The active geometric shape model: A new robust deformable shape model and its applications

Computer Vision and Image Understanding, December 2012

Quan Wang, Kim L. Boyer

Signal Analysis and Machine Perception Laboratory

Department of Electrical, Computer, and Systems Engineering

Rensselaer Polytechnic Institute

Abstract

- WHAT: we present a novel approach for <u>fitting a geometric shape</u> in images

- WHY: we can <u>detect an object</u> described with a geometric shape, represented by **parametric equations**

- HOW: we <u>adjust shape parameters</u> according to integrals of a force field along the shape contour

- APPLICATION: we use this model to detect the cross-sections of subarachnoid spaces containing cerebrospinal fluid (CSF) in phase-contrast magnetic resonance (PC-MR) image sequences

Background: Model-based image analysis

- Existing well known models:
  - Active Shape Model (ASM)
    - Statistics of point distribution
  - Active Appearance Model (AAM)
    - Statistics of point distribution + appearance

a model point is also called a landmark

- Two major steps of such models:
  1. Train the model parameters (*e.g.* PCA shapes)
  2. Fit the model to new images
- Drawbacks:
  - Need accurate annotation of landmark points
  - Need a large training dataset

Background: Geometric shape fitting

- Least squares / weighted least squares
  - Difficult to solve for complicated shapes
  - For set of points, not suited for images

- Hough transform / generalized Hough transform
  - Brute-force search on a high dimensional parameter space – cost increases exponentially when the number of parameters increases
  - Suited for black & white images, not gray/color

Important concept: Force field

- To fit a deformable model, model points move along the *force field* in each iteration

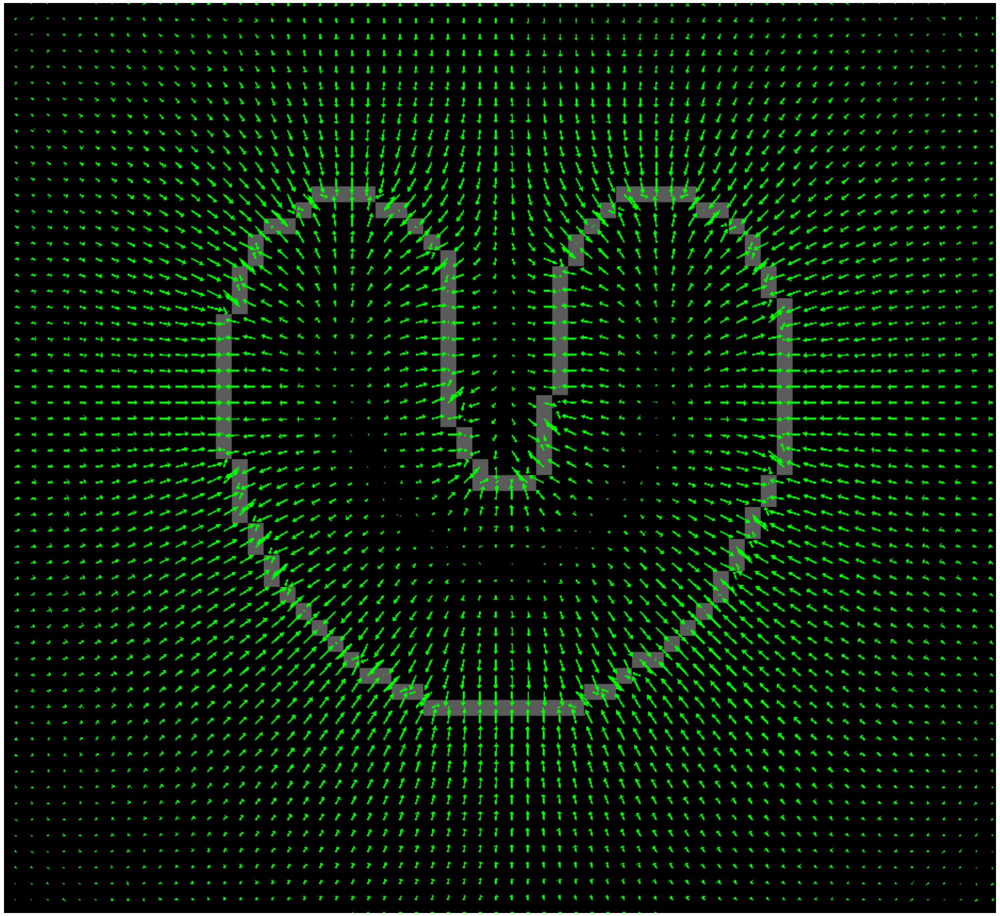
- A good force field needs to:
  1. Respect the gradient
  2. Be smooth and have a large capture range

- Gradient vector flow (GVF) is most widely used:
  - GVF $\mathbf{v}(x,y) = [u(x,y), v(x,y)]$ minimizes an energy functional (f is the smoothed image)

$$\mathcal{E} = \iint \left(\mu(u_x^2 + u_y^2 + v_x^2 + v_y^2) + \|\nabla f\|^2 \|\mathbf{v} - \nabla f\|^2 \right) \mathrm{d}x\mathrm{d}y$$

Deformable models and force field

- Biggest advantage of gradient vector flow (GVF) – large capture range



Overview of our AGSM

- Our problem
  - Training set is too small for statistical analysis 
  - Shape has a good geometric representation: parametric equations 

1. We associate each parameter with a force or torque
   - *Force* for position/size/shape parameters
   - *Torque* for orientation parameters

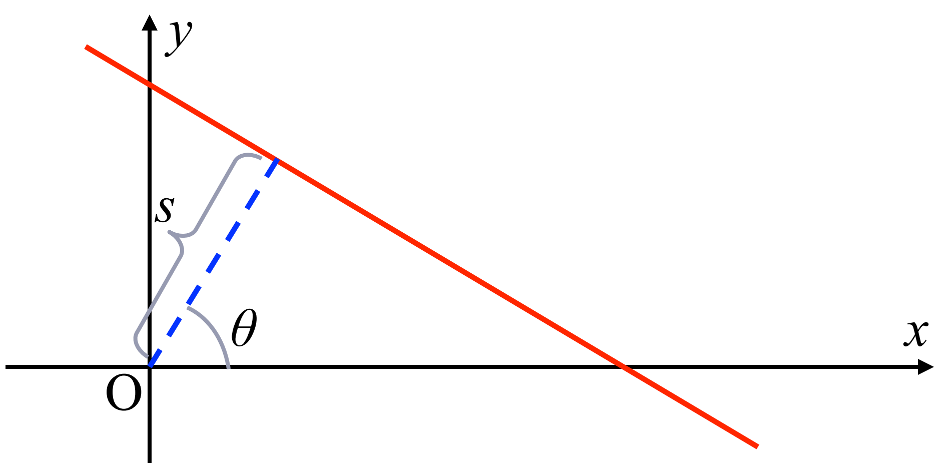
2. We adjust the parameter according to this force or torque

Example: Line-fitting

- Parametric equation for a line:

$$x\cos\theta + y\sin\theta - s = 0$$

- Two parameters: s and θ
- Geometric understanding:
  - s: the distance from the origin to the line
  - θ: the orientation
- Let the GVF force field be $\mathbf{F}(x,y) = [F_x(x,y), F_y(x,y)]$

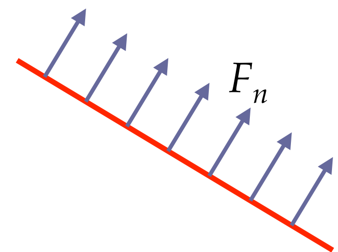


Example: Line-fitting (define the force)

- The <u>normal force</u> for parameter *s*:

The dot product indicates whether the force is pushing the line or pulling the line

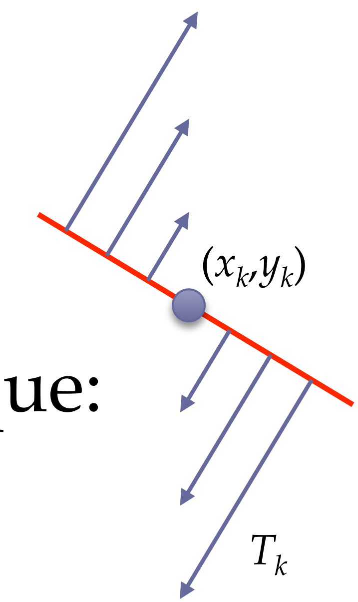
$$F_n = \frac{1}{N}\sum_{i=1}^{N} \mathbf{F}(x_i, y_i) \cdot \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}$$



- The <u>torque</u> around pivot point (x_k, y_k):

$$T_k = \frac{1}{N^2}\sum_{i=1}^{N} \operatorname{sgn}(k-i)\, d_{ik} \mathbf{F}(x_i, y_i) \cdot \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}$$

$$d_{ik} = \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2}$$



- The *k* is selected to <u>maximize</u> the torque:

$$\tilde{k} = \arg\max_k |T_k|$$

Example: Line-fitting (update parameters)

- Parameters are updated according to the force/torque:

