MEDICAL IMAGE ANALYSIS

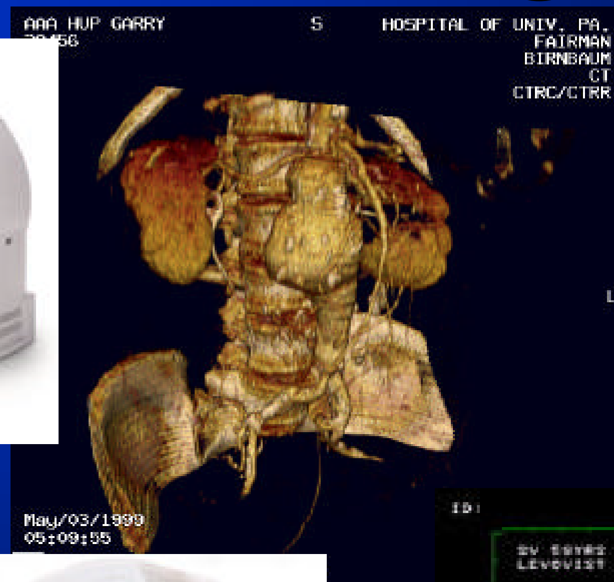
Marek Brejl

Vital Images, Inc.

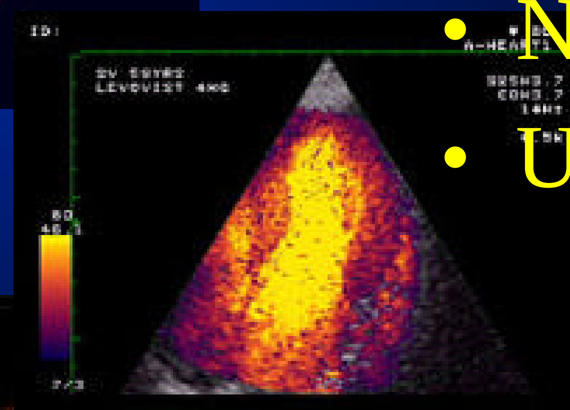
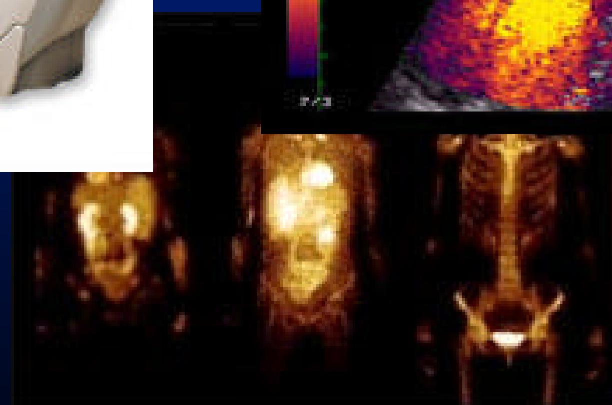
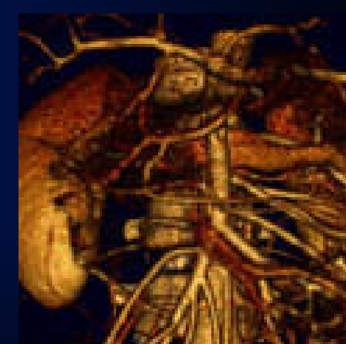
Outline

- Brief introduction to Medical Image Analysis
- Recent development in:
  - Model based detection and segmentation
  - Automated training/model generation

Medical Image Data

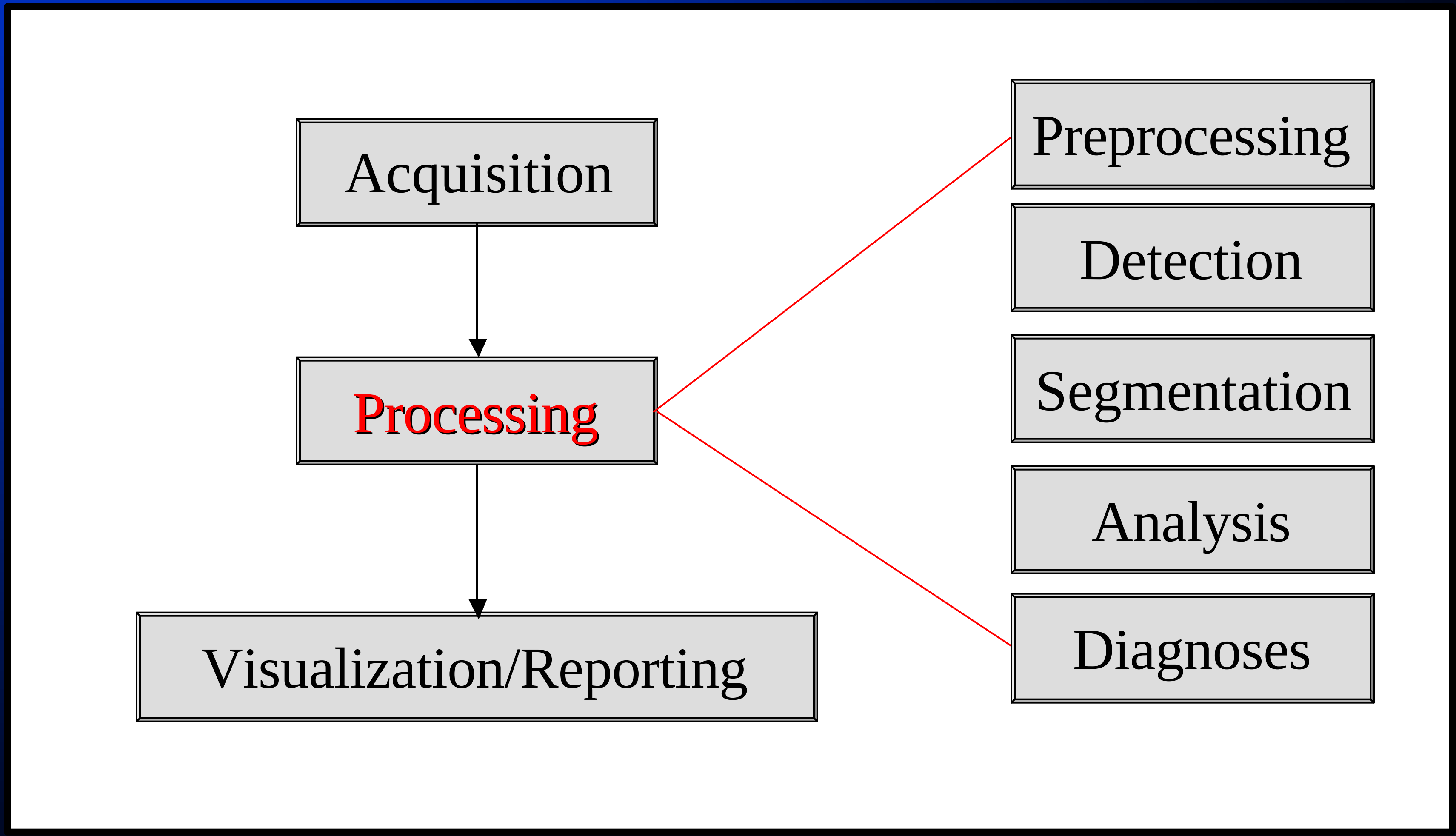





- CT
- MR
- X-ray
- Nuclear
- Ultrasound

Image Processing in Medical Settings

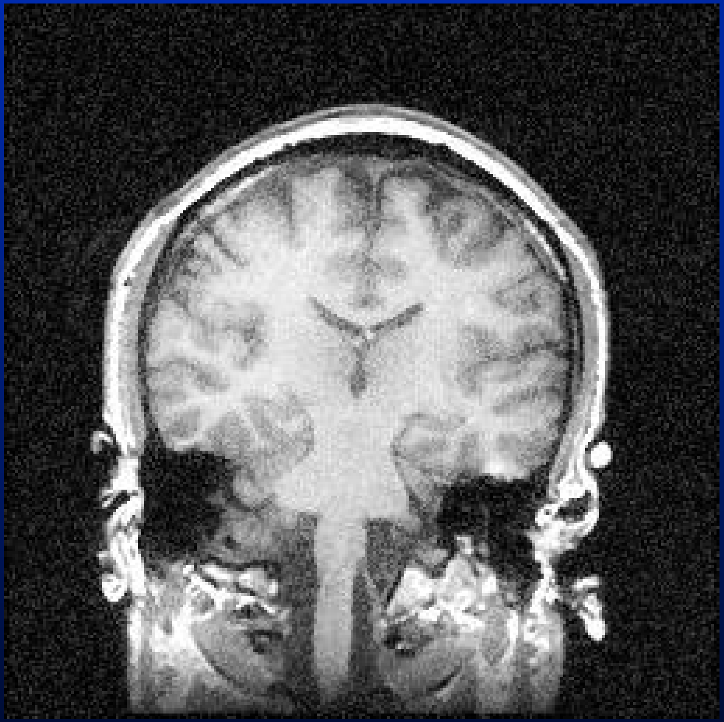


Data Processing

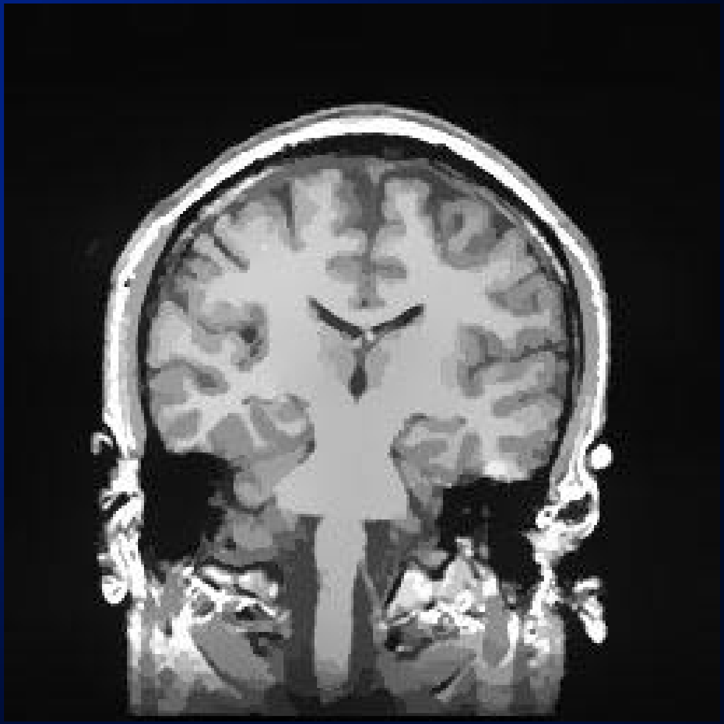
1. Preprocessing
   - Filtering, registration
2. Detection
   - Finding objects (nodules, polyps, organs)
3. Segmentation
   - Exact delimitation of objects
4. Analysis
   - Measurement (volume, curvature); Functional Imaging (Perfusion)
5. Classification/diagnoses

