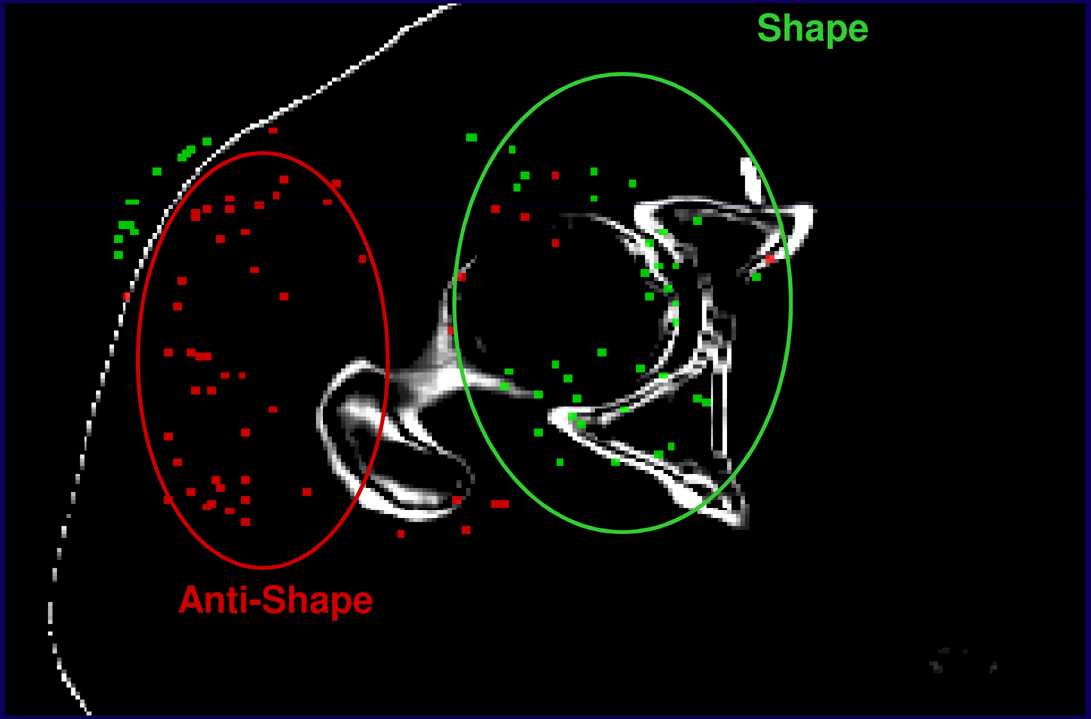
Discriminative Optimization of 3D Shape Models for the Generalized Hough Transform



Hauke Schramm, Ana Belén Martín Recuero, Peter Beyerlein

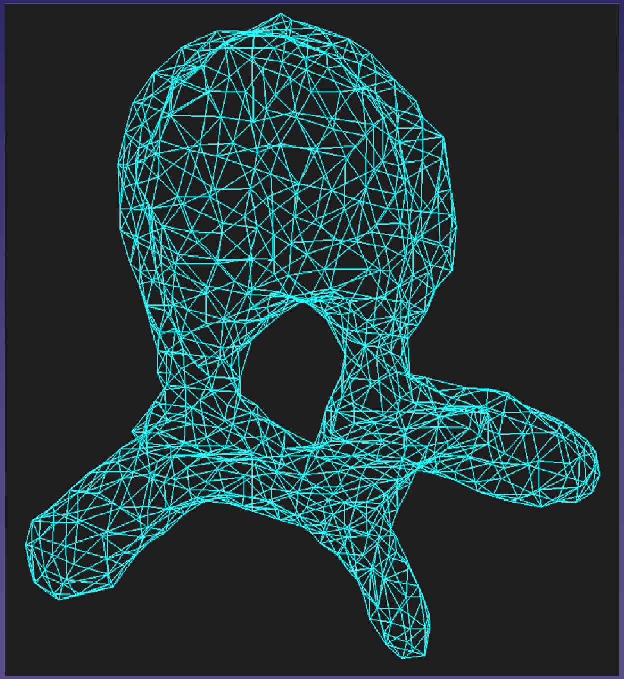
Overview

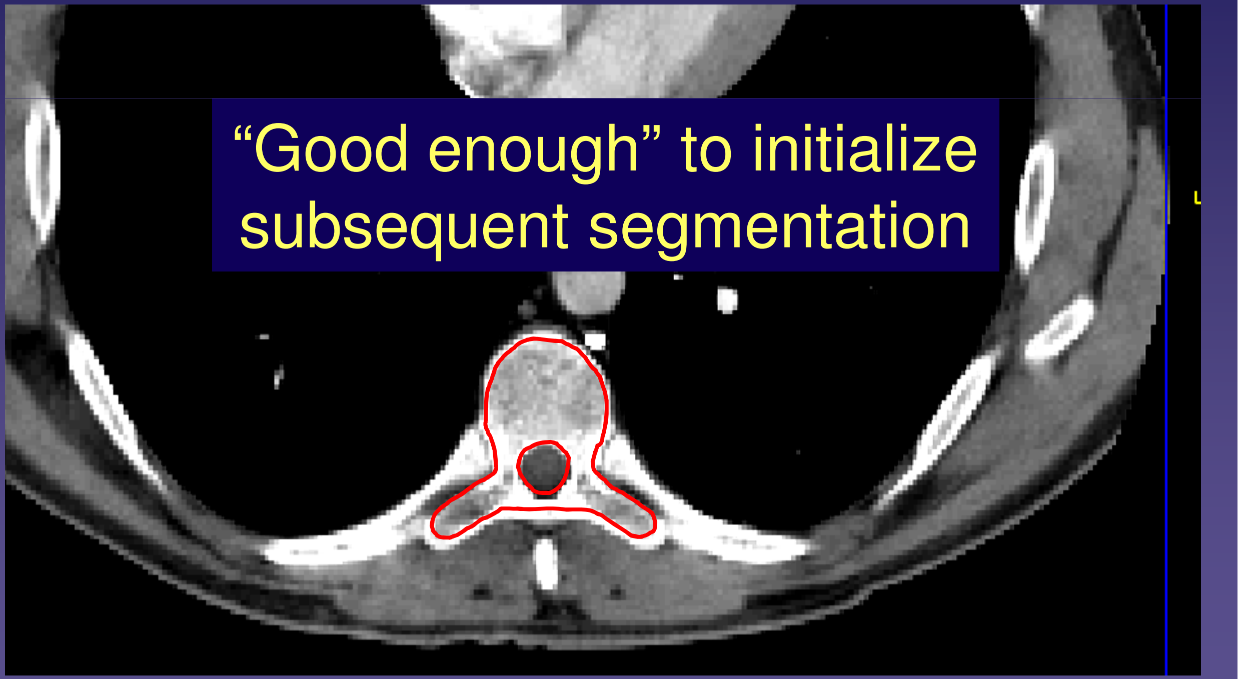
1. Introduction
2. Generalized Hough Transform
   a) Principle
   b) Experimental results
3. Discriminative Shape Model Optimization
   a) Motivation
   b) Theory
   c) First Experimental Results
4. Summary

The Shape Finder Task

Given: 3D mesh model of an anatomical object

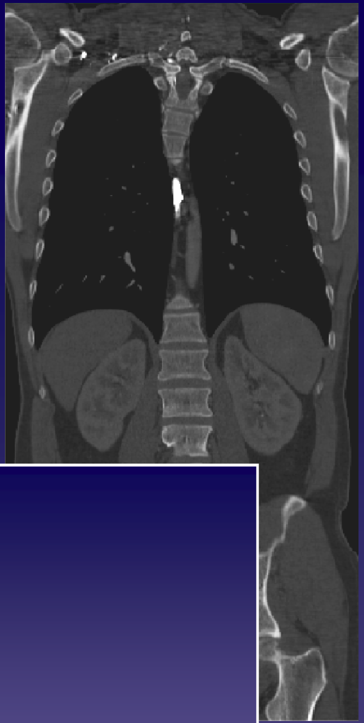
Task: Identify object in unknown individual





“Good enough” to initialize subsequent segmentation

Research Challenges

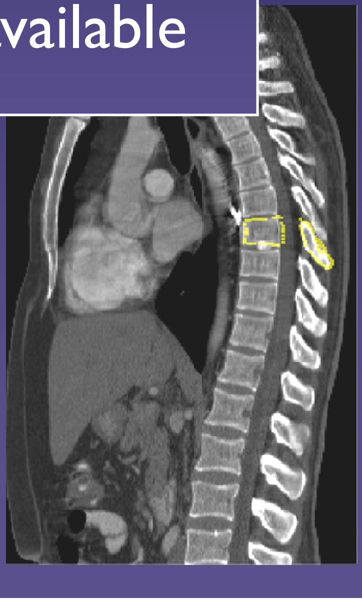


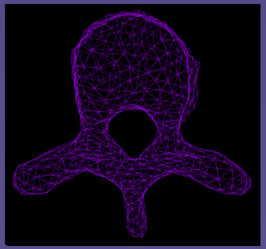
- Fast & reliable object detection in large images
- Deal with substantial object variability

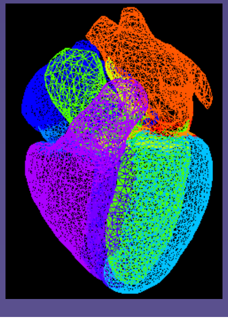
State of the art:

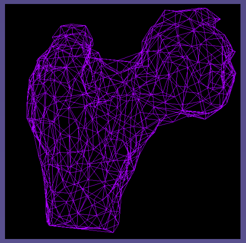
- Specific solutions exist (e.g. lung or heart finder)
- General, portable, automated approach not yet available

- Discriminate between similar objects
- Address new objects with minimal manual effort









The Generalized Hough Transform

– Hough transform : Detection of analytical curves

– Generalized HT : Generalization to arbitrary objects

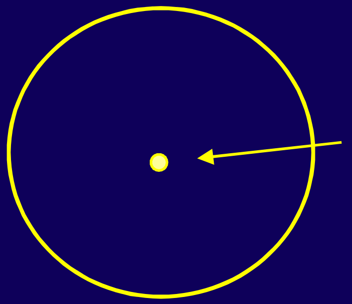
Pro's:

- Robust to occlusions, deformations
- Robust to noise and artefacts
- Able to find multiple occurrences
- Well established (in 2-D)

Con's:

- High computational complexity
- Large memory requirement

Generalized Hough Transform: Principle

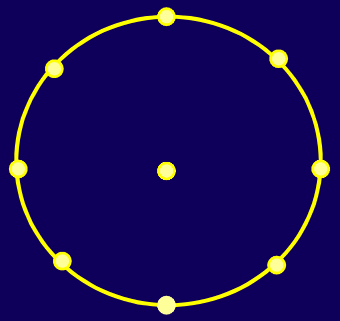


reference point
(e.g. center)

Template Object

Generalized Hough Transform: Principle

Determine model points
(e.g. triangle centers of mesh)

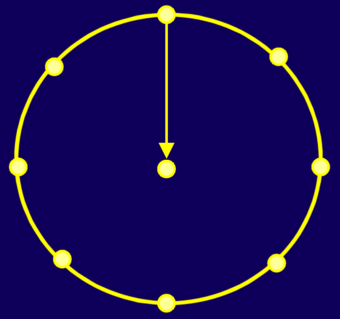


Template Object

Generalized Hough Transform: Principle

For each model point:

Store position relative to reference point

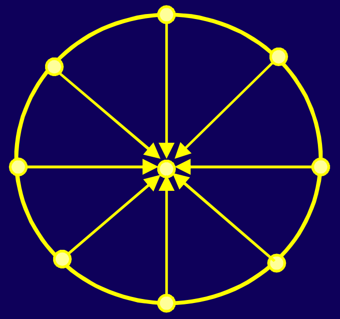


Template Object

Generalized Hough Transform: Principle

For each model point:

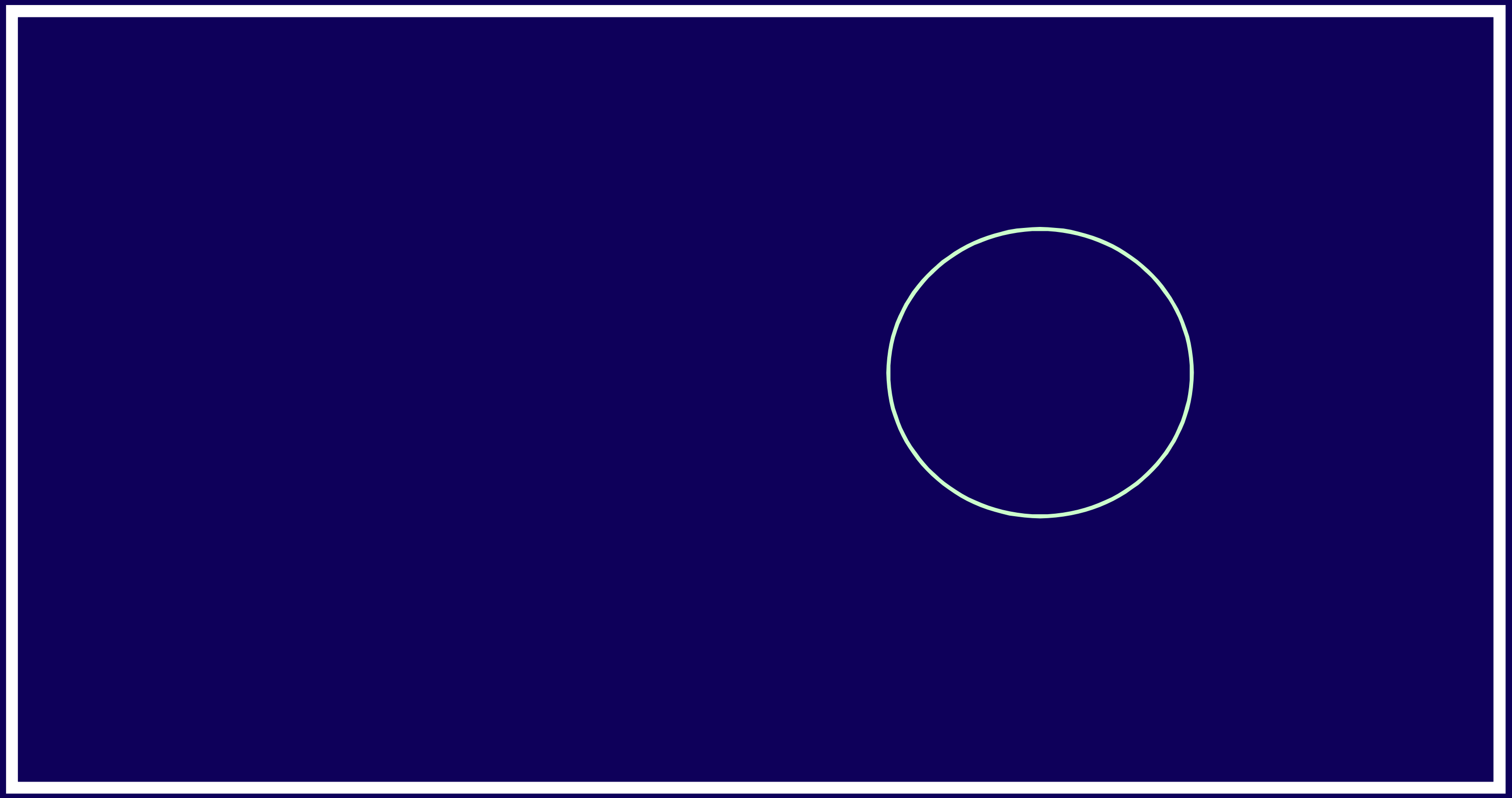
Store position relative to reference point