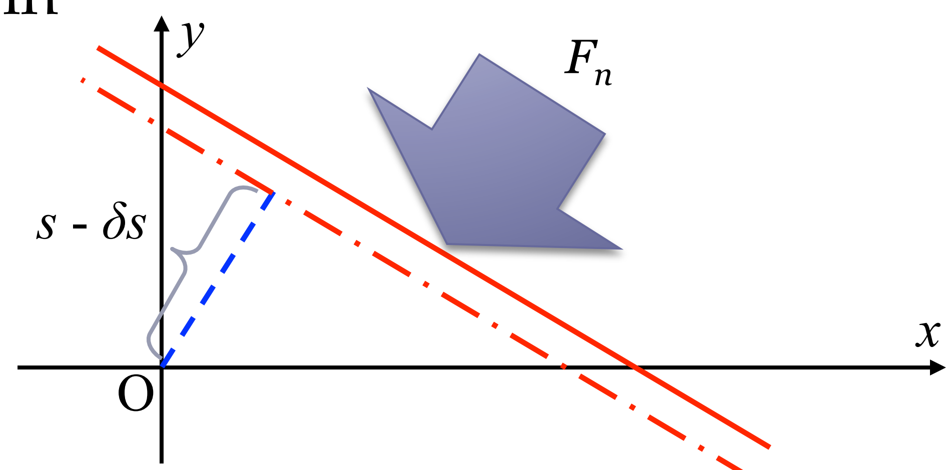
$$\begin{cases} s_{\text{new}} = s + \delta s & \text{if } F_n > t_s \\ s_{\text{new}} = s - \delta s & \text{if } F_n < -t_s \end{cases}$$

step size

threshold

$$\begin{cases} \theta_{\text{new}} = \theta - \delta\theta & \text{if } T > t_\theta \\ \theta_{\text{new}} = \theta + \delta\theta & \text{if } T < -t_\theta \end{cases}$$

- Explanation: if the force pushes the line towards the origin, then we change the parameters to move it closer to the origin



Generalization from the line example

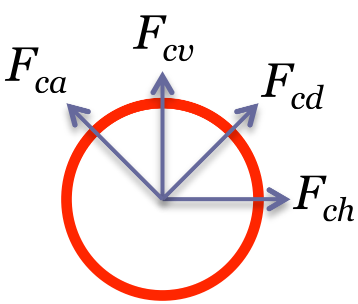
1. For each parameter, we define a force/torque for it according to its **geometric meaning**
   - This force/torque tends to directly change the value of this parameter
2. We adjust the parameter according to the **sign** of the force/torque
3. All parameters are adjusted in arbitrary order (order does not matter) in one iteration
4. After many iterations we get a good fit to the image

Fitting a circle

- Parametric equations:

$$\begin{cases} x = x_c + r\cos\theta \\ y = y_c + r\sin\theta \end{cases}$$

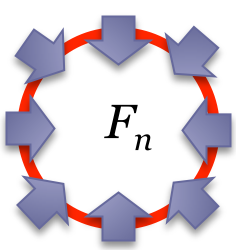
- For the center (x_c, y_c), we define horizontal (ch), vertical (cv), diagonal (cd), and anti-diagonal (ca) forces:



$$F_{ch} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{F}(x_i, y_i) \cdot [1, 0]^{\mathsf{T}}, \qquad F_{cd} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{F}(x_i, y_i) \cdot \left[\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\right]^{\mathsf{T}},$$

$$F_{cv} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{F}(x_i, y_i) \cdot [0, 1]^{\mathsf{T}}, \qquad F_{ca} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{F}(x_i, y_i) \cdot \left[-\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\right]^{\mathsf{T}}.$$

- For the radius r, we define the normal force:



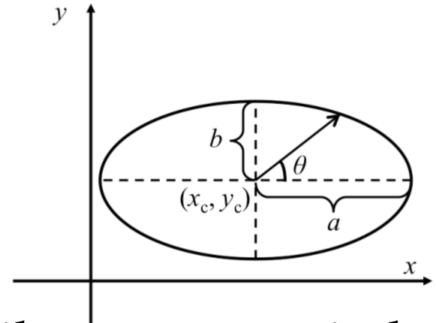
$$F_n = \frac{1}{N}\sum_{i=1}^{N} \mathbf{F}(x_i, y_i) \cdot \begin{bmatrix} \cos\theta_i \\ \sin\theta_i \end{bmatrix}$$

The dot product indicates whether the force makes the circle expand or shrink

Fitting an ellipse in standard orientation

- Parametric equations:

$$\begin{cases} x = x_c + a\cos\theta \\ y = y_c + b\sin\theta \end{cases}$$



- The center (x_c, y_c) can be fitted in a similar way to a circle
- The force for the shape parameters a and b are defined on part of the ellipse:

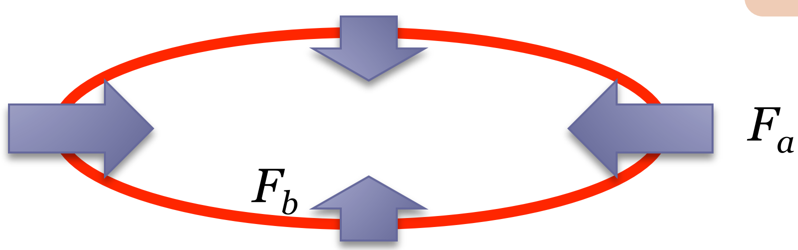
$$F_a = \frac{1}{N_a}\left(\sum_{\frac{3\pi}{4}<\theta_i<\frac{5\pi}{4}} \mathbf{F}(x_i, y_i)\cdot[1, 0]^T + \sum_{\theta_i<\frac{\pi}{4} \text{ or } \theta_i>\frac{7\pi}{4}} \mathbf{F}(x_i, y_i)\cdot[-1, 0]^T\right)$$

$$F_b = \frac{1}{N_b}\left(\sum_{\frac{5\pi}{4}<\theta_i<\frac{7\pi}{4}} \mathbf{F}(x_i, y_i)\cdot[0, 1]^T + \sum_{\frac{\pi}{4}<\theta_i<\frac{3\pi}{4}} \mathbf{F}(x_i, y_i)\cdot[0, -1]^T\right)$$

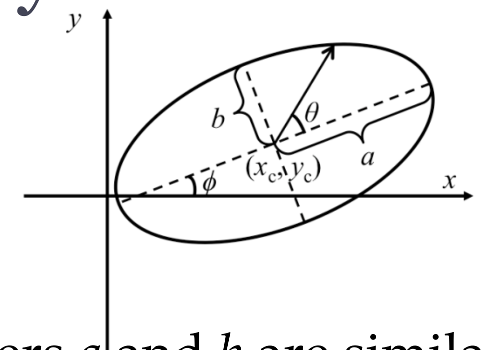
$$N_a = \sum_{\frac{3\pi}{4}<\theta_i<\frac{5\pi}{4}} 1 + \sum_{\theta_i<\frac{\pi}{4} \text{ or } \theta_i>\frac{7\pi}{4}} 1,$$

$$N_b = \sum_{\frac{5\pi}{4}<\theta_i<\frac{7\pi}{4}} 1 + \sum_{\frac{\pi}{4}<\theta_i<\frac{3\pi}{4}} 1.$$

Normalization numbers



Fitting an ellipse in arbitrary orientation



- Parametric equations:

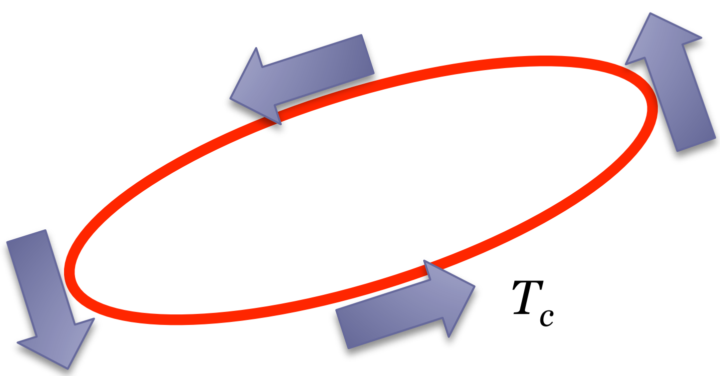
$$\begin{cases} x = x_c + a\cos\theta\cos\phi - b\sin\theta\sin\phi \\ y = y_c + a\cos\theta\sin\phi + b\sin\theta\cos\phi \end{cases}$$

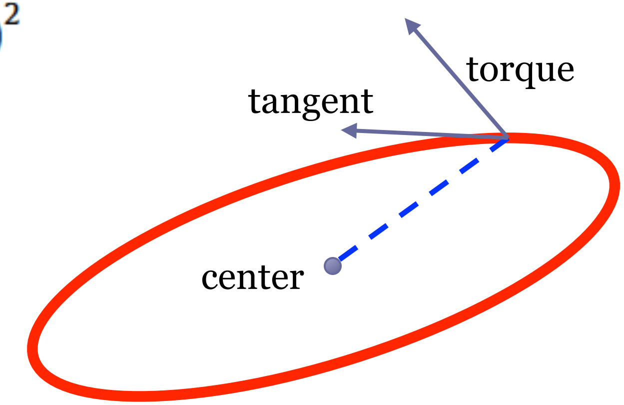
- The center (x_c, y_c) and the shape parameters a and b are similar to a standard ellipse
- The torque for the shape orientation ϕ:

$$T_c = \frac{1}{N^2}\sum_{i=1}^{N} d_i\,\mathbf{F}(x_i, y_i) \cdot \begin{bmatrix} -\sin(\theta+\phi) \\ \cos(\theta+\phi) \end{bmatrix}$$

$$d_i = \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2}$$

The dot product can be thought of something similar to a shear stress, but not necessarily in a tangent direction!