Examples: Preprocessing
Filtering/image enhancement

Original

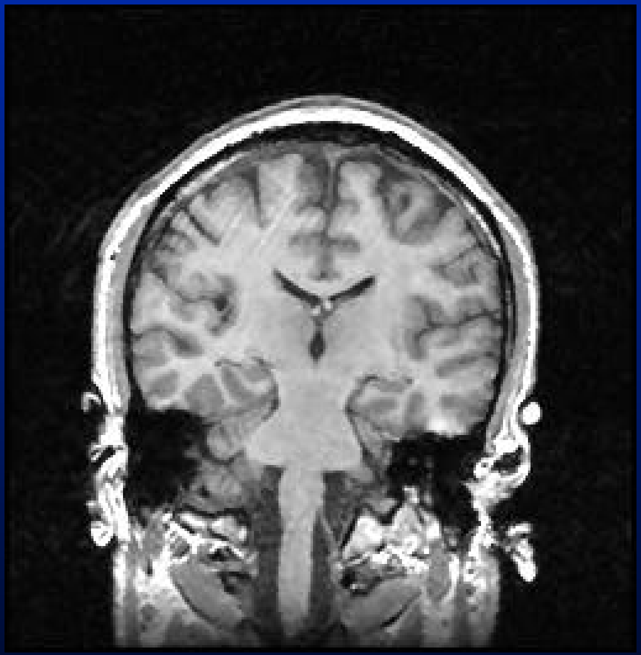


Enhanced

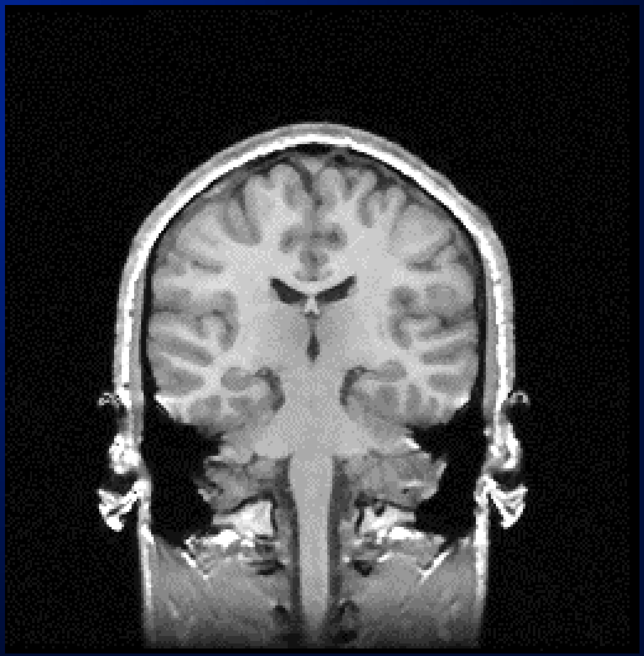


Examples: Preprocessing Registration

Target



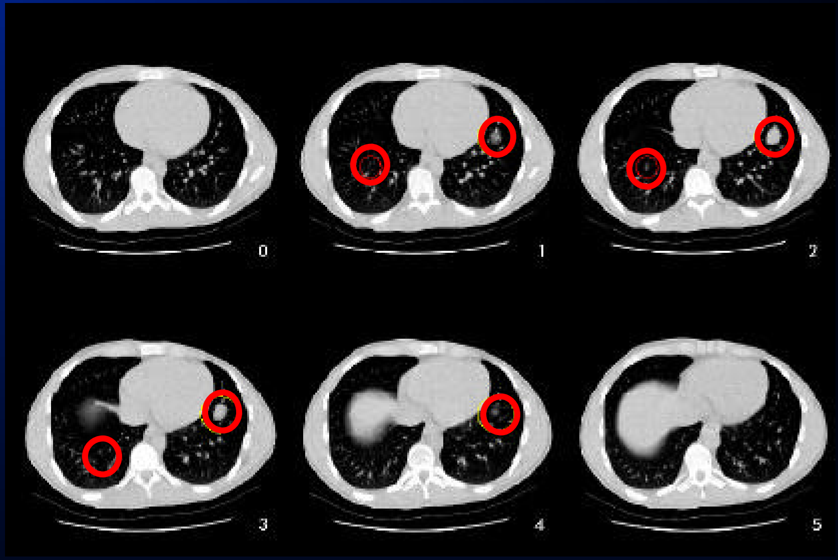
Template registration



Detection

- Find location of objects of interest

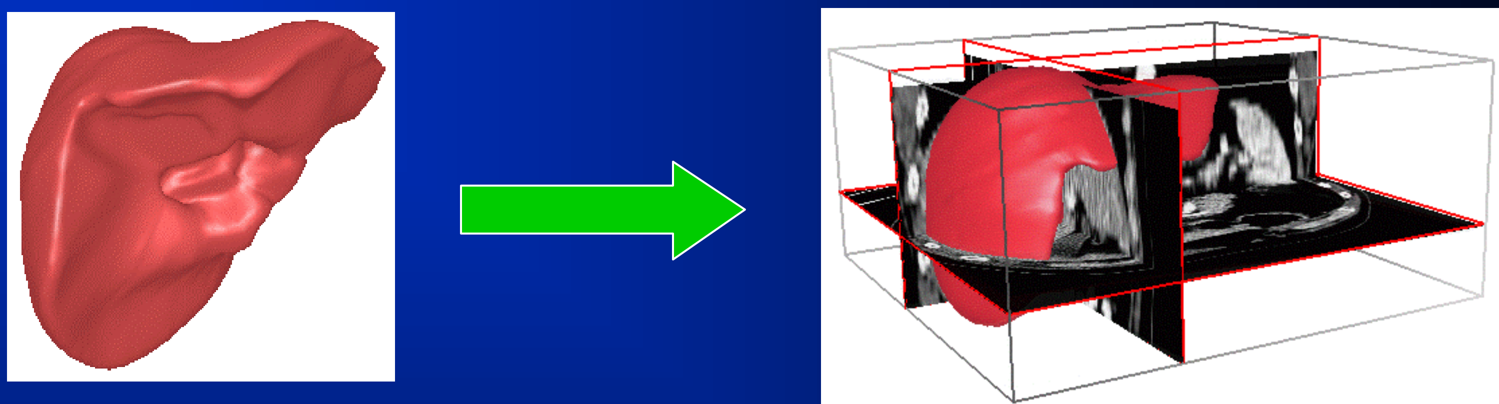
  (find or detect objects without prior knowledge about their location/existence)
  - Bones
  - Organs
  - Polyps in colon
  - Nodules in lungs

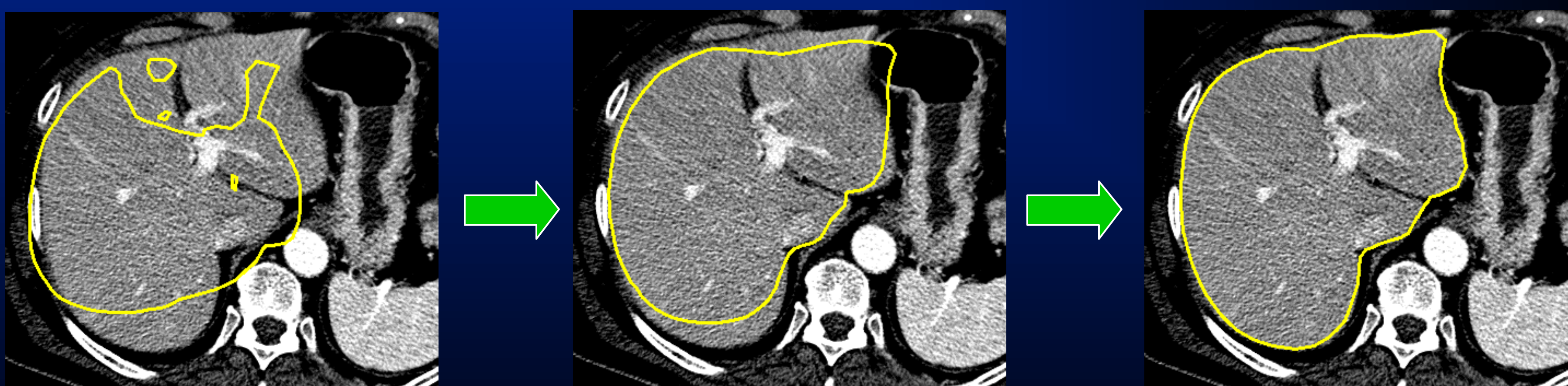


Segmentation

- Exactly delimitate objects, once they are detected (found)
  - Any object of predictable shape (organs, bones, vessel segments)
  - Liver
  - Cardiac imaging (left ventricle)
  - Brain

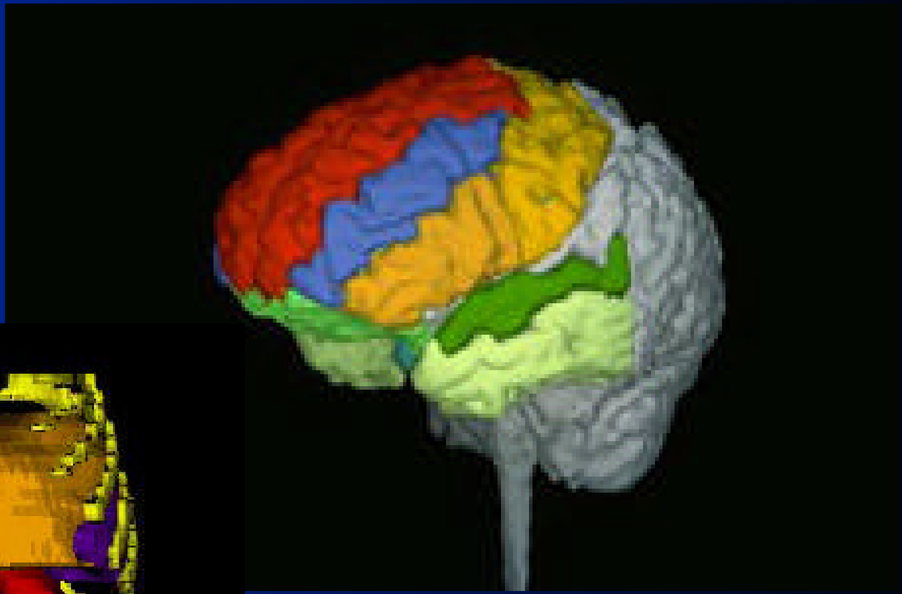
Segmentation

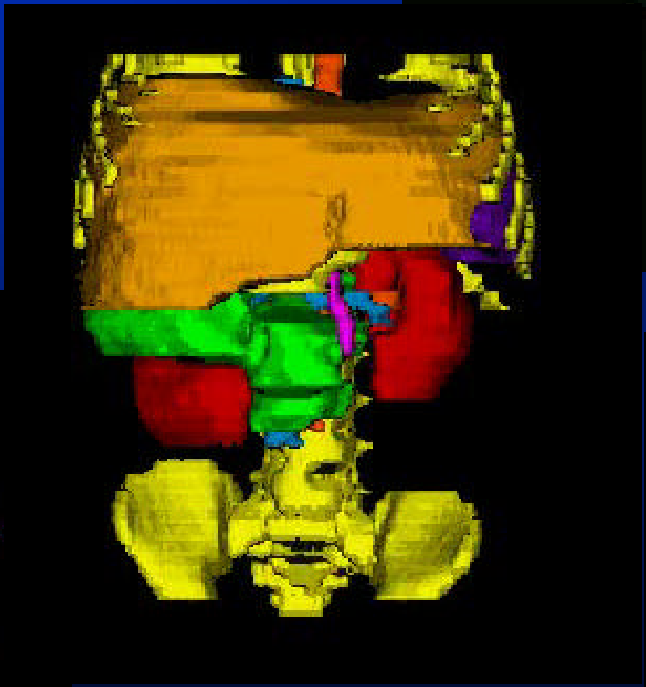


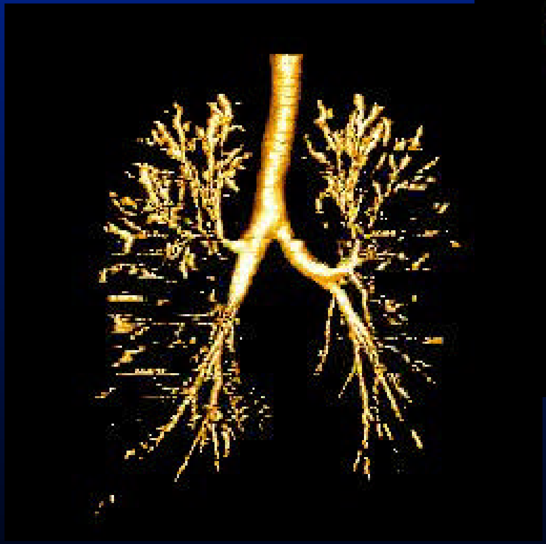


Segmentation

- Brain segmentation
- Heart
- Lung
- Body

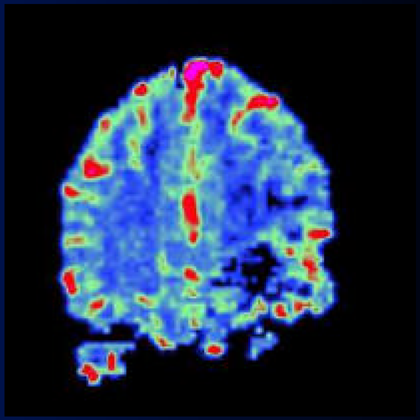


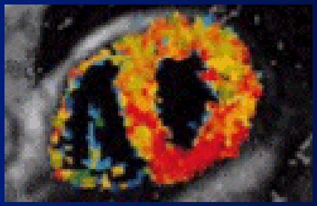


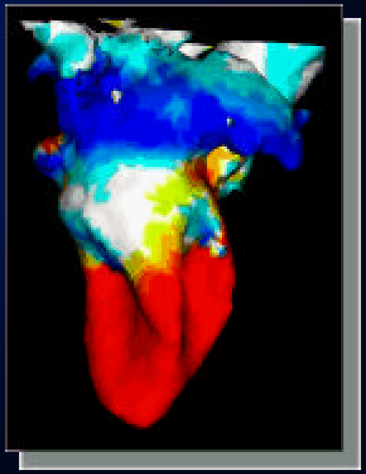


Analyses

- Measurement
  - Volume - growth rate
  - Vessel stenosis
- Functional imaging
  - Stroke
  - Cardiac perfusion
  - Tumor perfusion
- Cardiac function
  - LV motion
  - Injection fraction



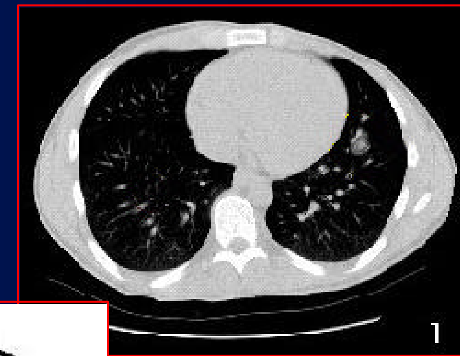


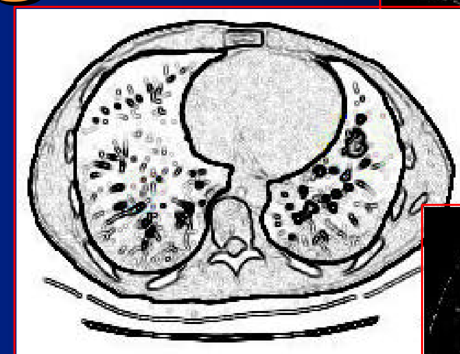


Classification/Diagnoses

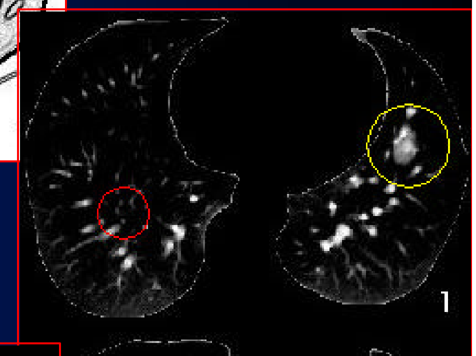
- Comparison to developed atlases
- Use of knowledge databases
  - Classify as normal/abnormal (brain structure)
  - Classify lung nodules as benign/malignant
  - Determine cancer/non-cancer

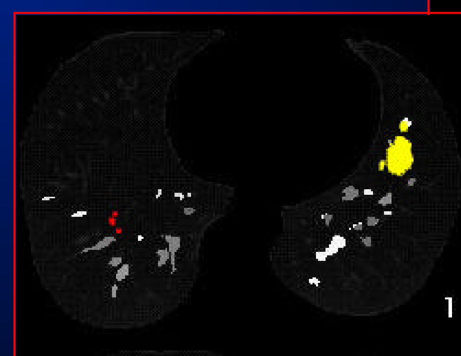
Data Processing – Example
Lung screening





