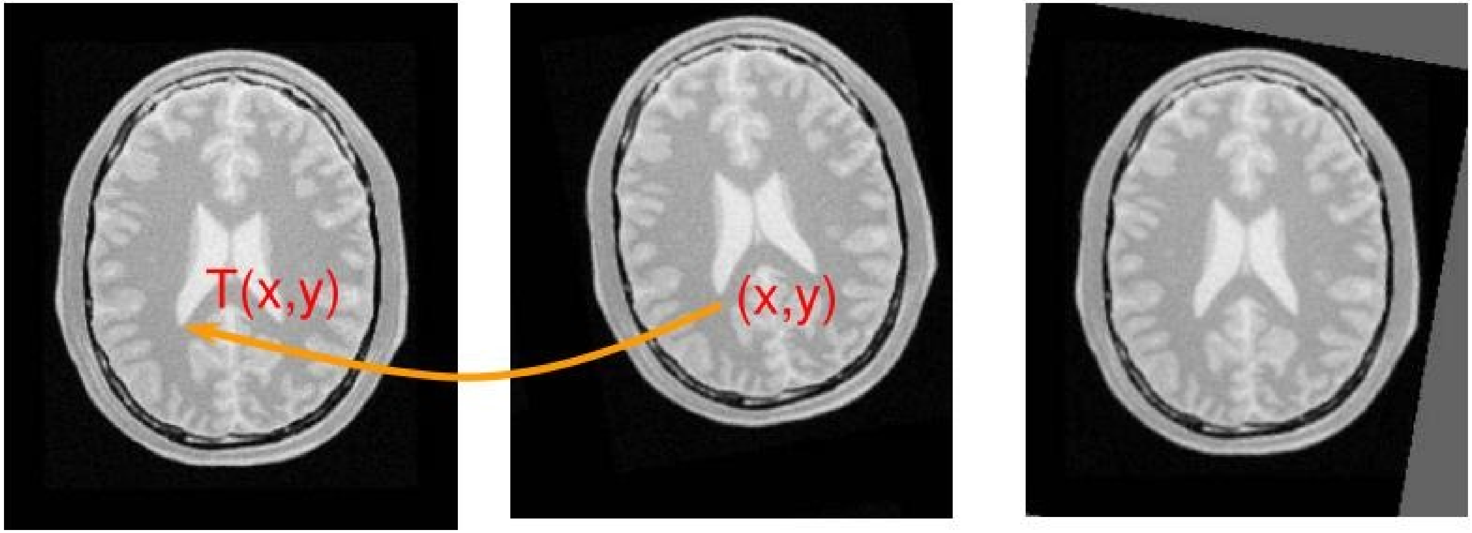
Digital Image Processing Lecture 7: Geometric Transformation

Prof. Charlene Tsai

Review

- A geometric transform of an image consists of two basic steps
  - Step1: determining the pixel co-ordinate transformation
  - Step2: determining the brightness of the points in the digital grid.



- In the previous lecture, we discussed brightness interpolation and some variations of affine transformation.

Affine Transformation (con'd)

- An affine transformation is equivalent to the composed effects of translation, rotation and scaling.
- The general affine transformation is commonly expressed as below:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \underset{\boxed{0^{th} \text{ order coefficients}}}{B} + \underset{\boxed{1^{st} \text{ order coefficients}}}{A} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$$

General Formulation

- A geometric transform is a vector function T that maps the pixel (x,y) to a new position (x',y'):

$$x' = T_x(x, y) \qquad y' = T_y(x, y)$$

- $T_x(x,y)$ and $T_y(x,y)$ are usually polynomial equations.

$$T_x(x, y) = \sum_{r=0}^{m} \sum_{k=0}^{m-r} a_{rk} x^r y^k \qquad T_y(x, y) = \sum_{r=0}^{m} \sum_{k=0}^{m-r} b_{rk} x^r y^k$$

- This transform is linear with respect to the coefficients a_{rk} and b_{rk}.

Finding Coefficients

- How to find a_{rk} and b_{rk}, which are often unknown?
  - Finding pairs of corresponding points (x,y), (x',y') in both images,
  - Determining a_{rk} and b_{rk} by solving a set of linear equations.
  - More points than coefficients are usually used to get robustness. Least-squares fitting is often used.

Jacobian

- A geometric transform applied to the whole image may change the co-ordinate system, and a Jacobian J provides information about how the co-ordinate system changes

$$J(x,y)=\frac{\partial(T_x,T_y)}{\partial(x,y)}=\begin{bmatrix} \partial T_x / \partial x & \partial T_x / \partial y \\ \partial T_y / \partial x & \partial T_y / \partial y \end{bmatrix}$$

- The area of the image is invariant if and only if $|J|=1$ ($|J|$ is the determinant of J).
- What is the Jacobian of an affine transform?

Variation of Affine (2D)

- Translation: displacement

$$T_x(x,y) = x + a_{00} \qquad T_y(x,y) = y + b_{00}$$

- Euclidean (rigid): translation + rotation

$$T_x(x,y) = a_{00} + x\cos\phi + y\sin\phi$$
$$T_y(x,y) = b_{00} - x\sin\phi + y\cos\phi$$

- Similarity: Euclidean + scaling

$$T_x(x,y) = a_{00} + s_x(x\cos\phi + y\sin\phi)$$
$$T_y(x,y) = b_{00} + s_y(-x\sin\phi + y\cos\phi)$$

(for isotropic scaling, sx = sy)

Types of transformations (2D)

- Affine: Similarity + shearing

$$T_x(x,y) = a_{00} + a_{10}x + a_{01}y$$
$$T_y(x,y) = b_{00} + b_{10}x + b_{01}y$$

