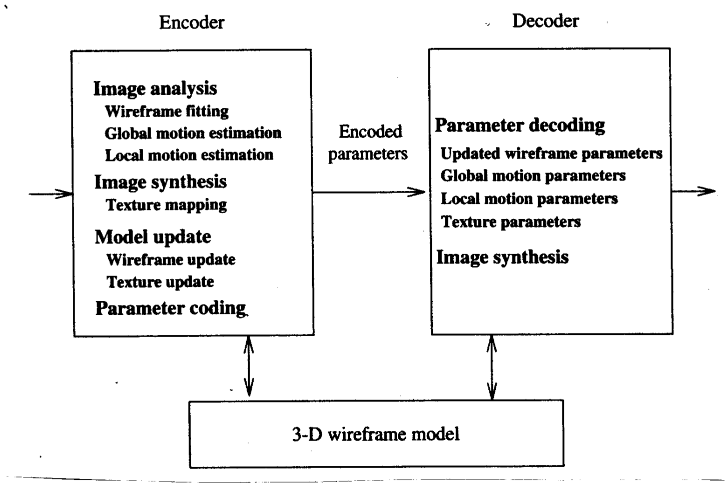
Lecture-9

Model-Based Image Coding



Model-Based Image Coding



Model-Based Image Coding

- The transmitter and receiver both posses the same 3D face model and texture images.
- During the session, at the transmitter the facial motion parameters: global and local, are extracted.
- At the receiver the image is synthesized using estimated motion parameters.
- The difference between synthesized and actual image can be transmitted as residuals.

Candide Model

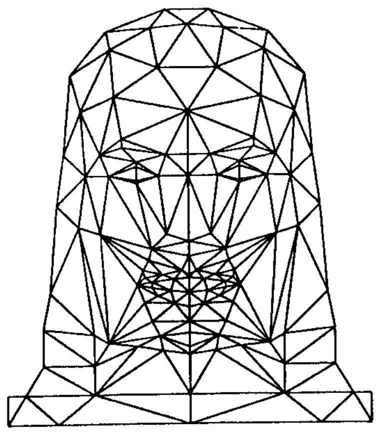


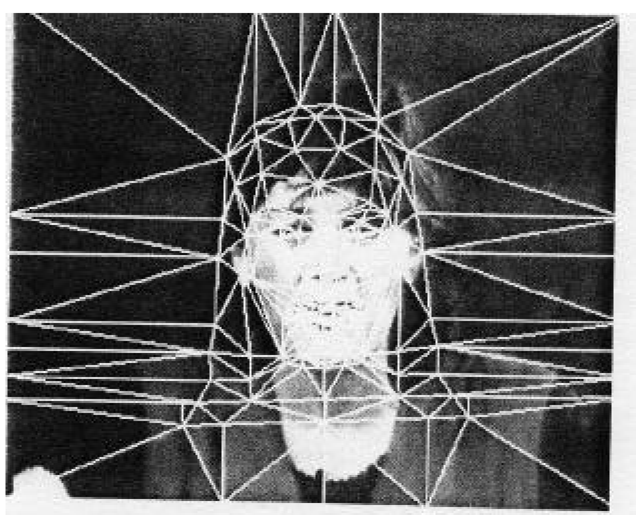
Fig. 2. Wire-frame model of the face.