1. Preprocessing
   - Reduce noise, threshold image
2. Detection
   - Find lungs, find nodules location
3. Segmentation
   - Accurately segment out nodules
4. Analysis
   - Measure volume, texture, curvature
5. Classification/diagnoses
   - Classify as benign or malignant





$V=35mm^3$
$S=25mm^2$
$V=12mm^3$
$S=5mm^2$

- benign
- malignant

Model-based detection and segmentation

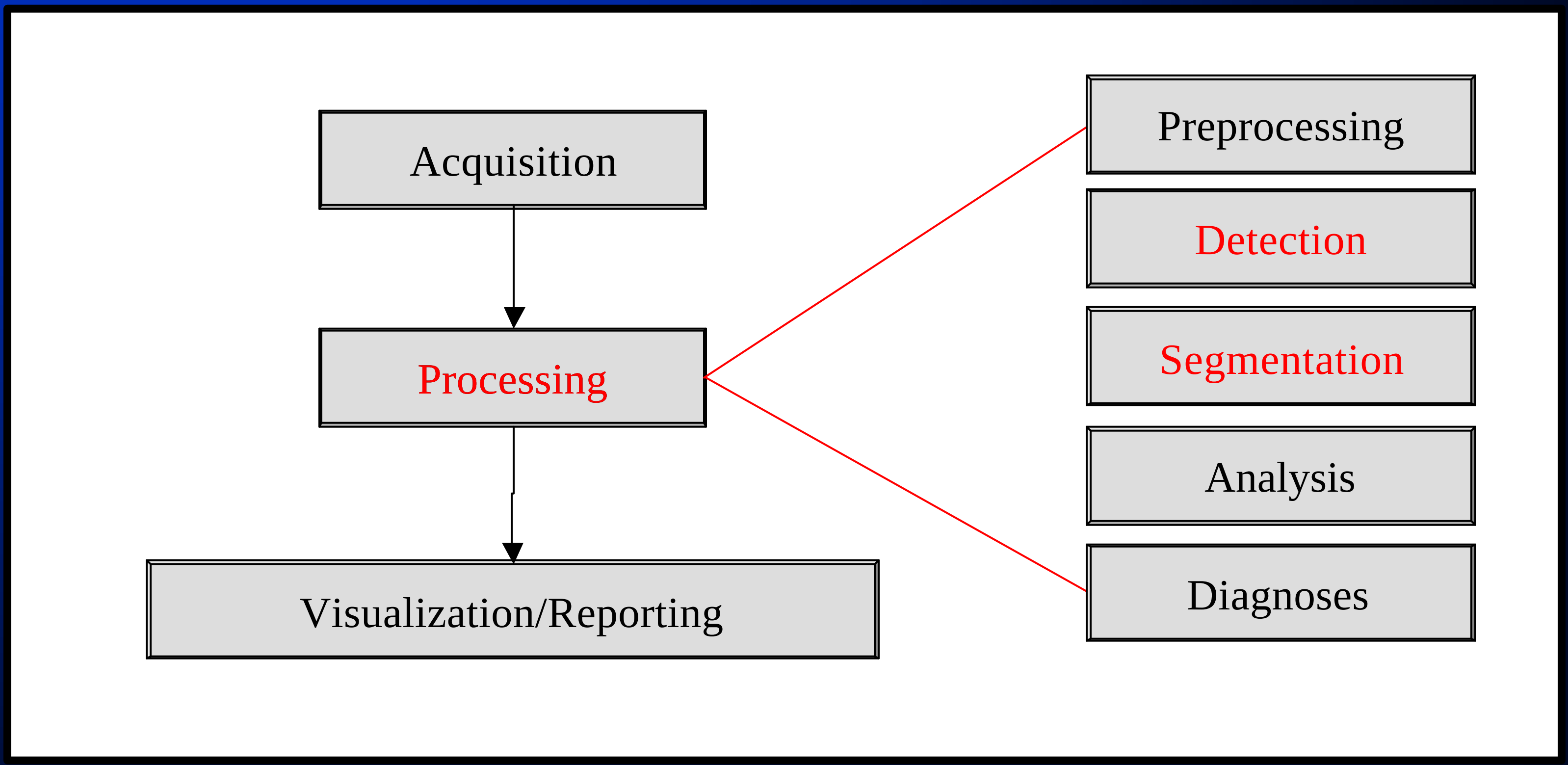
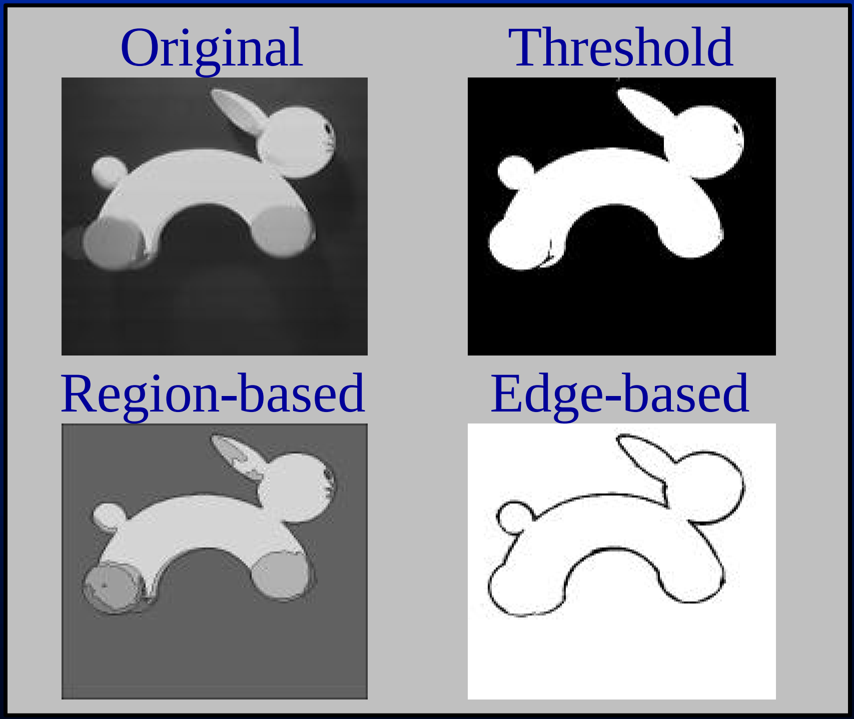
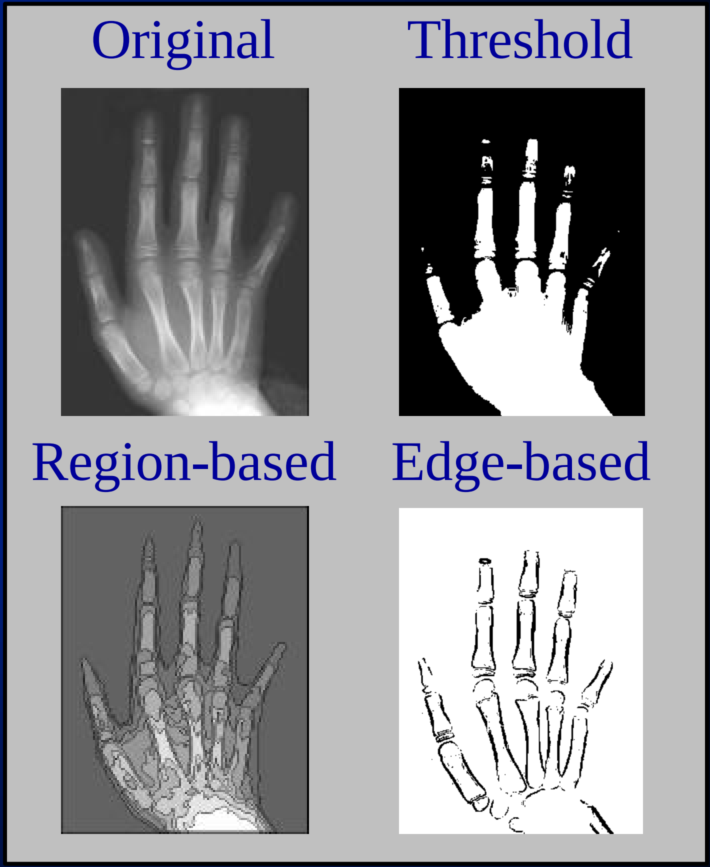


Image Segmentation

1. Thresholding
2. Region-based
3. Edge-based





Advanced (edge-based)

Advanced Image Segmentation

1. ***Segmentation criterion*** design

$$E(v(s)) = \int \left(E_I(v(s)) + E_E(v(s))\right) ds$$

$$C(P) = \sum_i c(x_i) + \sum_i w(x_i, x_j)$$

$$C(g,h) = \int_\Omega |T(h(x)) - S(x)|^2 \, dx + \int_\Omega |S(g(x)) - T(x)|^2 \, dx$$

2. ***Segmentation criterion*** optimization
   - graph searching
   - gradient descent

Goals

Design of methodology for automated model-based image segmentation (segmentation via boundary detection)

Model-Based Segmentation

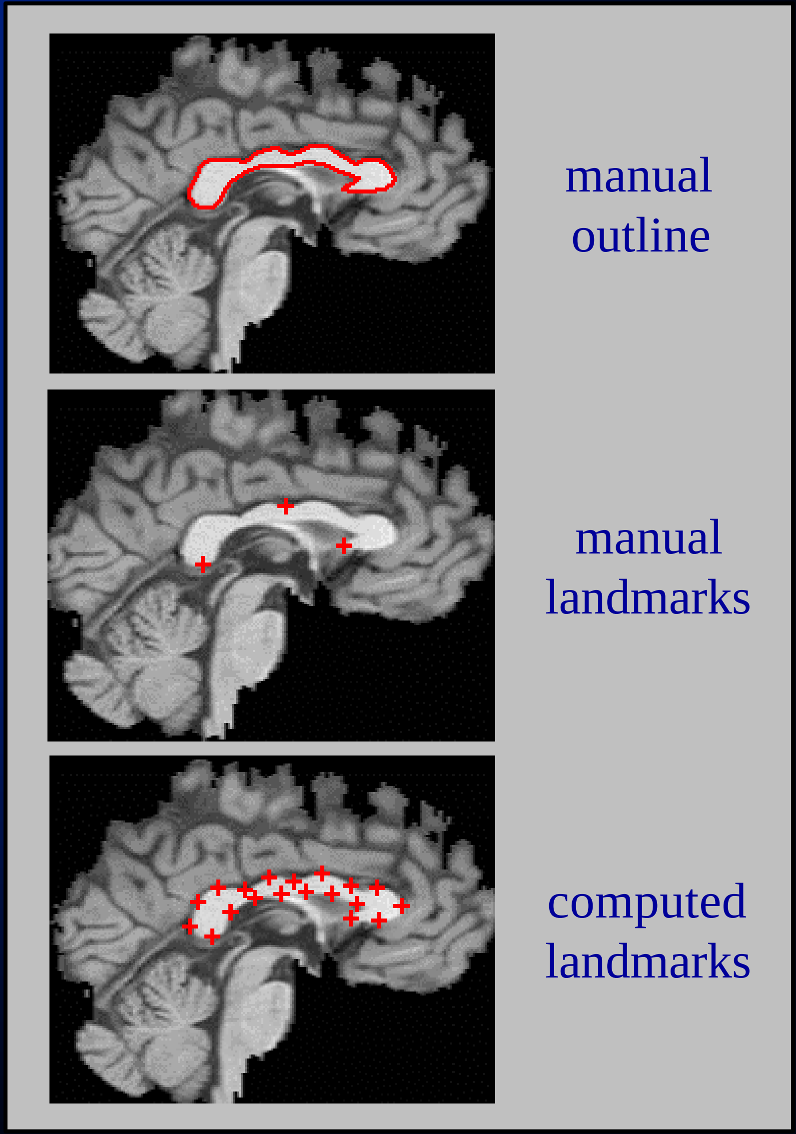
1. Training set design
2. Training
   a. Shape Model
   b. Border Appearance Model
3. Segmentation

   Step 1: Approximate object location

   Step 2: Accurate boundary detection

Training Set Design

- Selection of examples
  - Contains expected variability in local border appearance
  - Contains expected variability in object shape
- Segmentation examples
  - Outlined objects
  - Registration landmarks



Training

- Purpose:
  - To create representation of the objects of interest presented in the training set
- Statistical models:

  (*mean, variance*)
  - Shape Model
  - Border Appearance Model

Automated Segmentation

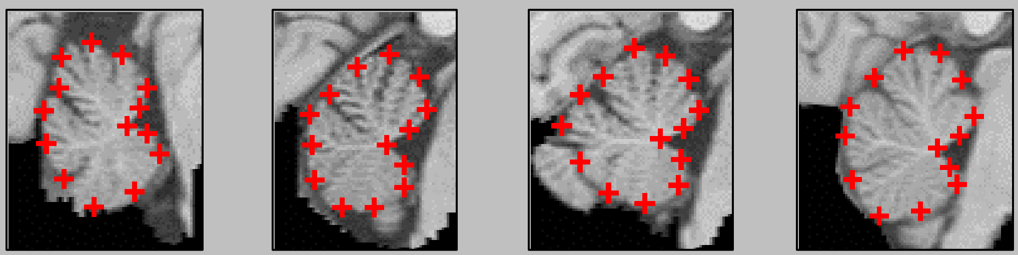
1. Training set design
2. Training
   a. Shape Model
   b. Border Appearance Model
3. Segmentation
   Step 1: Approximate object location
   Step 2: Accurate boundary detection

Shape Model

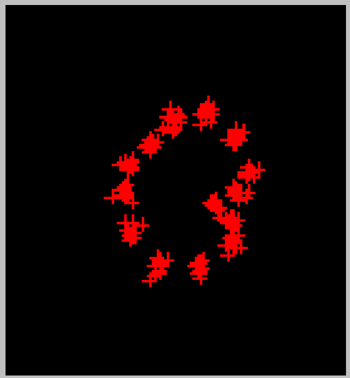
Point Distribution Model (PDM)