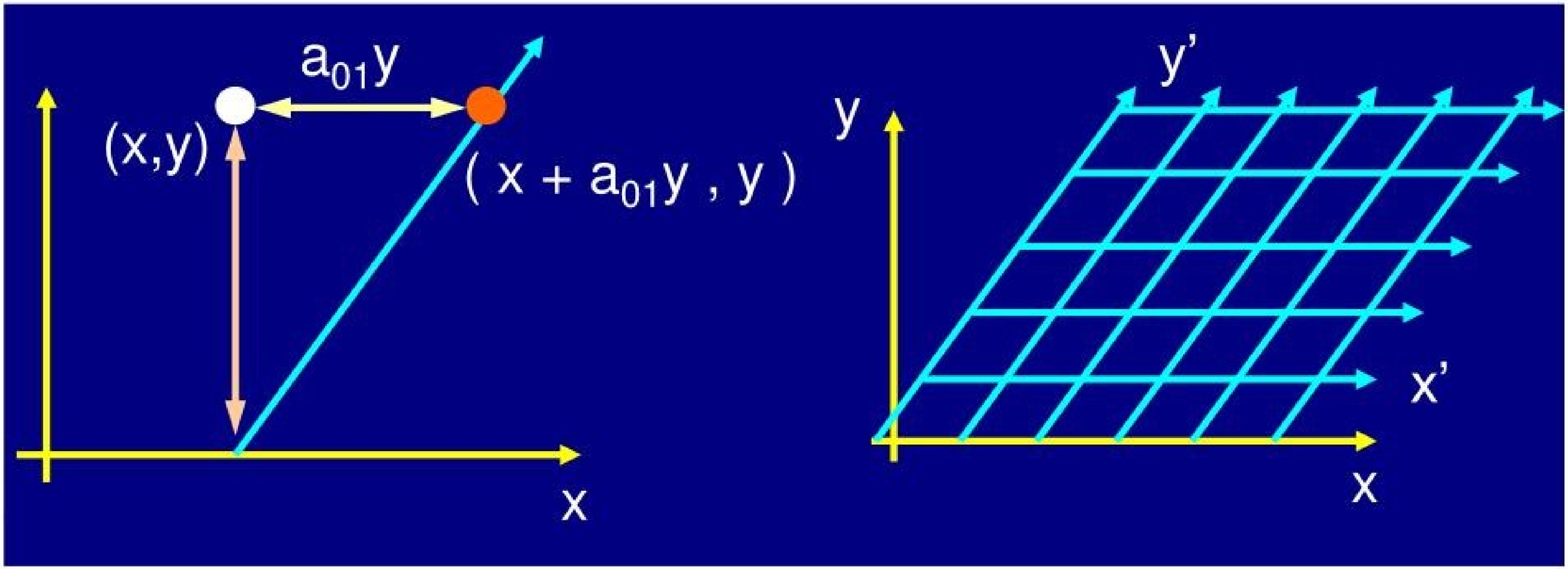
  - Some illustrations of shearing effects (courtesy of Luis Ibanez)

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a_{00} \\ b_{00} \end{bmatrix} + \begin{bmatrix} a_{10} & a_{01} \\ b_{10} & b_{01} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Affine transform

- Shearing in x-direction

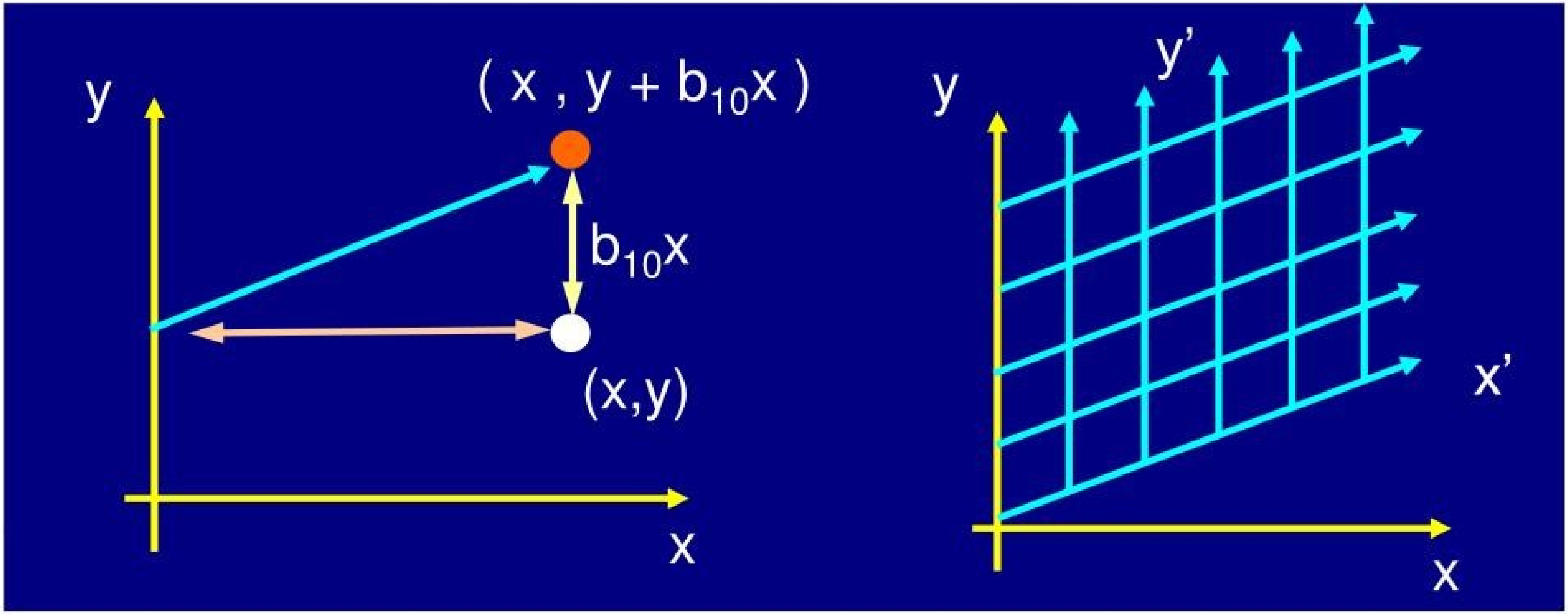
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & a_{01} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$