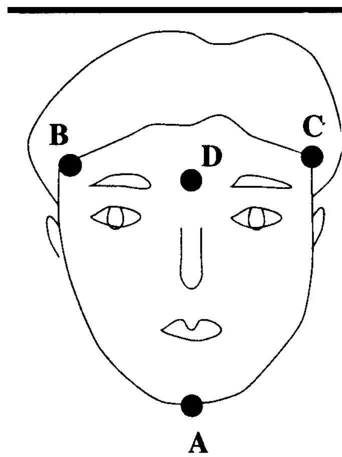
Face Model

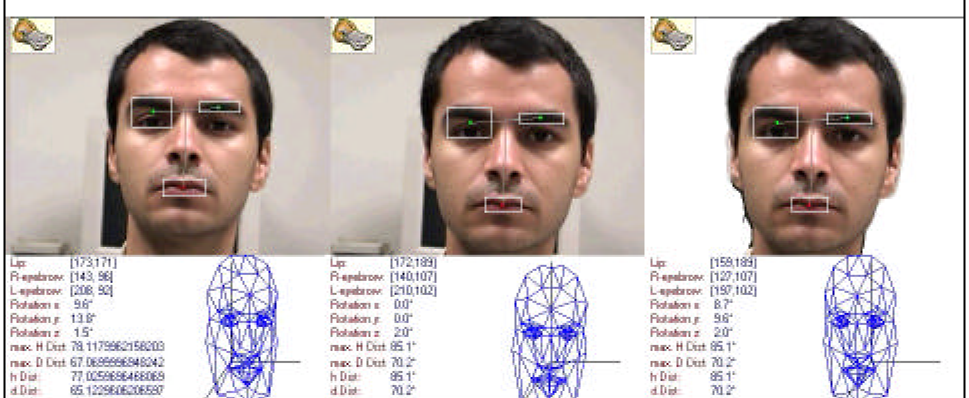
- Candide model has 108 nodes, 184 polygons.
- Candide is a generic head and shoulder model. It needs to be conformed to a particular person's face.
- Cyberware scan gives head model consisting of 460,000 polygons.

Wireframe Model Fitting

- Fit orthographic projection of wireframe to the frontal view of speaker using Affine transformation.
  - Locate three to four features in the image and the projection of a model.
  - Find parameters of Affine transformation using least squares fit.
  - Apply Affine to all vertices, and scale $\sqrt{(a_1^2 + a_4^2)^2 / 2}$ depth.



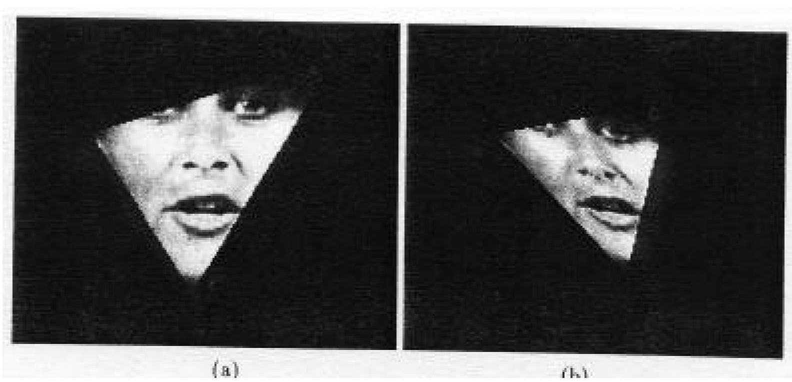

B
C
D
A


Lip: [173,171]
R-eyebrow: [143, 96]
L-eyebrow: [208, 92]
Rotation x: 9.6°
Rotation y: 13.8°
Rotation z: 1.5°
max. H Dist: 78.1179962158203
max. D Dist: 67.0695996948242
h Dist: 77.0259696466069
d Dist: 65.1229606206597
Lip: [172,189]
R-eyebrow: [140,107]
L-eyebrow: [210,102]
Rotation x: 0.0°
Rotation y: 0.0°
Rotation z: 2.0°
max. H Dist: 85.1°
max. D Dist: 70.2°
h Dist: 85.1°
d Dist: 70.2°
Lip: [159,189]
R-eyebrow: [127,107]
L-eyebrow: [197,102]
Rotation x: 8.7°
Rotation y: 9.6°
Rotation z: 2.0°
max. H Dist: 85.1°
max. D Dist: 70.2°
h Dist: 85.1°
d Dist: 70.2°

Synthesis

- Collapse initial wire frame onto the image to obtain a collection of triangles.
- Map observed texture in the first frame into respective triangles.
- Rotate and translate the initial wire frame according to global and local motion, and collapse onto the next frame.
- Map texture within each triangle from first frame to the next frame by interpolation.