Fitting a distorted ellipse

- Parametric equations ($p > 1$):

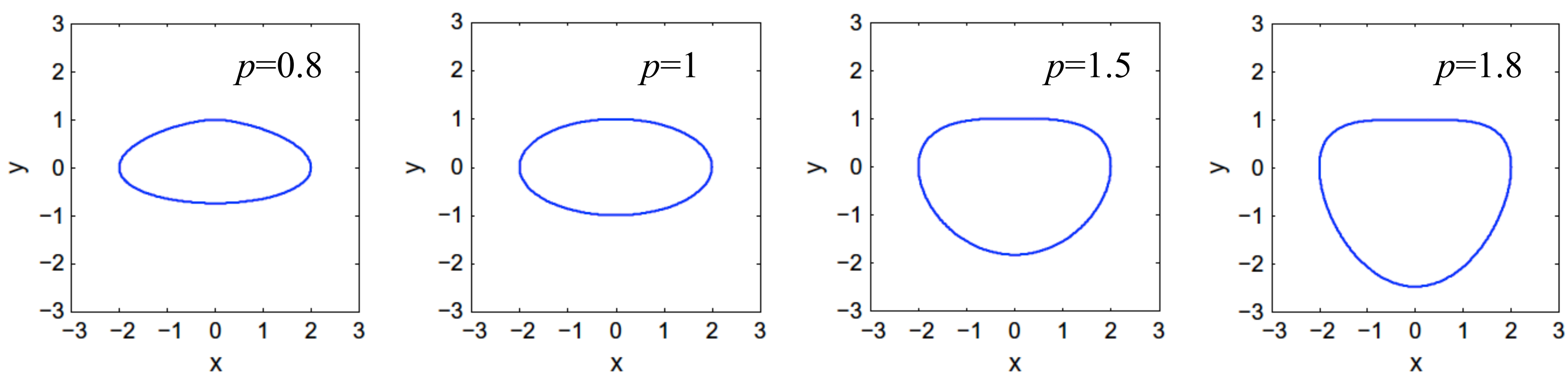
$$\begin{cases} x = x_c + a\cos\theta \\ y = y_c + b\left(1 - (1 - \sin\theta)^p\right) \end{cases}$$

This is the problem that motivated this work

- The force for the distortion parameter p:

$$F_p = \frac{1}{N_p} \sum_{\frac{11\pi}{8} < \theta_i < \frac{13\pi}{8}} \mathbf{F}(x_i, y_i) \cdot [0,\ 1]^{\mathrm{T}}$$

Defined on the lower part (the most protruding part) of the shape



Fitting a cubic spline contour

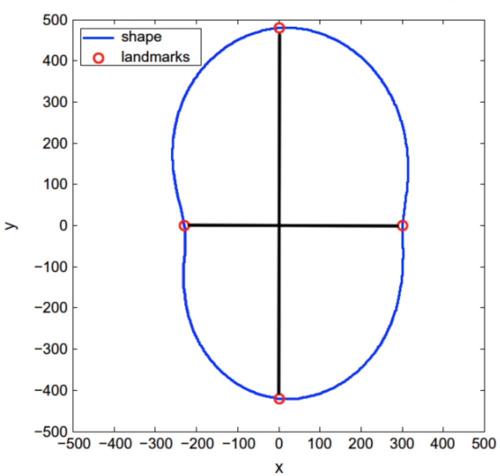
- Shape is obtained by cubic spline interpolation using N_{lm} landmark points:

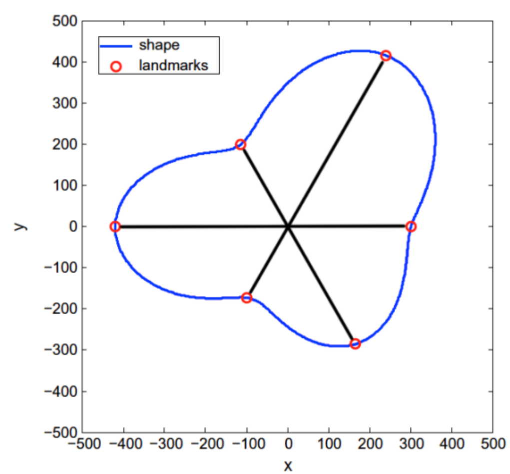
$$\begin{cases} x_{P_k} = x_c + D_k \cos \Theta_k \\ y_{P_k} = y_c + D_k \sin \Theta_k \end{cases} \qquad \Theta_k = (k-1)\frac{2\pi}{N_{lm}}$$

- Parameters: (x_c, y_c) and $D = (D_1, D_2, \ldots, D_{Nlm})$
- Force for D_k:

Dot product defined on local arc: expand or shrink

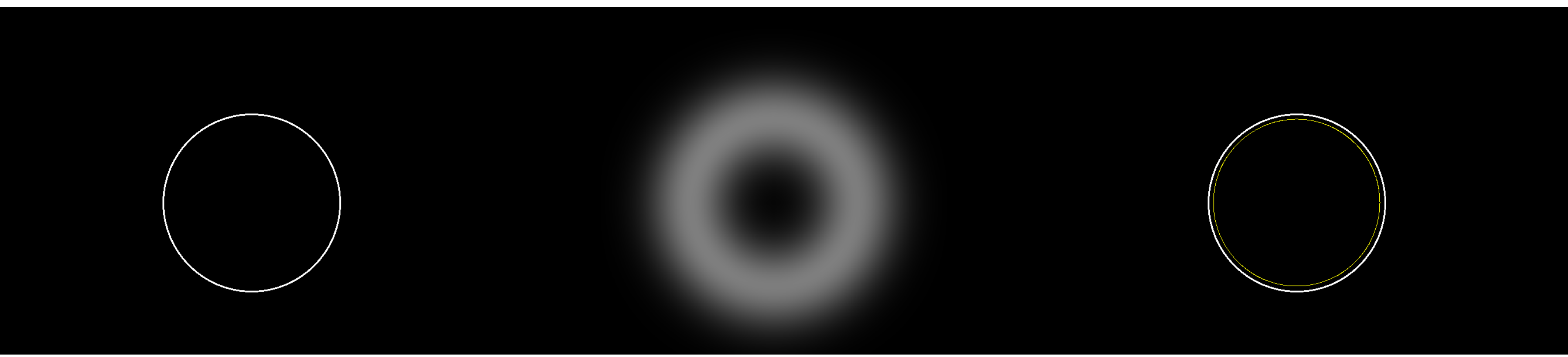
$$F_{D_k} = \frac{1}{N_{D_k}} \sum_{\Theta_k - \frac{\pi}{N_{lm}} < \theta_i < \Theta_k + \frac{\pi}{N_{lm}}} \mathbf{F}(x_i, y_i) \cdot [\cos \theta_i, \sin \theta_i]^{\mathrm{T}}$$





Correction of curvature

- To increase the capture range of the force field, the gradient is computed on the **smoothed** version of the image (standard practice)
- This smoothing operation dislocates the local maxima (where the model converges to) from original positions