(Cootes *et al.*, 1992)

Example training shapes (N=12)



Alligned shapes



<u>Automated Segmentation</u>

1. Training set design
2. Training
   a. Shape Model
   b. Border Appearance Model
3. Segmentation
   Step 1: Approximate object location
   Step 2: Accurate boundary detection

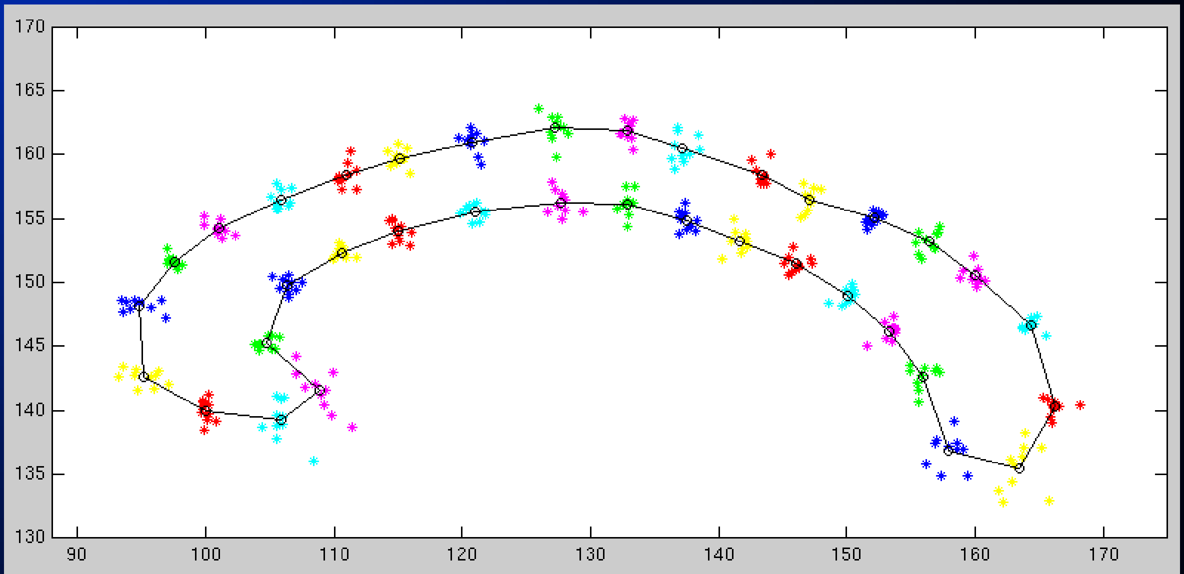
Shape Model

Point Distribution Model

- Object representation:

$$v_i = (x_{i0}, y_{i0}, x_{i1}, y_{i1}, \ldots, x_{in-1}, y_{in-1}) \qquad i \in \{1, \ldots M\}$$

- Shape Model:

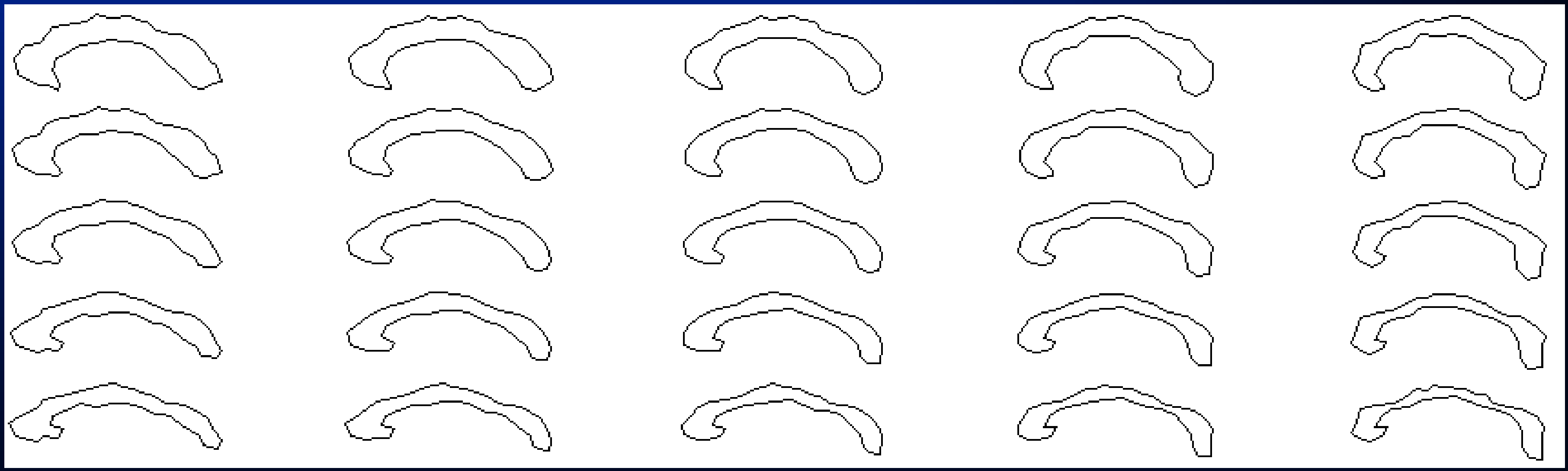


$$\bar{v} = \frac{1}{M} \sum_{i=1}^{M} v_i$$

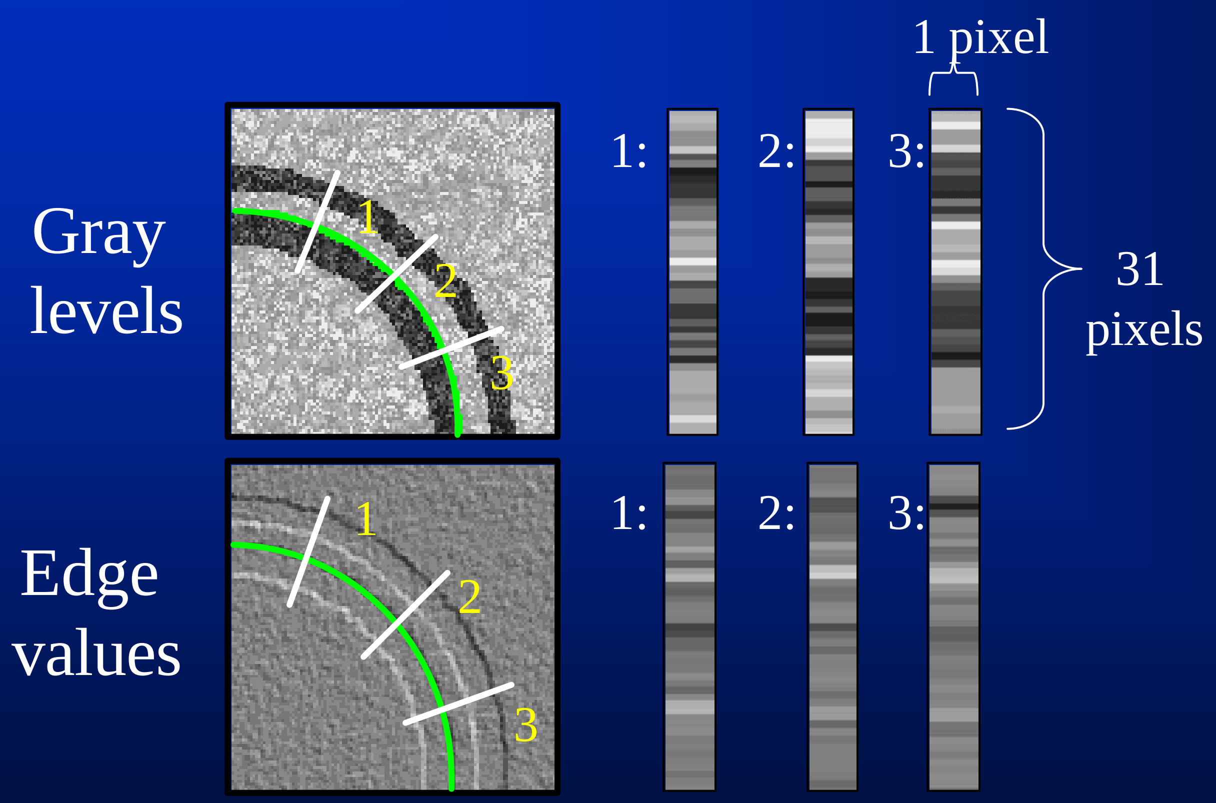
$$S = \frac{1}{M} \sum_{i=1}^{M} (v_i - \bar{v})(v_i - \bar{v})^T$$

Model Size Reduction

- Eigenvalues and eigenvectors of the variance matrix
  - Eigenvectors – directions of the main modes of variation
  - Eigenvalues – importance of the variation
- Size reduction – keep only most important modes



Border Appearance Model

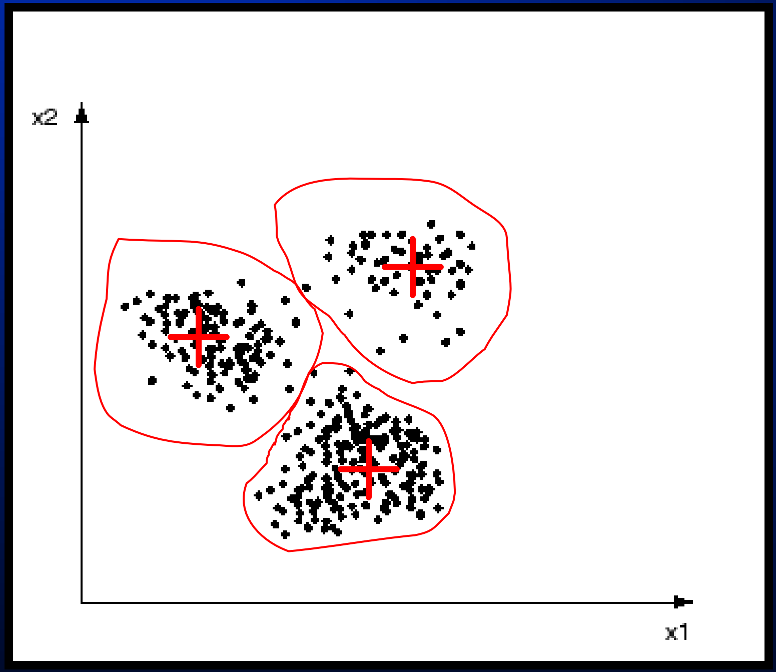


<u>Automated Segmentation</u>

1. Training set design
2. Training
   a. Shape Model
   b. Border Appearance Model
3. Segmentation
   Step 1: Approximate object location
   Step 2: Accurate boundary detection

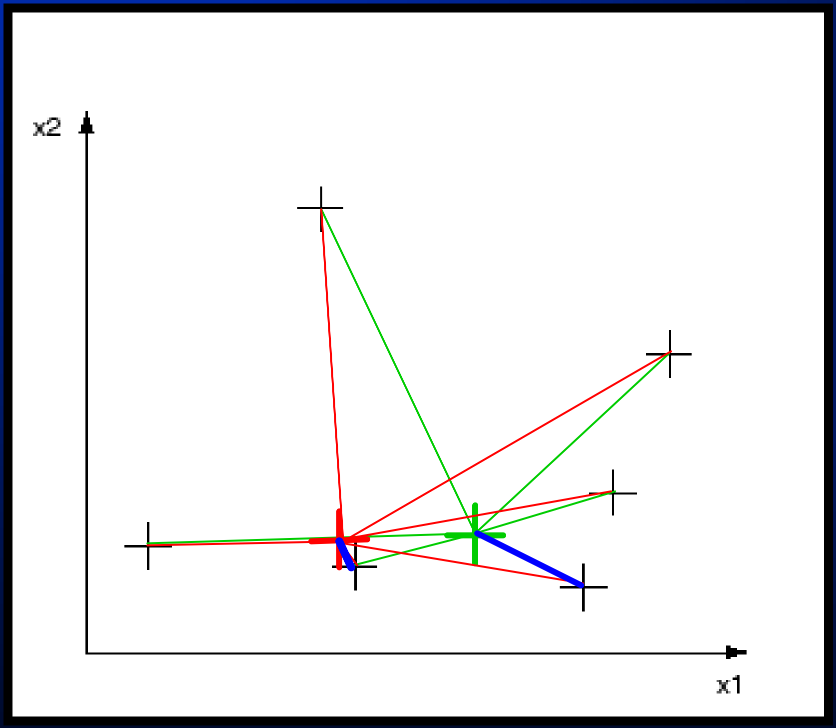
BAM – Basic Idea

- BA - represented by <u>*appearance vector*</u>
- Training vectors grouped to create the model



BAM – Basic Idea

- Comparison of image data to the BA model



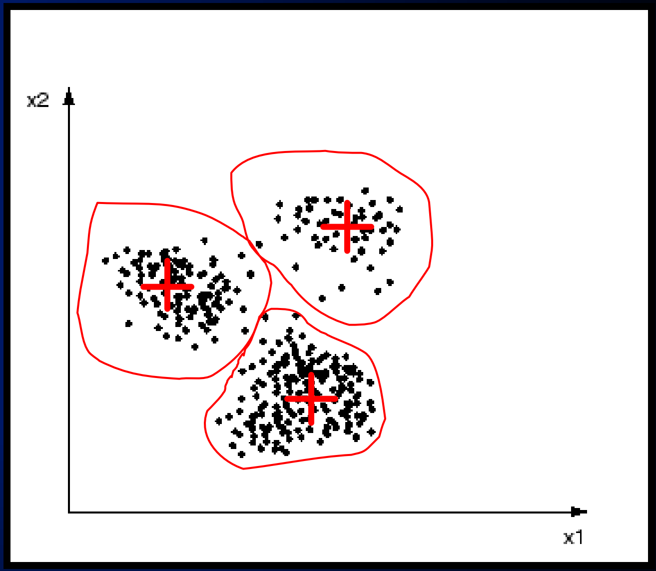
BAM - Clustering

Fuzzy *c-means* clustering

(Bezdek *et al.*,1981)