Affine transform

- Shearing in y-direction

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ b_{10} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$



Invariant Properties

- Translation:
- Euclidean:
- Similarity:
- Affine:

Types of transformations (2D)

- Bilinear: Affine + warping

$$T_x(x, y) = a_{00} + a_{10}x + a_{01}y + a_{11}xy$$
$$T_y(x, y) = b_{00} + b_{10}x + b_{01}y + b_{11}xy$$

- Quadratic: Affine + warping

$$T_x(x, y) = a_{00} + a_{10}x + a_{01}y + a_{11}xy + a_{20}x^2 + a_{02}y^2$$
$$T_y(x, y) = b_{00} + b_{10}x + b_{01}y + b_{11}xy + b_{20}x^2 + b_{02}y^2$$

Least-Squares Estimation

- Because of noise in the images, if there are more correspondence pairs than minimally required, it is often not possible to find a transformation that satisfies all pairs.
- Objective: To minimize the sum of the Euclidean distances of the correspondence set $C=\{p_i,q_i\}$, where $p_i=\{x_i,y_i\}$, and q_i is the corresponding point.

$$F(\theta;C)=\sum_i \left\| T(p_i;\theta)-q_i \right\|^2$$

Least-Squares Estimation - Affine

- For affine model, $\theta = [a_{00}, b_{00}, a_{10}, a_{01}, b_{10}, b_{01}]^T$
- The Euclidean error $e_i = X_i\theta - q_i$, where

$$X_i = \begin{pmatrix} 1 & 0 & p_i^T & O^T \\ 0 & 1 & O^T & p_i^T \end{pmatrix}$$

- With the new notation,

$$F(\theta; C) = \sum_i [X_i\theta - q_i]^T [X_i\theta - q_i]$$

- To find θ which minimize $F(\theta; C)$ we want

$$\frac{\partial F}{\partial \theta} = 0$$

Least-Squares Estimation - Affine

- This leads to

$$\sum_i \left[2X_i^T X_i \theta - 2X_i^T q_i \right] = 0$$

- Therefore,

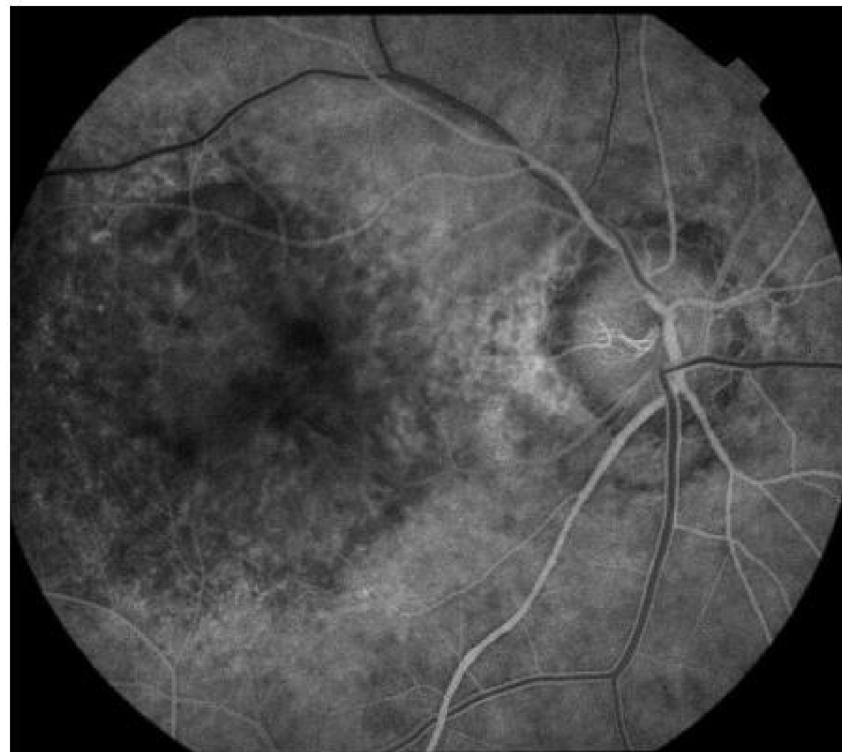
$$\theta = \left[\sum_i X_i^T X_i \right]^{-1} \left[\sum_i X_i^T q_i \right]$$

Some remarks

- So far, we only discussed global transformation (applied to entire image)
- It is possible to approximate complex geometric transformations (distortion) by partitioning an image into smaller rectangular sub-images.
- for each sub-image, a simple geometric transformation, such as the affine, is estimated using pairs of corresponding pixels.
- geometric transformation (distortion) is then performed separately in each sub-image.