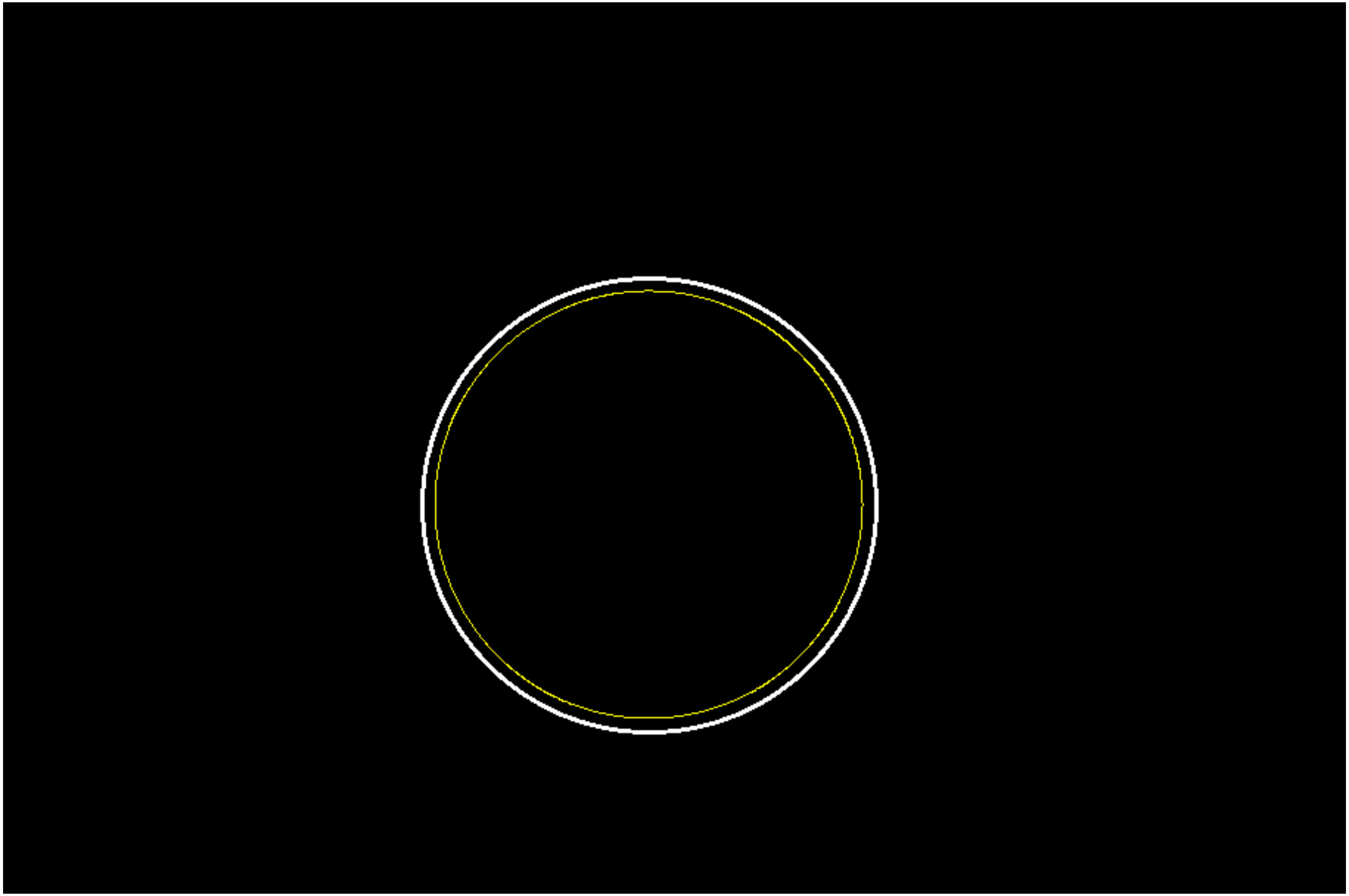


A circle

The Gaussian smoothed circle (enhanced for visualization)

The local maxima of the smoothed circle are on a smaller circle (yellow)

Correction of curvature



Correction for a circle

- In the polar coordinate system (ρ,θ), we define a <u>disk</u> with radius R as $M(\rho,\theta) = U(R - \rho)$, where $U(\bullet)$ is the unit step

convolution

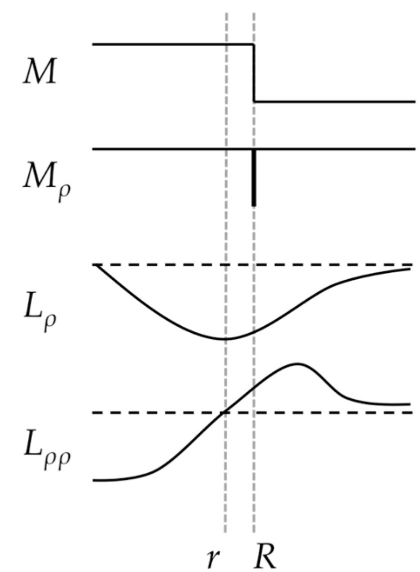
- The convolution with Gaussian kernel $G_\sigma(\rho,\theta)$ is $L(\rho,\theta) = G_\sigma * M$

standard deviation

- The derivative of M in the radial direction is $M_\rho = - \delta(R - \rho)$
- Based on the work of Bouma *et al.* (PAMI 2005), we can compute the first order and second order derivatives of $L(\rho,\theta)$:

$$L_\rho(\rho,\theta) = G_\sigma * M_\rho = -\frac{R}{\sigma^2} e^{-\frac{R^2+\rho^2}{2\sigma^2}} I_1\left(\frac{\rho R}{\sigma^2}\right)$$

$$L_{\rho\rho}(\rho,\theta) = e^{-\frac{R^2+\rho^2}{2\sigma^2}} \left(-\frac{R^2}{\sigma^4} I_0\left(\frac{\rho R}{\sigma^2}\right) + \left(\frac{\rho R}{\sigma^4} + \frac{R}{\rho\sigma^2}\right) I_1\left(\frac{\rho R}{\sigma^2}\right)\right)$$



- $I_n(\bullet)$ is the modified Bessel function of the first kind

Correction for a circle

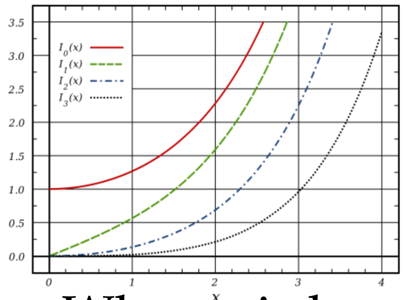
- If $L_{\rho\rho}(r,\theta) = 0$, then r is the dislocated radius of the disk $M(\rho,\theta) = U(R - \rho)$ whose true radius is R

M: disk
M_ρ: circle
$M_{\rho\rho}$: derivative of circle
L: smoothed disk
L_ρ: smoothed circle
$L_{\rho\rho}$: derivative of smoothed circle

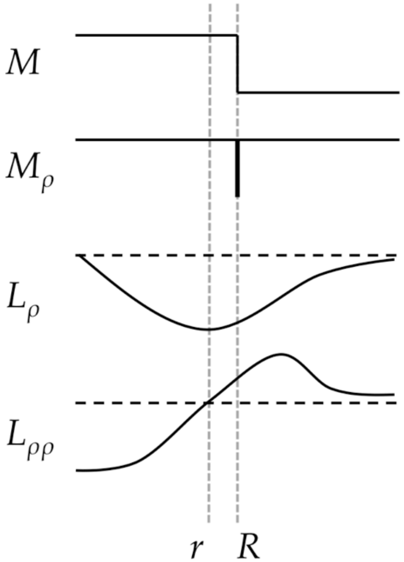
- The equation $L_{\rho\rho}(r,\theta) = 0$ can be rewritten as:

$$\frac{R}{\sigma^2} I_0\left(\frac{rR}{\sigma^2}\right) = \left(\frac{r}{\sigma^2} + \frac{1}{r}\right) I_1\left(\frac{rR}{\sigma^2}\right)$$

- We solve for $R = \Omega(r,\sigma)$ using numeric iterations:



$$R^{(k+1)} = \left(r + \frac{\sigma^2}{r}\right) \frac{I_1\left(\frac{rR^{(k)}}{\sigma^2}\right)}{I_0\left(\frac{rR^{(k)}}{\sigma^2}\right)}$$