⇒ Object representation

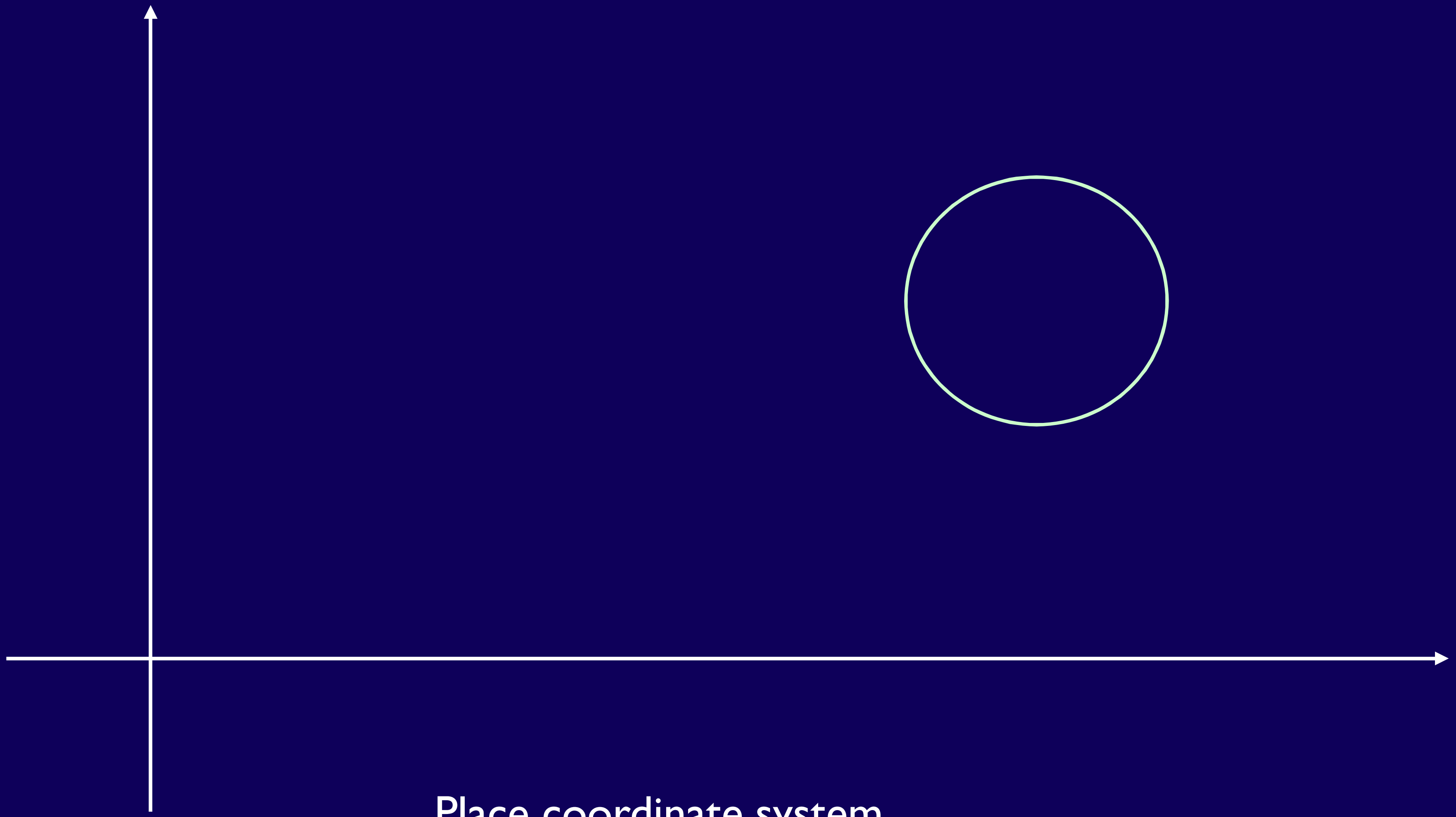
Template Object

Generalized Hough Transform: Principle



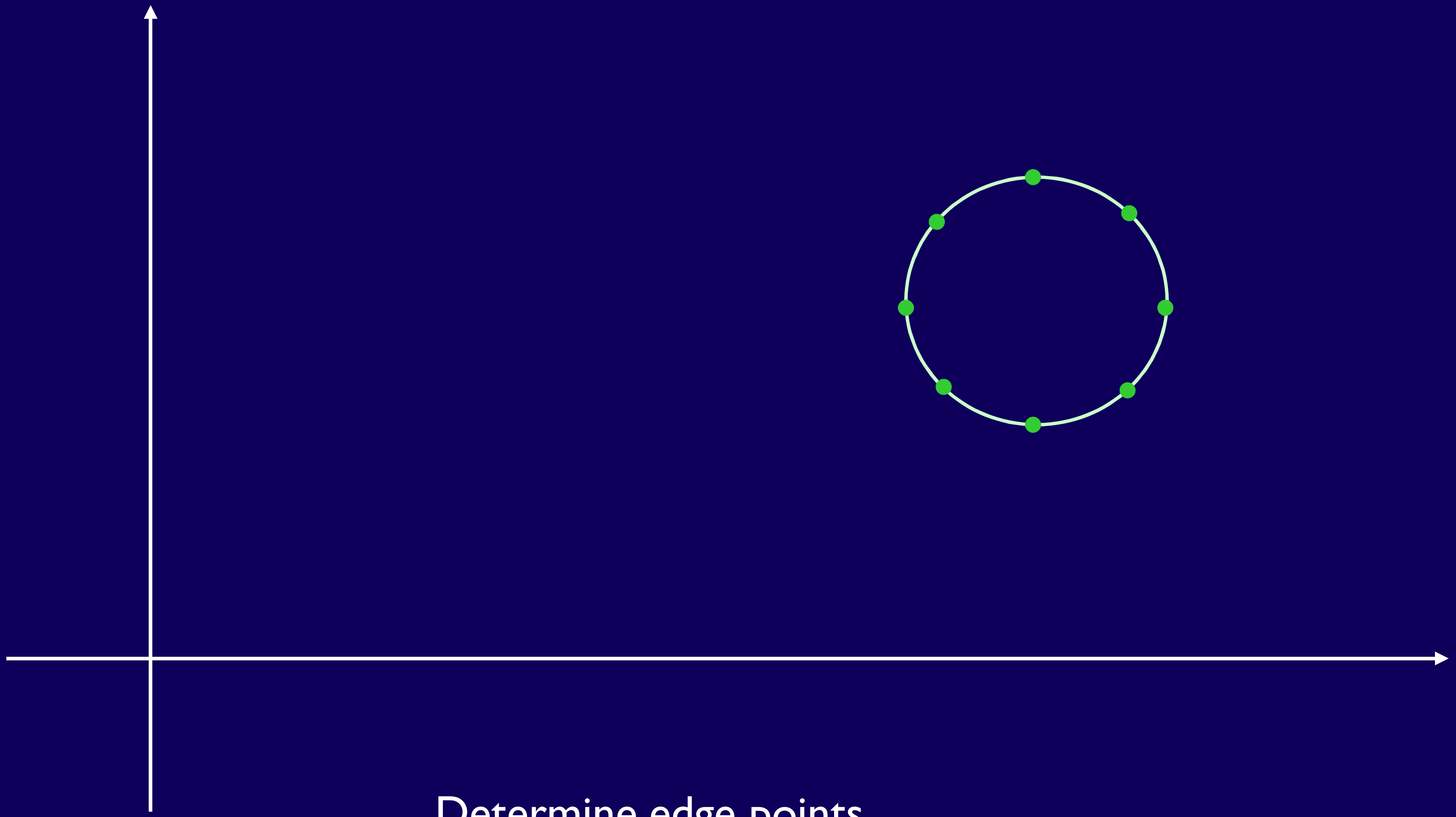
Unknown Image

Generalized Hough Transform: Principle



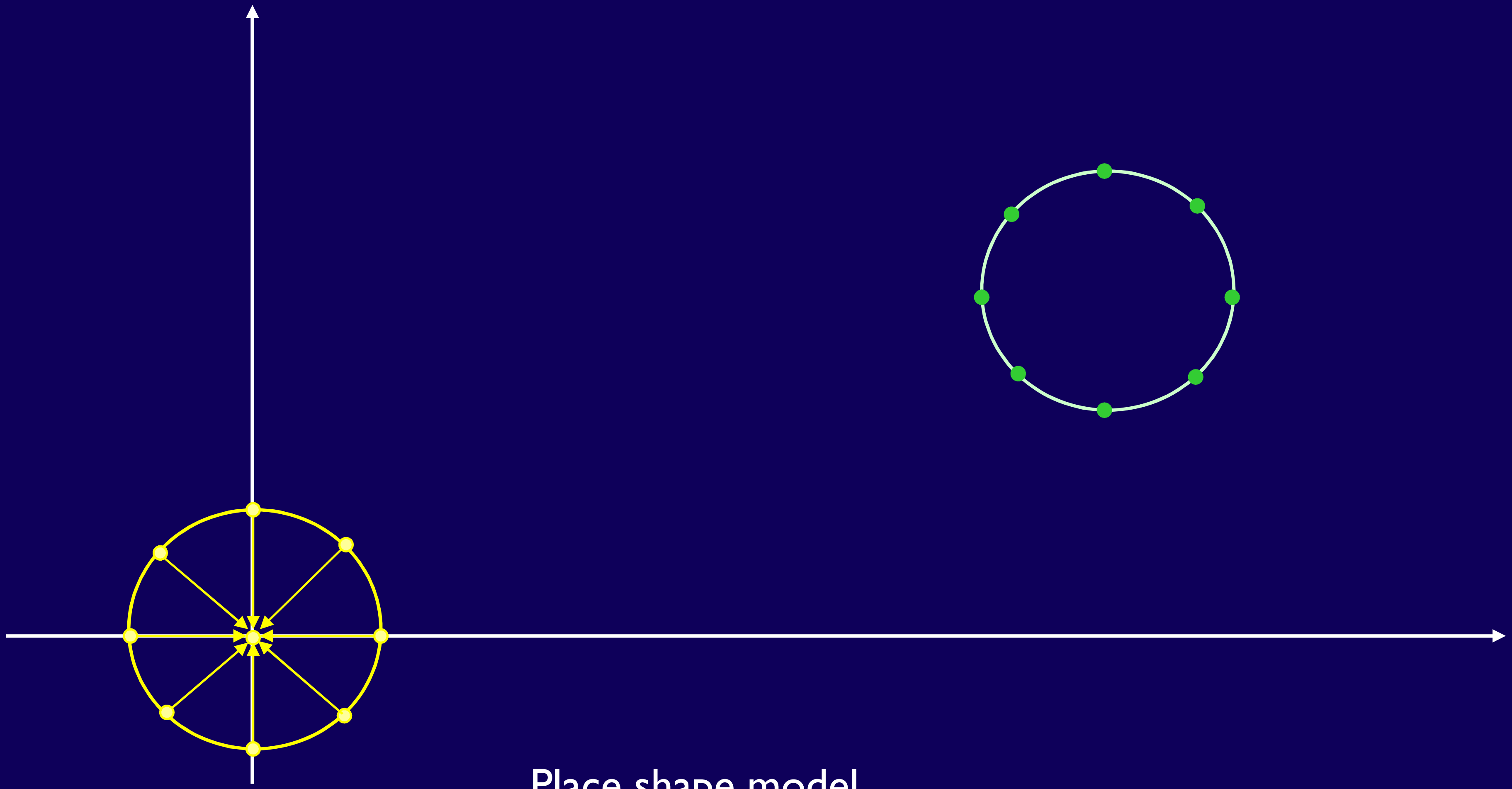
Place coordinate system

Generalized Hough Transform: Principle



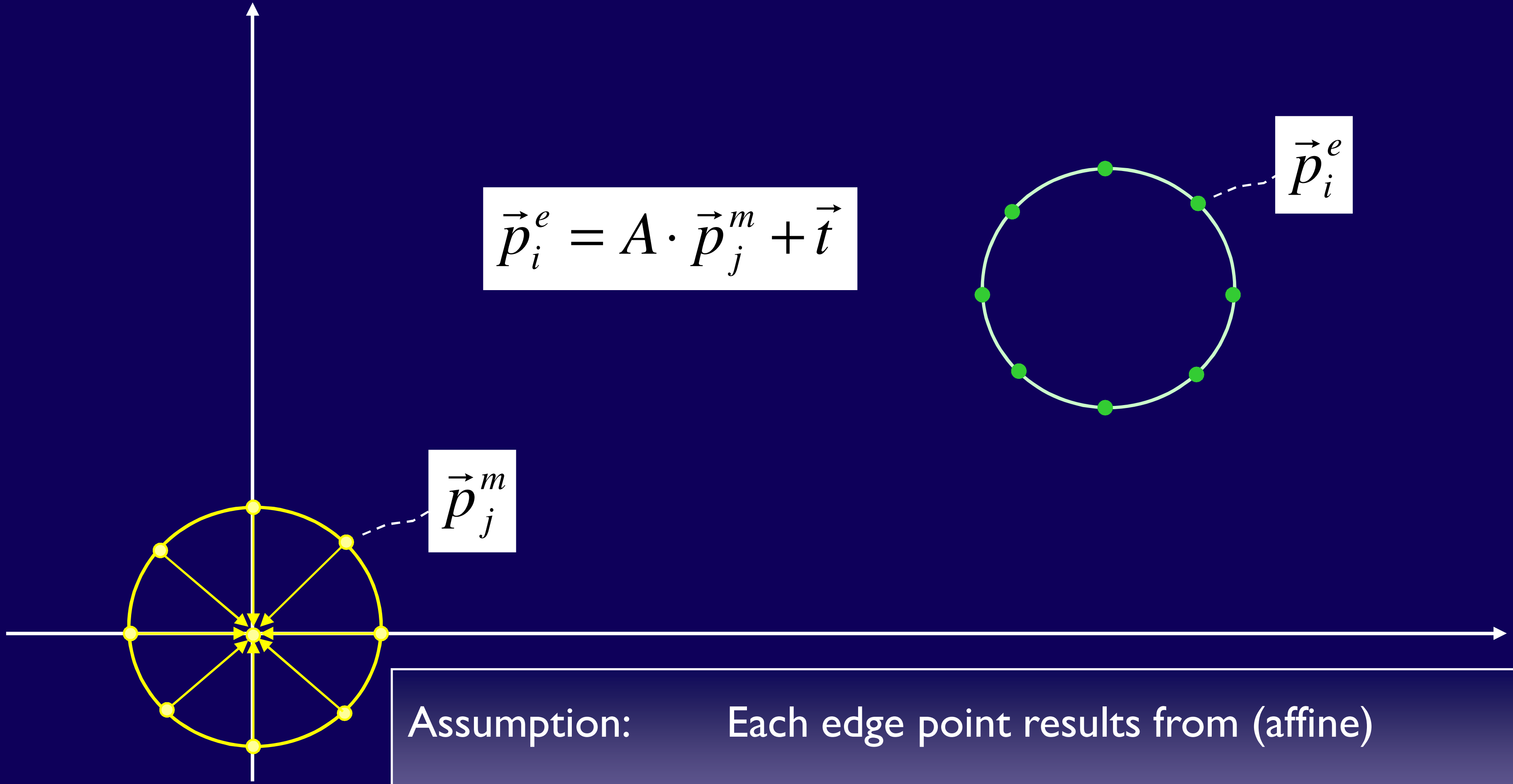
Determine edge points

Generalized Hough Transform: Principle



Place shape model

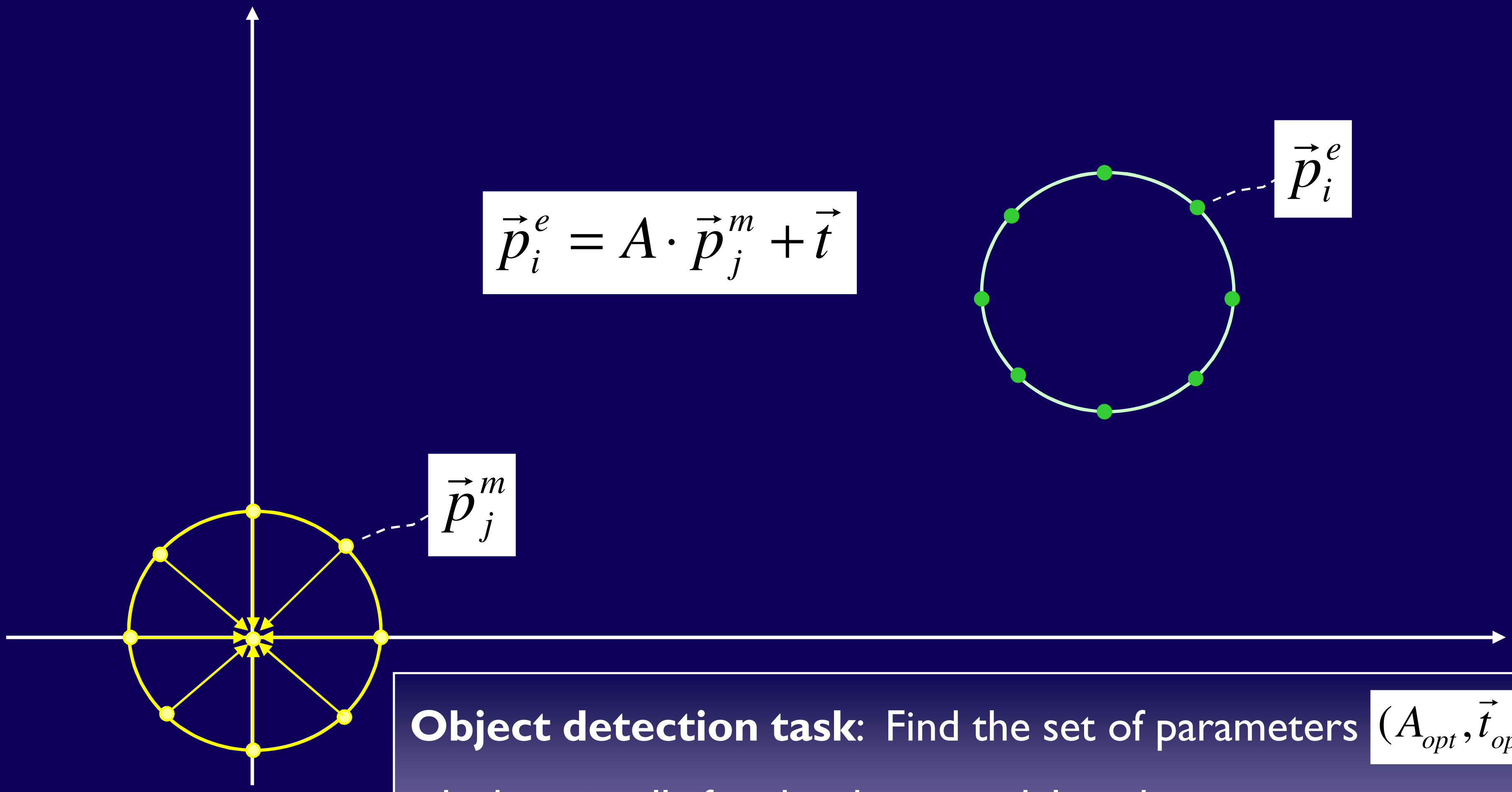
Generalized Hough Transform: Principle



$$\vec{p}_i^{\,e} = A \cdot \vec{p}_j^{\,m} + \vec{t}$$

Assumption: Each edge point results from (affine) transformation of a model point

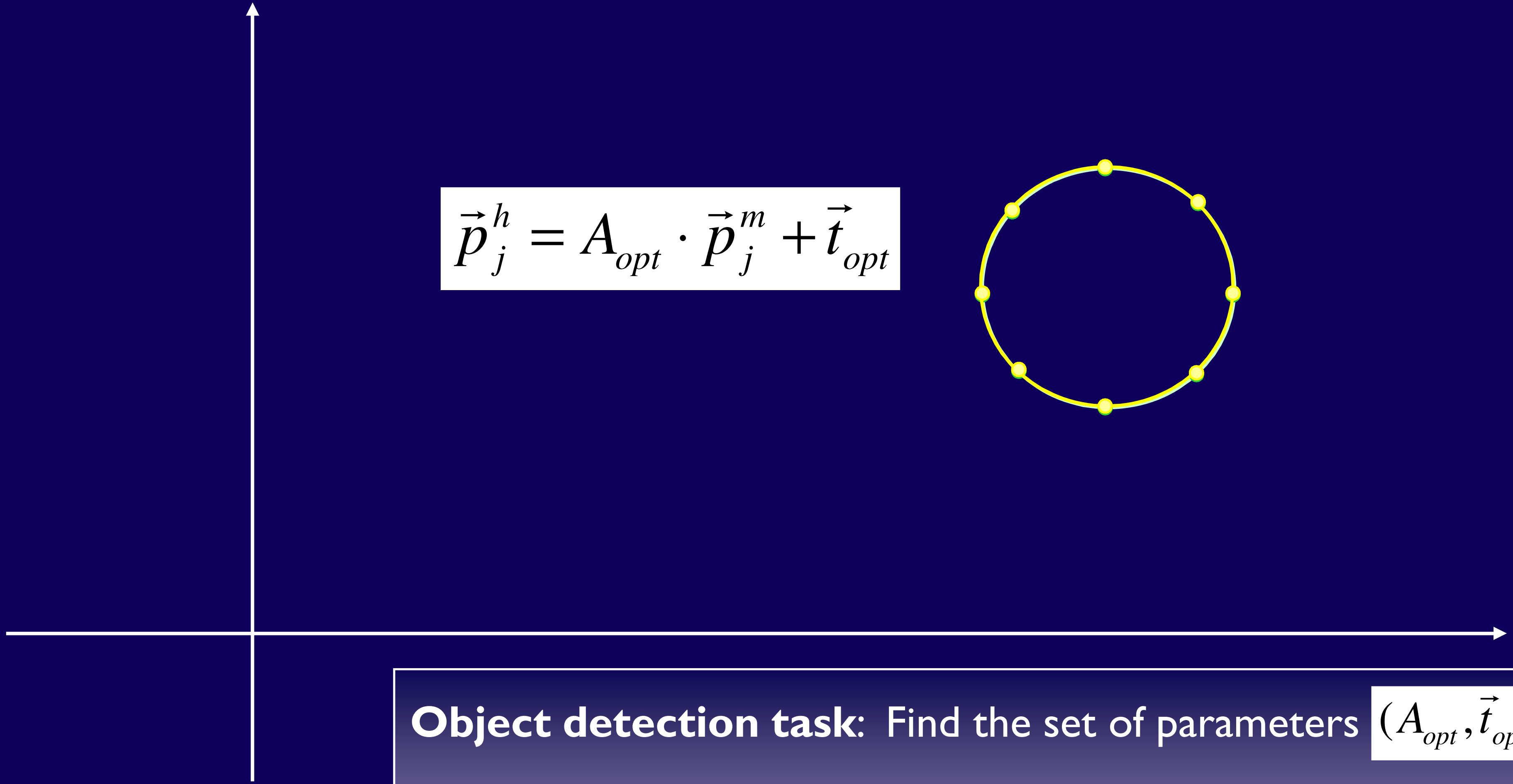
Generalized Hough Transform: Principle



$$\vec{p}_i^{\,e} = A \cdot \vec{p}_j^{\,m} + \vec{t}$$

Object detection task: Find the set of parameters $(A_{opt}, \vec{t}_{opt})$ which optimally fits the shape model to the image.

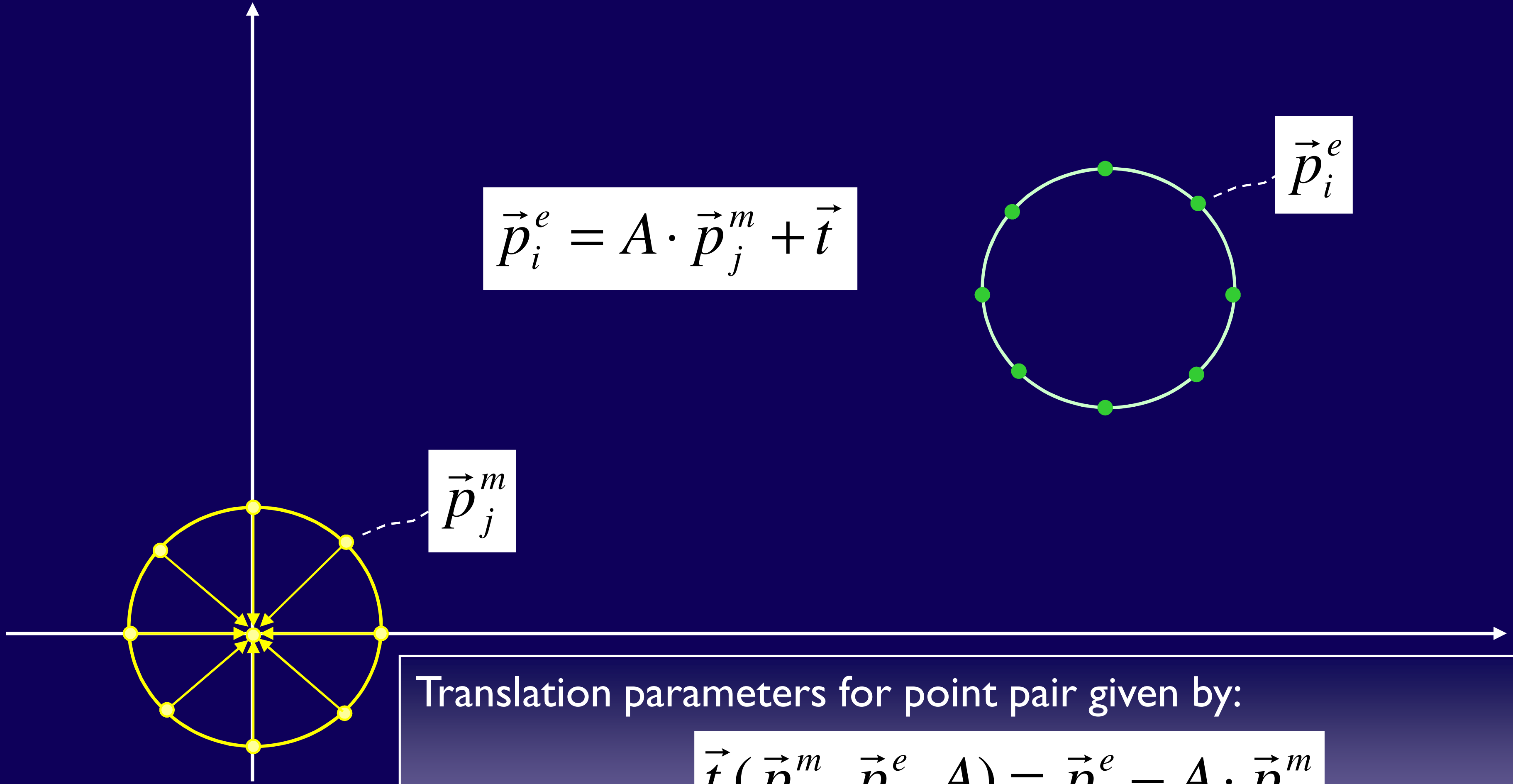
Generalized Hough Transform: Principle



$$\vec{p}_j^{\,h} = A_{opt} \cdot \vec{p}_j^{\,m} + \vec{t}_{opt}$$

Object detection task: Find the set of parameters $(A_{opt}, \vec{t}_{opt})$ which optimally fits the shape model to the image.