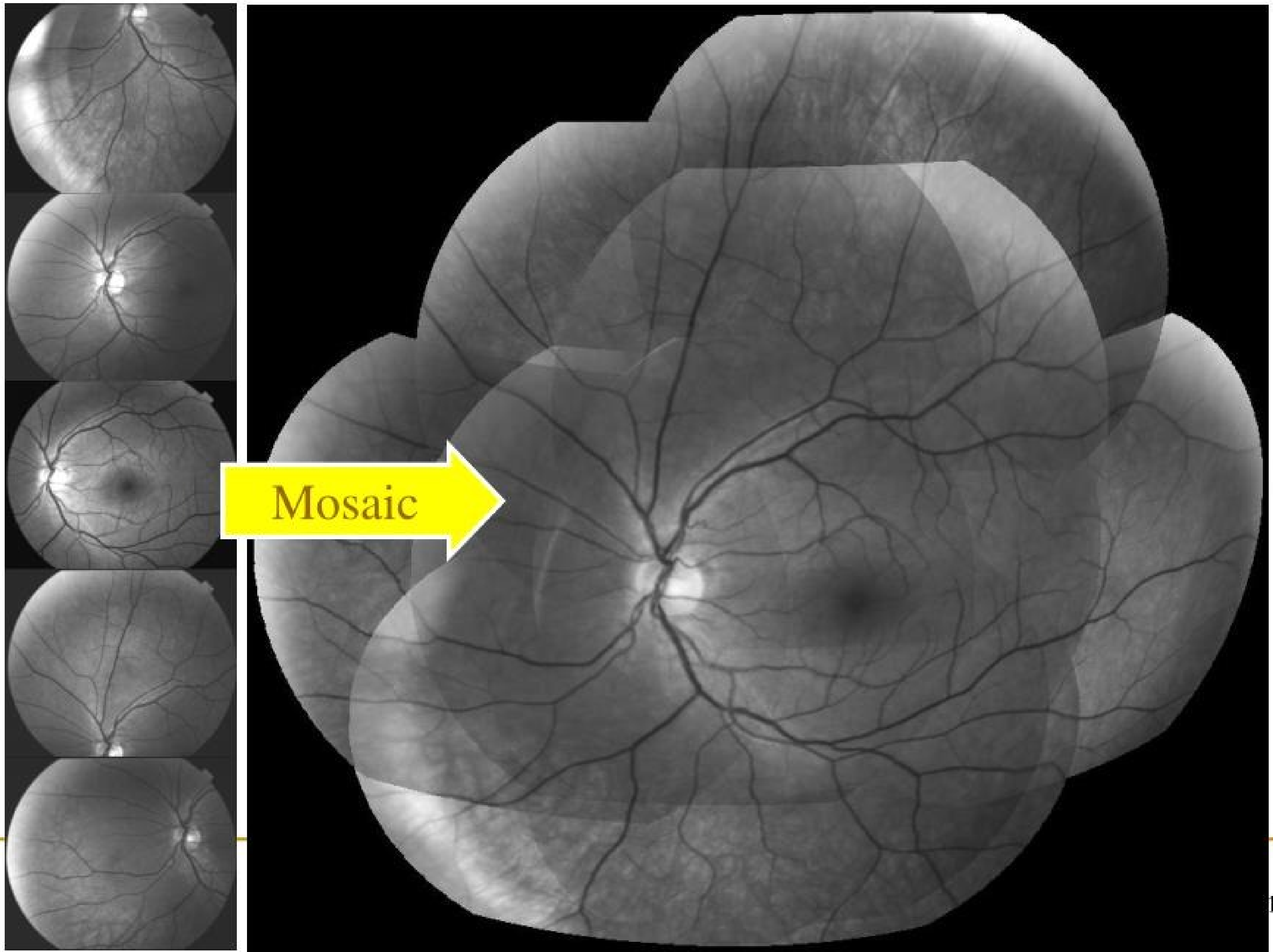
Real Application

- Registration of retinal images of different modalities
- Importance?