Texture Mapping


(a) (b)

Video Phones

Motion Estimation

Perspective Projection (optical flow)

$$u = f\left(\frac{V_1}{Z} + \Omega_2\right) - \frac{V_3}{Z}x - \Omega_3 y - \frac{\Omega_1}{f}xy + \frac{\Omega_2}{f}x^2$$

$$v = f\left(\frac{V_2}{Z} - \Omega_1\right) + \Omega_3 x - \frac{V_3}{Z}y + \frac{\Omega_2}{f}xy - \frac{\Omega_1}{f}y^2$$

Optical Flow Constraint Eq

$$f_x u + f_y v + f_t = 0$$

$$f_x\left(f\left(\frac{V_1}{Z}+\Omega_2\right)-\frac{V_3}{Z}x-\Omega_3 y-\frac{\Omega_1}{f}xy+\frac{\Omega_2}{f}x^2\right)+f_y$$

$$\left(f\left(\frac{V_2}{Z}-\Omega_1\right)+\Omega_3 x-\frac{V_3}{Z}y+\frac{\Omega_2}{f}xy-\frac{\Omega_1}{f}y^2\right)+f_t=0$$

$$\left(f_x\frac{f}{Z}\right)V_1+\left(f_y\frac{f}{Z}\right)V_2+\left(\frac{f}{Z}(f_x x-f_y y)\right)V_3+$$

$$\left(-f_x\frac{xy}{f}+f_y\frac{y^2}{f}-f_y f\right)\Omega_1+\left(f_x f+f_x\frac{x^2}{f}+f_y\frac{xy}{f}\right)\Omega_2+$$

$$(f_x y+f_y x)\Omega_3=-f_t$$

$$(f_x \frac{f}{Z})V_1 + (f_y \frac{f}{Z})V_2 + (\frac{f}{Z}(f_x x - f_y y)V_3 +$$
$$(-f_x \frac{xy}{f} + f_y \frac{y^2}{f} - f_y f)\Omega_1 + (f_x f + f_x \frac{x^2}{f} + f_y \frac{xy}{f})\Omega_2 +$$
$$(f_x y + f_y x)\Omega_3 = -f_t$$

$$\mathbf{Ax} = \mathbf{b}$$

Solve by Least Squares

$$\mathbf{x} = (V_1, V_2, V_3, \Omega_1, \Omega_2, \Omega_3)$$

$$\mathbf{Ax} = \mathbf{b}$$

$$\begin{bmatrix} & & \vdots & & & \\ (f_x \frac{f}{Z}) & (f_y \frac{f}{Z}) & (\frac{f}{Z}(f_x x - f_y y) & (-f_x \frac{xy}{f} + f_y \frac{y^2}{f} - f_y f) & (f_x f + f_x \frac{x^2}{f} + f_y \frac{xy}{f}) & (f_x y + f_y x) \\ & & \vdots & & & \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ \Omega_1 \\ \Omega_2 \\ \Omega_3 \end{bmatrix} = \begin{bmatrix} \vdots \\ f_t \\ \vdots \end{bmatrix}$$

Comments

- This is a simpler (linear) problem than sfm because depth is assumed to be known.
- Since no optical flow is computed, this is called “direct method”.
- Only spatiotemporal derivatives are computed from the images.

Problem

- We have used 3D rigid motion, but face is not purely rigid!
- Facial expressions produce non-rigid motion.
- Use global rigid motion and non-rigid deformations.