- Input vectors: f_i
- Partitioning matrix: U
- Cluster centers: $\bar{f}$
- Objective function to be minimized:

$$J(U,\bar{f}) = \sum_i \sum_j u_{ij}{}^m \left\| f_i - \bar{f}_j \right\|^2$$



fit Value Computation

- Fuzzy model

$$fit_k(f) = \left[\sum_l \left(\frac{\|f - \bar{f}_k\|^2}{\|f - \bar{f}_l\|^2} \right)^{\frac{1}{m-1}} \right]^{-1}$$

- Euclidean distance

$$fit_k(f) = F\left\{ \|f - \bar{f}_k\|^2 \right\}$$

- Gaussian model

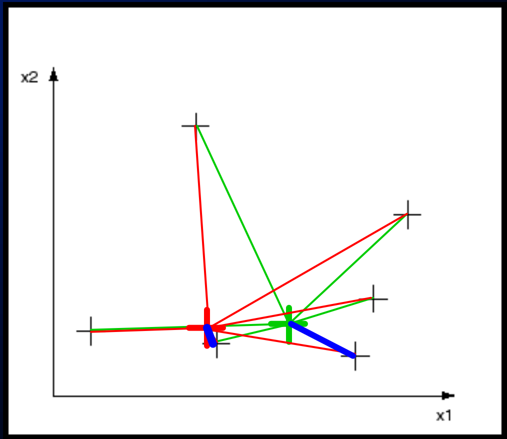
$$\boxed{fit_k(f) = F\left\{ (f - \bar{f}_k)^T S_{\bar{f}_k}^{-1} (f - \bar{f}_k) \right\}}$$

F performs some inversion

$$F(x) = MAX - x$$

Best fit:

$$\boxed{fit(f) = \max_k fit_k(f)}$$



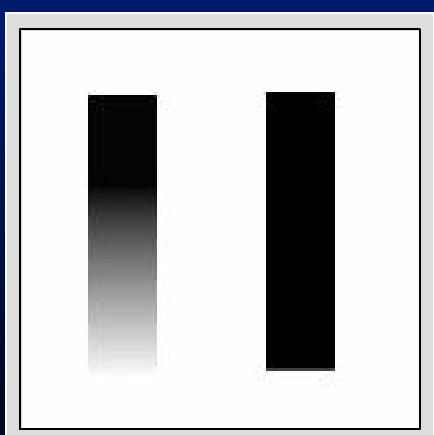
Border Appearance Model

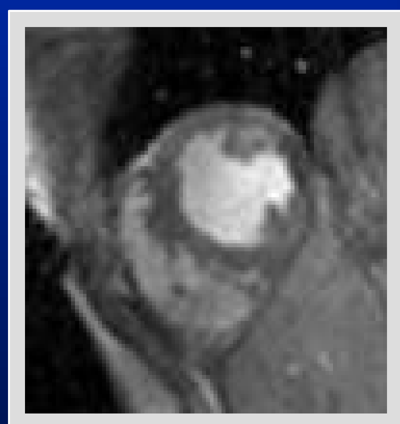
- BAM – tool for:
  - Modeling local border properties
  - Comparing image data to the model
  - Computing cost values based on the comparison
- Further improvements
  - Spatial information
  - Counter-examples

Automated Segmentation
1. Training set design
2. Training
   a. Shape Model
   b. Border Appearance Model
3. Segmentation
   Step 1: Approximate object location
   Step 2: Accurate boundary detection

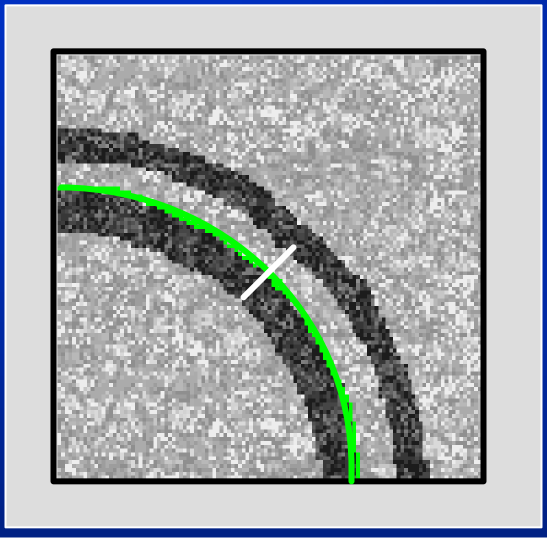
BAM – Spatial Information

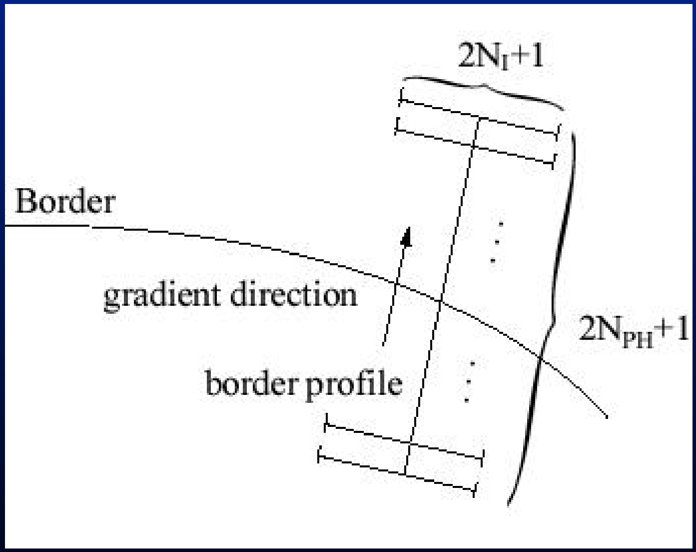
- BAM records:
  - Appearance vectors
  - Location of every vector on the original border

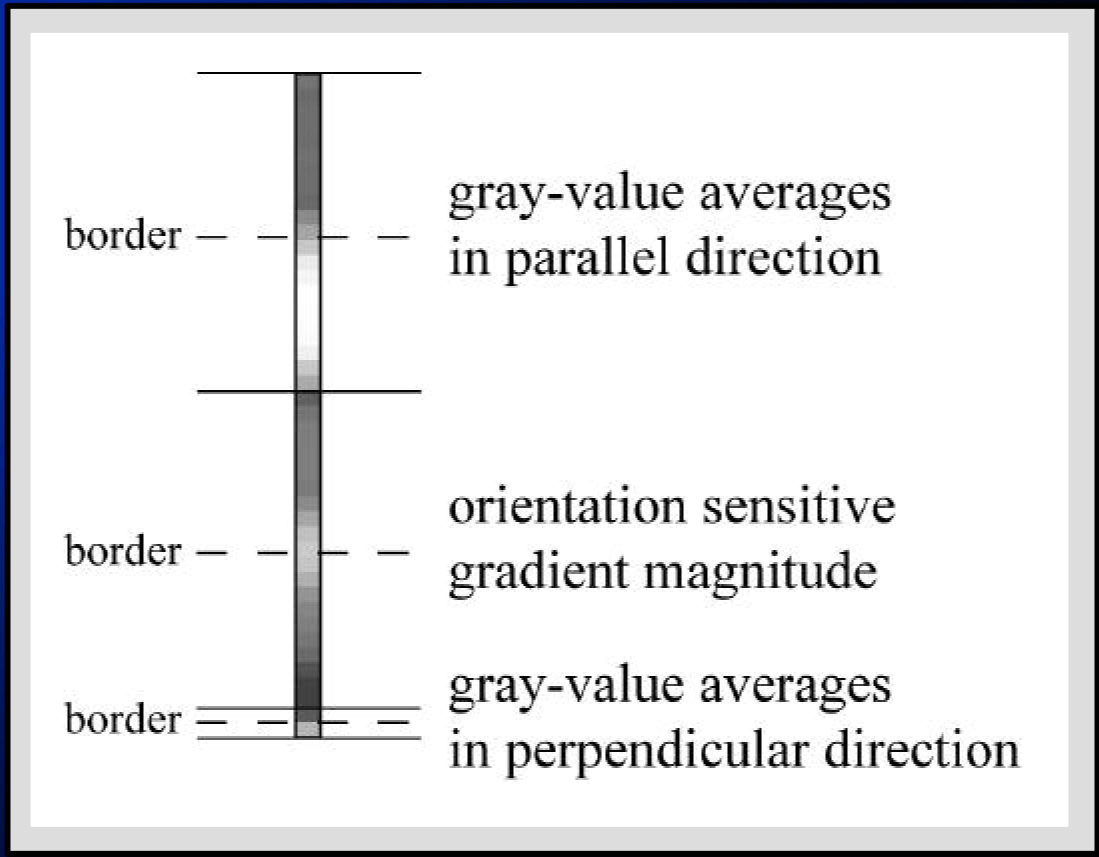




Border Appearance Vector







Training - Summary

- Shape Model
- Border Appearance Model
  - border appearance
  - appearance spatial information

Automated Segmentation

1. Training set design
2. Training
   a. Shape Model
   b. Border Appearance Model
3. Segmentation
   Step 1: Approximate object location
   Step 2: Accurate boundary detection

Approximate Object Location

- Algorithm for detection of approximate object location
- Requirements:
  - detects objects defined by the models
  - detects objects modified by rigid transforms (rotation, translation, scaling)
  - Insensitive to shape variability (captured in the training set)

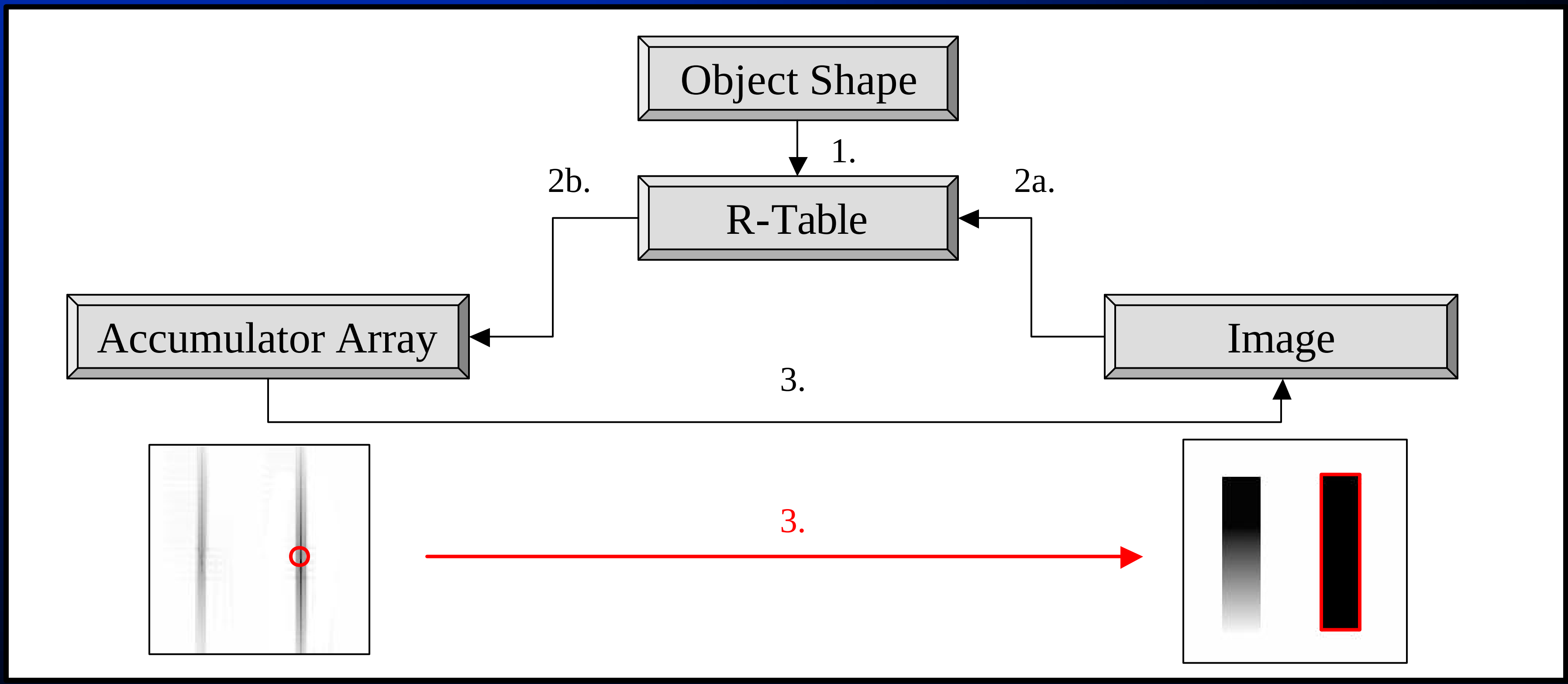
<u>Automated Segmentation</u>

1. Training set design
2. Training
   a. Shape Model
   b. Border Appearance Model
3. Segmentation
   Step 1: Approximate object location
   Step 2: Accurate boundary detection

Generalized Hough Transform

(Ballard, 1981)