Generalized Hough Transform: Principle

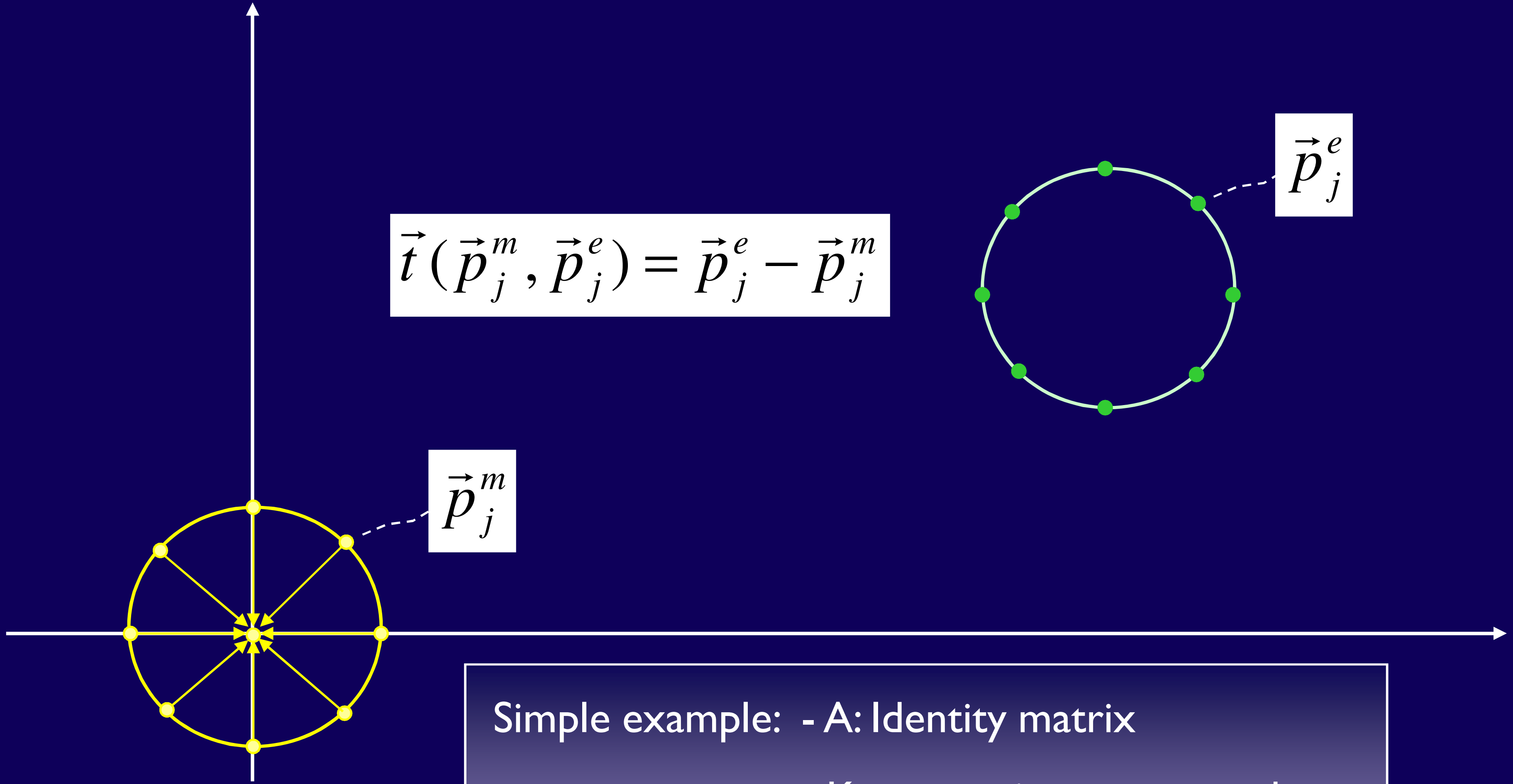


$$\vec{p}_i^{\,e} = A \cdot \vec{p}_j^{\,m} + \vec{t}$$

Translation parameters for point pair given by:

$$\vec{t}\,(\vec{p}_j^{\,m}, \vec{p}_i^{\,e}, A) = \vec{p}_i^{\,e} - A \cdot \vec{p}_j^{\,m}$$

Generalized Hough Transform: Principle

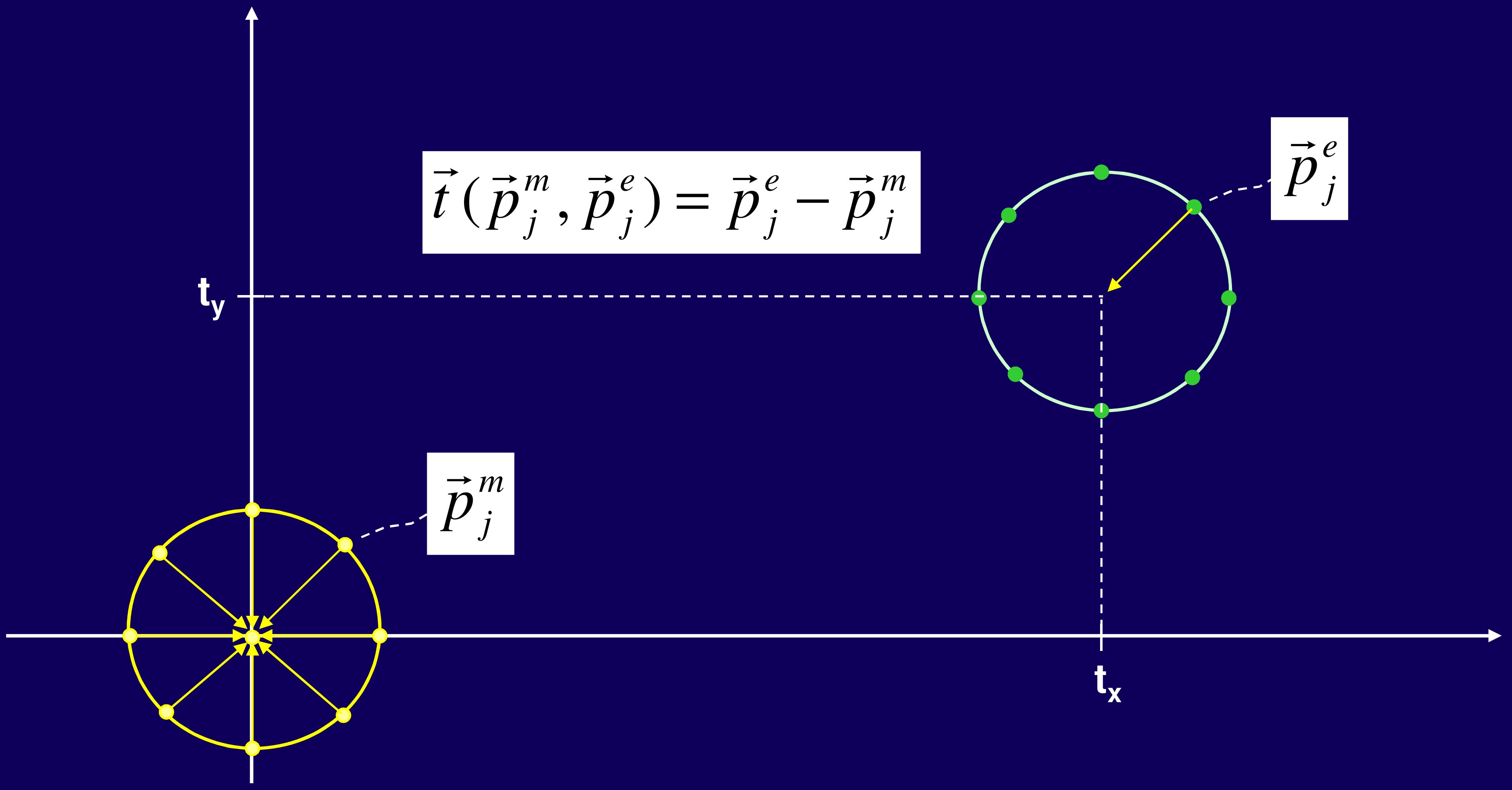


$$\vec{t}\,(\vec{p}_j^{\,m}, \vec{p}_j^{\,e}) = \vec{p}_j^{\,e} - \vec{p}_j^{\,m}$$

Simple example: - A: Identity matrix

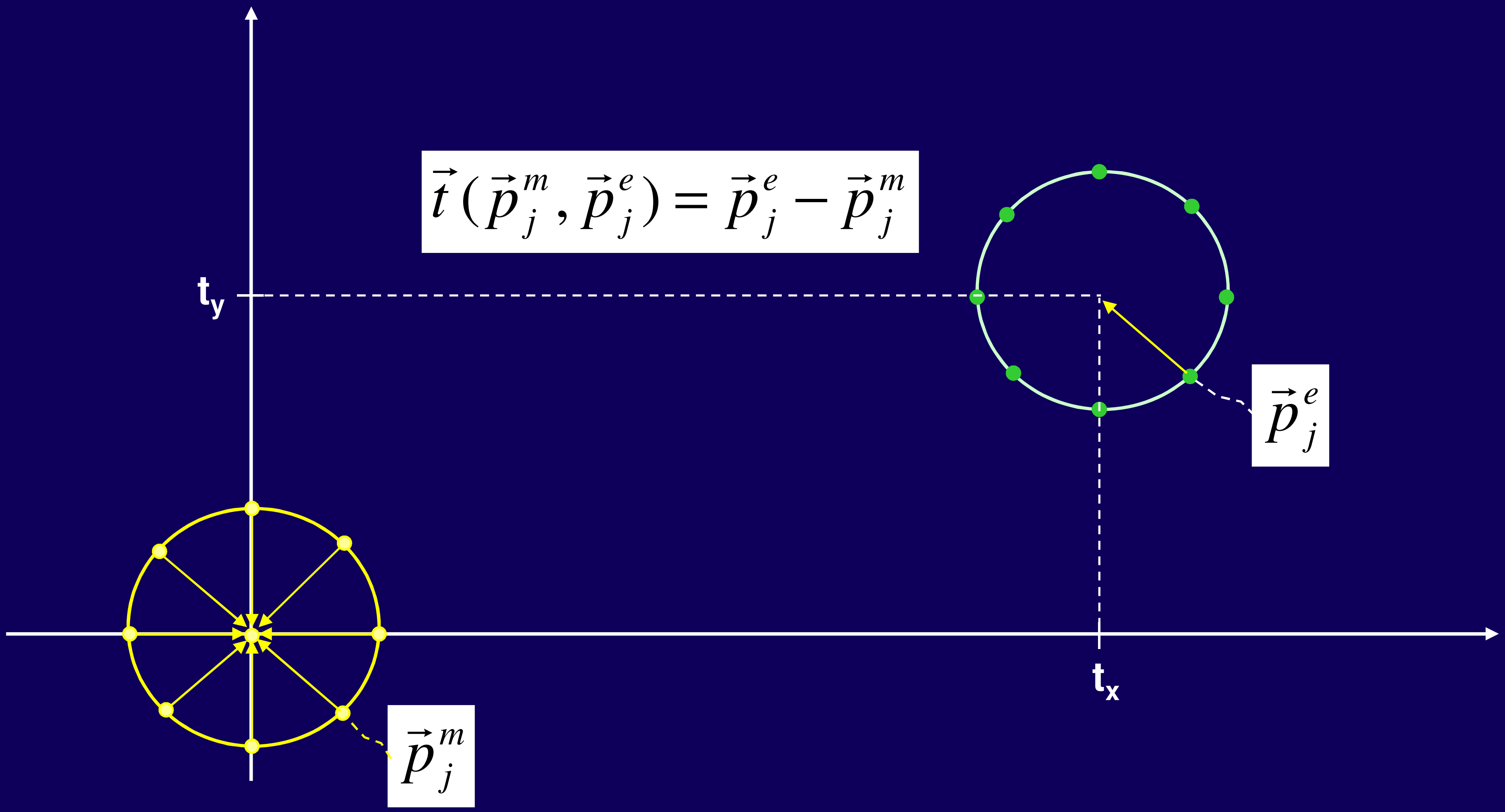
- Known point correspondence

Generalized Hough Transform: Principle



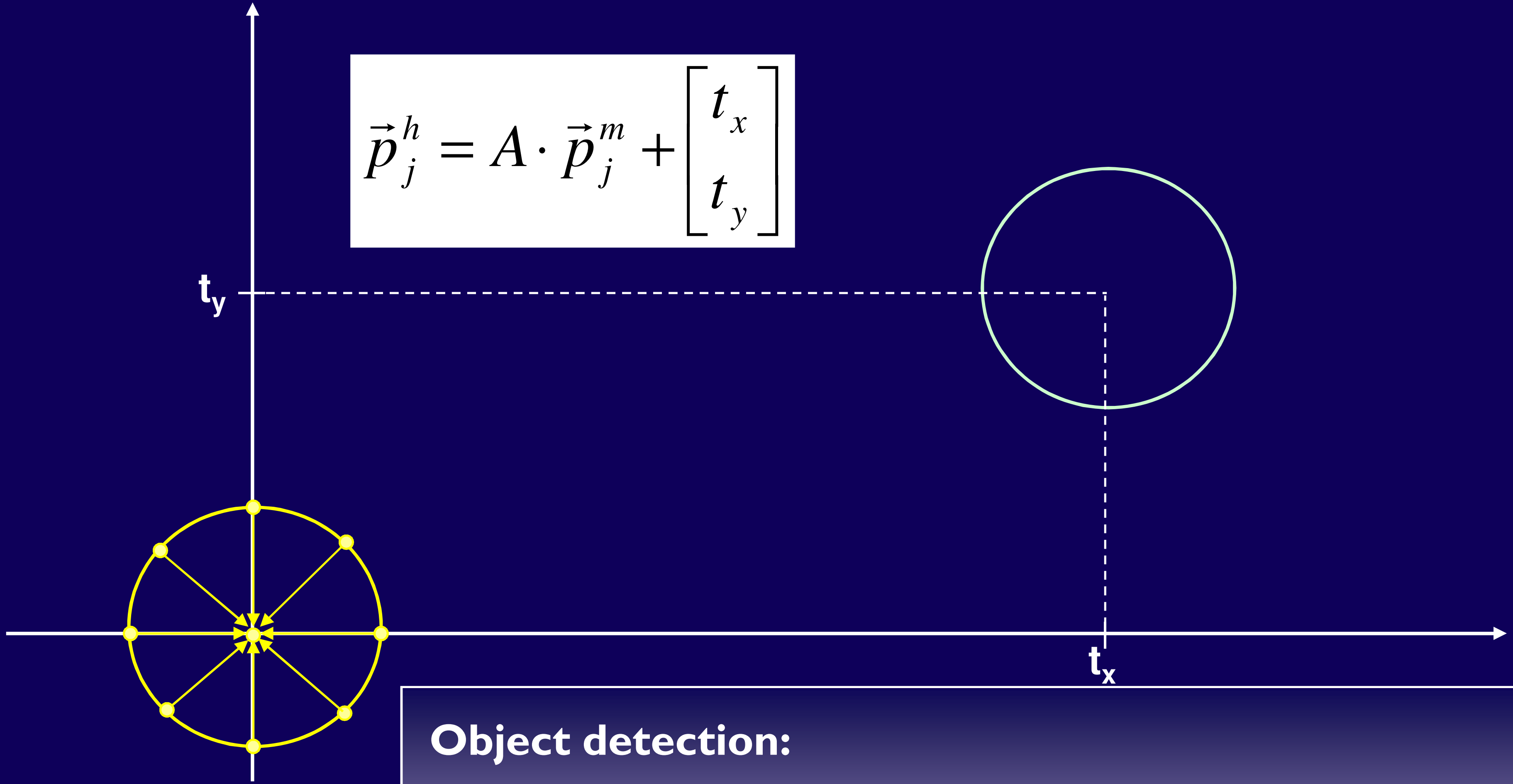
$$\vec{t}\,(\vec{p}_j^{\,m}, \vec{p}_j^{\,e}) = \vec{p}_j^{\,e} - \vec{p}_j^{\,m}$$

Generalized Hough Transform: Principle



$$\vec{t}\,(\vec{p}_j^{\,m}, \vec{p}_j^{\,e}) = \vec{p}_j^{\,e} - \vec{p}_j^{\,m}$$

Generalized Hough Transform: Principle

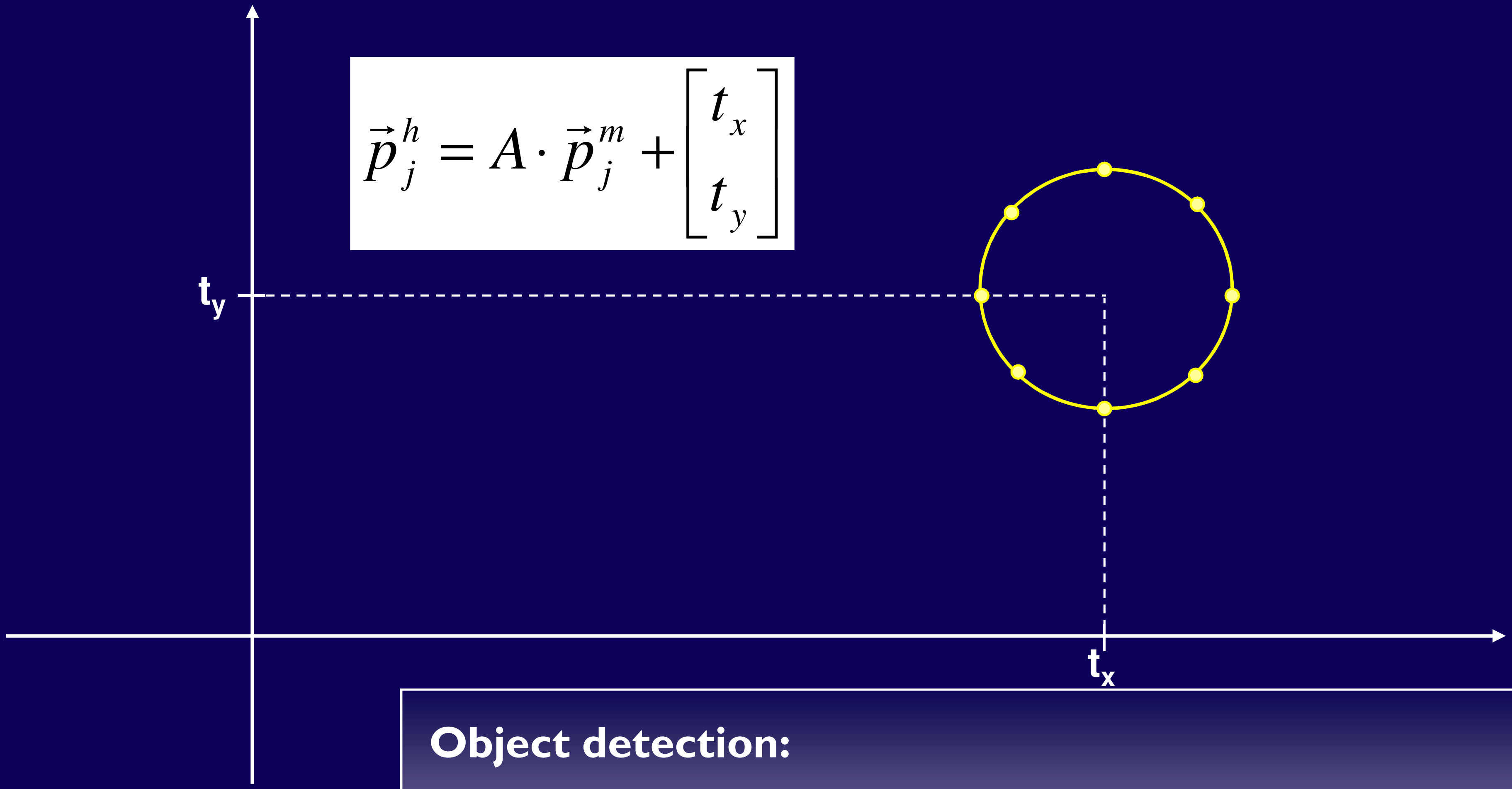


$$\vec{p}_j^{\,h} = A \cdot \vec{p}_j^{\,m} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

Object detection:

Transformation of shape model with identified parameters.