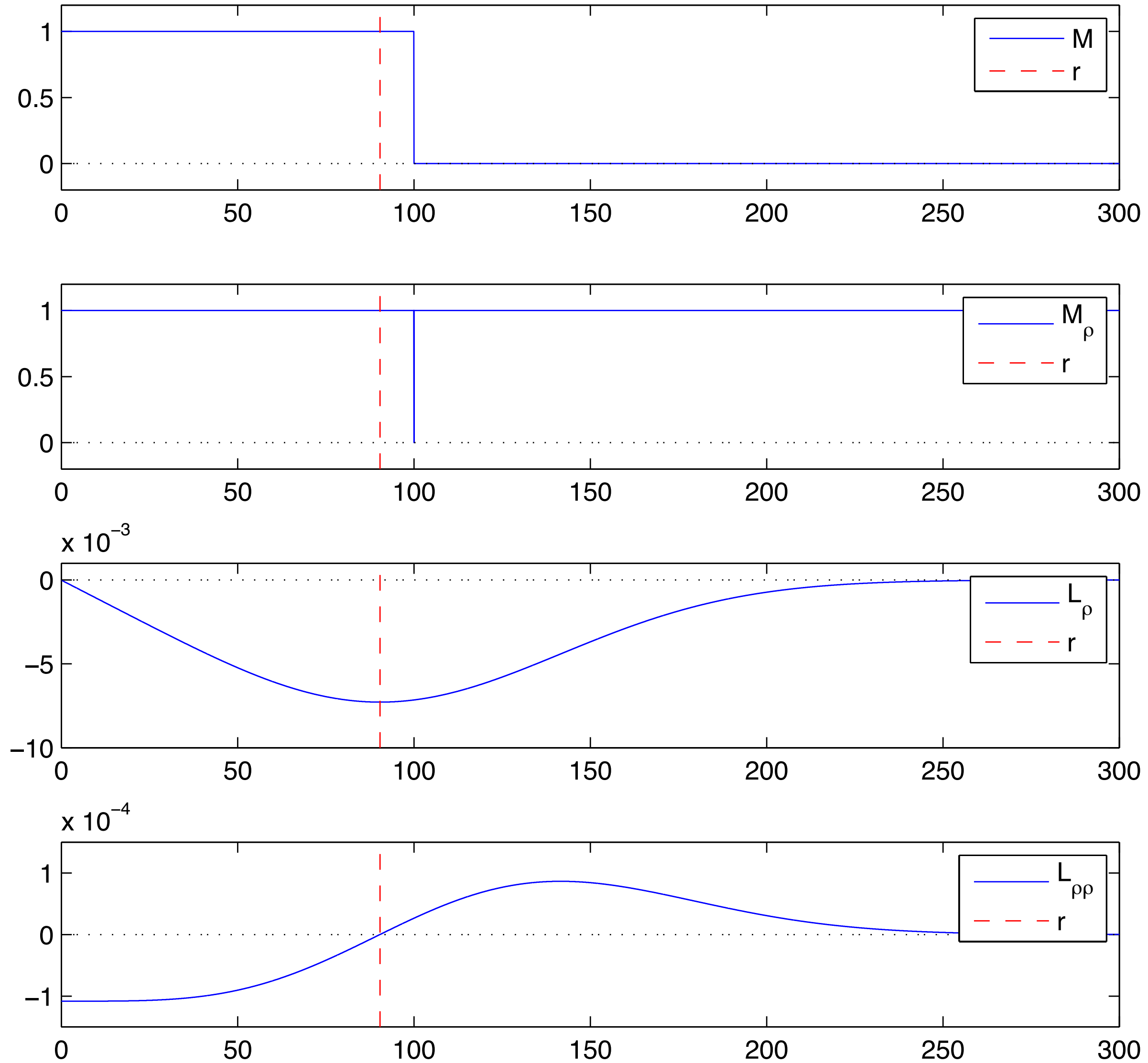


- When x is large, we make use of the fact:

$$\frac{I_1(x)}{I_0(x)} \approx \frac{128x^2 - 48x - 15}{128x^2 + 16x + 9}$$

Correction for a circle

- Example :
  - $R = 100$
  - $\sigma = 50$
  - $r = 90.42$



Correction for other shapes

- If the shape is not a circle, it is difficult to analytically determine the dislocation using equations of mathematical physics

- Thus we approximately make corrections according to <u>local curvature</u>

- Example – approximate correction for an ellipse
  - For an ellipse, we correct *a* and *b* for the curvature at $\theta = k\pi/2$
  - Let the solution of the equation for a circle be $R = \Omega(r,\sigma)$

$$\frac{b'^2}{a'} = R_1 = \Omega\left(\frac{b^2}{a}, \sigma\right)$$

$$\frac{a'^2}{b'} = R_2 = \Omega\left(\frac{a^2}{b}, \sigma\right)$$

$$a' = \sqrt[3]{R_2^2 R_1}$$

$$b' = \sqrt[3]{R_1^2 R_2}$$

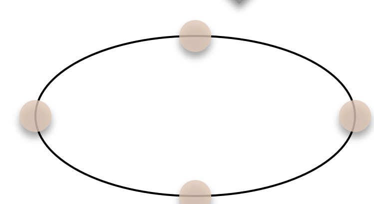
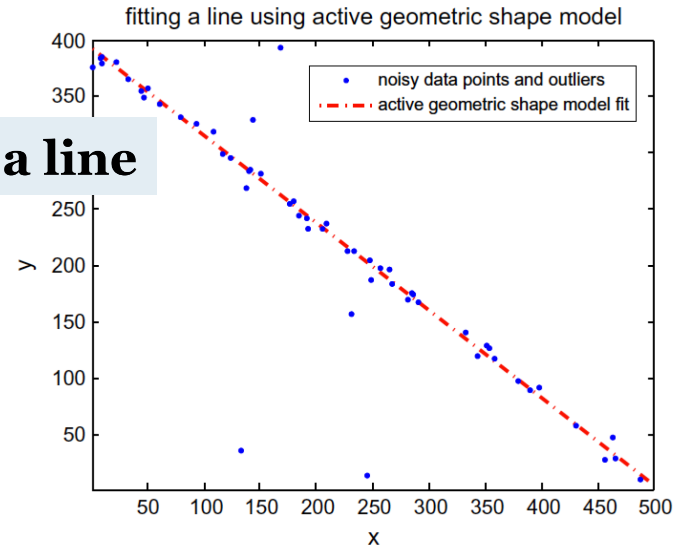


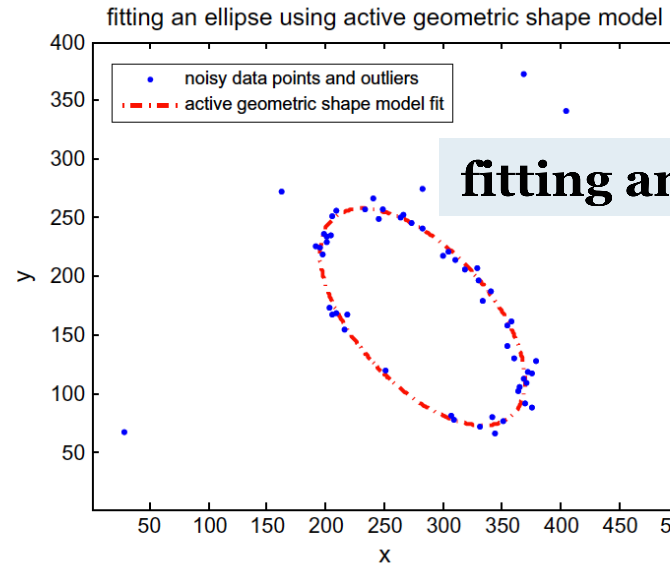
Fig. The 4 positions to be corrected.