3-D Rigid Motion

$$\mathbf{X}' = \mathbf{R}\mathbf{X} + \mathbf{T}$$

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} 1 & -a & b \\ a & 1 & -g \\ -b & g & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} T_X \\ T_Y \\ T_Z \end{bmatrix}$$

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \left(\begin{bmatrix} 0 & -a & b \\ a & 0 & -g \\ -b & g & 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right) \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} T_X \\ T_Y \\ T_Z \end{bmatrix}$$

3-D Rigid Motion

$$\begin{bmatrix} X' - X \\ Y' - Y \\ Z' - Z \end{bmatrix} = \begin{bmatrix} 0 & -a & b \\ a & 0 & -g \\ -b & g & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} T_X \\ T_Y \\ T_Z \end{bmatrix}$$

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = \begin{bmatrix} 0 & -\Omega_3 & \Omega_2 \\ \Omega_3 & 0 & -\Omega_1 \\ -\Omega_2 & \Omega_1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} T_X \\ T_Y \\ T_Z \end{bmatrix}$$

$$\dot{\mathbf{X}} = \Omega \times \mathbf{X} + \mathbf{V}$$

3-D Rigid+Non-rigid Motion

$$\mathbf{X}' = \mathbf{R}\mathbf{X} + \mathbf{T} + \mathbf{E}\Phi$$

Facial expressions

$$\mathbf{E} = \begin{bmatrix} e_{11} & e_{12} & \dots & e_{1m} \\ e_{21} & e_{22} & \dots & e_{2m} \\ e_{31} & e_{32} & \dots & e_{3m} \end{bmatrix}$$

Action Units:
-opening of a mouth
-closing of eyes
-raising of eyebrows

$$\Phi = (f_1, f_2, \dots, f_m)^T$$

3-D Rigid+Non-rigid Motion

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} 1 & -\boldsymbol{a} & \boldsymbol{b} \\ \boldsymbol{a} & 1 & -\boldsymbol{g} \\ -\boldsymbol{b} & \boldsymbol{g} & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} T_X + \sum_{i=1}^{m} e_{1i} f_i \\ T_Y + \sum_{i=1}^{m} e_{2i} f_i \\ T_Z + \sum_{i=1}^{m} e_{3i} f_i \end{bmatrix}$$

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \left(\begin{bmatrix} 0 & -\boldsymbol{a} & \boldsymbol{b} \\ \boldsymbol{a} & 0 & -\boldsymbol{g} \\ -\boldsymbol{b} & \boldsymbol{g} & 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right) \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} T_X + \sum_{i=1}^{m} e_{1i} f_i \\ T_Y + \sum_{i=1}^{m} e_{2i} f_i \\ T_Z + \sum_{i=1}^{m} e_{3i} f_i \end{bmatrix}$$

3-D Rigid+Non-rigid Motion

$$\begin{bmatrix} X'-X \\ Y'-Y \\ Z'-Z \end{bmatrix} = \begin{bmatrix} 0 & -a & b \\ a & 0 & -g \\ -b & g & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} T_X + \sum_{i=1}^{m} e_{1i} f_i \\ T_Y + \sum_{i=1}^{m} e_{2i} f_i \\ T_Z + \sum_{i=1}^{m} e_{3i} f_i \end{bmatrix}$$

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = \begin{bmatrix} 0 & -\Omega_3 & \Omega_2 \\ \Omega_3 & 0 & -\Omega_1 \\ -\Omega_2 & \Omega_1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} T_X + \sum_{i=1}^{m} e_{1i} f_i \\ T_Y + \sum_{i=1}^{m} e_{2i} f_i \\ T_Z + \sum_{i=1}^{m} e_{3i} f_i \end{bmatrix}$$

$$\dot{\mathbf{X}} = \Omega \times \mathbf{X} + \mathbf{D}$$

3-D Rigid+Non-rigid Motion

$$\dot{X} = -\Omega_3 Y + \Omega_2 Z + V_1 + \sum_{i=1}^{m} e_{1i} f_i$$

$$\dot{Y} = \Omega_3 X - \Omega_1 Z + V_2 + \sum_{i=1}^{m} e_{2i} f_i$$

$$\dot{Z} = -\Omega_2 X + \Omega_1 Z + V_3 + \sum_{i=1}^{m} e_{3i} f_i$$

Perspective Projection (arbitrary flow)