Generalized Hough Transform: Principle

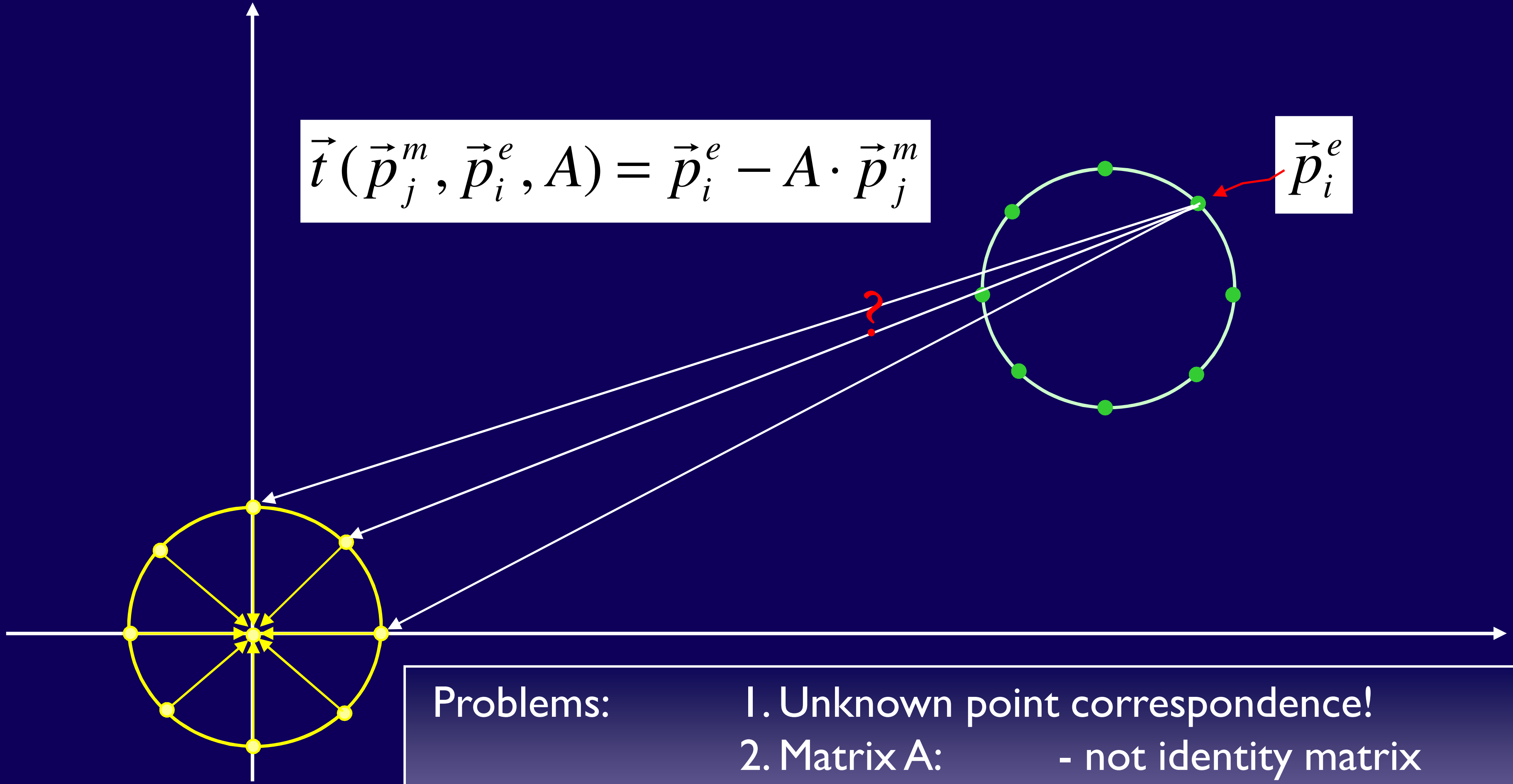


$$\vec{p}_j^{\,h} = A \cdot \vec{p}_j^{\,m} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

Object detection:

Transformation of shape model with identified parameters.

Generalized Hough Transform: Principle



$$\vec{t}\,(\vec{p}_j^{\,m}, \vec{p}_i^{\,e}, A) = \vec{p}_i^{\,e} - A \cdot \vec{p}_j^{\,m}$$

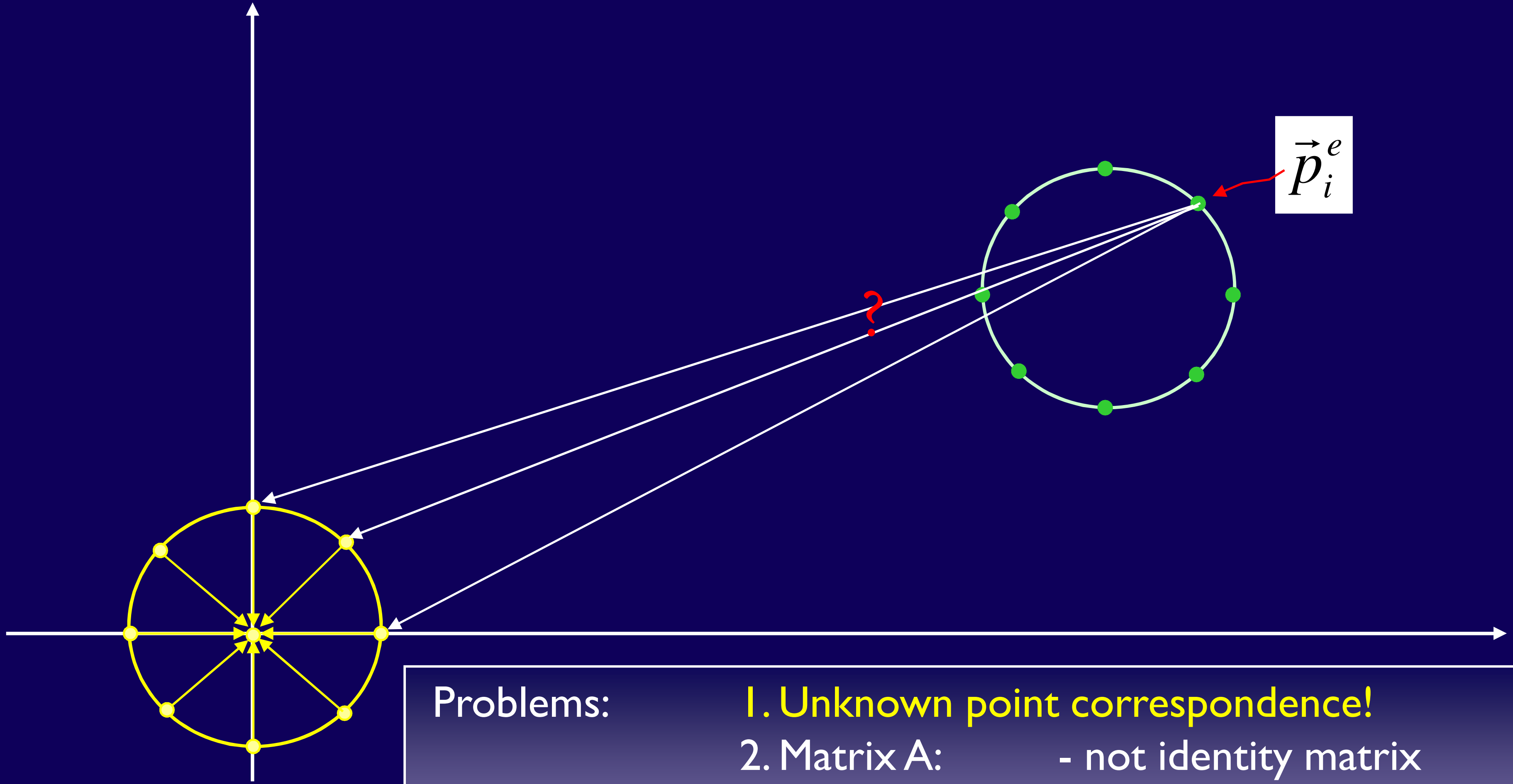
$\vec{p}_i^{\,e}$

Problems: 1. Unknown point correspondence!

2. Matrix A: - not identity matrix

- unknown

Generalized Hough Transform: Principle

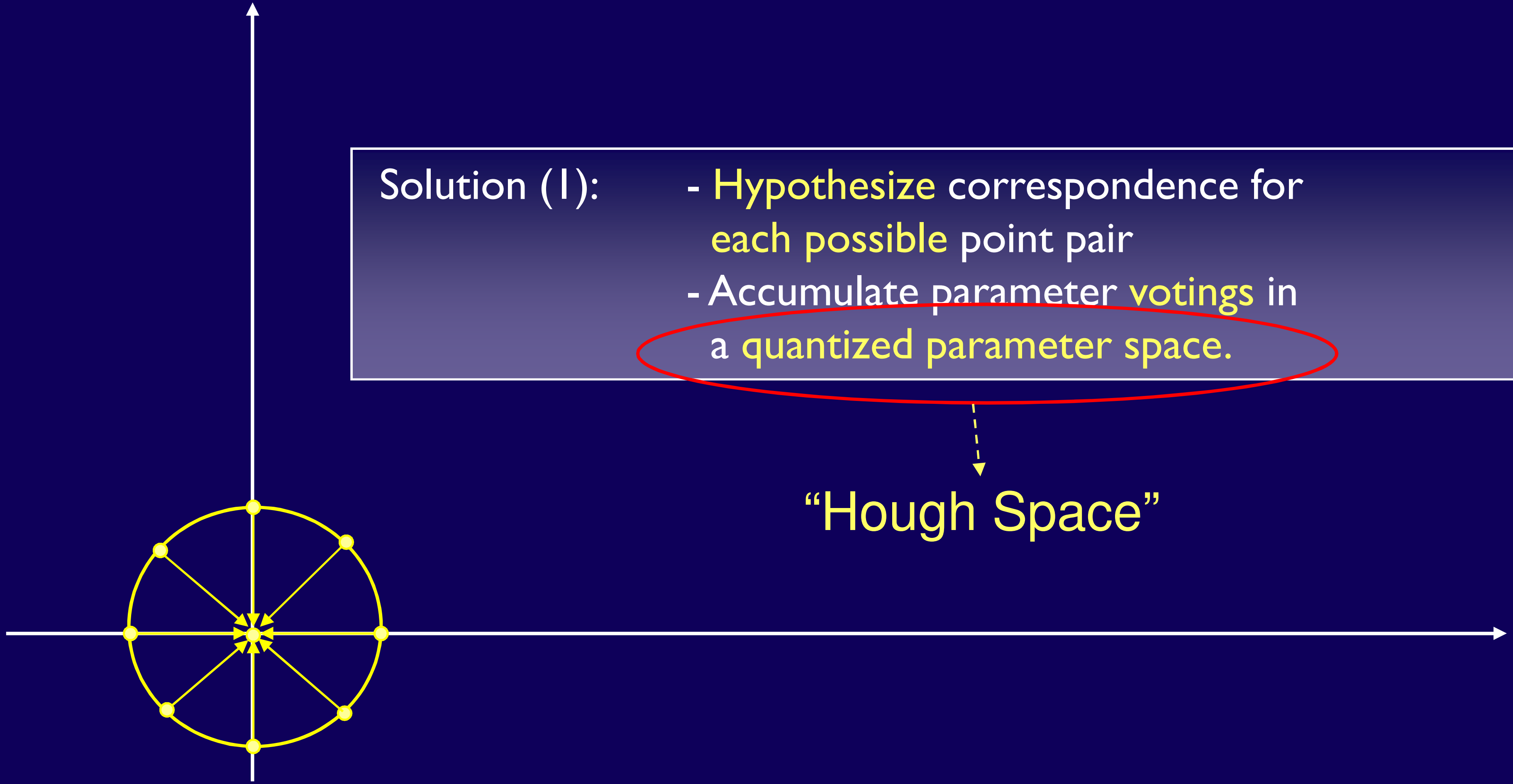


Problems: 1. Unknown point correspondence!

2. Matrix A: - not identity matrix

- unknown

Generalized Hough Transform: Principle



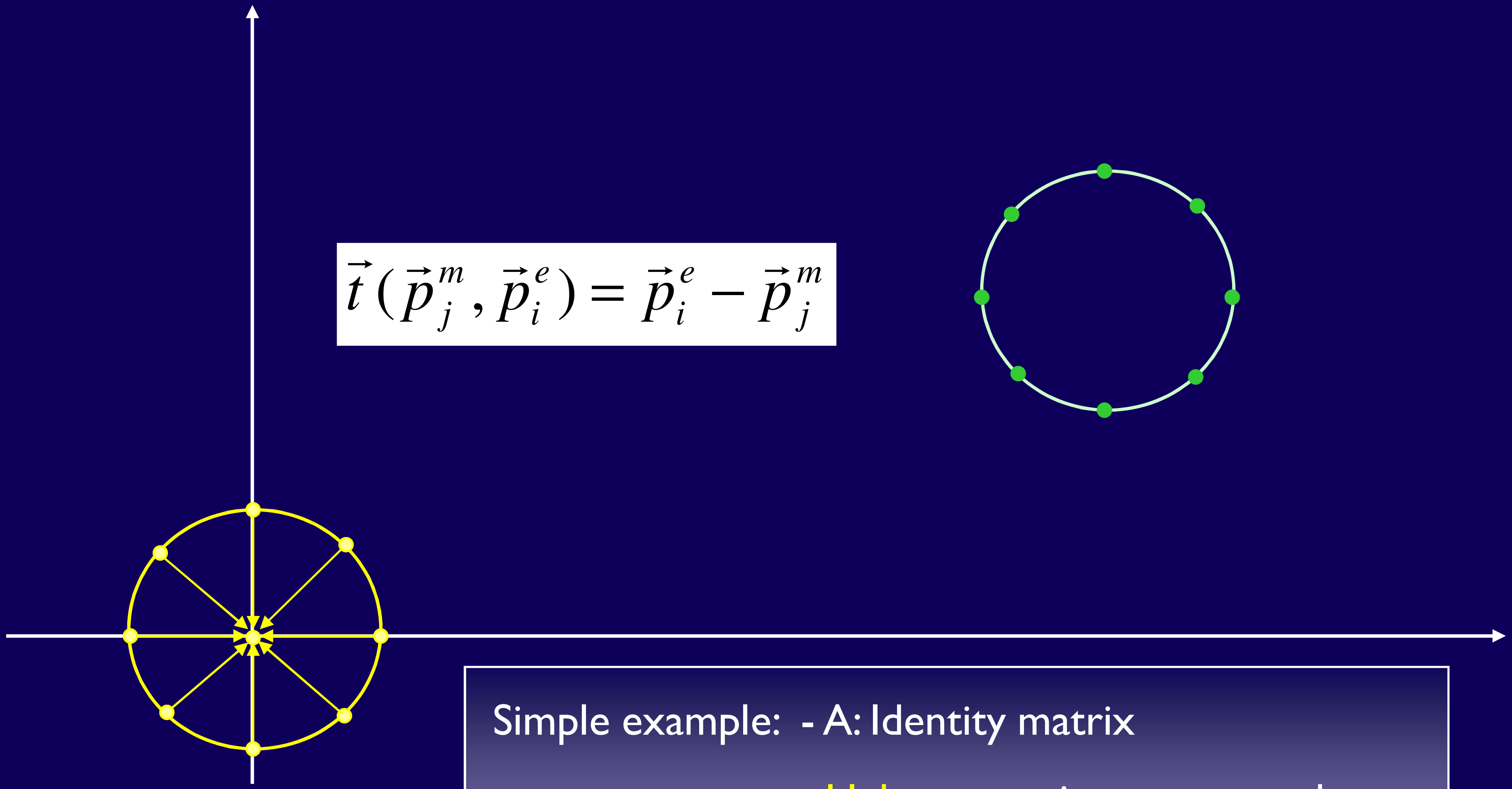
Solution (1): - Hypothesize correspondence for each possible point pair

- Accumulate parameter votings in a quantized parameter space.

"Hough Space"

Generalized Hough Transform: Principle

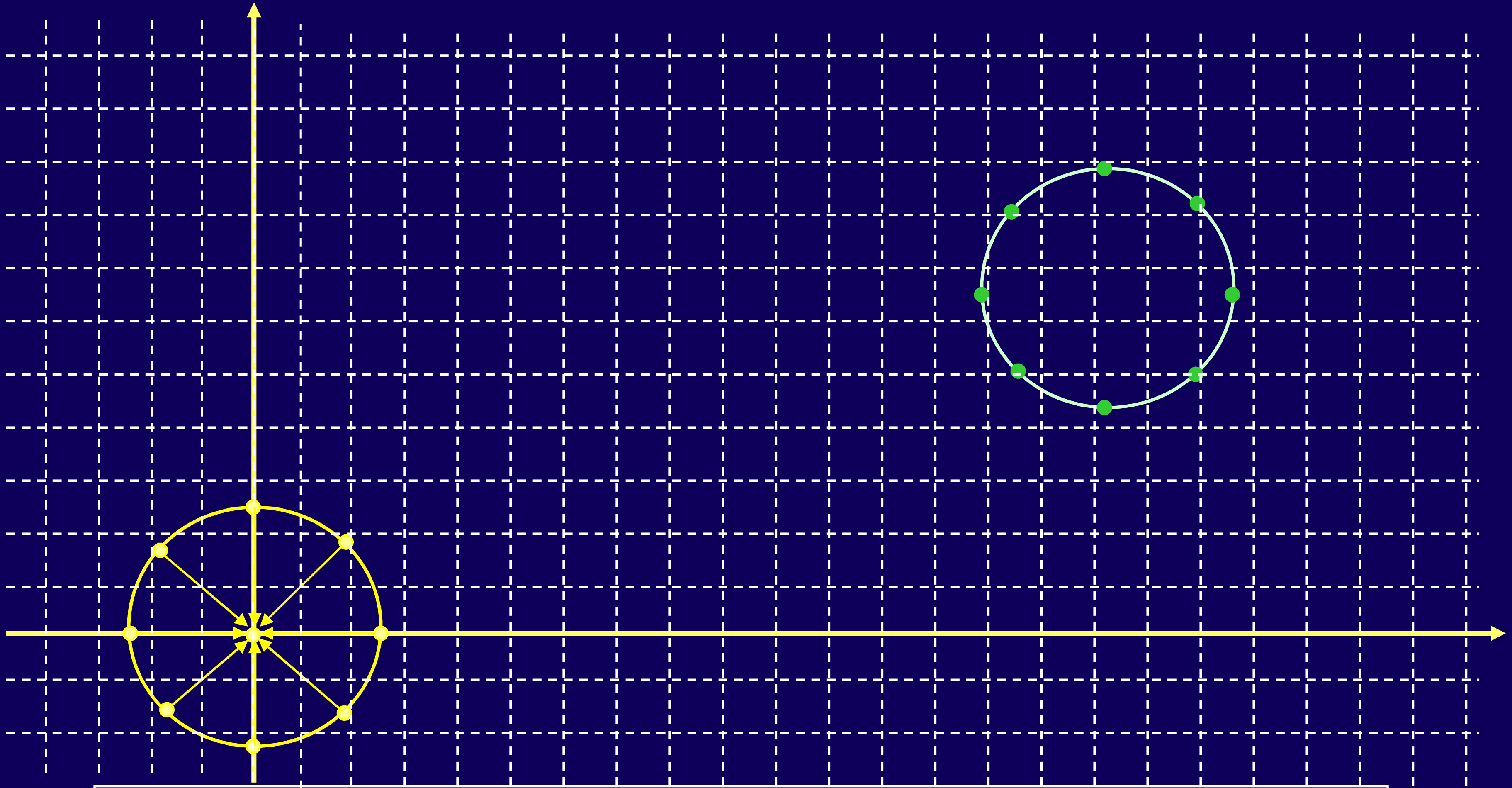
$$\vec{t}\,(\vec{p}_j^{\,m}, \vec{p}_i^{\,e}) = \vec{p}_i^{\,e} - \vec{p}_j^{\,m}$$