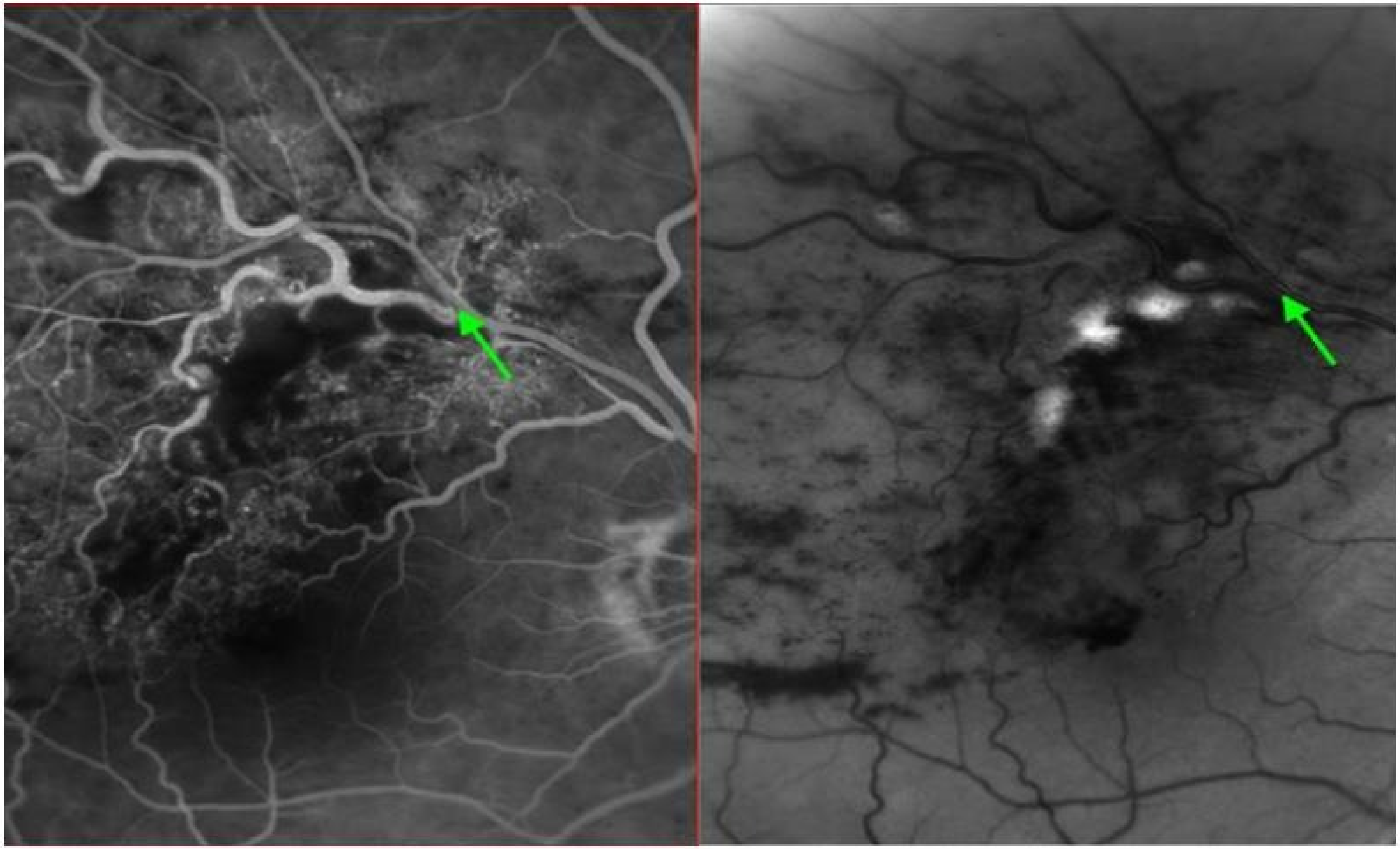


Color slide **Fluorocein Angiogram**

Mosaicing

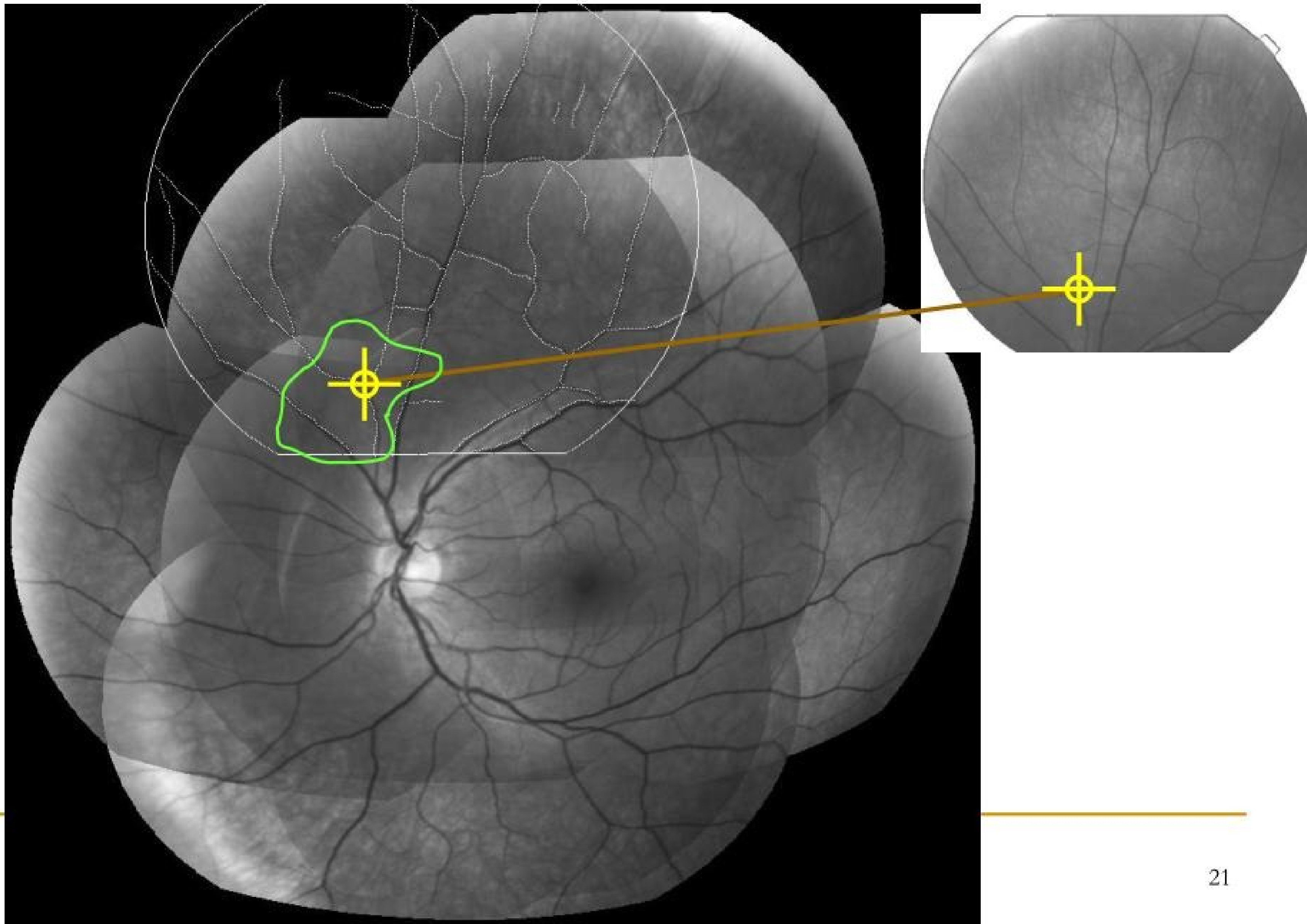


Fusion of medical information



Change Detection

Computer-assisted surgery



What is needed?

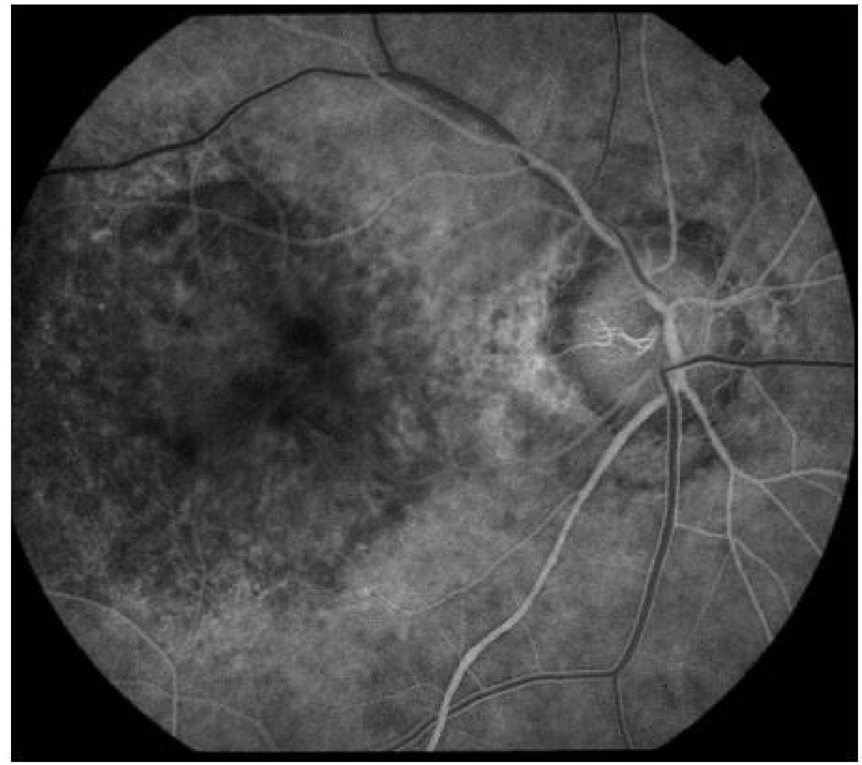
- A known transformation, which is less likely