$$x = \frac{fX}{Z}$$

$$y = \frac{fY}{Z}$$

$$u = \dot{x} = \frac{fZ\dot{X} - fX\dot{Z}}{Z^2} = f\frac{\dot{X}}{Z} - x\frac{\dot{Z}}{Z}$$

$$v = \dot{y} = \frac{fZ\dot{Y} - fY\dot{Z}}{Z^2} = f\frac{\dot{Y}}{Z} - y\frac{\dot{Z}}{Z}$$

Perspective Projection (arbitrary flow)

$$\dot{X} = -\Omega_3 Y + \Omega_2 Z + V_1 + \sum_{i=1}^{m} e_{1i} f_i$$

$$\dot{Y} = \Omega_3 X - \Omega_1 Z + V_2 + \sum_{i=1}^{m} e_{2i} f_i$$

$$\dot{Z} = -\Omega_2 X + \Omega_1 Z + V_3 + \sum_{i=1}^{m} e_{3i} f_i$$

$$u = \dot{x} = \frac{fZ\dot{X} - fX\dot{Z}}{Z^2} = f\frac{\dot{X}}{Z} - x\frac{\dot{Z}}{Z}$$

$$v = \dot{y} = \frac{fZ\dot{Y} - fY\dot{Z}}{Z^2} = f\frac{\dot{Y}}{Z} - y\frac{\dot{Z}}{Z}$$

$$u = f\left(\frac{V_1 + \sum_{i=1}^{m} e_{1i} f_i}{Z} + \Omega_2\right) - \frac{V_3 + \sum_{i=1}^{m} e_{3i} f_i}{Z} x - \Omega_3 y - \frac{\Omega_1}{f} x + \frac{\Omega_2}{f} x^2$$

$$v = f\left(\frac{V_2 + \sum_{i=1}^{m} e_{2i} f_i}{Z} - \Omega_1\right) + \Omega_3 x - \frac{V_3 + \sum_{i=1}^{m} e_{3i} f_i}{Z} y + \frac{\Omega_2}{f} xy - \frac{\Omega_1}{f} y$$

Optical Flow Constraint Eq

$$f_x u + f_y v + f_t = 0$$

$$\mathbf{Ax} = \mathbf{b}$$

$$\mathbf{x} = (V_1, V_2, V_3, \Omega_1, \Omega_2, \Omega_3, f_1, f_2, \ldots, f_m)$$

Estimation Using Flexible Wireframe Model

Main Points

- Model photometric effects
- Simultaneously compute 3D motion and adapt the wireframe model.

Generalized Optical Flow Constraint

Lambertian Model

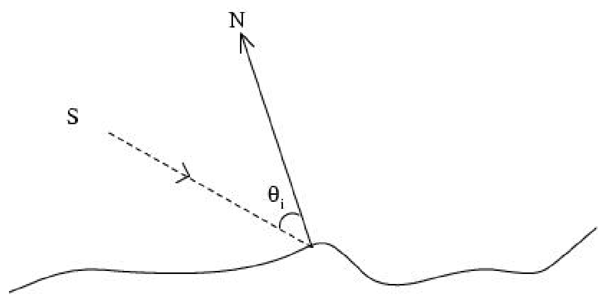
$$f(x,y,t) = rN(t).L$$

Albedo
Surface Normal
(-p,-q,1)

$$\frac{df(x,y,t)}{dt} = rL.\frac{dN}{dt}$$

$$f_x u + f_y v + f_t = rL.\frac{dN}{dt}$$

Lambertian Model



S=L, light source

$$f(x,y) = n.L = (n_x, n_y, n_z).(l_x, l_y, l_z)$$

$$f(x,y) = n.L = \left(\frac{1}{\sqrt{p^2 + q^2 + 1}}(-p, -q, 1)\right).(l_x, l_y, l_z)$$