- Detection of objects of arbitrary *a priori known* shapes



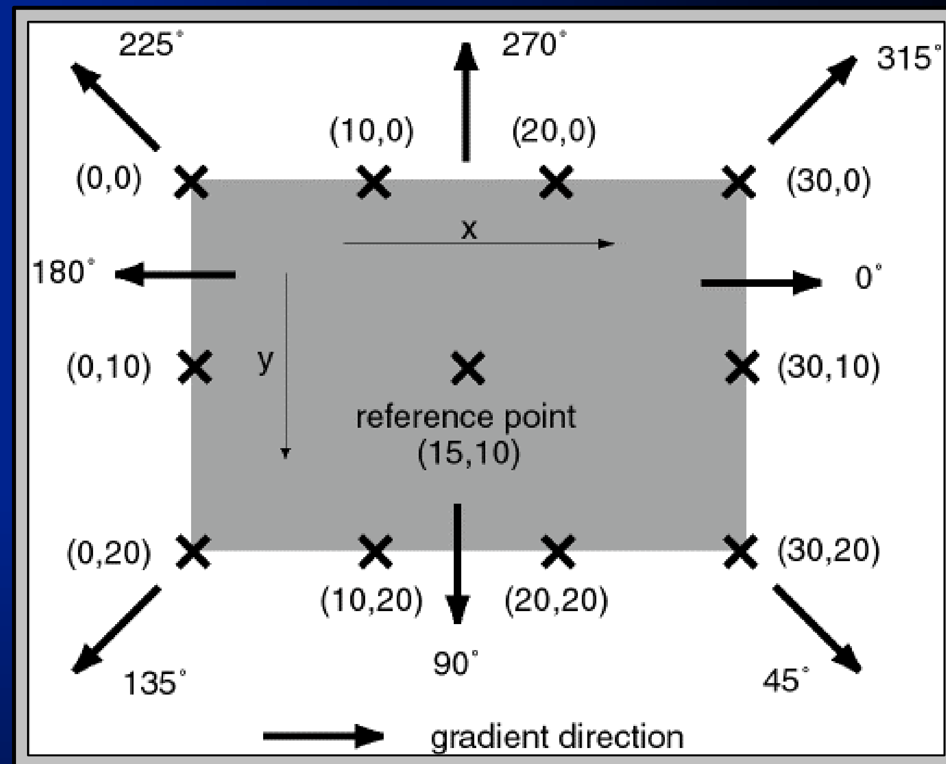
R-Table

1. R-Table construction:

$R =$

	a_{1x}	a_{1y}	a_{2x}	a_{2y}
$<0,45)$	15	0	–	–
$<45,90)$	15	10	–	–
$<90,135)$	5	10	−5	10
$<135,180)$	−15	10	–	–
$<180,225)$	−15	0	–	–
$<225,270)$	−15	−10	–	–
$<270,315)$	−5	−10	5	−10
$<315,360)$	15	−10	–	–

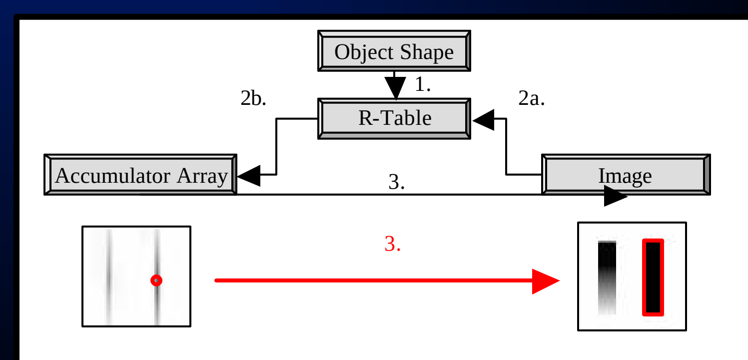
$$(a_x, a_y) = (x_i - x^R, x_i - y^R)$$



2. Accumulator array updates:

$$x^A = x + s(a_x \cos t - a_y \sin t)$$
$$y^A = y + s(a_y \cos t - a_x \sin t)$$

$$A(x^A, y^A, s, t) = A(x^A, y^A, s, t) + strength$$

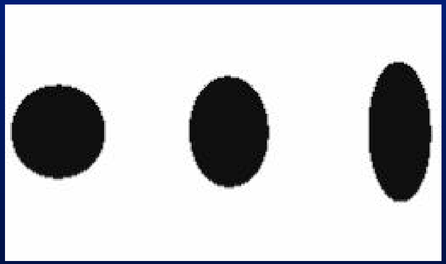


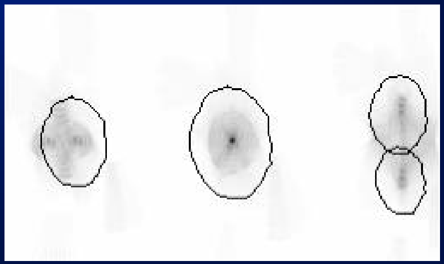
Vital Images, Inc. 2001

Problems of Generalized HT

- Searches for strong edges

- Cannot handle shape variance





- Remedy: Shape-Variant Hough Transform

Shape-variant Hough Transform

- Use of Border Appearance Model:

$$\boxed{A(x^A, y^A, s, t) = A(x^A, y^A, s, t) + strength} \longrightarrow \boxed{A(x^A, y^A, s, t) = A(x^A, y^A, s, t) + fit}$$

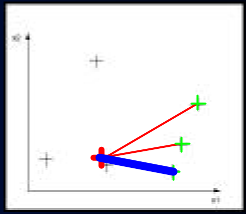
$$fit_k(f) = F\left\{(f - \bar{f}_k)^T S_{\bar{f}_k}^{-1}(f - \bar{f}_k)\right\}$$

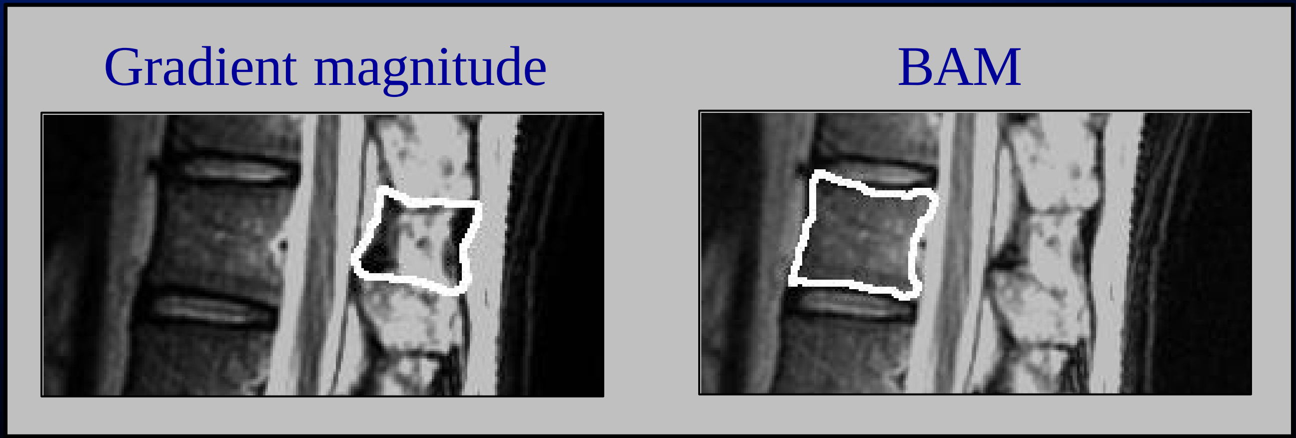
General

$$fit(f) = \max_k fit_k(f)$$

Spatial information

$$fit(f) = \max_{k \in P_{seg}} fit_k(f)$$

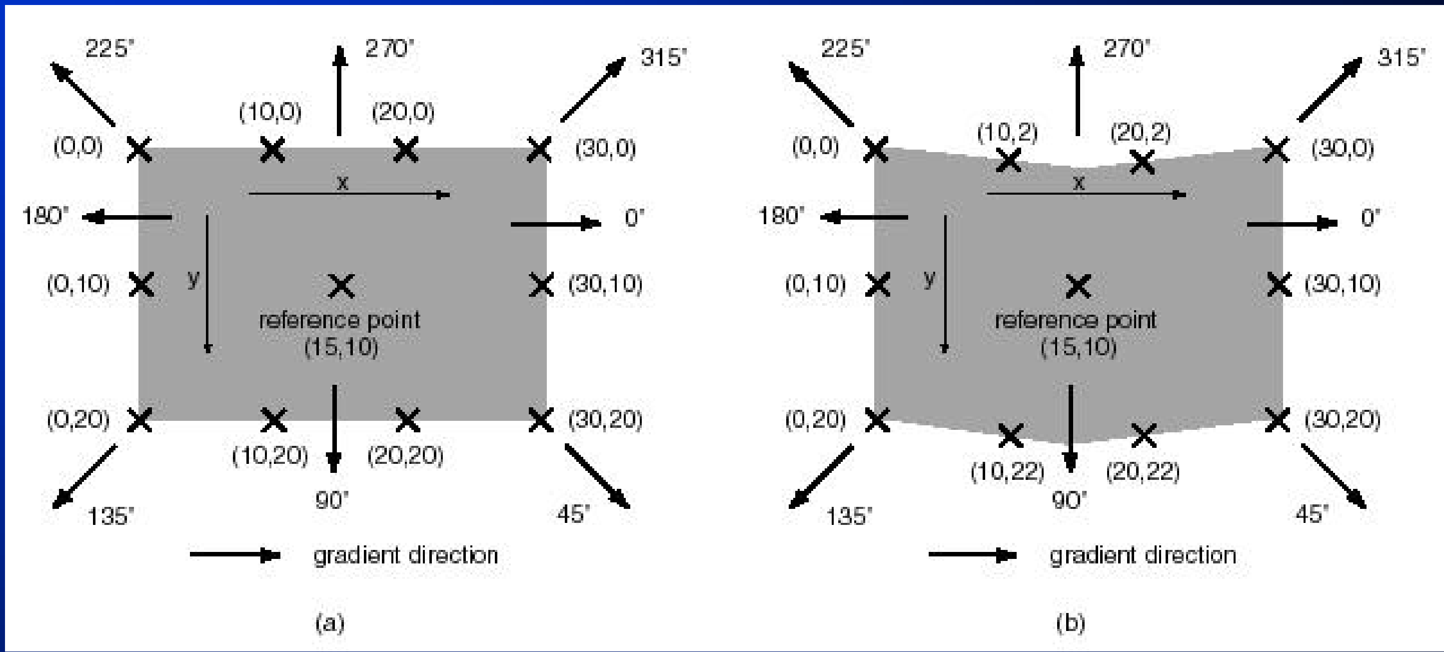




SVHT – Shape Variance

1. Alignment of training shapes
   - Same algorithm as for PDM
2. Encoding shape variance in the R-Table
   - All border directions are encoded
3. Shape reconstruction
   - Several shapes available for reconstruction
   - Selection of the most probable shape

SVHT – R-Table Construction



$R =$

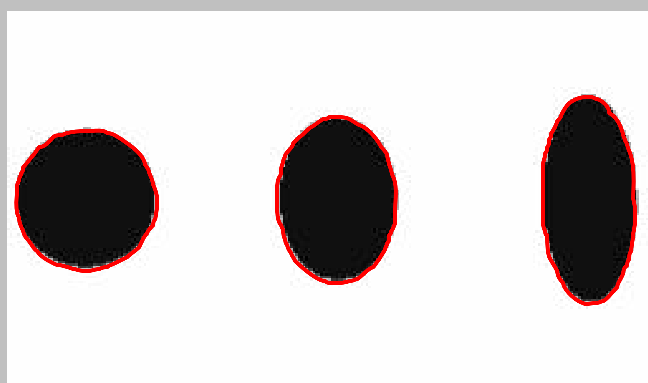
	a_{1x}	a_{1y}	a_{2x}	a_{2y}	a_{3x}	a_{3y}
$<0,45)$	$15^{a,b}$	$0^{a,b}$	$-$	$-$	$-$	$-$
$<45,90)$	$15^{a,b}$	$10^{a,b}$	5^{b}	12^{b}	$-$	$-$
$<90,135)$	5^{a}	10^{a}	-5^{a}	10^{a}	-5^{b}	12^{b}
$<135,180)$	$-15^{a,b}$	$10^{a,b}$	$-$	$-$	$-$	$-$
$<180,225)$	$-15^{a,b}$	$0^{a,b}$	$-$	$-$	$-$	$-$
$<225,270)$	$-15^{a,b}$	$-10^{a,b}$	5^{b}	-8^{b}	$-$	$-$
$<270,315)$	-5^{a}	-10^{a}	5^{a}	-10^{a}	-5^{b}	-8^{b}
$<315,360)$	$15^{a,b}$	$-10^{a,b}$	$-$	$-$	$-$	$-$

SVHT – Object Reconstruction

- Reconstructed all, selected the best fit

$$\max \sum_j fit\left(f\,(x_{ij}^{\;M}, y_{ij}^{\;M})\right)$$

Original Image

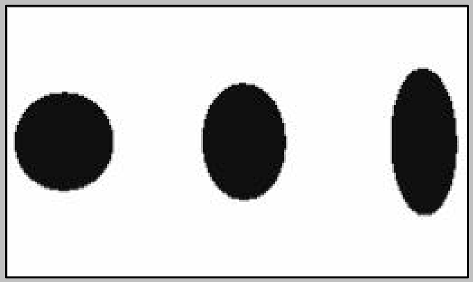


fit values

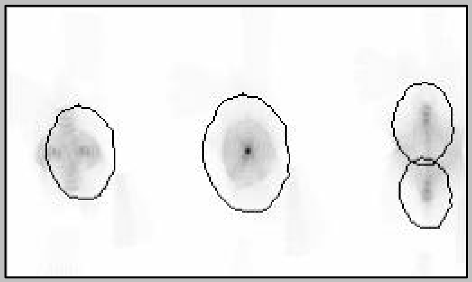
1. fit=98	fit=45	fit=8
2. fit=65	fit=97	fit=30
3. fit=6	fit=32	fit=94