Simple example: - A: Identity matrix

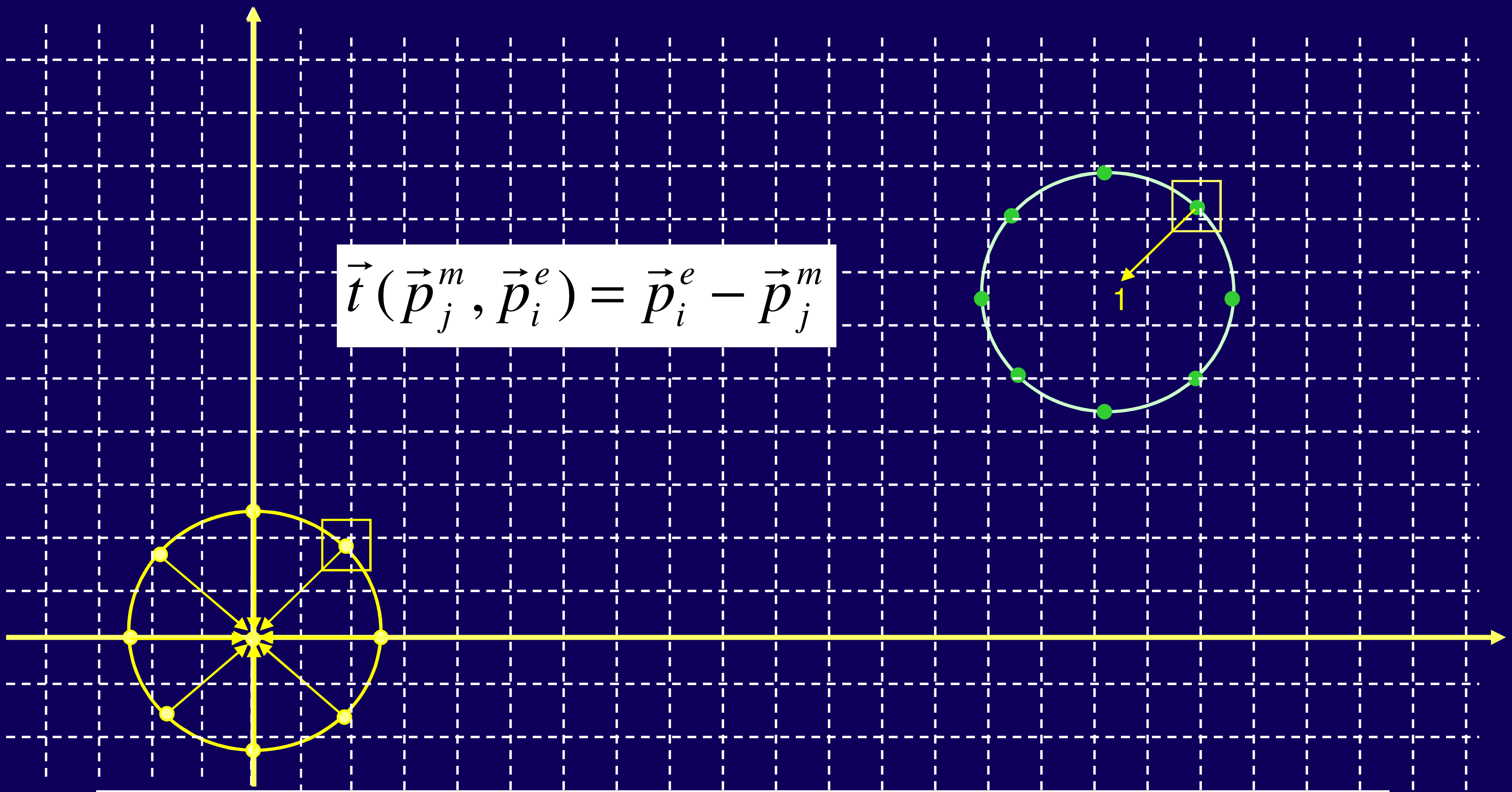
- Unknown point correspondence

Generalized Hough Transform: Principle



Step 1: Quantize (translation) parameter space

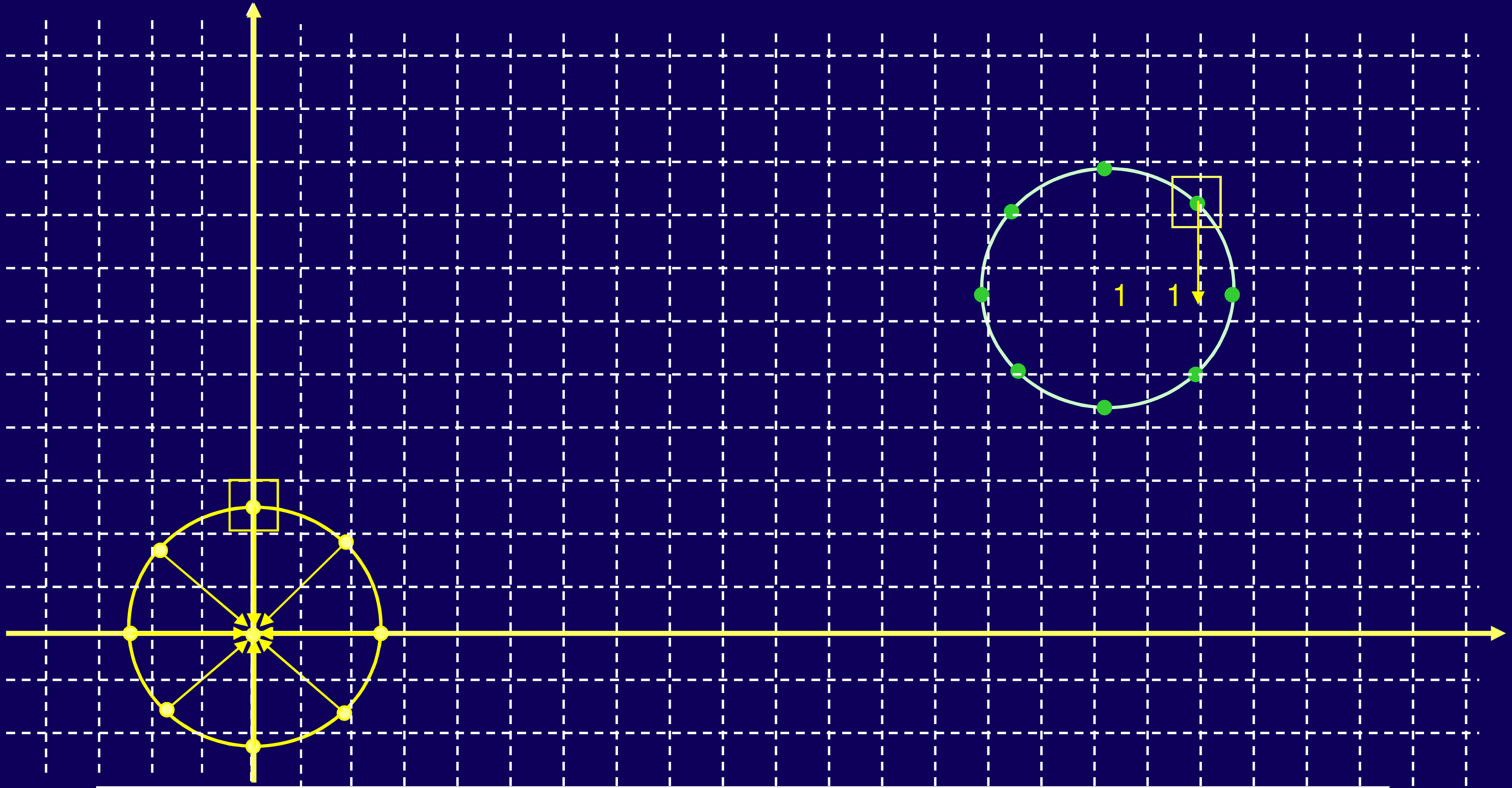
Generalized Hough Transform: Principle



$$\vec{t}\,(\vec{p}_j^{\,m}, \vec{p}_i^{\,e}) = \vec{p}_i^{\,e} - \vec{p}_j^{\,m}$$

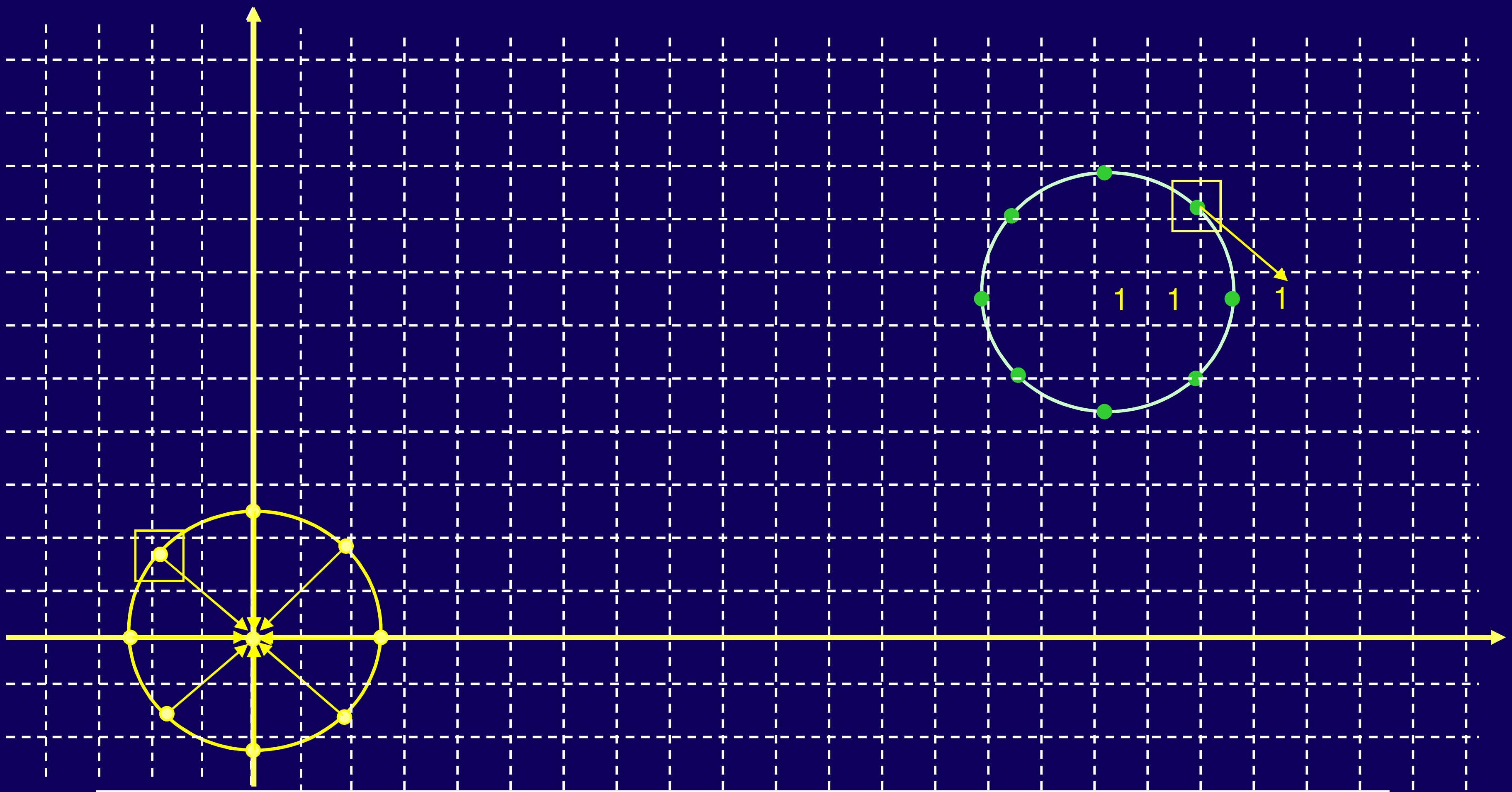
Step 2: Hypothesize point correspondences + accumulate votes

Generalized Hough Transform: Principle



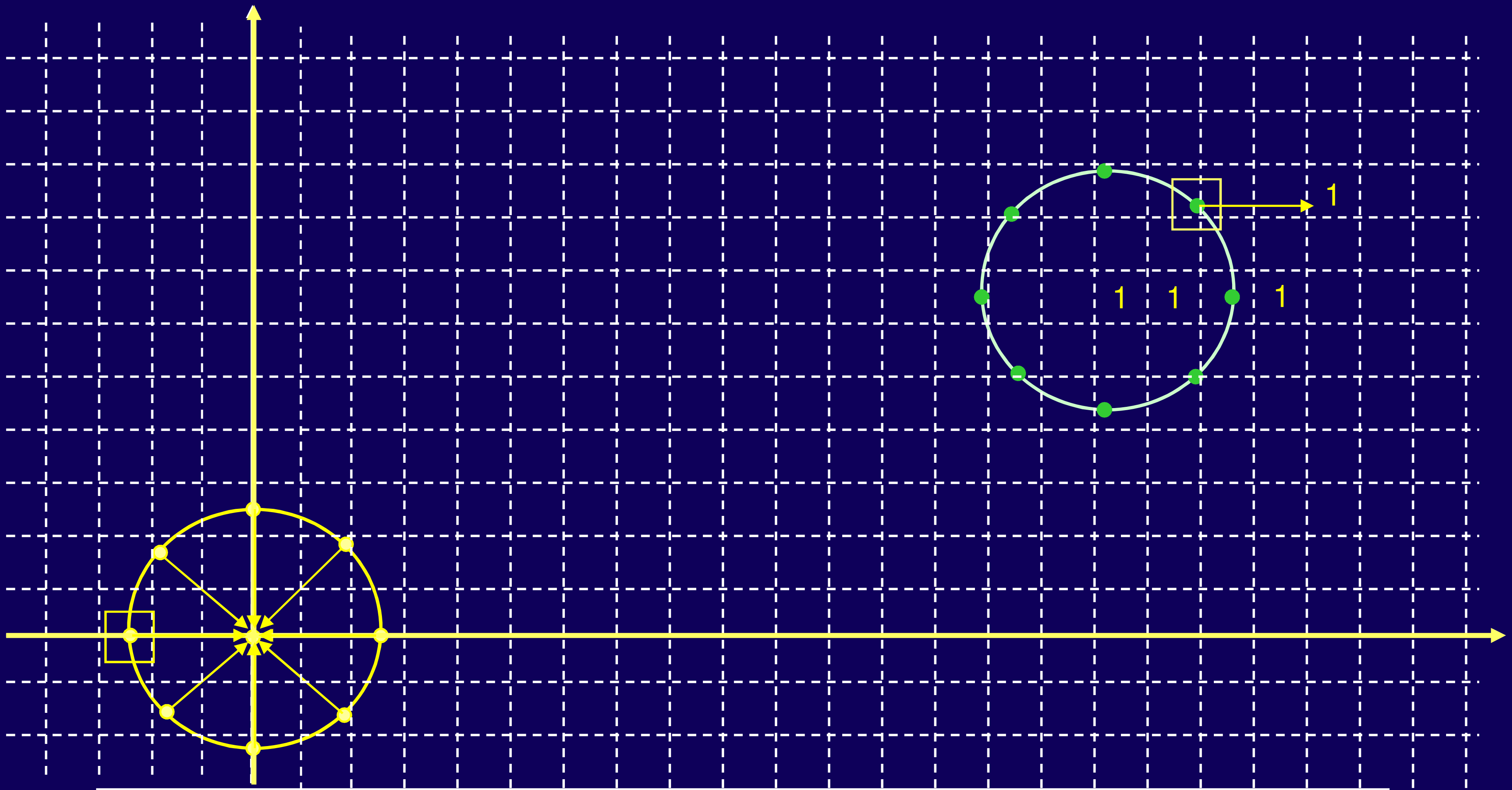
Step 2: Hypothesize point correspondences + accumulate votes

Generalized Hough Transform: Principle



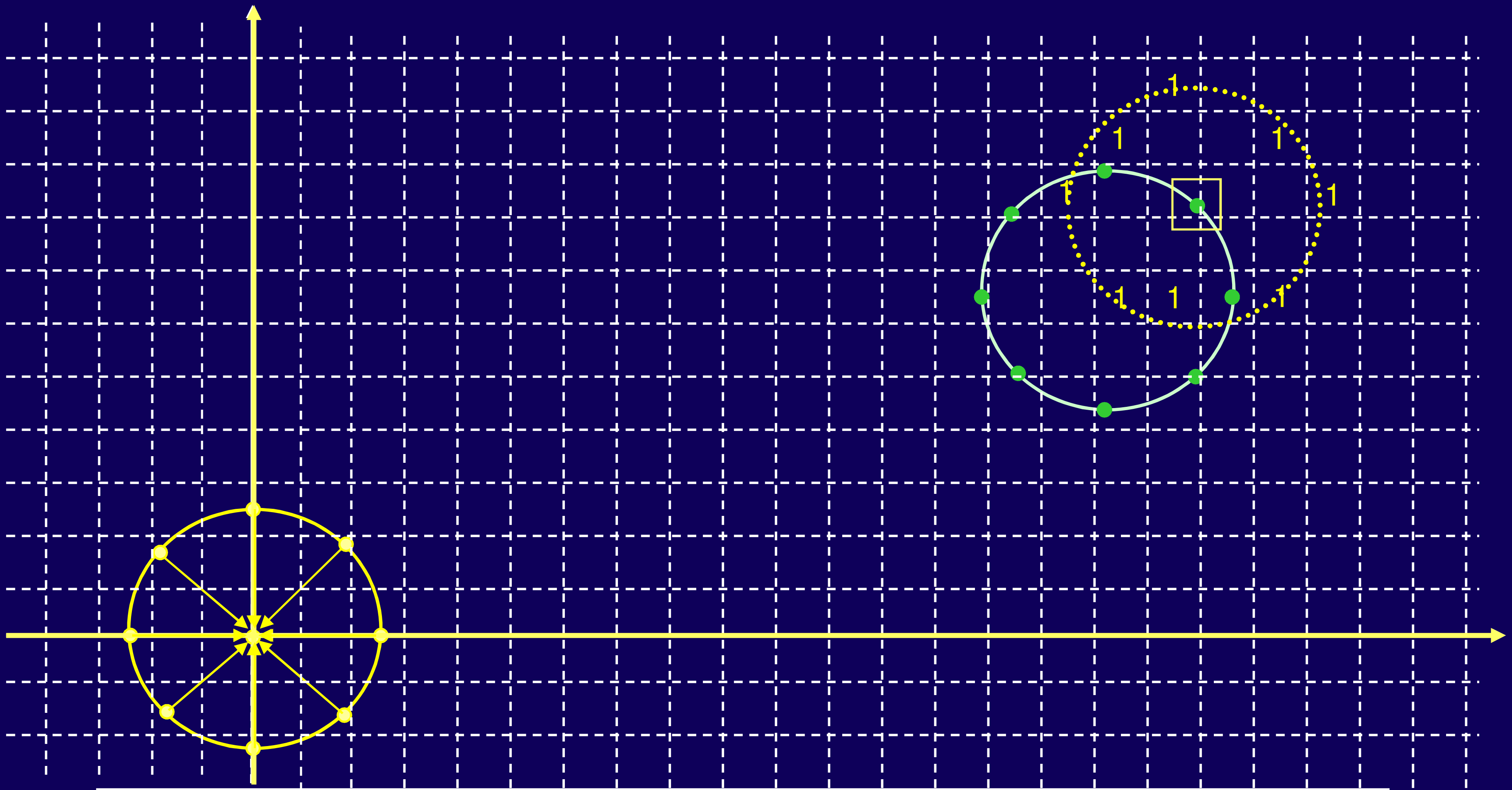
Step 2: Hypothesize point correspondences + accumulate votes

Generalized Hough Transform: Principle



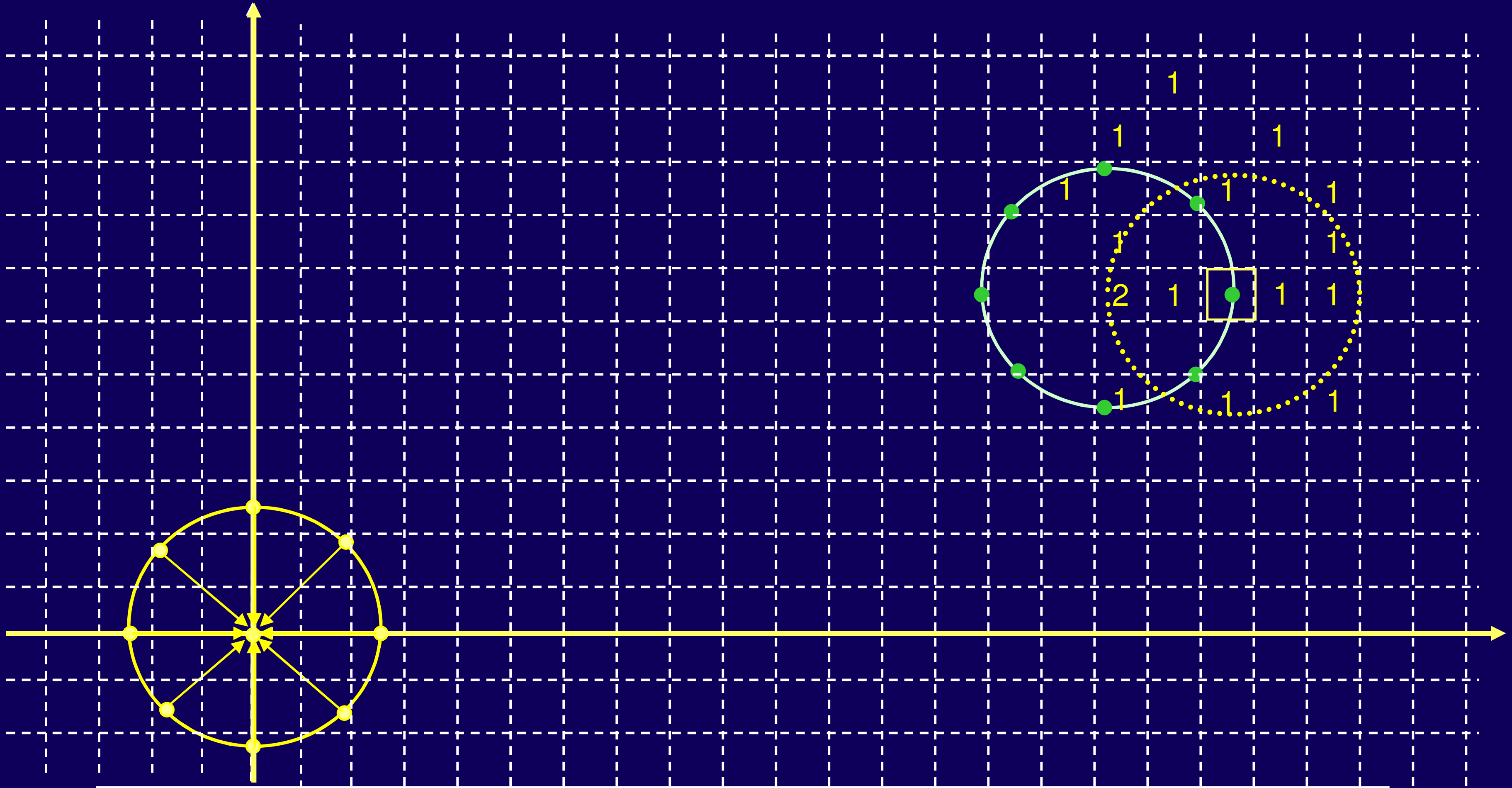
Step 2: Hypothesize point correspondences + accumulate votes