Experiments on synthetic data

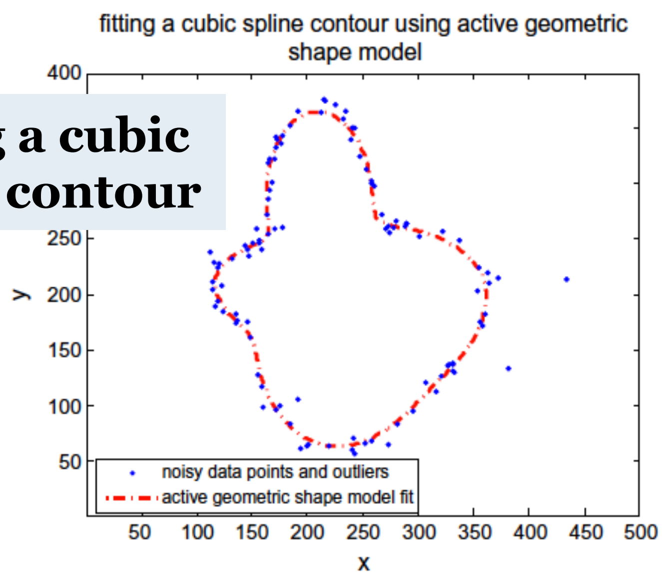
fitting a line



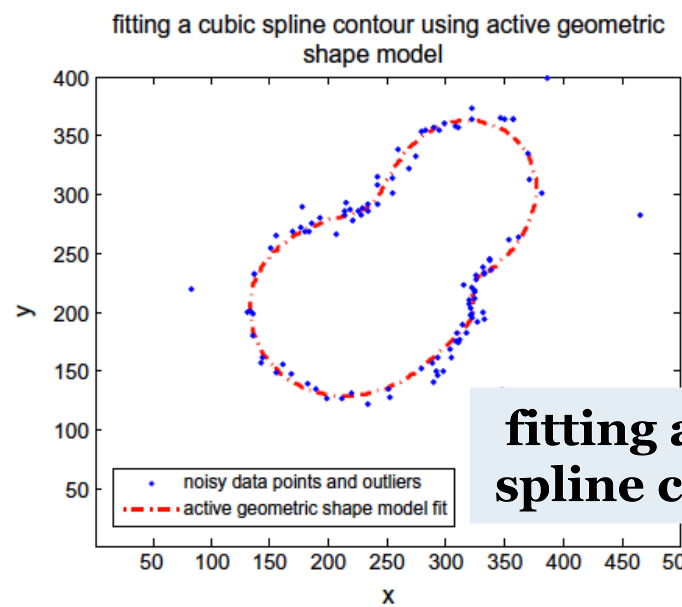
fitting an ellipse



fitting a cubic spline contour

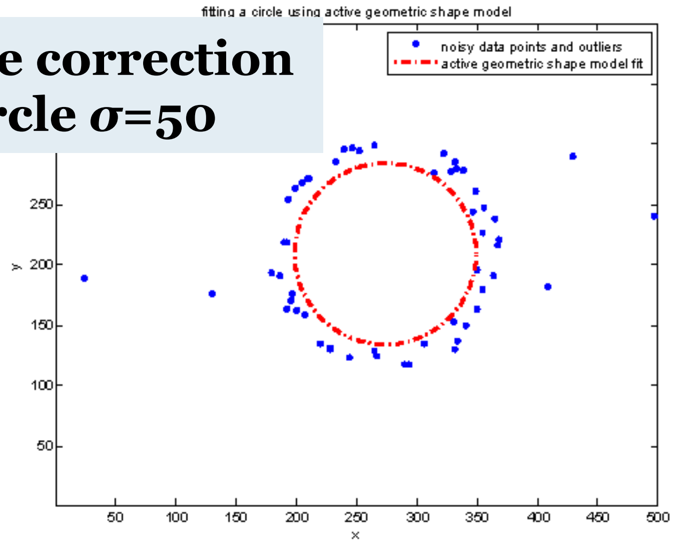


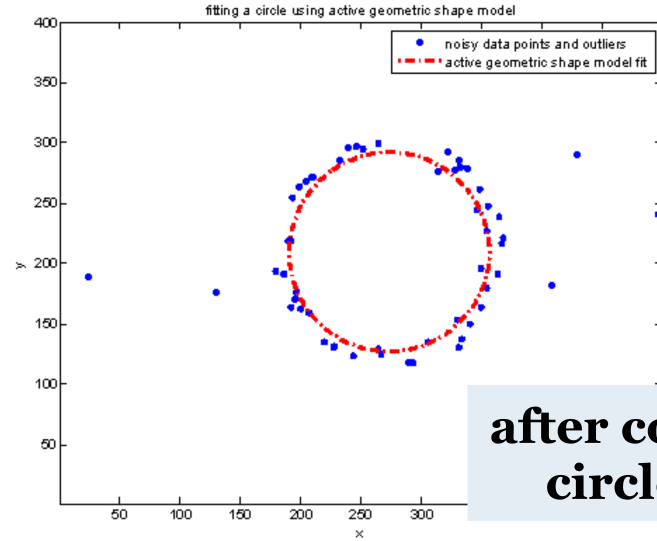
fitting a cubic spline contour



Before and after correction of curvature

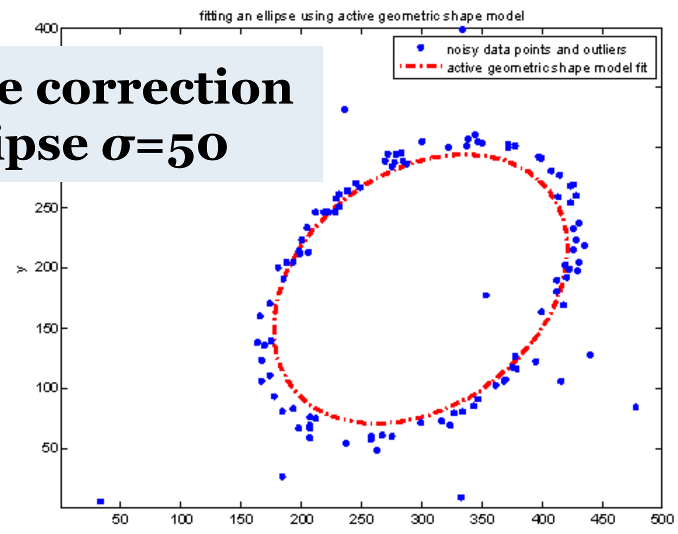
before correction
circle σ=50

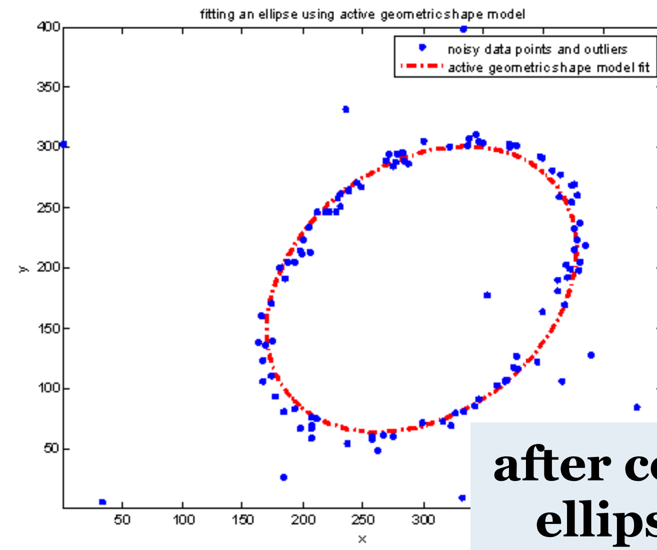




after correction
circle σ=50

before correction
ellipse σ=50

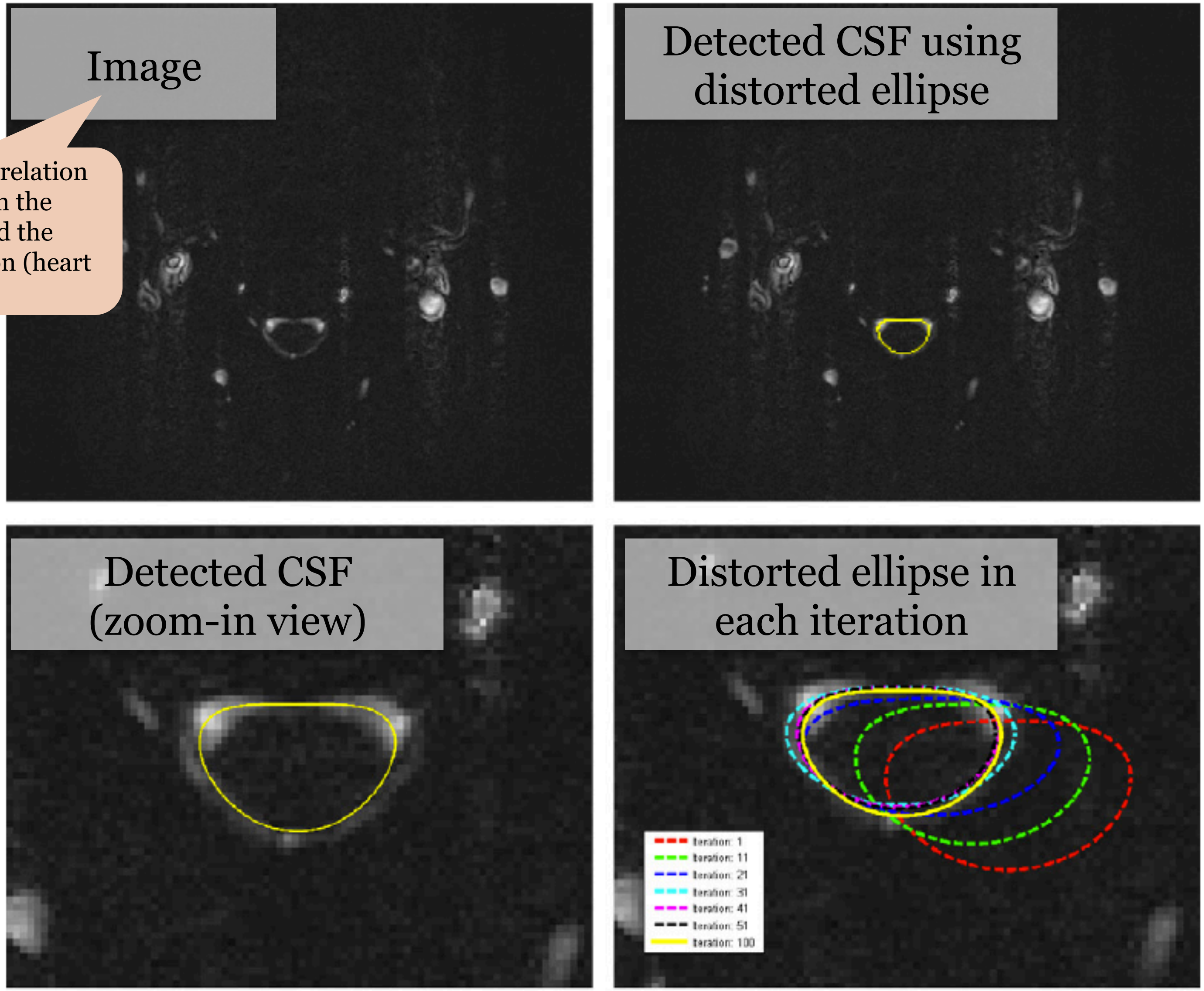




after correction
ellipse σ=50

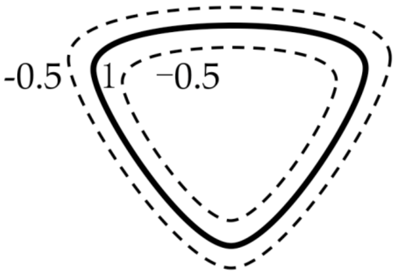
Experiments on PC-MR images

Actually the correlation map between the sequence and the sinusoid function (heart cycle)