Sphere

$$z = \sqrt{(R^2 - x^2 - y^2)}$$

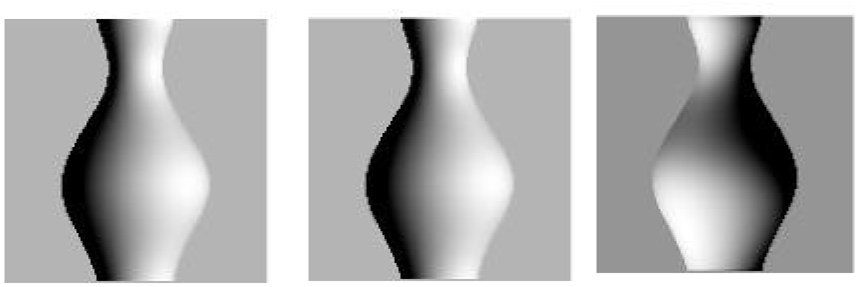
$$p = \frac{\partial z}{\partial x} = -\frac{x}{z}$$

$$q = \frac{\partial z}{\partial y} = -\frac{y}{z}$$

$$(n_x, n_y, n_z) = \frac{1}{R}(x, y, z)$$

Sphere

Vase



(1, 0, 1) (-1, 1, 1) (-1,-1, 1)

Orthographic Projection

$$u = \dot{x} = \Omega_2 Z - \Omega_3 y + V_1$$
$$v = \dot{y} = \Omega_3 x - \Omega_1 Z + V_2$$

(u,v) is optical flow

$$\dot{\mathbf{X}} = \Omega \times \mathbf{X} + \mathbf{V}$$
$$\dot{X} = \Omega_2 Z - \Omega_3 Y + V_1$$
$$\dot{Y} = \Omega_3 X - \Omega_1 Z + V_2$$
$$\dot{Z} = \Omega_1 Y - \Omega_2 X + V_3$$

Optical flow equation

$$f_x(\Omega_2 Z - \Omega_3 y + V_1) + f_y(\Omega_3 x - \Omega_1 Z + V_2) + f_t = rL.\frac{dN}{dt}$$

$$f_x(\Omega_2 Z - \Omega_3 y + V_1) + f_y(\Omega_3 x - \Omega_1 Z + V_2) + f_t =$$
$$rL.\left[\frac{(-p', -q', 1)^T}{\sqrt{p'^2 + q'^2 + 1}} - \frac{(-p, -q, 1)^T}{\sqrt{p^2 + q^2 + 1}}\right]$$

Homework 3.1
Show this.

Equation 24.8, page 473.

Error Function

$$E = \sum_i \sum_{(x,y) e ith patch} e_i^2$$

$$p_i x_1^{(ij)} + q_i x_2^{(ij)} + c_i = p_j x_1^{(ij)} + q_j x_2^{(ij)} + c_j$$

constraint

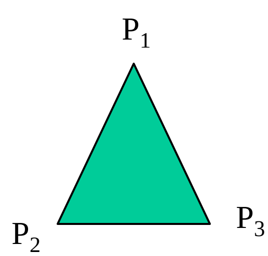
$$e_i(x, y) = f_x(\Omega_3 y - \Omega_2 (p_i x + q_1 y + c_i) + V_1)$$
$$+ f_y(-\Omega_3 x + \Omega_1 (p_i x + q_1 y + c_i) + V_2) + f_t$$
$$- \mathbf{r}(L_1, L_2, L_3).(\frac{(-\frac{-\Omega_2 + p_i}{1+\Omega_2 p_i}, \frac{-\Omega_1 + q_i}{1-\Omega_1 q_i})}{((\frac{-\Omega_2 + p_i}{1+\Omega_2 p_i})^2 + (\frac{-\Omega_1 + q_i}{1-\Omega_1 q_i})^2 + 1)^{1/2}} -$$
$$\frac{(-p_i, -q_i, 1)}{(p_i^2 + q_i^2 + 1)^{1/2}}$$

Homework 3.3
Show this.