Shape Variance - Comparisons

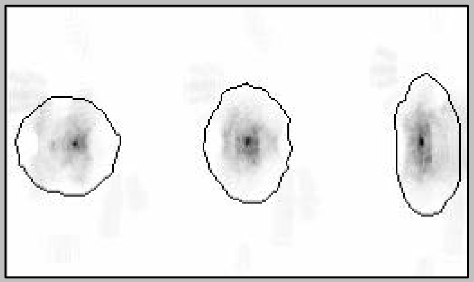
Original Image



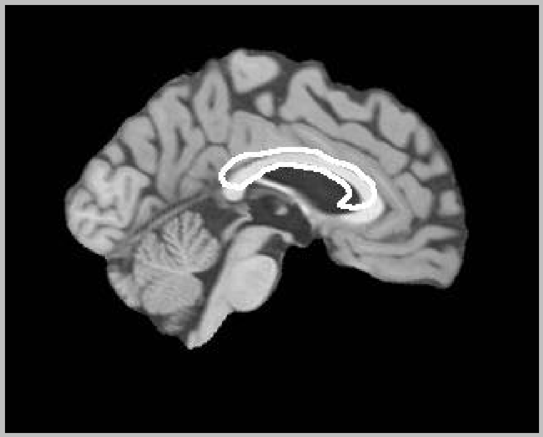
Hough Transform



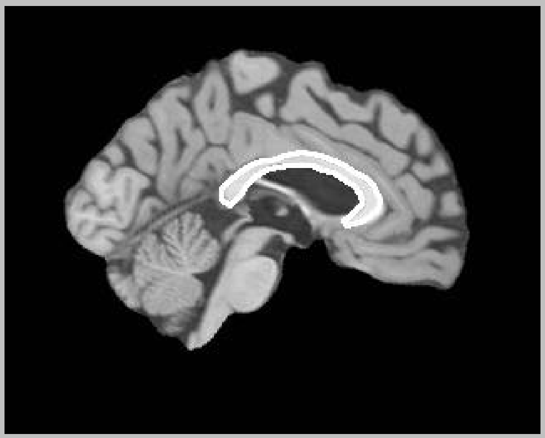
Active Hough Transform



Hough Transform



Active Hough Transform



Shape-Variant Hough Transform Summary

- Algorithm for detection of approximate object location
  - Automated, information derived form training set
  - BAM removed dependency on high gradient magnitude
  - Insensitive to shape variability

<u>Automated Segmentation</u>

1. Training set design
2. Training
   a. Shape Model
   b. Border Appearance Model
3. Segmentation
   Step 1: Approximate object location
   Step 2: Accurate boundary detection

Accurate Boundary Detection

- Any edge-based image segmentation algorithm
  - Automated design of cost function - BAM
  - Fit values replace gradient magnitude
- Examples
  - Dynamic Programming

$$C(P)=\sum_i c(x_i)+\sum_i w(x_i,x_j)$$

$$C(P)=\sum_i F^{-1}\big(fit(f(x_i))\big)+\sum_i w(x_i,x_j)$$

  - Snakes

$$E_E(v(s))=-\left\|\nabla I(v(s))\right\|^2$$

$$E_E(v(s))=-fit\big(f(v(s))\big)$$

<u>Automated Segmentation</u>

1. Training set design
2. Training
   a. Shape Model
   b. Border Appearance Model
3. Segmentation
   Step 1: Approximate object location
   Step 2: Accurate boundary detection

Segmentation - Summary

- Active Hough Transform
  - Automated detection of approximate object location
  - Provides initialization for accurate border detection algorithms
- Accurate border detection
  - Any algorithm
  - *fit* values based on BAM substitutes gradient-based terms in segmentation criterion

<u>Automated Segmentation</u>

1. Training set design
2. Training
   a. Shape Model
   b. Border Appearance Model
3. Segmentation
   Step 1: Approximate object location
   Step 2: Accurate boundary detection

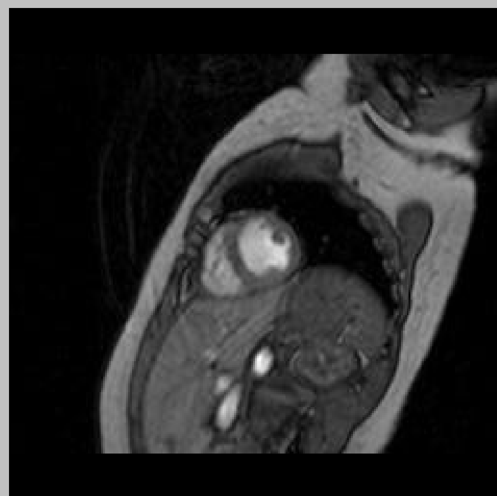
Experimental Methods

5 segmentation tasks

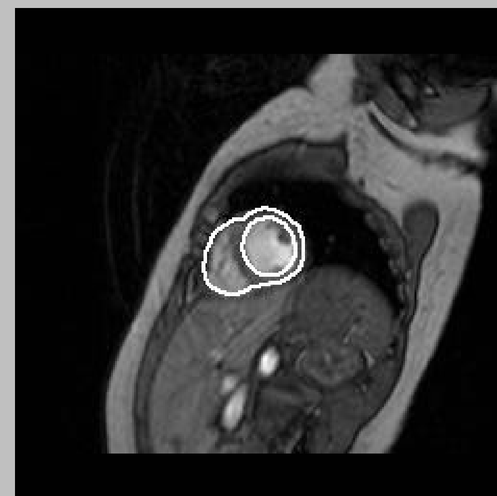
1. Epicardial border in MR images of thorax
   - 58 images, 21 used for training
2. Endocardial border in MR images of thorax
   - 58 images, 21 used for training
3. Corpus Callosum in MR images of brain
   - 90 images, 15 used for training
4. Cerebellum in MR images of brain
   - 90 images, 6 used for training
5. Vertebrae in MR images of spine
   - 55 images , 15 used for training, (235 vertebrae to segment)

Results

Original Image



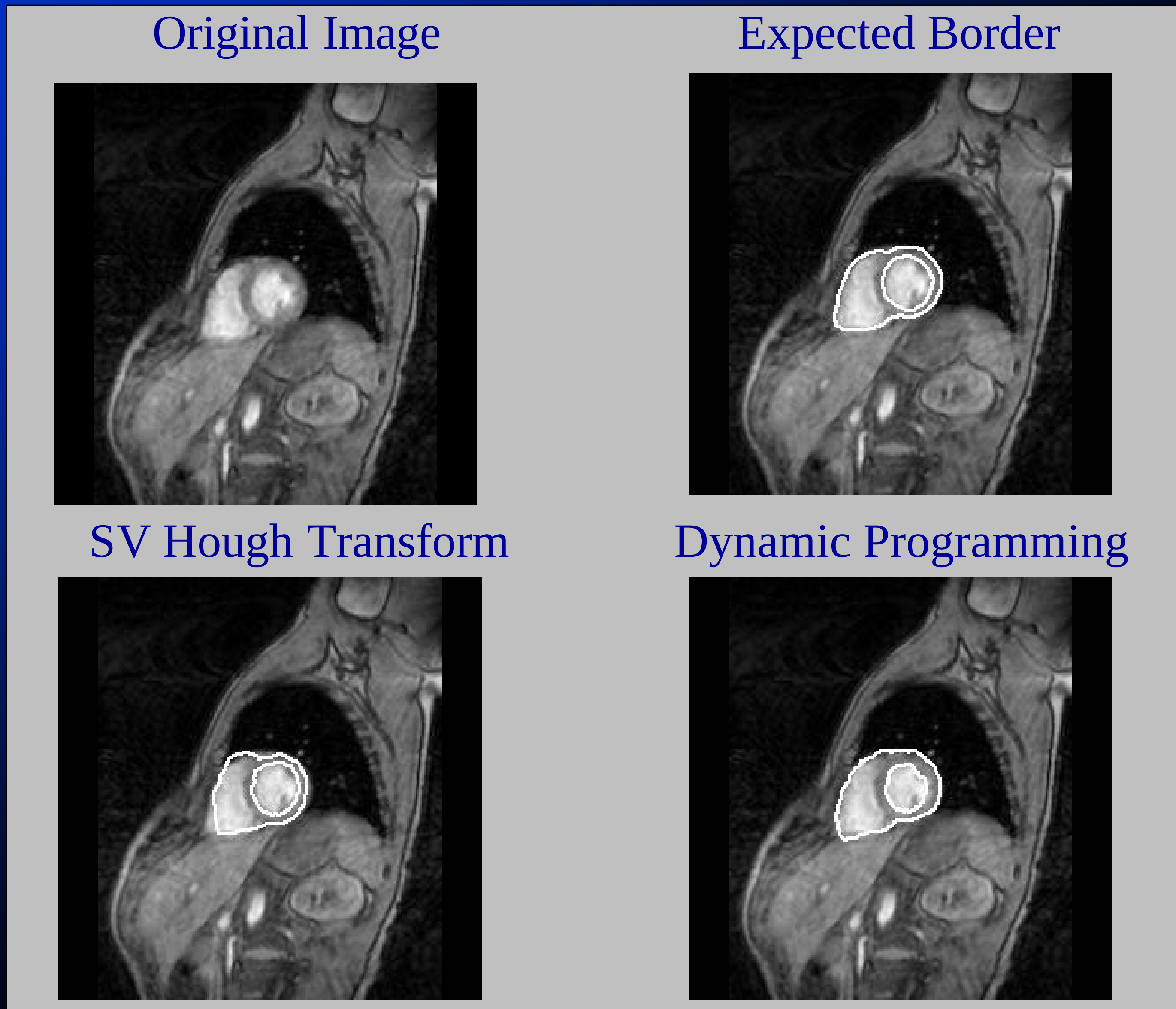
Expected Border



SV Hough Transform

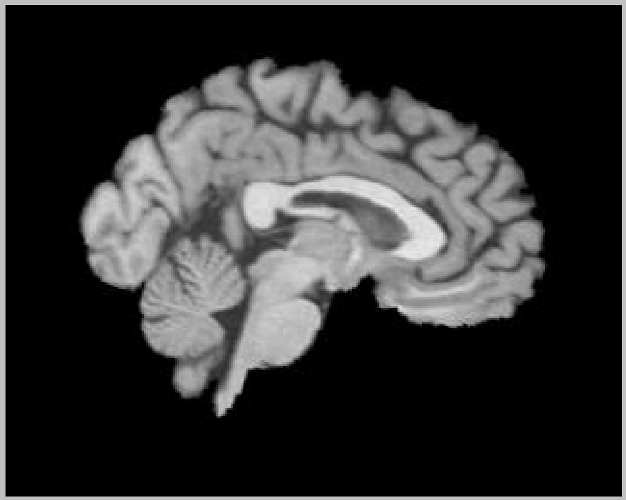
Dynamic Programming

Results


Original Image
Expected Border
SV Hough Transform
Dynamic Programming

Results

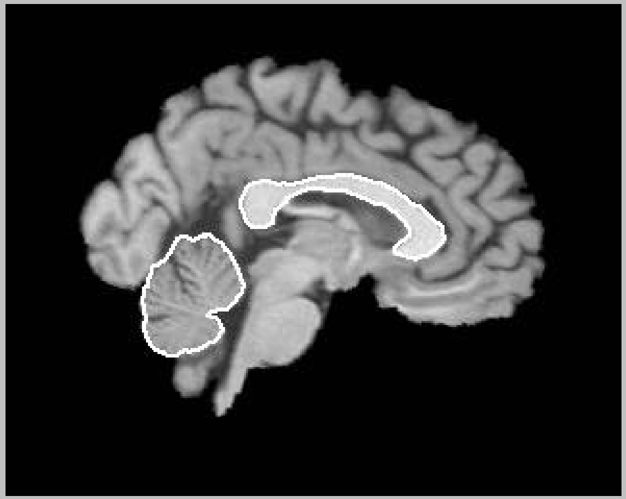
Original Image



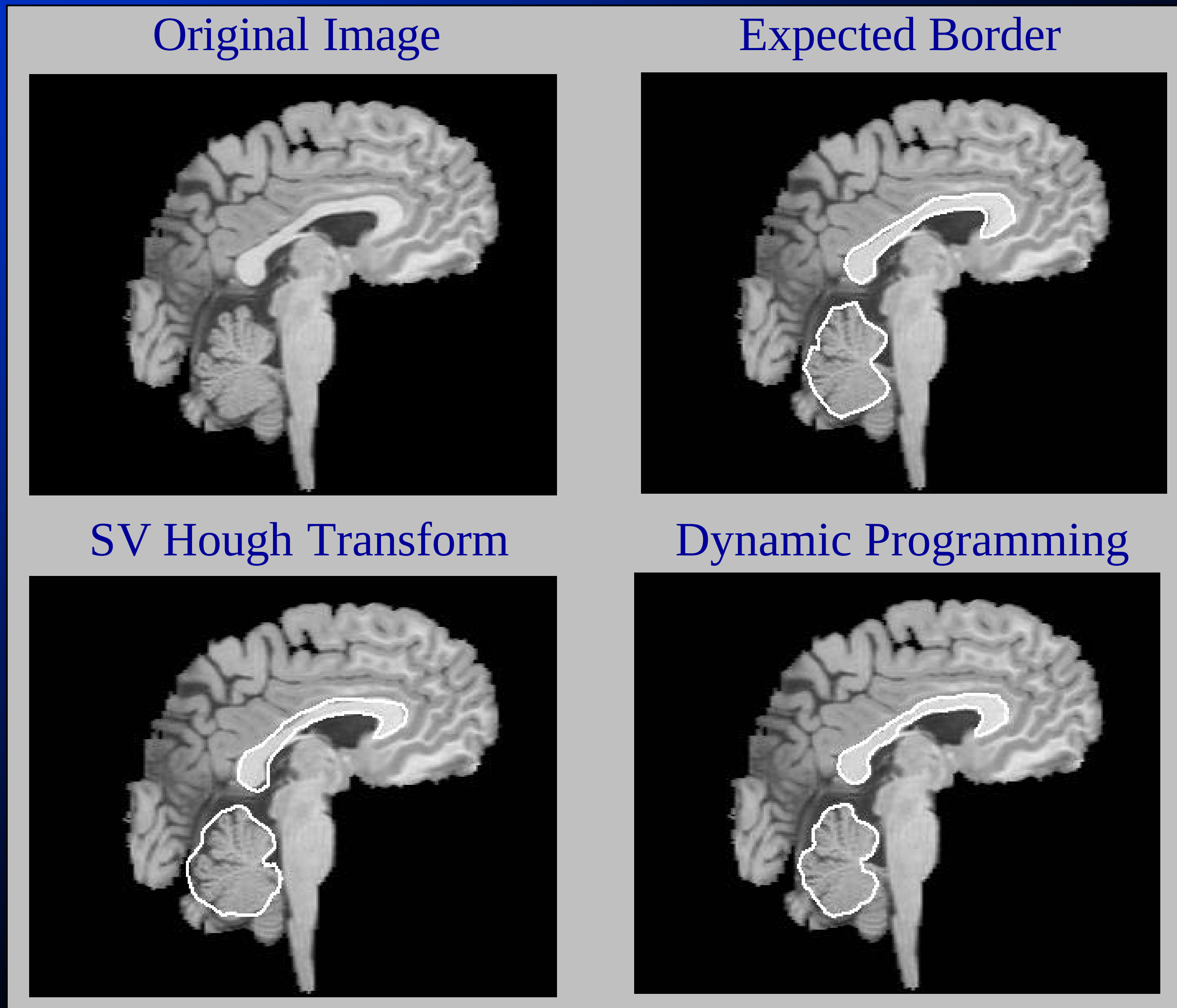
Expected Border

SV Hough Transform

Dynamic Programming



Results


Original Image
Expected Border
SV Hough Transform
Dynamic Programming

Results

Original Image

Expected Border

SV Hough Transform

Dynamic Programming

Results

Original Image

Expected Border

SV Hough Transform

Dynamic Programming

Detection Accuracy

Errors of approximate position detection: (mean+/-std)[pixels]					
Structure	**Method**	**rms**	**mean**	**sgn mean**	**max**
Cerebellum		2.4+/-0.8	1.9+/-0.6	-0.4+/-1.0	5.9+/-1.8
Corpus Callosum		1.8+/-0.4	1.4+/-0.3	-0.2+/-0.6	5.5+/-1.4
Spine		2.3+/-0.8	1.8+/-0.6	-0.3+/-1.0	5.8+/-2.1
Heart-epicardium		1.6+/-0.7	1.3+/-0.6	+0.1+/-1.1	3.5+/-1.5
Heart-endocardium		2.1+/-0.8	1.6+/-0.6	+0.4+/-1.0	5.5+/-2.0