Experiments on PC-MR images

- Goodness measurement
  - We generate 50 seed shapes to evolve, and select the best fit
  - Goodness is measured by



$$\mathcal{F}(\mathcal{P}) = \frac{1}{N}\sum_{i=1}^{N}\|\mathbf{F}(x_i, y_i)\| - \frac{1}{2N'}\sum_{i=1}^{N'}\|\mathbf{F}(x'_i, y'_i)\| - \frac{1}{2N''}\sum_{i=1}^{N''}\|\mathbf{F}(x''_i, y''_i)\|$$

Current shape | Shrunken shape | Expanded shape

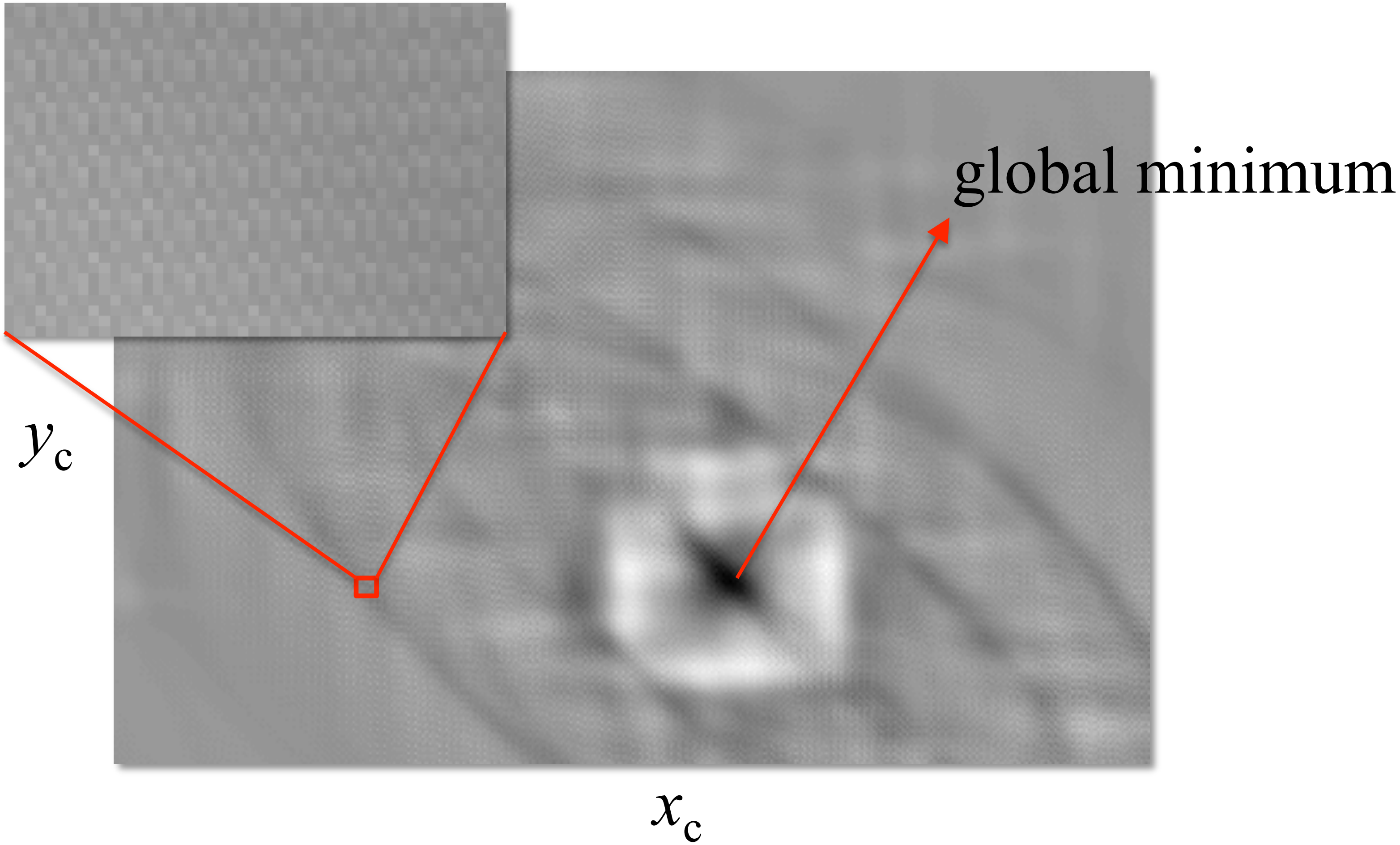
- CSF segmentation
  - Detection + Graph cuts → Segmentation
  - We have achieved a mean Dice similarity coefficient (DSC) of 86.4% on our dataset (unsupervised!)

Difficulties of non-heuristic methods

- Our AGSM method is heuristic (inspired by physics)
- AGSM iteratively adjust parameters
- Question: Can we directly minimize the fitness function using gradient descent or genetic algorithms?
- Answer: It sounds feasible. But actually the fitness function:
    - Is not continuous
    - Is non-convex
    - Has local minimums almost everywhere
    - Is slow to compute (render three shapes)

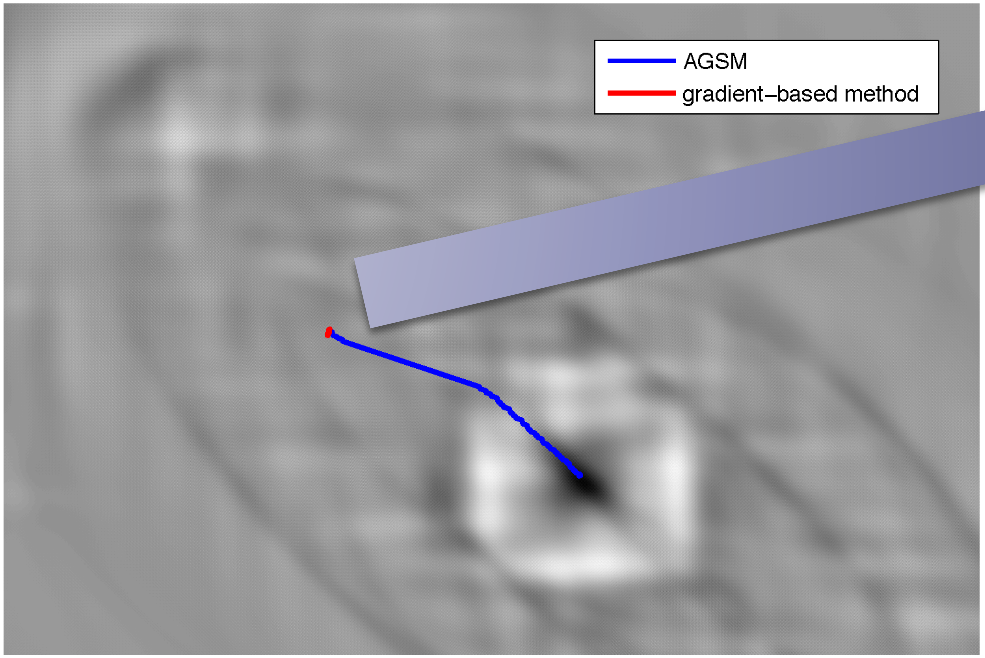
Fitness function of ellipse

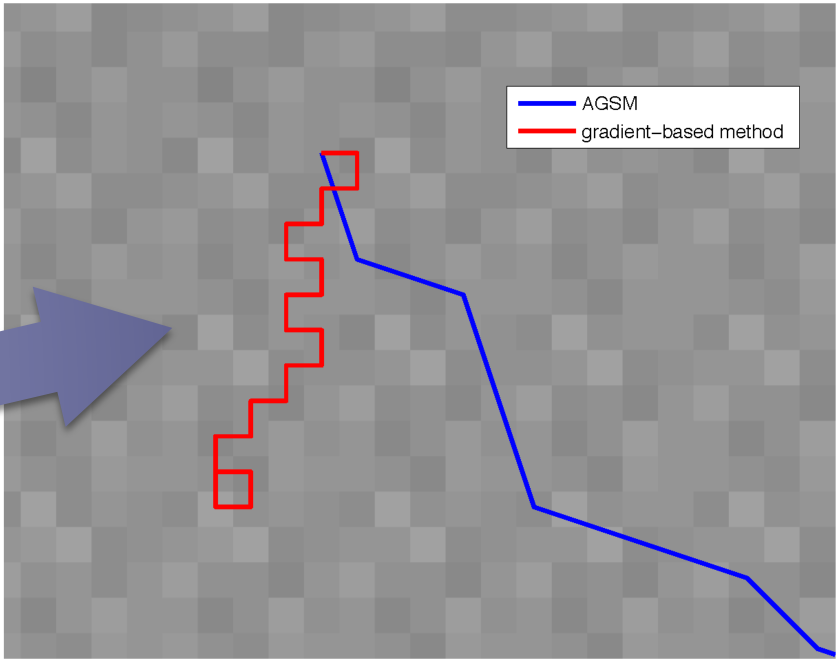
- If we know the ground truths of a, b and φ
- The fitness function with respect to x_c and y_c:



Fitness function of ellipse

- The solution paths of AGSM and gradient-based method on the fitness map

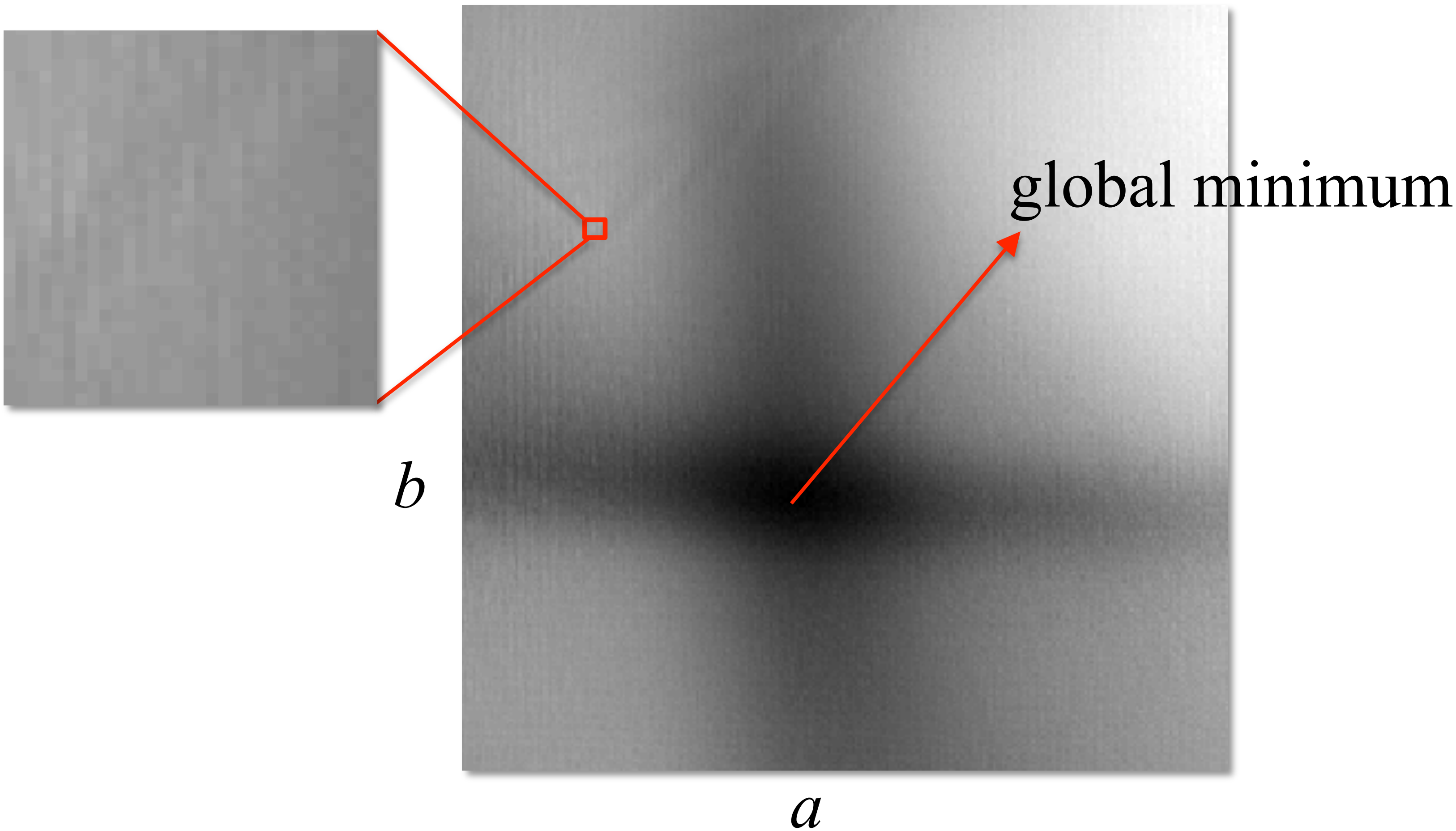




The zoom-in view
around the initial solution

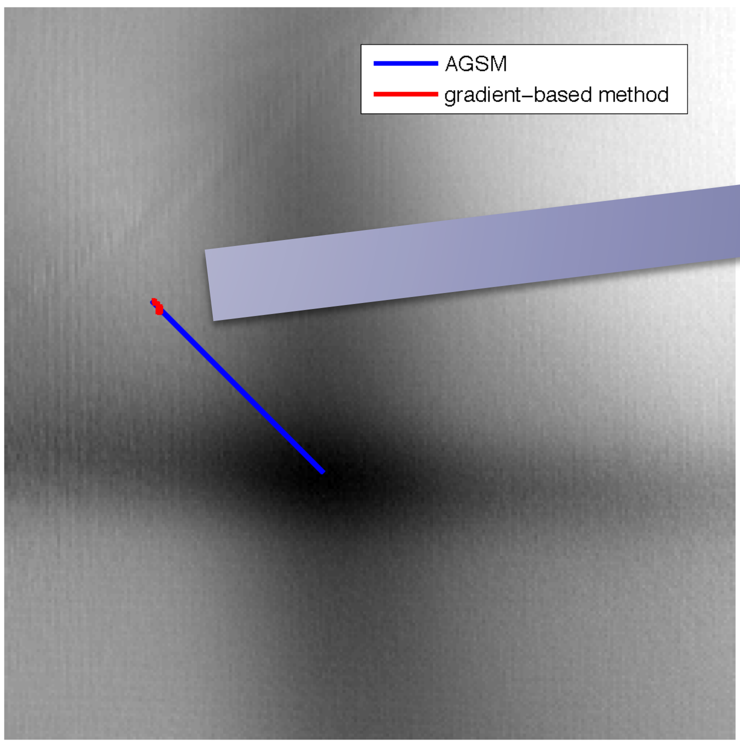
Fitness function of ellipse

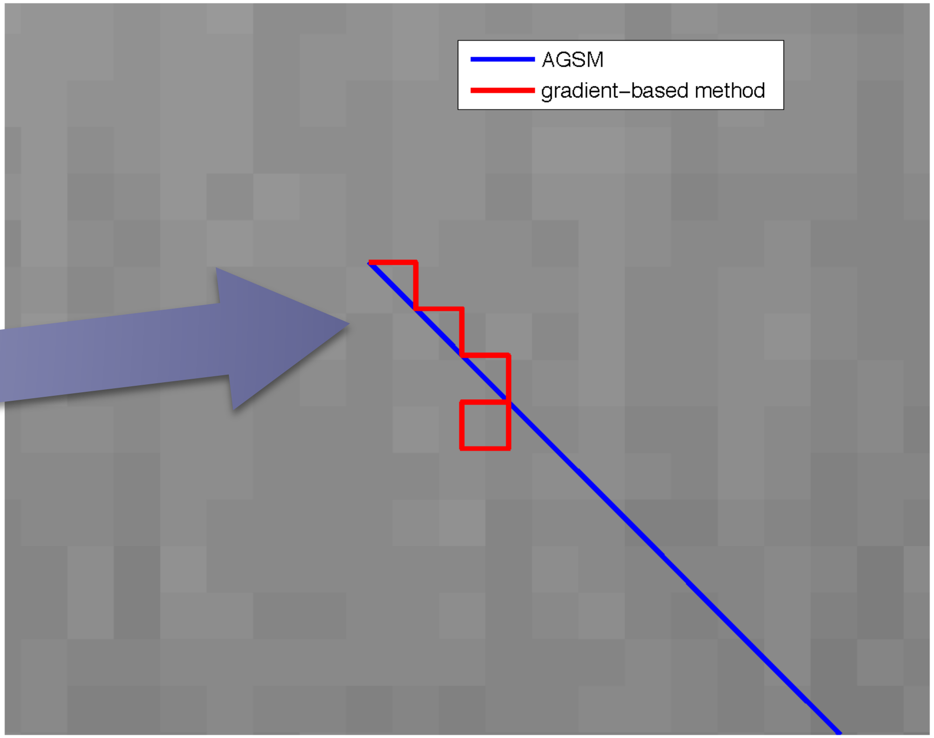
- If we know the ground truths of x_c, y_c and φ
- The fitness function with respect to a and b:



Fitness function of ellipse

- The solution paths of AGSM and gradient-based method on the fitness map





The zoom-in view
around the initial solution

Conclusion

- Our active geometric shape model (AGSM) is a novel and powerful approach to fit a geometric shape to image

- This model is validated on both synthetic data and PC-MR image sequences

- These slides are only a quick view of the work. For more technical details (some are very important) and more experiments, please look at our CVIU paper, and check our website:
  - https://sites.google.com/site/agsmwiki/

The active geometric shape model: A new robust deformable shape model and its applications

Computer Vision and Image Understanding, December 2012

Quan Wang, Kim L. Boyer

Signal Analysis and Machine Perception Laboratory

Department of Electrical, Computer, and Systems Engineering

Rensselaer Polytechnic Institute