Equation of a Planar Patch

$$P_1^{(i)} = (X_1^{(i)}, Y_1^{(i)}, Z_1^{(i)})$$
$$P_2^{(i)} = (X_2^{(i)}, Y_2^{(i)}, Z_2^{(i)})$$
$$P_3^{(i)} = (X_3^{(i)}, Y_3^{(i)}, Z_3^{(i)})$$

$$P^{(i)} = (X^{(i)}, Y^{(i)}, Z^{(i)})$$



$$\overline{P^{(i)} P_1^{(i)}} . (\overline{P_2^{(i)} P_1^{(i)}} \times \overline{P_3^{(i)} P_1^{(i)}}) = 0$$

Equation of a Planar Patch

$$Z^{(i)} = p_i X^{(i)} + q_i Y^{(i)} + c_i$$

Homework 3.2
Show this.

$$p_i = \frac{(Y_2^{(i)} - Y_1^{(i)})(Z_3^{(i)} - Z_1^{(i)}) - (Z_2^{(i)} - Z_1^{(i)})(Y_3^{(i)} - Y_1^{(i)})}{(X_2^{(i)} - X_1^{(i)})(Y_3^{(i)} - Y_1^{(i)}) - (Y_2^{(i)} - Y_1^{(i)})(X_3^{(i)} - X_1^{(i)})}$$

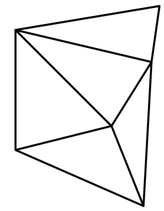
$$q_i = \frac{(Z_2^{(i)} - Z_1^{(i)})(X_3^{(i)} - X_1^{(i)}) - (X_2^{(i)} - X_1^{(i)})(Z_3^{(i)} - Z_1^{(i)})}{(X_2^{(i)} - X_1^{(i)})(Y_3^{(i)} - Y_1^{(i)}) - (Y_2^{(i)} - Y_1^{(i)})(X_3^{(i)} - X_1^{(i)})}$$

$$c_i = Z_1^{(i)} + X_1^{(i)} \frac{(Y_2^{(i)} - Y_1^{(i)})(Z_3^{(i)} - Z_1^{(i)}) - (Z_2^{(i)} - Z_1^{(i)})(Y_3^{(i)} - Y_1^{(i)})}{(X_2^{(i)} - X_1^{(i)})(Y_3^{(i)} - Y_1^{(i)}) - (Y_2^{(i)} - Y_1^{(i)})(X_3^{(i)} - X_1^{(i)})} +$$

$$Y_1^{(i)} \frac{(Z_2^{(i)} - Z_1^{(i)})(X_3^{(i)} - X_1^{(i)}) - (X_2^{(i)} - X_1^{(i)})(Z_3^{(i)} - Z_1^{(i)})}{(X_2^{(i)} - X_1^{(i)})(Y_3^{(i)} - Y_1^{(i)}) - (Y_2^{(i)} - Y_1^{(i)})(X_3^{(i)} - X_1^{(i)})}$$

Structure of Wireframe Model

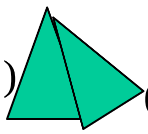
- Each triangular patch is either surrounded by two (if it is on the boundary of the wireframe) or three other triangles.