OR

- Computing the transformation from features.
  - Feature extraction
  - Features in correspondence
  - Transformation models
  - Objective function

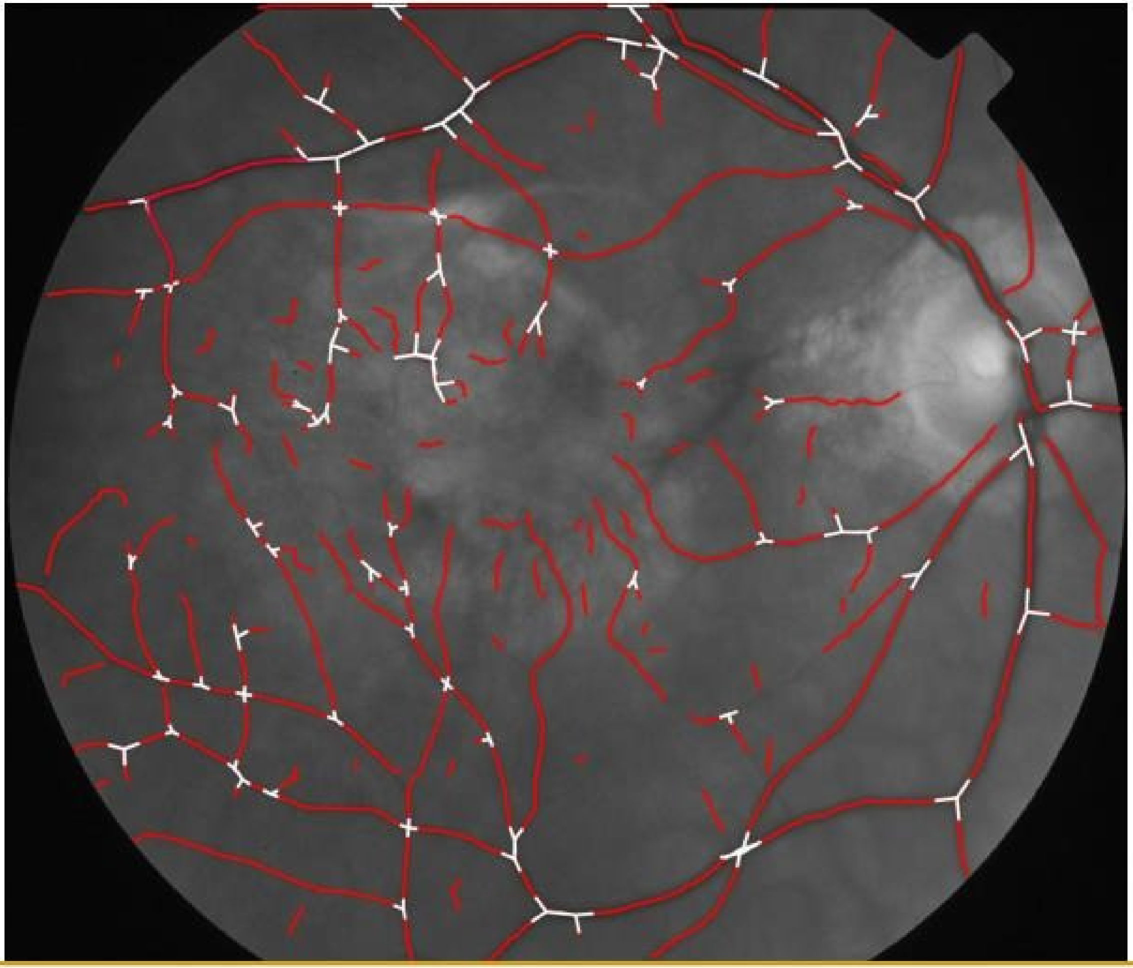
What might be a good feature?