Errors of accurate border detection: (mean+/-std)[pixels]					
Structure	**Method**	**rms**	**mean**	**sgn mean**	**max**
Cerebellum	Dynamic Pr.	1.2+/-0.7	0.9+/-0.5	-0.1+/-0.6	3.6+/-2.3
	Snakes	1.7+/-0.5	1.4+/-0.4	+0.1+/-0.6	4.3+/-1.4
Corpus Callosum	Dynamic Pr.	1.1+/-0.5	0.7+/-0.3	-0.3+/-0.3	3.9+/-2.1
	Snakes	1.5+/-0.4	1.2+/-0.3	-0.2+/-0.3	4.8+/-2.0
Spine	Dynamic Pr.	1.5+/-0.5	1.0+/-0.3	+0.1+/-0.5	4.5+/-1.9
	Snakes	2.3+/-0.7	1.8+/-0.5	+0.2+/-0.9	5.7+/-1.8
Heart-epicardium	Dynamic Pr.	1.4+/-0.6	1.2+/-0.5	+0.6+/-0.7	3.4+/-1.3
	Snakes	1.8+/-0.5	1.5+/-0.4	+1.1+/-0.6	4.0+/-1.3
Heart-endocardium	Dynamic Pr.	1.6+/-0.5	1.2+/-0.3	-0.2+/-0.8	4.5+/-0.3
	Snakes	1.8+/-0.6	1.5+/-0.5	-0.1+/-0.9	4.4+/-0.5

Comparison of Approximate and Accurate Detection

Paired two-sample t-test for means		
	AHT - DP	**AHT - Snakes**
Cerebellum	$p<0.001$	$p<0.001$
Corpus Callosum	$p<0.001$	$p<0.001$
Vertebrae	$p<0.001$	$p=0.74$
Heart-epicardium	$p=0.04$	$p=0.07$
Heart-endocardium	$p<0.001$	$p=0.04$
Overall	$p<0.001$	$p<0.001$

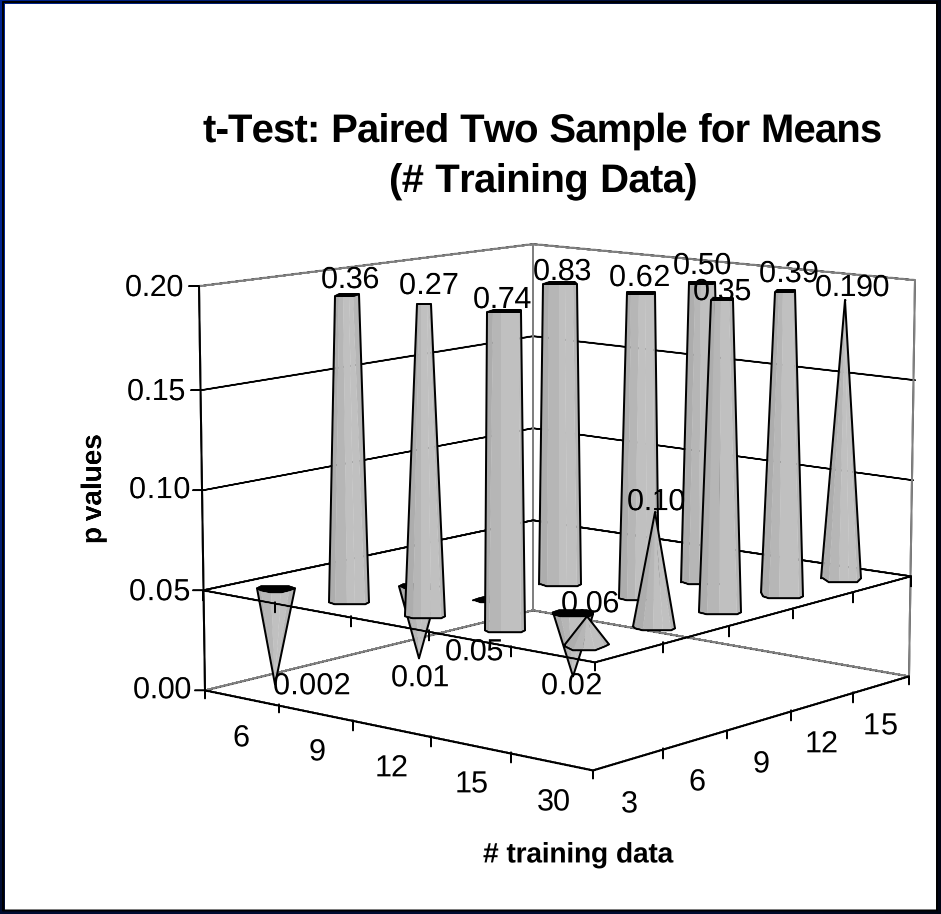
Study 1: Dependency on Number of Training Data

Errors of approximate position detection: (mean+/-std)[pixels]					
Structure	**Num. Data**	**rms**	**mean**	**sgn mean**	**max**
Cerebellum	3	2.4+/-0.6	1.9+/-0.5	-0.4+/-1.0	5.6+/-1.5
	6	2.4+/-0.8	1.9+/-0.6	-0.4+/-1.0	5.9+/-1.8
	9	2.3+/-1.0	1.8+/-0.7	-0.0+/-1.2	5.7+/-2.3
	12	2.3+/-0.7	1.8+/-0.5	-0.2+/-1.0	5.6+/-1.7
	15	2.2+/-0.6	1.8+/-0.5	-0.1+/-0.8	5.6+/-0.9
	30	2.0+/-0.6	1.6+/-0.5	-0.1+/-0.8	5.2+/-1.7

Errors of accurate border detection (DP): (mean+/-std)[pixels]					
Structure	**Num. Data**	**rms**	**mean**	**sgn mean**	**max**
Cerebellum	3	1.4+/-0.8	1.0+/-0.5	-0.3+/-0.7	4.4+/-2.5
	6	1.2+/-0.7	0.9+/-0.5	-0.1+/-0.6	3.6+/-2.3
	9	1.4+/-0.8	1.0+/-0.5	-0.2+/-0.7	4.1+/-2.7
	12	1.3+/-0.9	1.0+/-0.6	-0.1+/-0.7	3.9+/-2.6
	15	1.4+/-0.9	1.0+/-0.6	+0.1+/-0.7	4.1+/-2.8
	30	1.3+/-0.9	0.9+/-0.6	+0.1+/-0.7	3.9+/-2.6

Study 1: Statistical Significance

Dependency on Number of Training Data					
Paired two-sample t-test for means					
	6	9	12	15	30
3	0.002	0.36	0.27	0.74	0.06
6		0.01	0.05	0.02	0.10
9			0.83	0.62	0.35
12				0.50	0.39
15					0.19



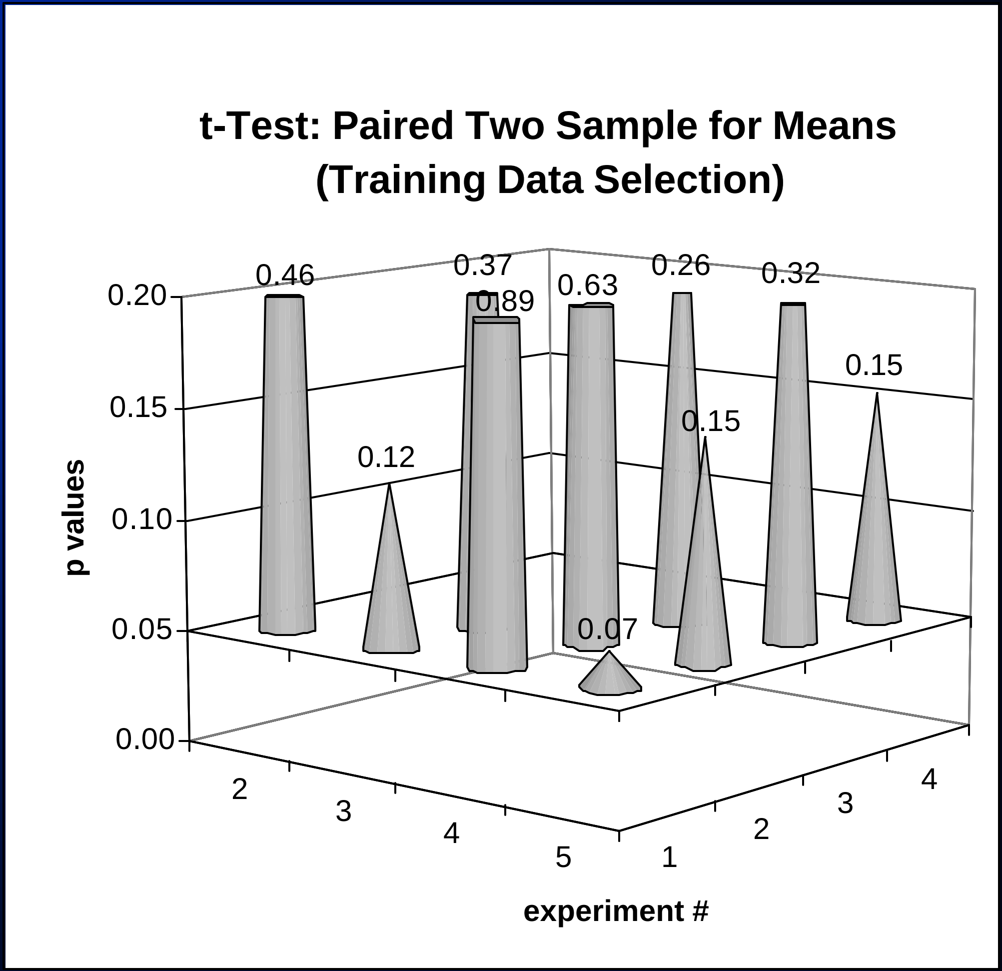
Study 2: Dependency on Training Data Selection

Errors of approximate position detection: (mean+/-std)[pixels]					
Structure	**Study #**	**rms**	**mean**	**sgn mean**	**max**
Cerebellum	1	2.4+/-0.8	1.9+/-0.6	-0.4+/-1.0	5.9+/-1.8
	2	2.3+/-0.7	1.8+/-0.5	-0.1+/-1.0	6.0+/-2.0
	3	2.2+/-0.6	1.8+/-0.5	+0.1+/-0.7	5.4+/-1.4
	4	2.4+/-0.8	1.8+/-0.6	+0.4+/-1.0	6.1+/-2.4
	5	2.5+/-0.8	1.9+/-0.6	-0.6+/-0.9	6.1+/-2.0

Errors of accurate border detection (DP): (mean+/-std)[pixels]					
Structure	**Study #**	**rms**	**mean**	**sgn mean**	**max**
Cerebellum	1	1.2+/-0.7	0.9+/-0.5	-0.1+/-0.6	3.6+/-2.3
	2	1.3+/-0.7	0.9+/-0.4	+0.2+/-0.6	3.6+/-2.2
	3	1.3+/-0.7	0.9+/-0.4	+0.1+/-0.6	4.0+/-2.4
	4	1.2+/-0.5	0.9+/-0.3	+0.2+/-0.4	3.7+/-1.9
	5	1.4+/-1.2	1.0+/-0.7	-0.2+/-0.9	4.2+/-3.5

Study 2: Statistical Significance

Dependency on Data Selection				
Paired two-sample t-test for means				
	2	3	4	5
1	0.46	0.12	0.89	0.07
2		0.37	0.63	0.15
3			0.26	0.32
4				0.15



Discussion - Conclusion

- Methodology for automated model-based image segmentation
  1. Training set design
  2. Training (Shape Model, Border Appearance Model)
  3. Segmentation (Shape-Variant Hough Transform, Accurate Border Detection)
- Fully automated – requires small set of training examples
- Two step segmentation
  - approximate object location
  - accurate boundary detection

Discussion - Conclusion

- The method is applicable to the following segmentation tasks:
  - Objects with well defined shape
  - Objects with reasonably consistent border appearance
  - Representative set of examples is available
  - Edge-based image segmentation method is appropriate for the segmentation task