Neighboring patches must intersect at a straight line.

$$Z^{(i)} = p_i X^{(i)} + q_i Y^{(i)} + c_i$$

$$p_i x^{(ij)} + q_i y^{(ij)} + c_i = p_j x^{(ij)} + q_j y^{(ij)} + c_j$$



Main Points of Algorithm

- Stochastic relaxation.
- Each iteration visit all patches in a sequential order.
  - If, at present iteration none of neighboring patches of *i* have been visited yet, then p_i, q_i, c_i are all independently perturbed.
  - If, only one of the neighbor, j, has been visited, then two parameters, say p_i, q_i, are independent and perturbed. The dependent variable c_i is calculated from the equation:

$$c_i = p_j x^{(ij)} + q_j y^{(ij)} + c_j - p_i x^{(ij)} - q_i y^{(ij)}$$

Main Points of Algorithm

– If two of the neighboring patches, say j and k, have already been visited, i.e., the variables p_k, q_k, c_{ik} and p_j, q_j, c_j have been updated, then only one variable p_i is independent, and is perturbed. q_i, c_i can be evaluated as

$$c_i = p_j x^{(ij)} + q_j y^{(ij)} + c_j - p_i x^{(ij)} - q_i y^{(ij)}$$

$$q_i = \frac{p_k x^{(ik)} = q_j y^{(ik)} + c_k - p_i x^{(ik)} - c_k}{y^{(ik)}}$$

Main Points of Algorithm

- The perturbation of structure parameters (surface normal) for each patch, results in the change of coordinates (X,Y,Z) of the nodes of wireframe.

Updating of (X,Y,Z):

Patches i, j, k intersect at node n.

$$p_i X^{(n)} + q_i Y^{(n)} + c_i = p_j X^{(n)} + q_j Y^{(n)} + c_j$$

$$p_i X^{(n)} + q_i Y^{(n)} + c_i = p_k X^{(n)} + q_k Y^{(n)} + c_k$$

$$\begin{bmatrix} X^{(n)} \\ Y^{(n)} \end{bmatrix} = \begin{bmatrix} p_i - p_j & p_j - p_k \\ q_i - q_j & q_j - q_k \end{bmatrix}^{-1} \begin{bmatrix} c_j - c_i \\ -c_j + c_k \end{bmatrix}$$

$$Z^{(i)} = p_i X^{(i)} + q_i Y^{(i)} + c_i$$

Algorithm

- Estimate light source direction
- Initialize coordinates of all nodes using approximately scaled wireframe model. Determine initial values of surface normals for each triangle.
- Determine initial motion parameters based on selected feature correspondences and their depth values from wireframe model.
- (A) Compute the value of error function E.

- If error E is less than some threshold, then stop
- Else
  - Perturb motion parameters (3 rotations and 2 translations) by random amount (zero mean Gaussian, s.d. equal to error E)
  - Perturb structure parameters (p,q,c):
    - Perturb p, q, and c of first patch by adding random amount (zero mean Gaussian, s.d. equal to error E)
    - Increment count for all neighbors of patch-1 by 1

- For patch 2 to n
  - If the count==1
    - » Perturb p and q
    - » Compute c using equation for c_i
    - » Increment the count

$$c_i = p_j x^{(ij)} + q_j y^{(ij)} + c_j - p_i x^{(ij)} - q_i y^{(ij)}$$

- If count==2
  - » Perturb p_i
  - » Compute c_i and q_i using equations

$$c_i = p_j x^{(ij)} + q_j y^{(ij)} + c_j - p_i x^{(ij)} - q_i y^{(ij)}$$

$$q_i = \frac{p_k x^{(ik)} = q_j y^{(ik)} + c_k - p_i x^{(ik)} - c_k}{y^{(ik)}}$$

  - » Increment the count
- If p, q and c for at least three patches intersecting at node are updated, then update the coordinates of the node using equation.

$$\begin{bmatrix} X^{(n)} \\ Y^{(n)} \end{bmatrix} = \begin{bmatrix} p_i - p_j & p_j - p_k \\ q_i - q_j & q_j - q_k \end{bmatrix}^{-1} \begin{bmatrix} c_j - c_i \\ -c_j + c_k \end{bmatrix}$$

$$Z^{(i)} = p_i X^{(i)} + q_i Y^{(i)} + c_i$$

- Go to step (A)