Color slide

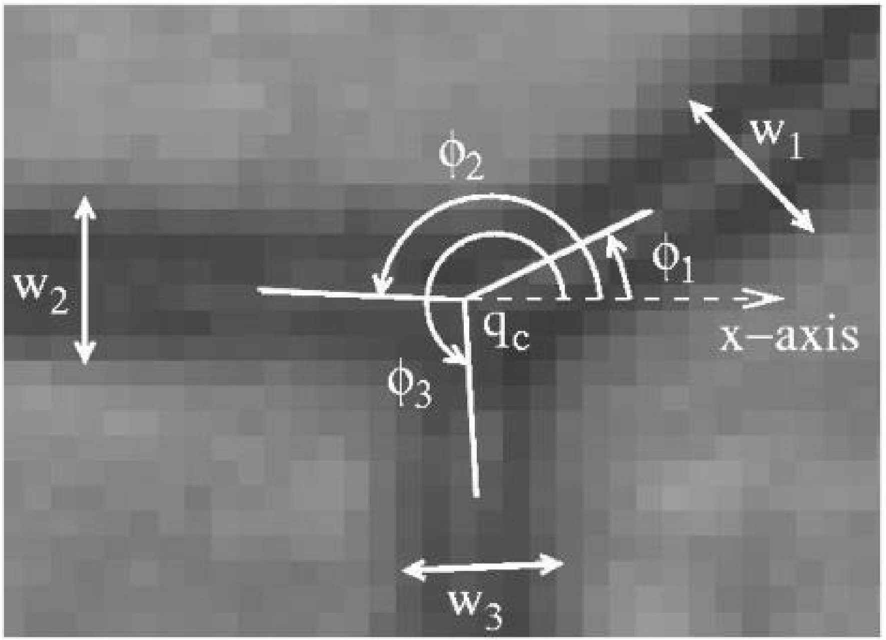


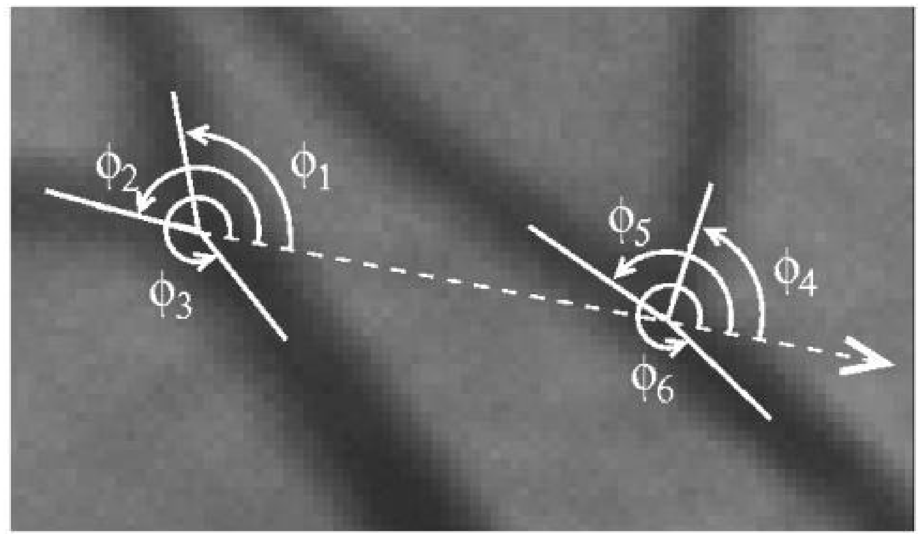
Fluorocein Angiogram

Feature Extraction

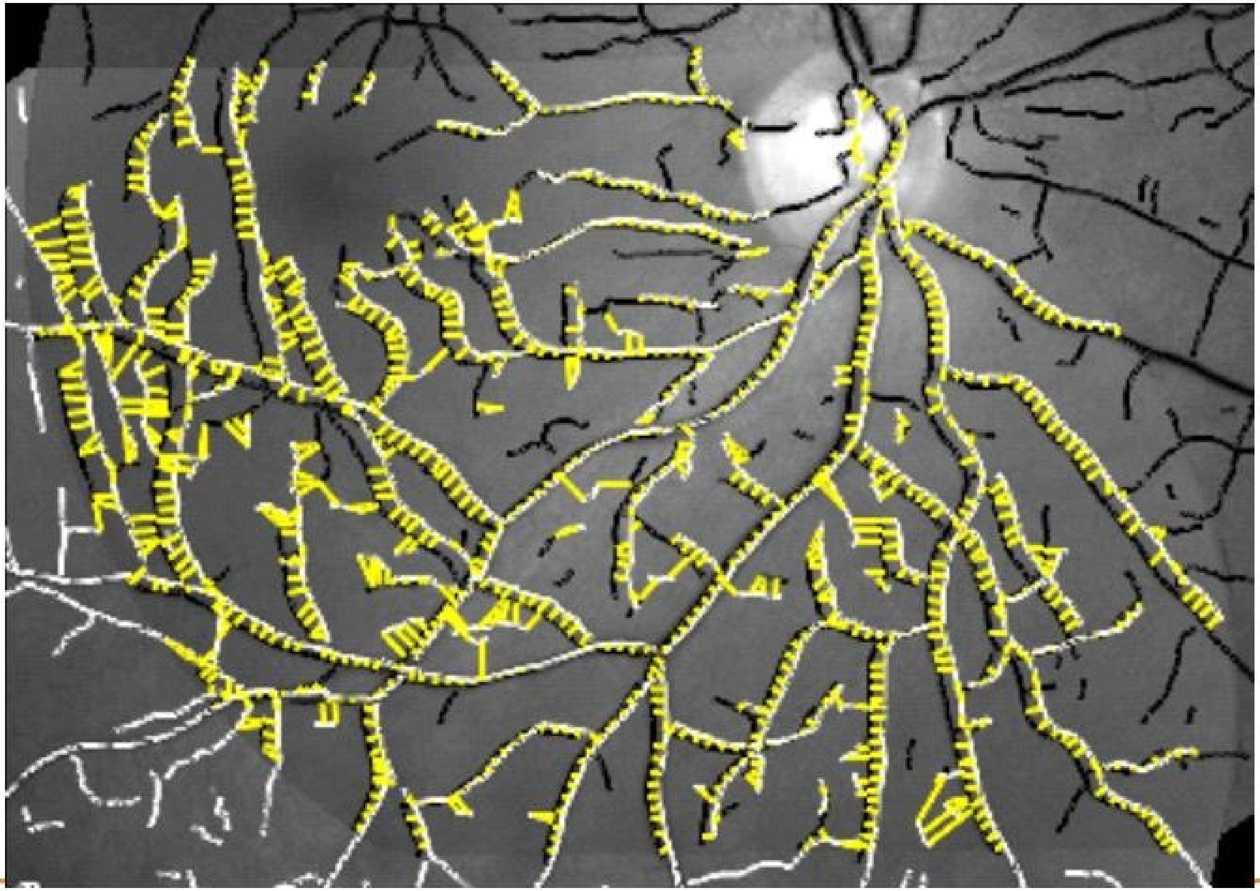


Features in Correspondence (crossover & branching points)

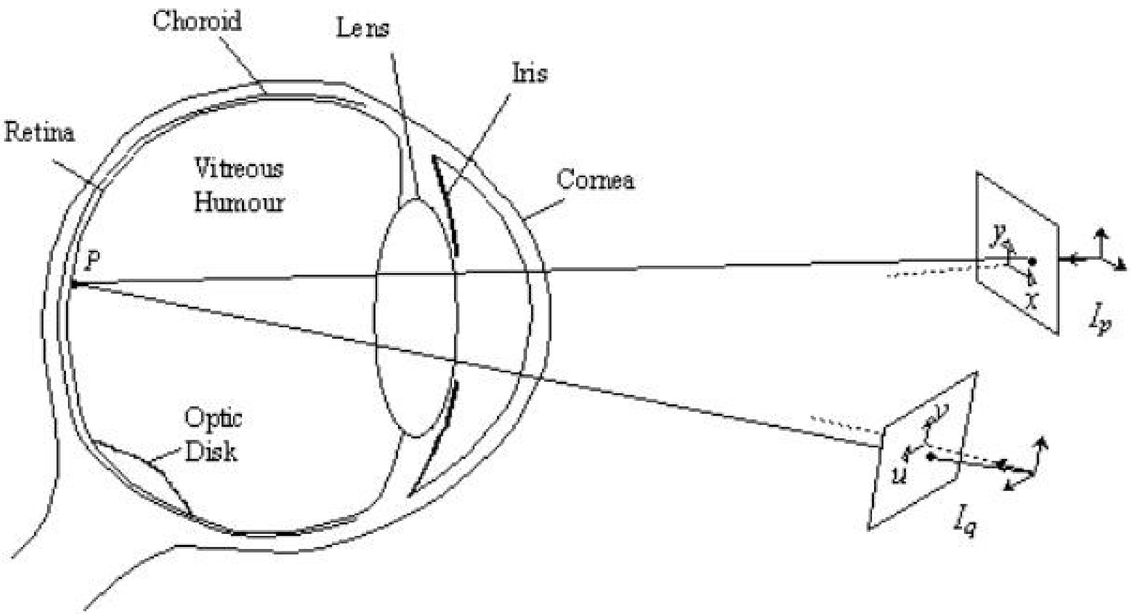


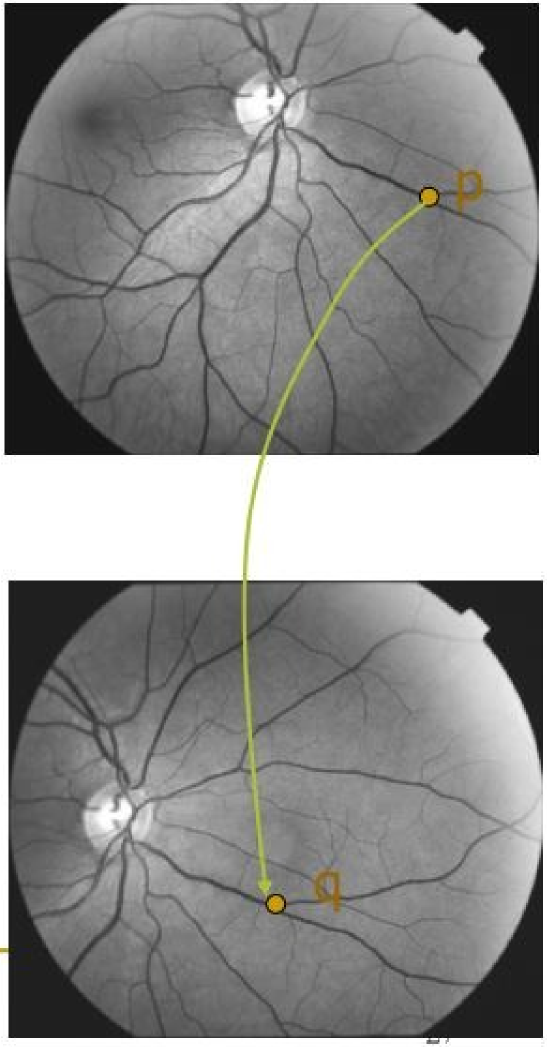


Features in Correspondence (vessel centerline points)



Transformation Models





$$\mathbf{q} = \mathbf{M}(\mathbf{p};\theta_q) = \theta_q \cdot \mathbf{X}(\mathbf{p})$$

$$\theta_q = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 \\ a_7 & a_8 & a_9 & a_{10} & a_{11} & a_{12} \end{bmatrix}$$

$$\mathbf{X}(\mathbf{p}) = \begin{bmatrix} x^2 & xy & y^2 & x & y & 1 \end{bmatrix}^T$$

$$\mathbf{p} = (x, y)^T$$

Transformation Model Hierarchy

$$q = M(p;\theta_s) = \begin{pmatrix} a & -b & t_x \\ b & a & t_y \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

$$q = M(p;\theta_a) = \begin{pmatrix} a & b & t_x \\ c & d & t_y \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

$$q = M(p;\theta_r) = \begin{pmatrix} c & a & -b & t_x \\ d & b & a & t_y \end{pmatrix} \begin{pmatrix} x^2 + y^2 \\ x \\ y \\ 1 \end{pmatrix}$$

$$q = M(p;\theta_q)$$

Objective Function

$$E(\theta;C) = \sum_{(\mathrm{p}_i,\mathrm{q}_i)\in C} \rho(\mathrm{d}(\mathrm{M}(\mathrm{p}_i;\theta),\mathrm{q}_i)/\sigma)$$

Transformation parameters

Set of correspondences $\neq \{\}$

Mapping of features from I_p to I_q based on θ

Feature point in image I_q

Feature point in image I_p

Result