Generalized Hough Transform: Principle



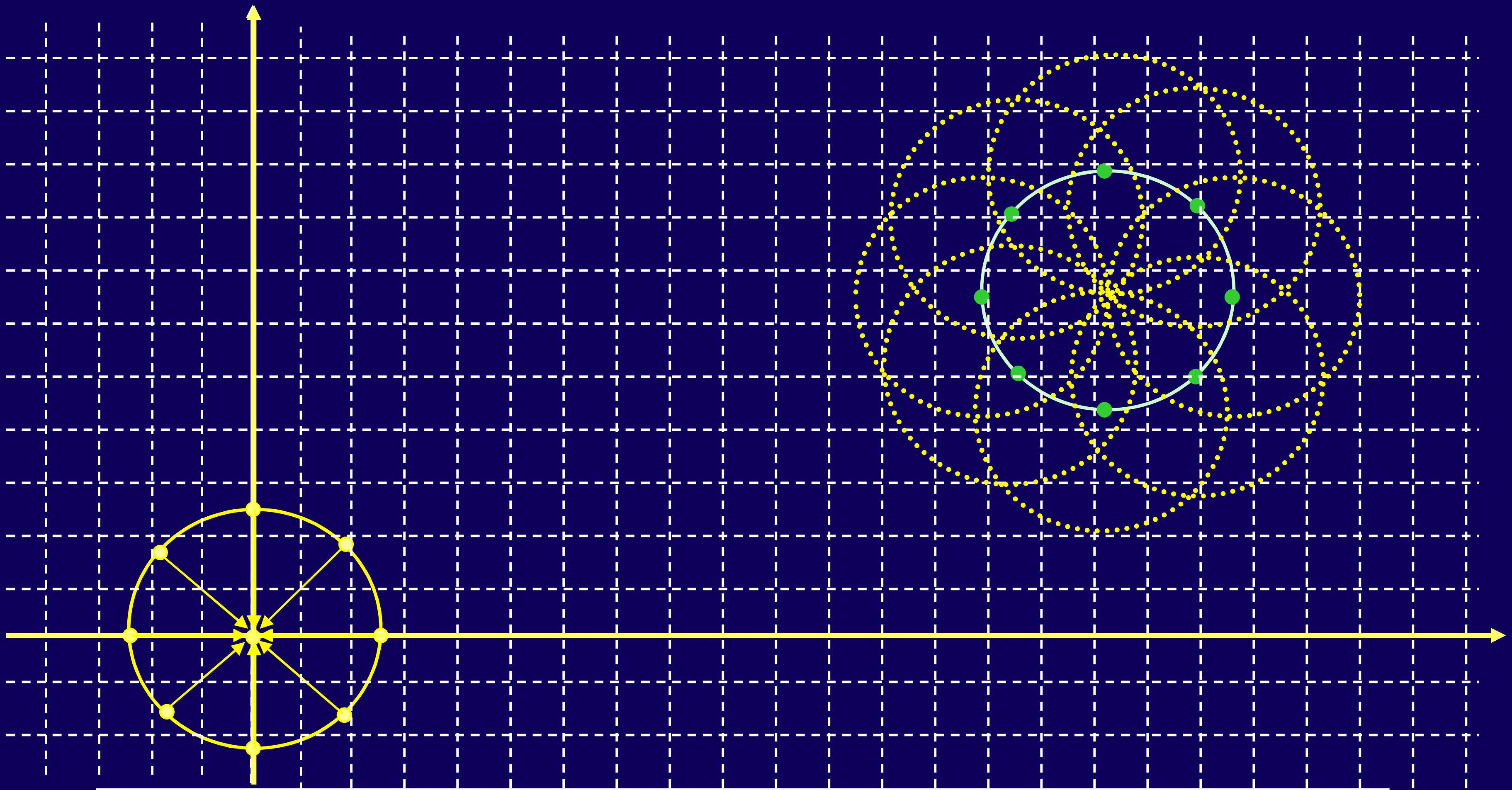
Step 2: Hypothesize point correspondences + accumulate votes

Generalized Hough Transform: Principle



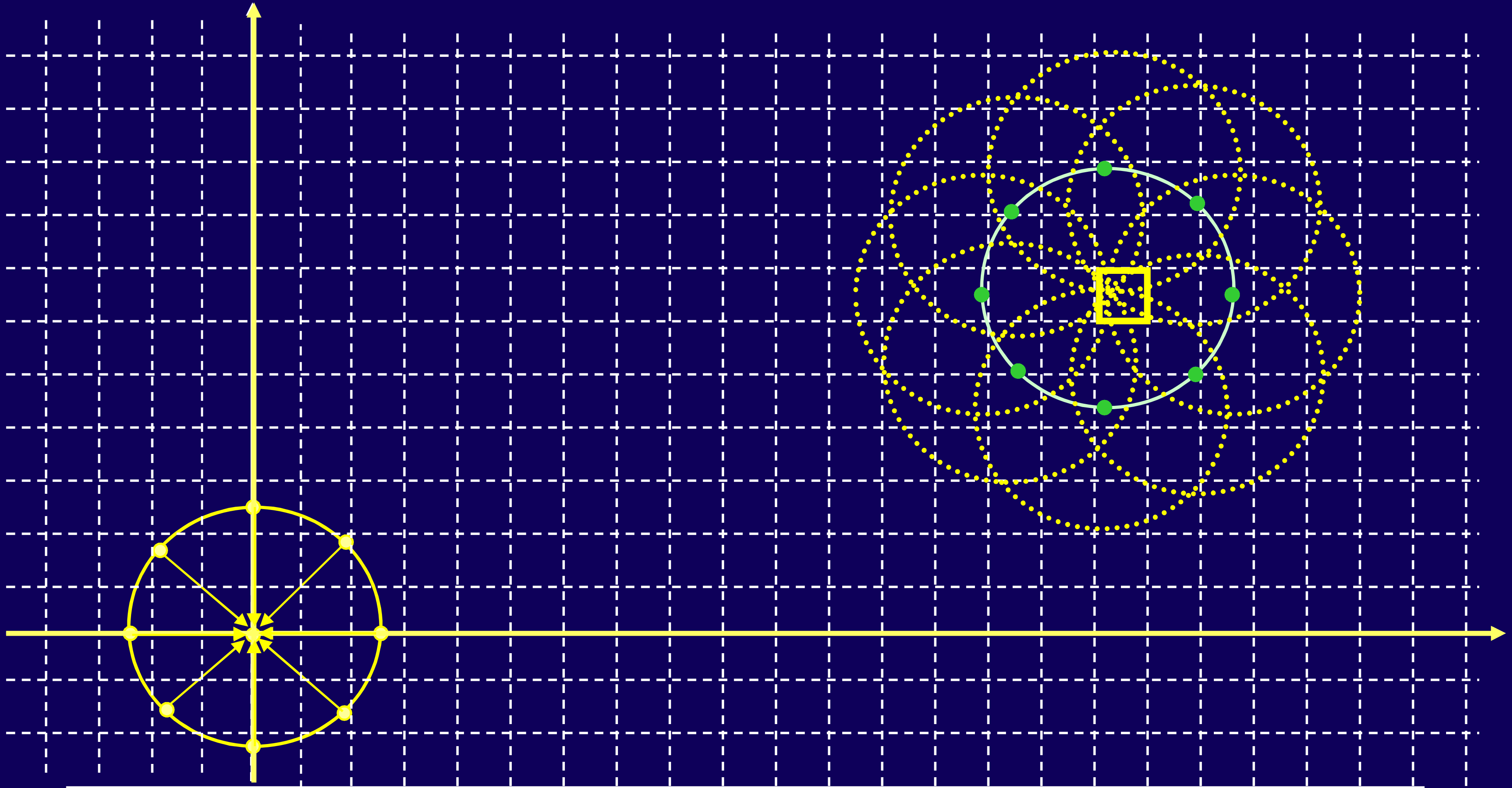
Step 2: Hypothesize point correspondences + accumulate votes

Generalized Hough Transform: Principle



Step 2: Hypothesize point correspondences + accumulate votes

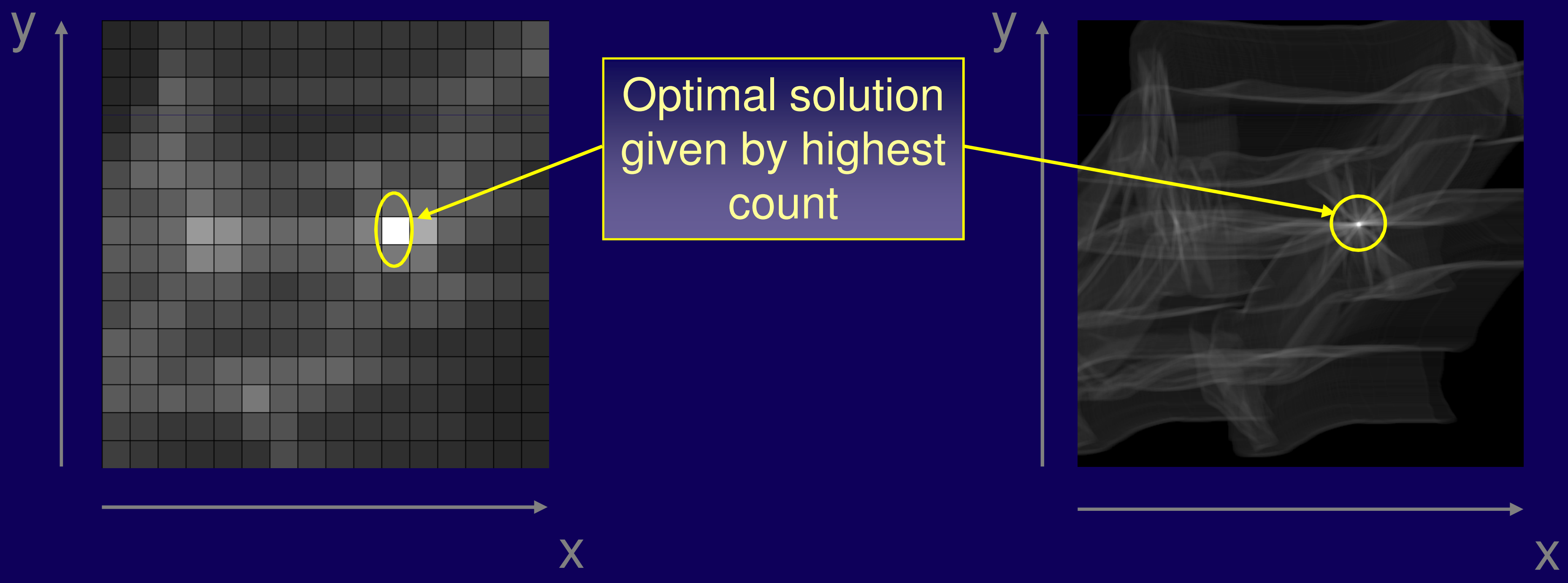
Generalized Hough Transform: Principle



Step 3: Search for cell with highest number of votes Ł optimal solution

Generalized Hough Transform: Principle

Examples for Hough Space appearance:

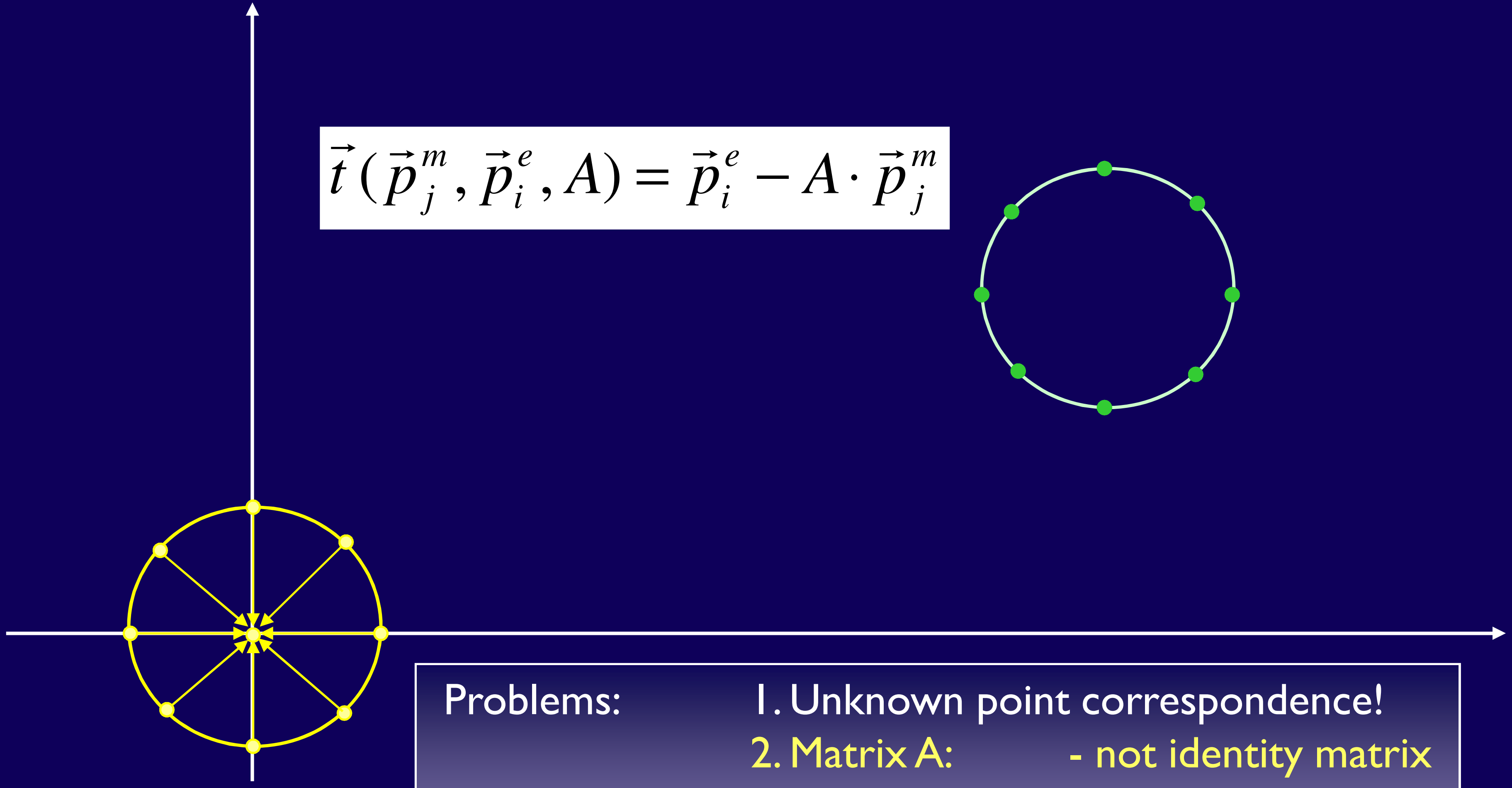


Optimal solution given by highest count

Large accumulator cells

Small accumulator cells

Generalized Hough Transform: Principle



$$\vec{t}\,(\vec{p}_j^{\,m}, \vec{p}_i^{\,e}, A) = \vec{p}_i^{\,e} - A \cdot \vec{p}_j^{\,m}$$

Problems: 1. Unknown point correspondence!

2. Matrix A: - not identity matrix

- unknown

Generalized Hough Transform: Principle

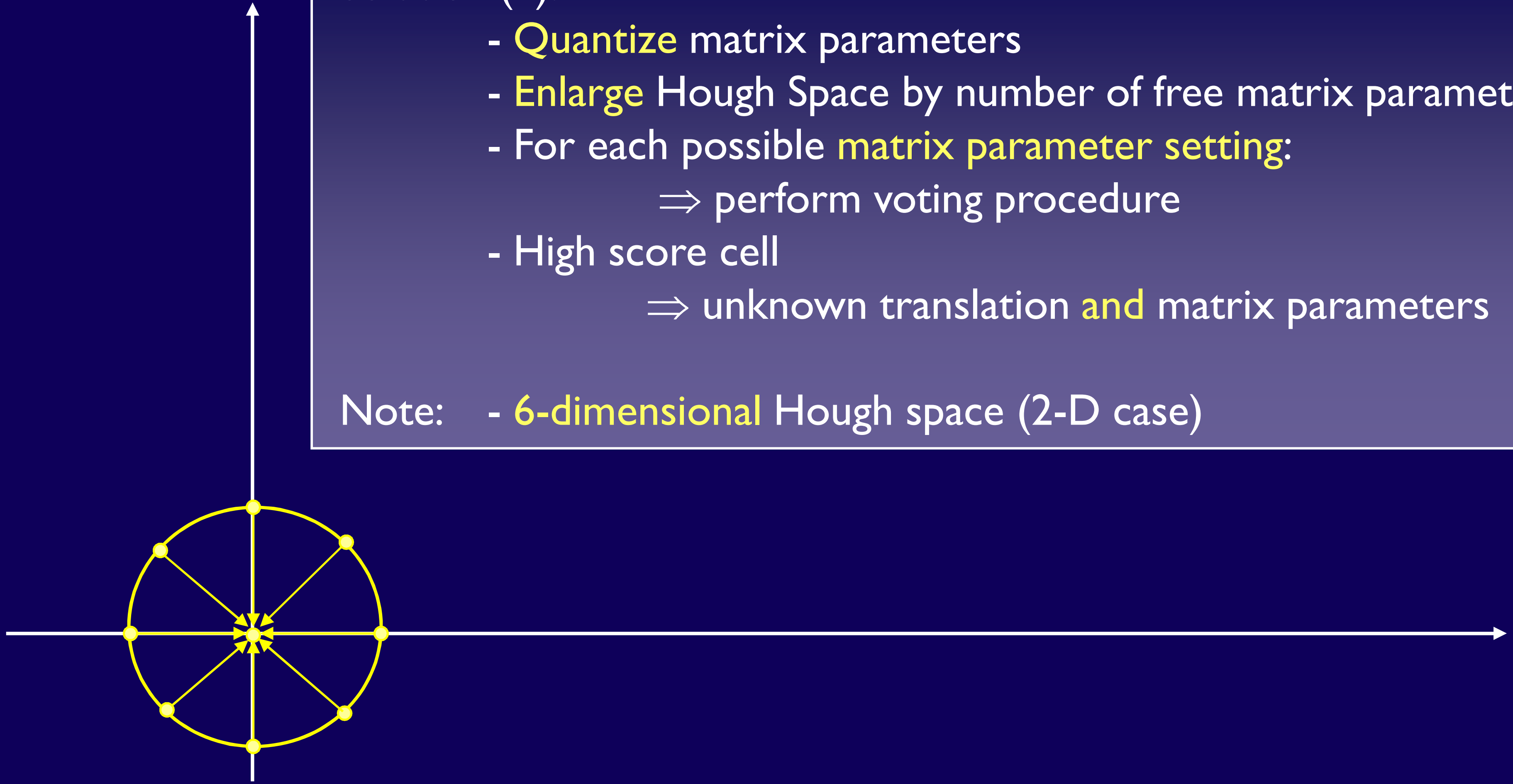
Solution (2):

- Quantize matrix parameters
- Enlarge Hough Space by number of free matrix parameters
- For each possible matrix parameter setting:

  ⇒ perform voting procedure
- High score cell

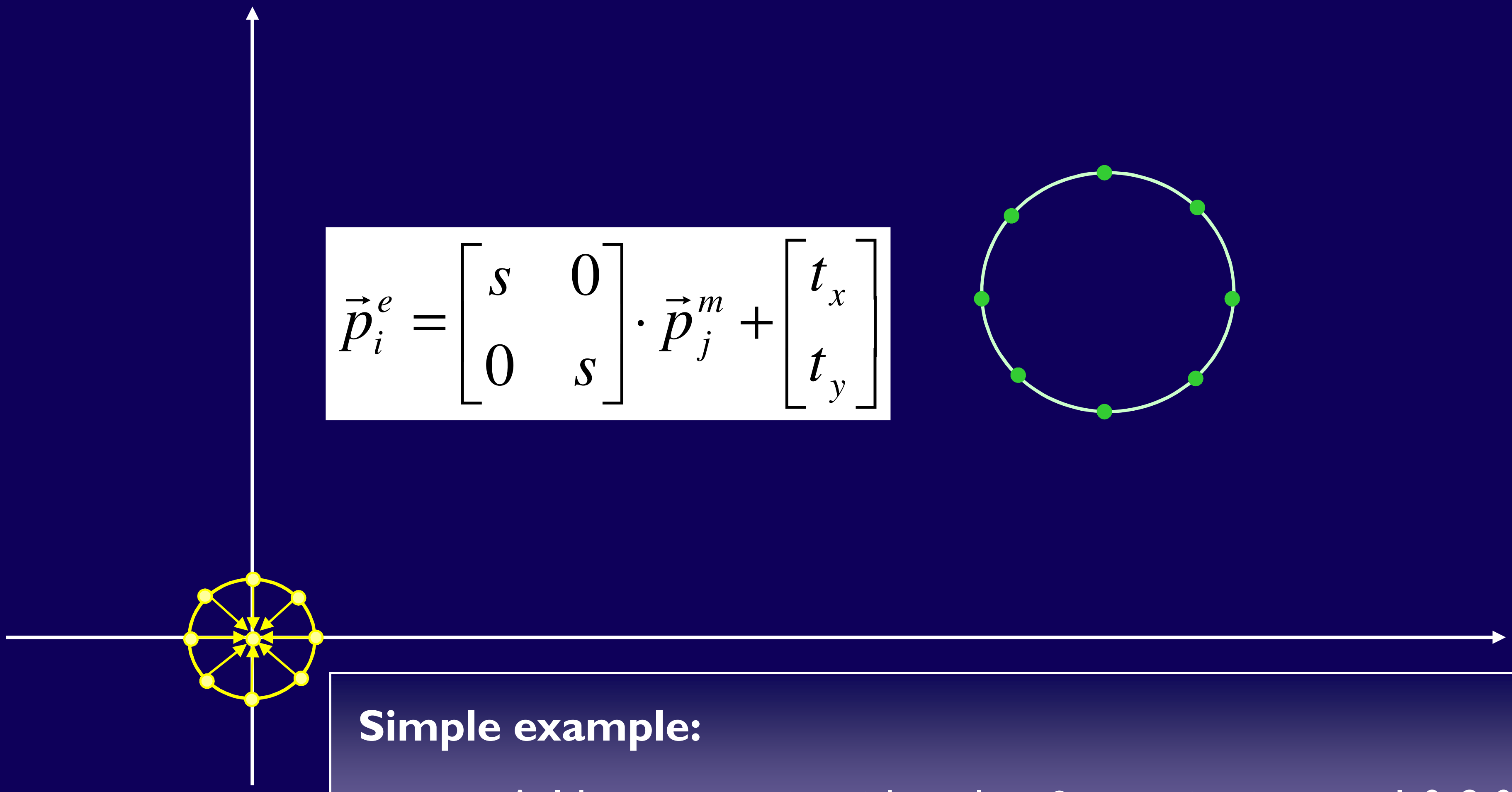
  ⇒ unknown translation and matrix parameters

Note: - 6-dimensional Hough space (2-D case)



Generalized Hough Transform: Principle

$$\vec{p}_i^{\,e} = \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} \cdot \vec{p}_j^{\,m} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

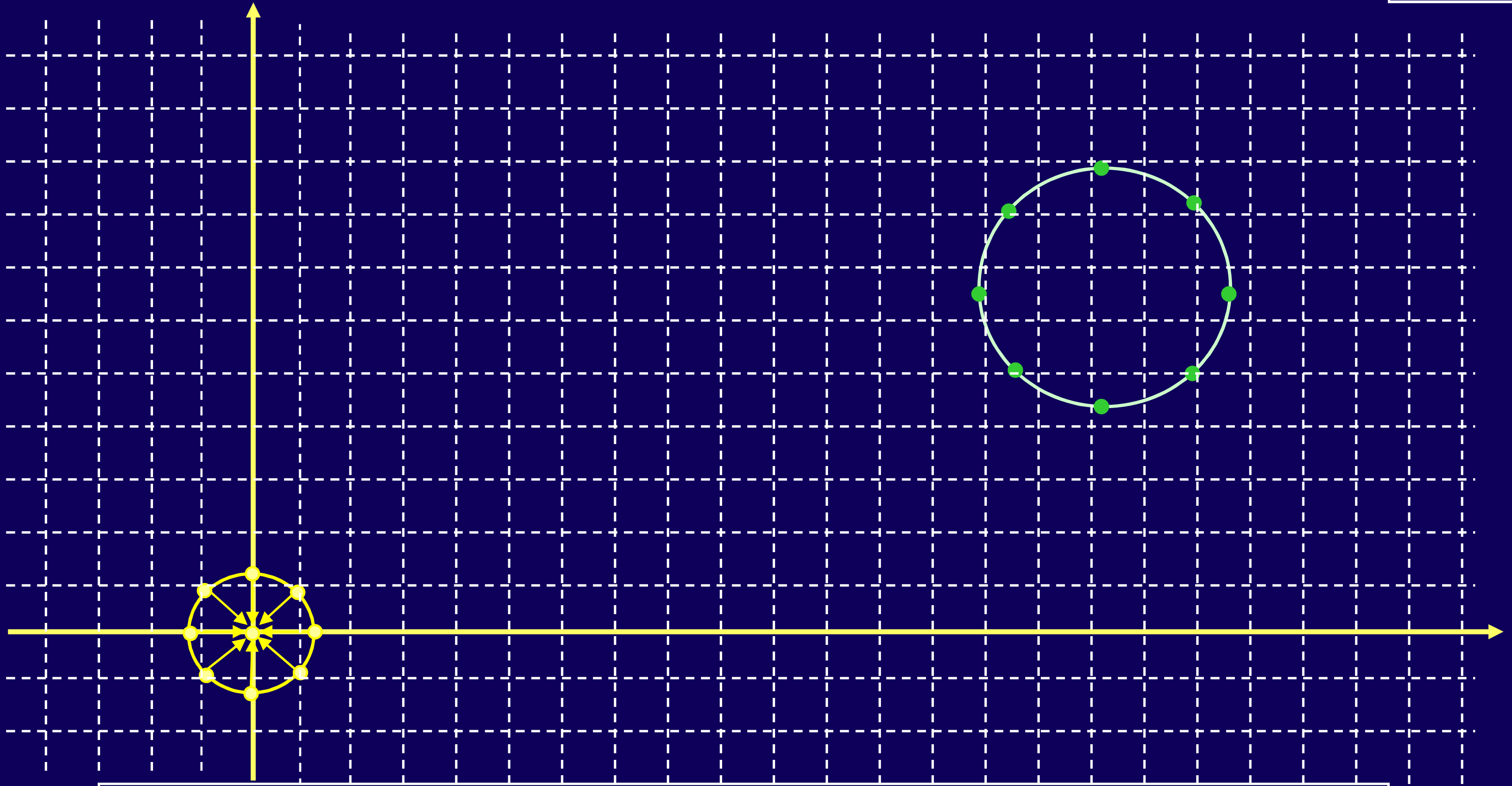


Simple example:

A: Identity matrix with scaling & quantization: s = 1.0, 2.0, 4.0

Generalized Hough Transform: Principle

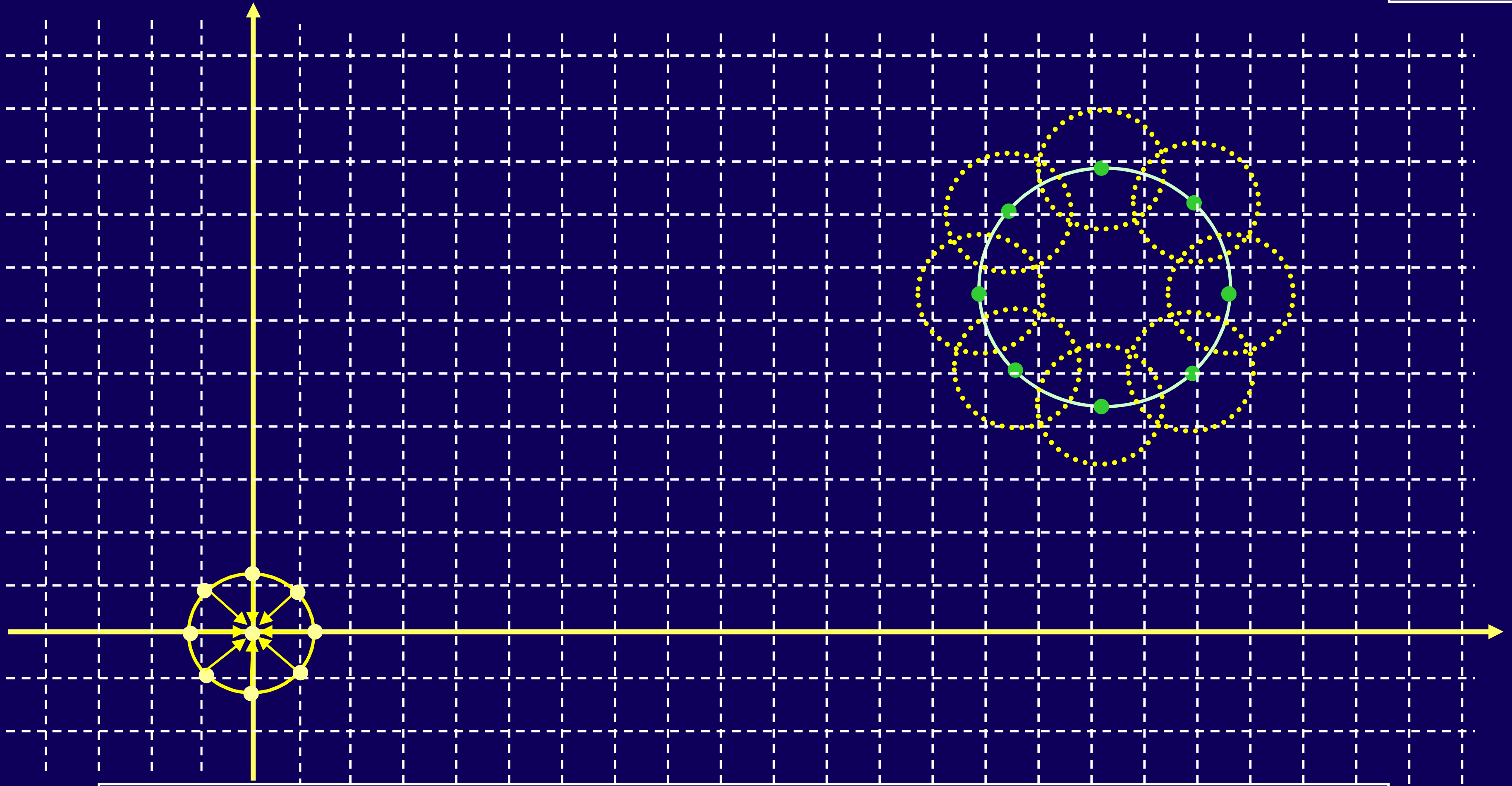
s = 1.0



Step 1: Quantize 3-D parameter space

Generalized Hough Transform: Principle

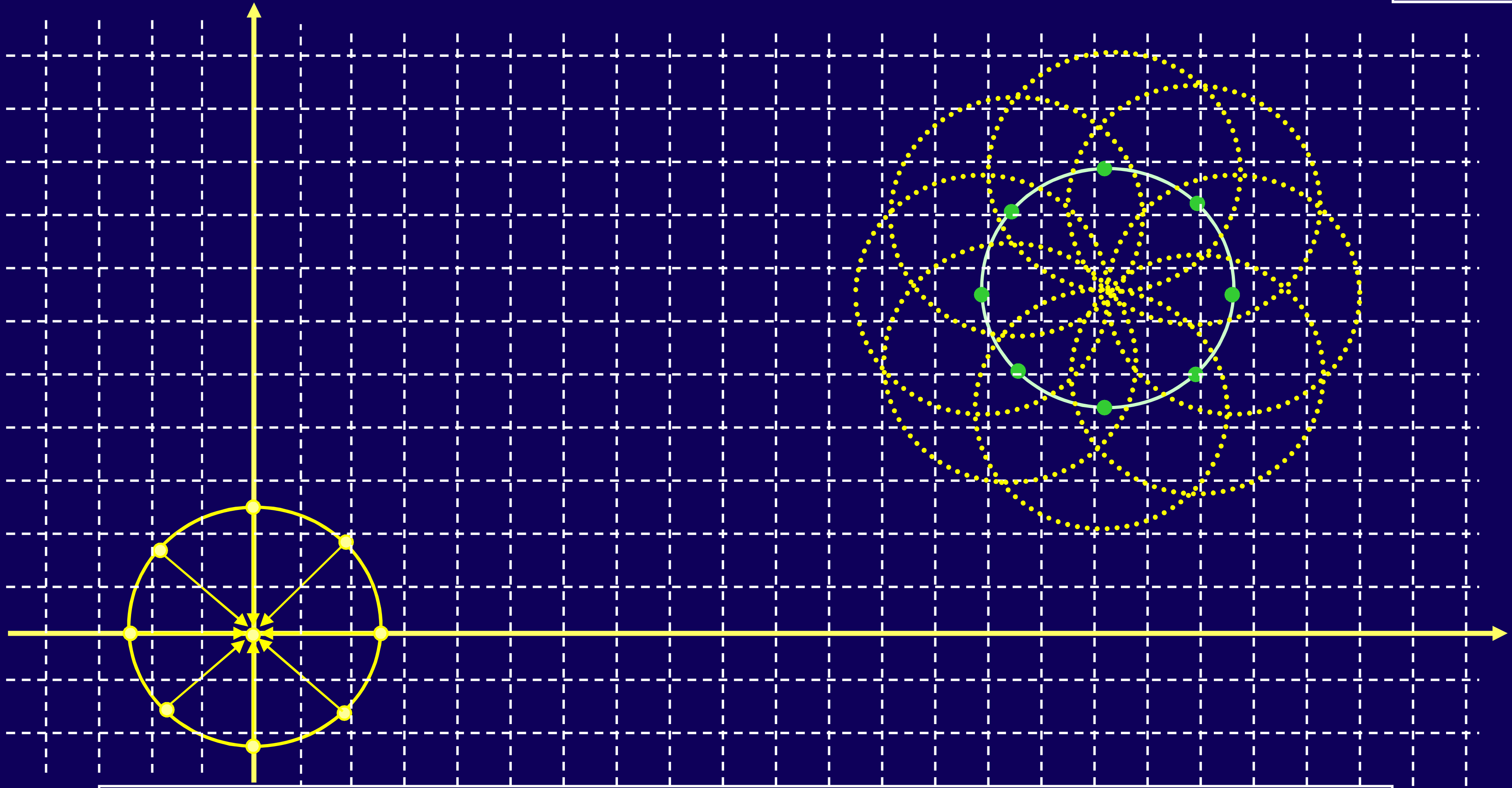
s = 1.0



Step 2: Perform voting for each scaling

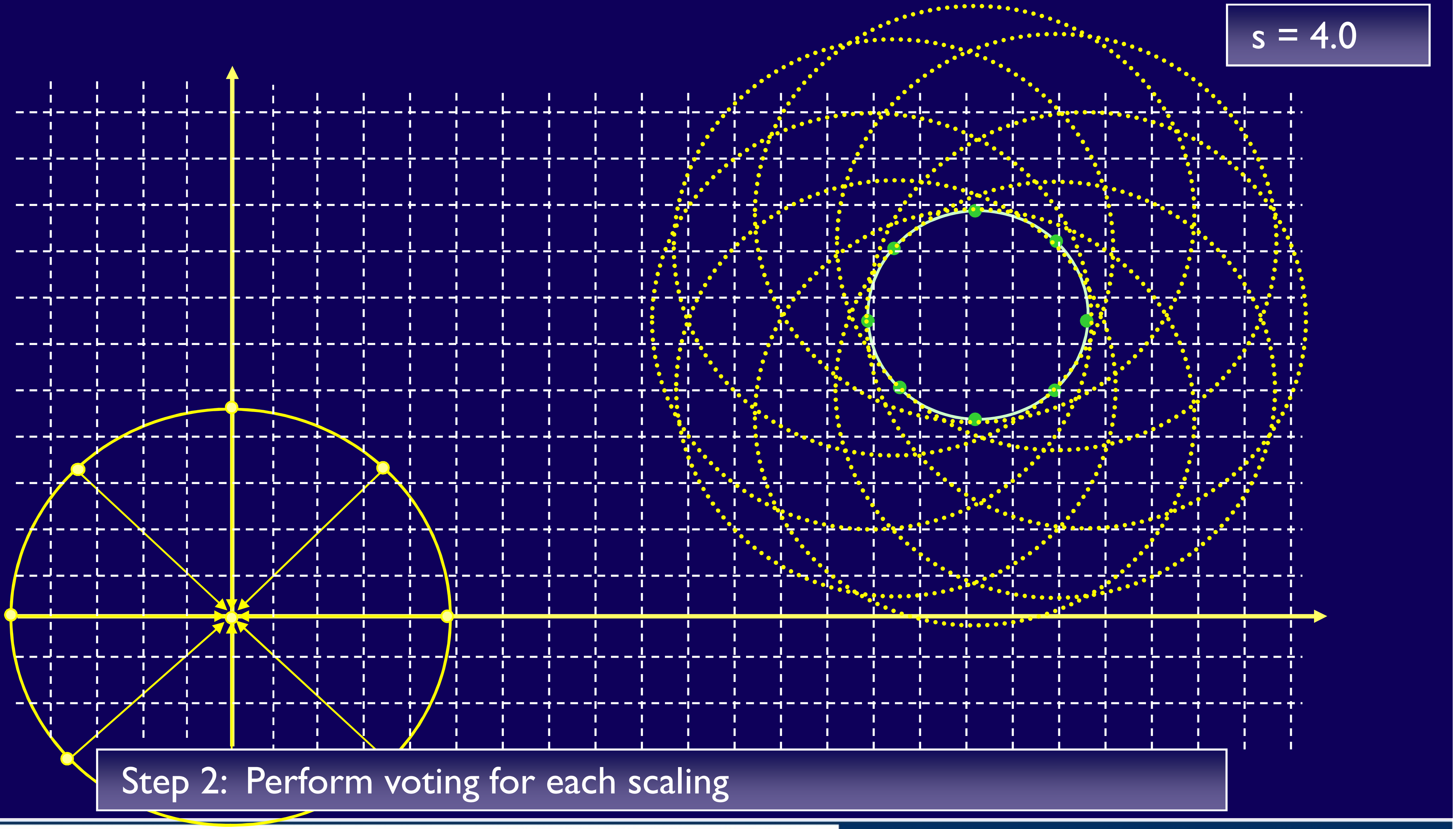
Generalized Hough Transform: Principle

s = 2.0



Step 2: Perform voting for each scaling

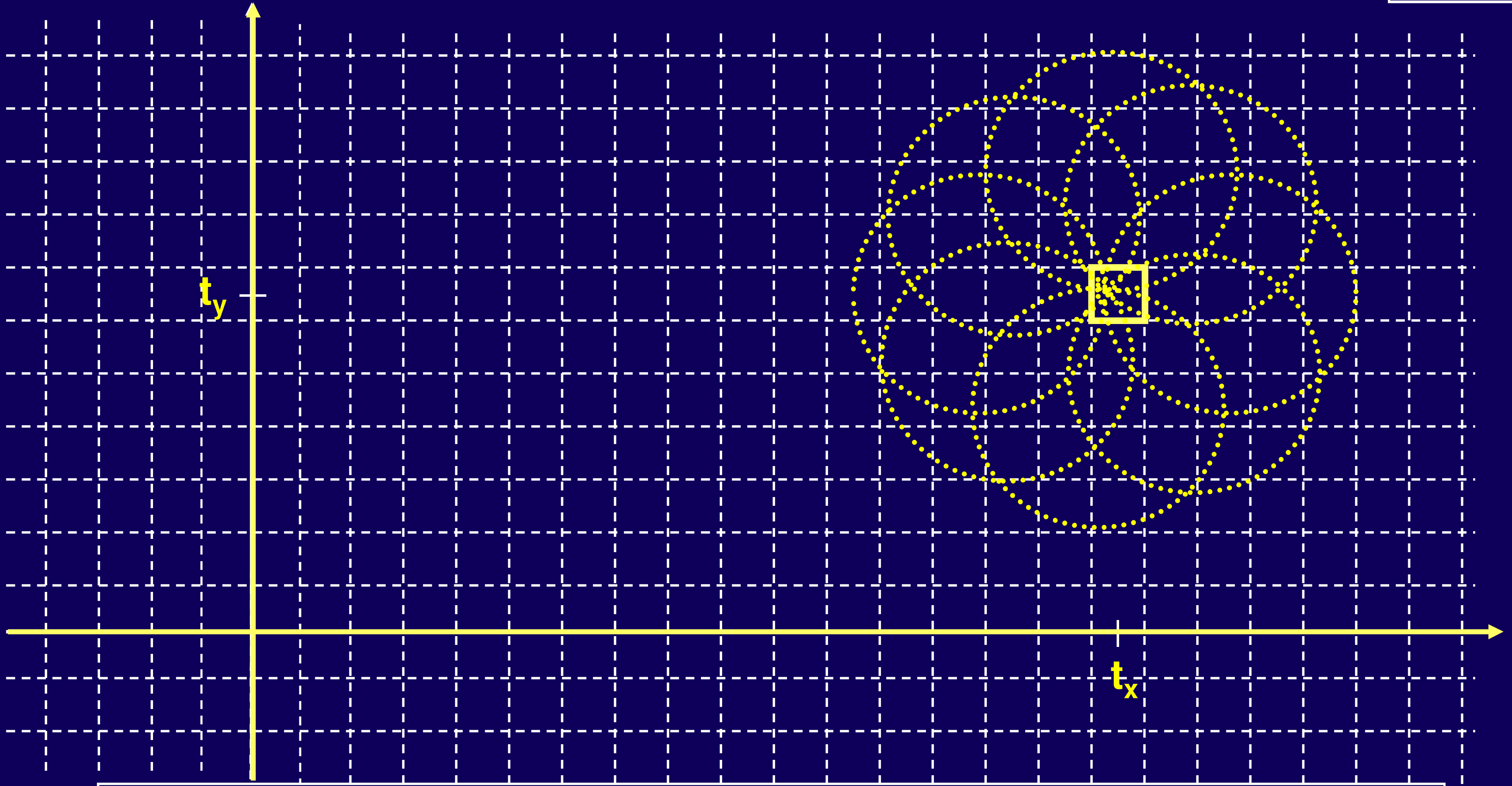
Generalized Hough Transform: Principle



Step 2: Perform voting for each scaling

Generalized Hough Transform: Principle

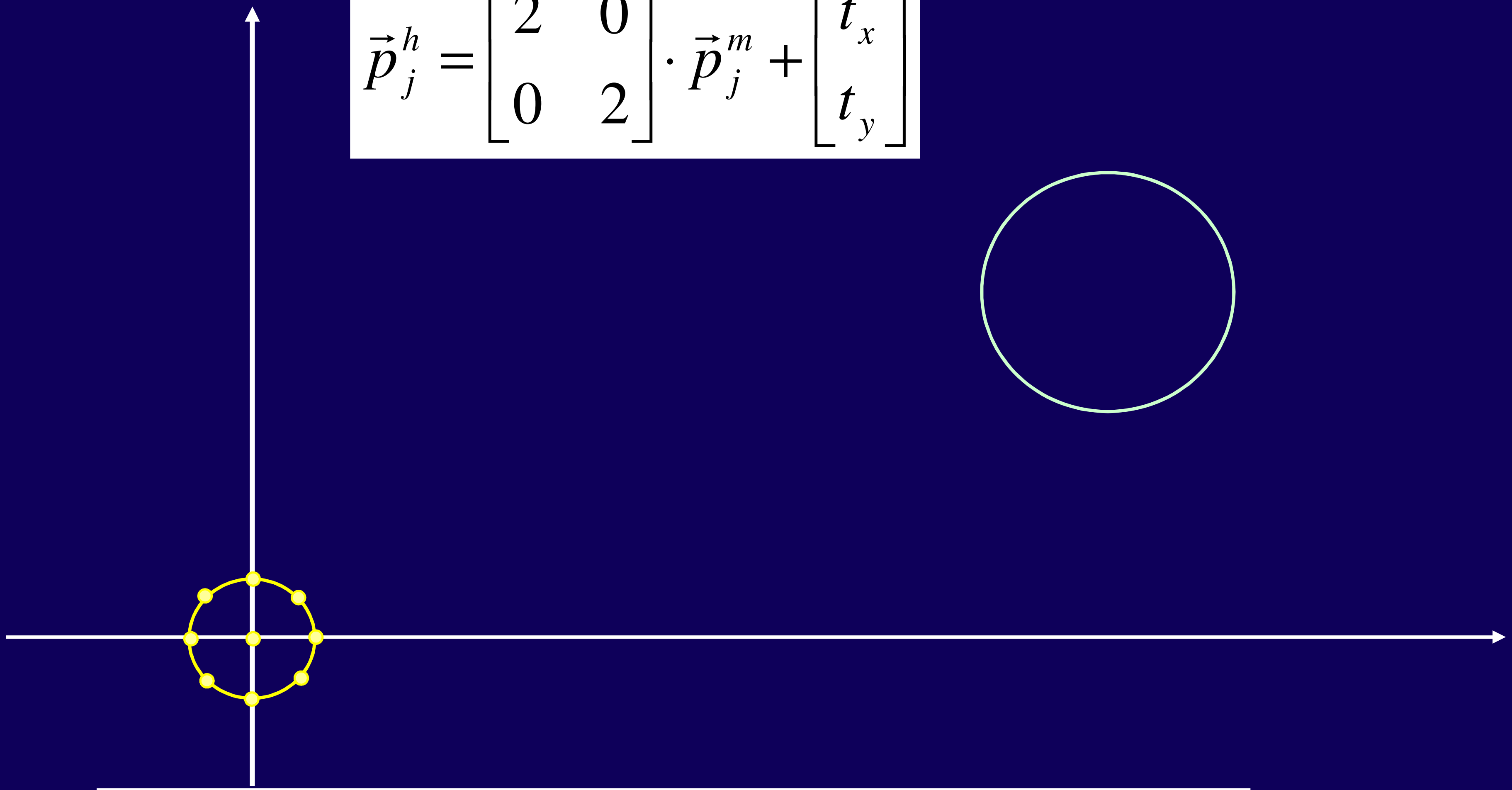
s = 2.0



Step 3: Search for cell with highest number of votes in 3-D Hough Space

Generalized Hough Transform: Principle

$$\vec{p}_j^{\,h} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \cdot \vec{p}_j^{\,m} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

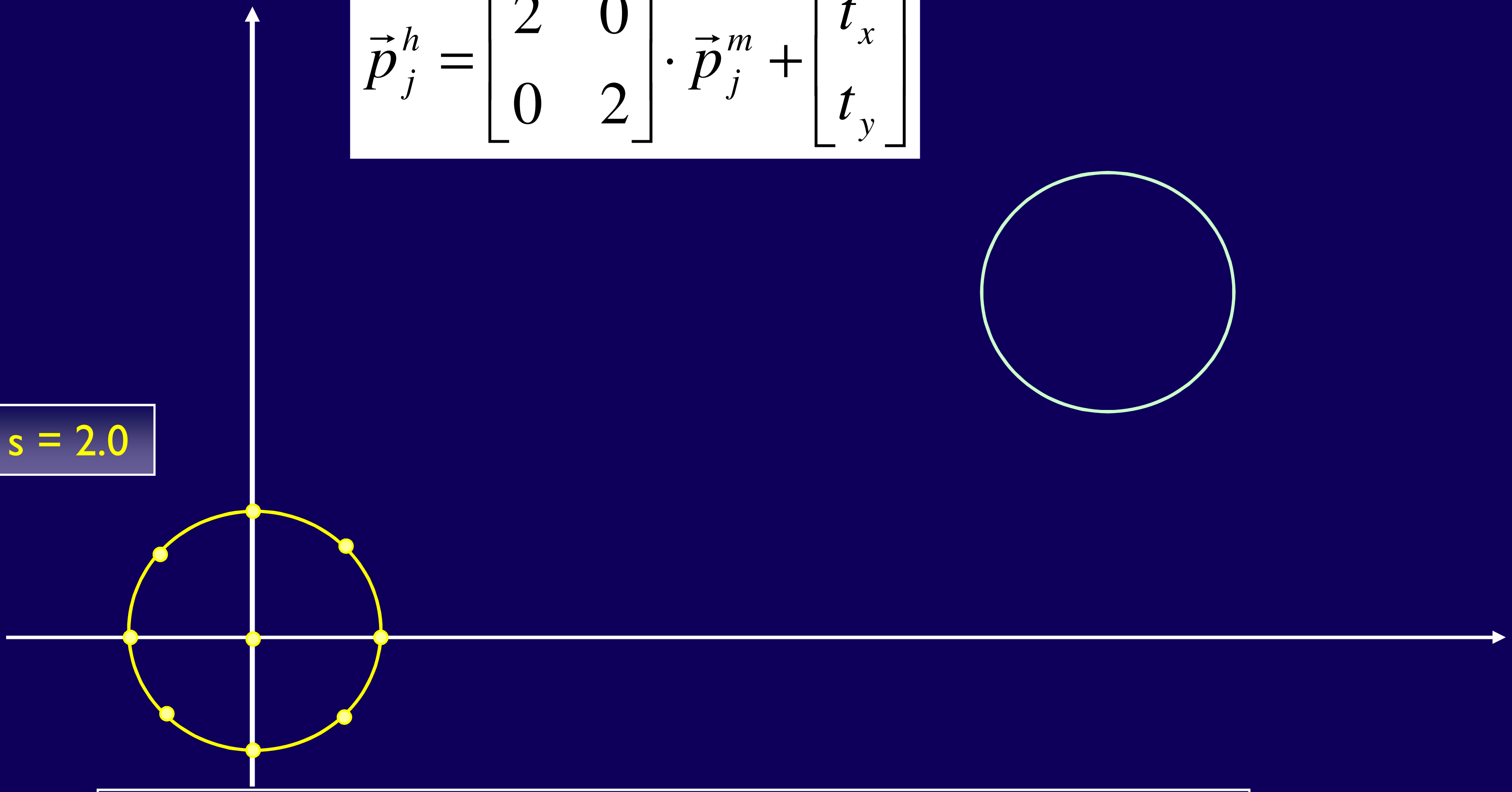


Step 4: Transform shape model with "optimal" parameter set

Generalized Hough Transform: Principle

$$\vec{p}_j^{\,h} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \cdot \vec{p}_j^{\,m} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

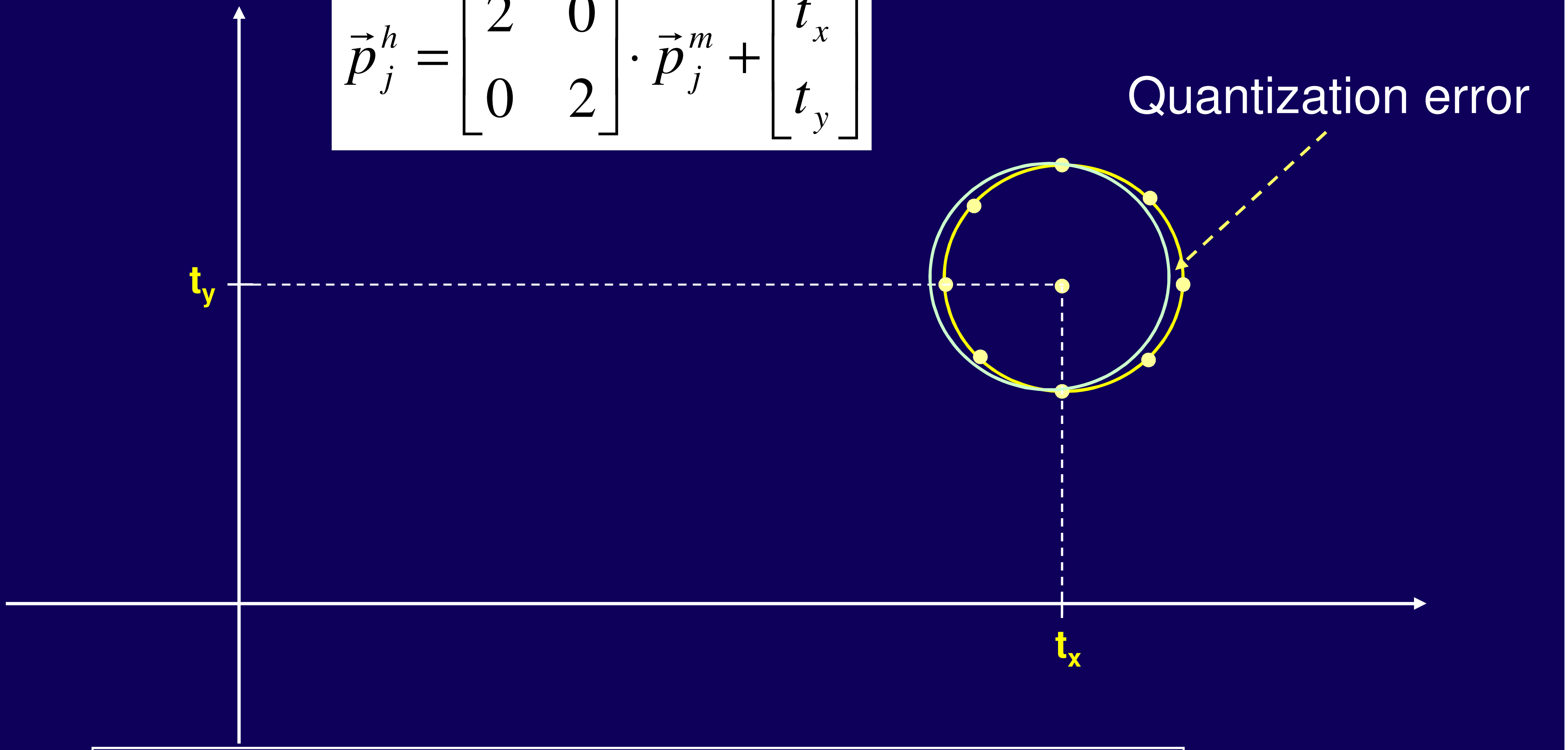
s = 2.0



Step 4: Transform shape model with “optimal” parameter set

Generalized Hough Transform: Principle

$$\vec{p}_j^{\,h} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \cdot \vec{p}_j^{\,m} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$



Step 4: Transform shape model with "optimal" parameter set

Preprocessing

Aim:

- Determine surface
- Suppress noise an
- Ideally: suppress o

Crucial for effici

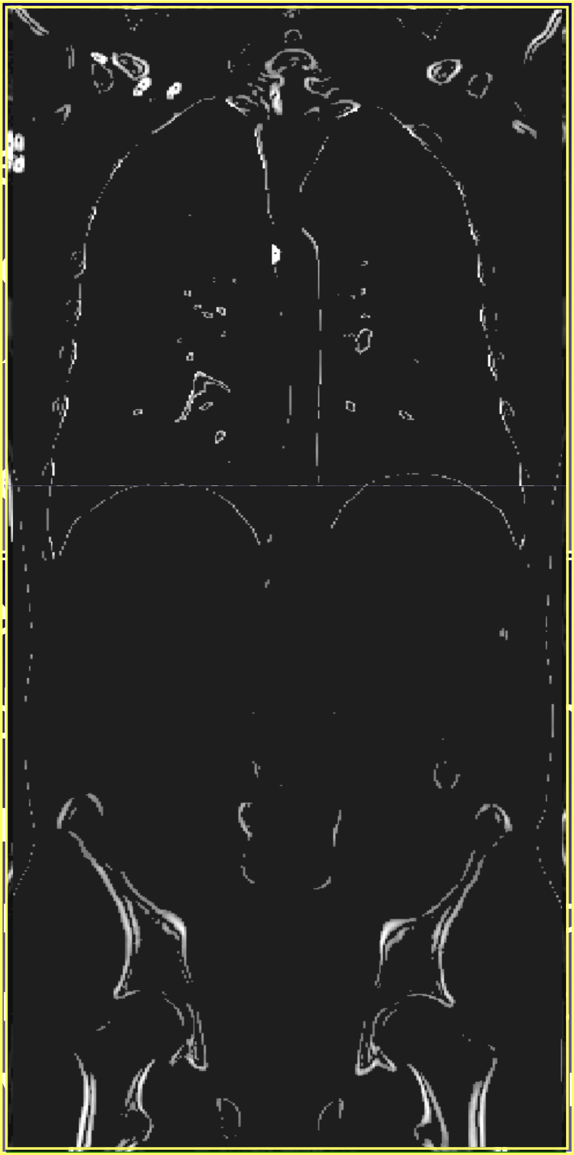
Technique

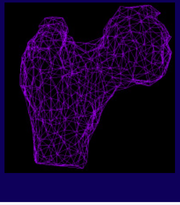
- Stanc n (e.g. Sobel, Canny)
- Supp using prior knowledge

(g in [M_{min}, M_{max}]) OR

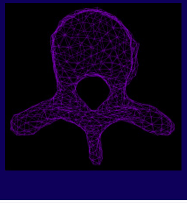
(g in [G_{min}, G_{max}])

- Inter raining data





Shape Model Generation



GHT is based on shape information – requires shape model

⇒ use triangulated meshes

Characteristic shape of anatomical objects

⇒ surface model in most cases sufficient

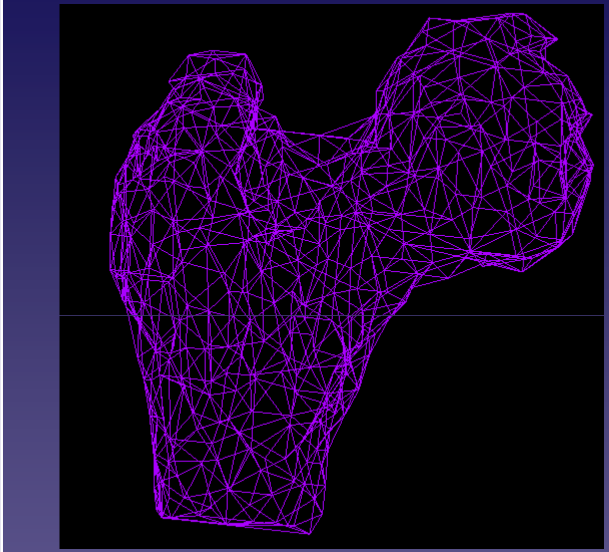
⇒ internal structures may, however, be helpful

Shape model generation:

- Femur/Vertebra: Based on one manual object delineation
- Heart model obtained from C. Lorenz and J. von Berg (Philips Research Europe – Hamburg)

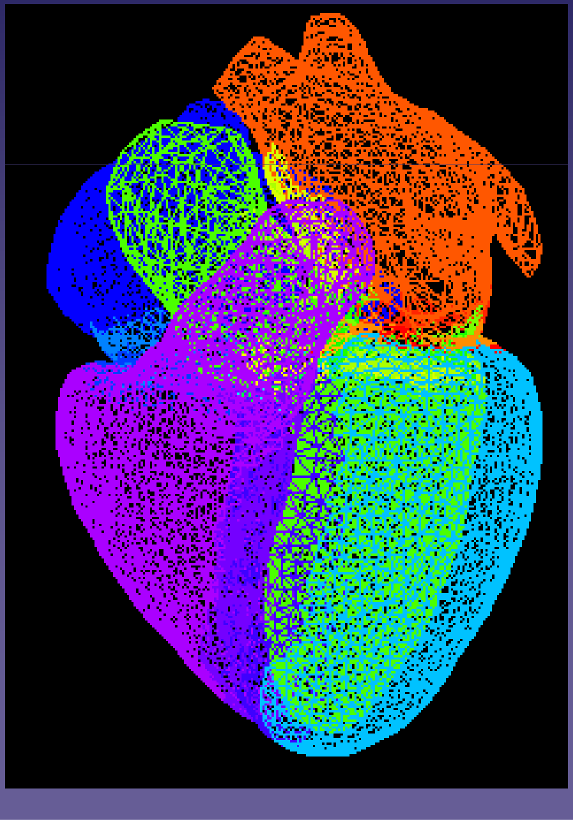
Shape Model Generation

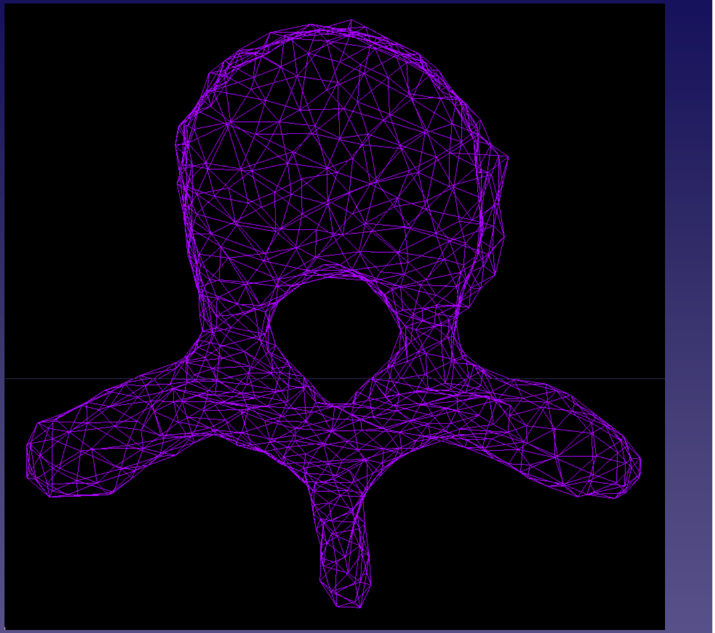
Used shape models:



Femur (1018 triangl.)

Heart (14771 triangl.)





Vertebra (1396 triangl.)

Experiments - Setup

- Femur, vertebra and heart detection in CT and MR images
- Usage of training images for:
  - shape model definition and selection of number of model points
  - parameter determination for preprocessing
  - learning of feature functions for segmentation

	Femur	Vertebra	Heart
Training			
Individuals	5	3	13
Images / Objects	5 / 5	3 / 11	28 / 28
Evaluation			
Individuals	9	5	10
Images / Objects	9 / 9	5 / 25	39 / 39
Image type	pelvis	cardiac	cardiac

Experiments - Results

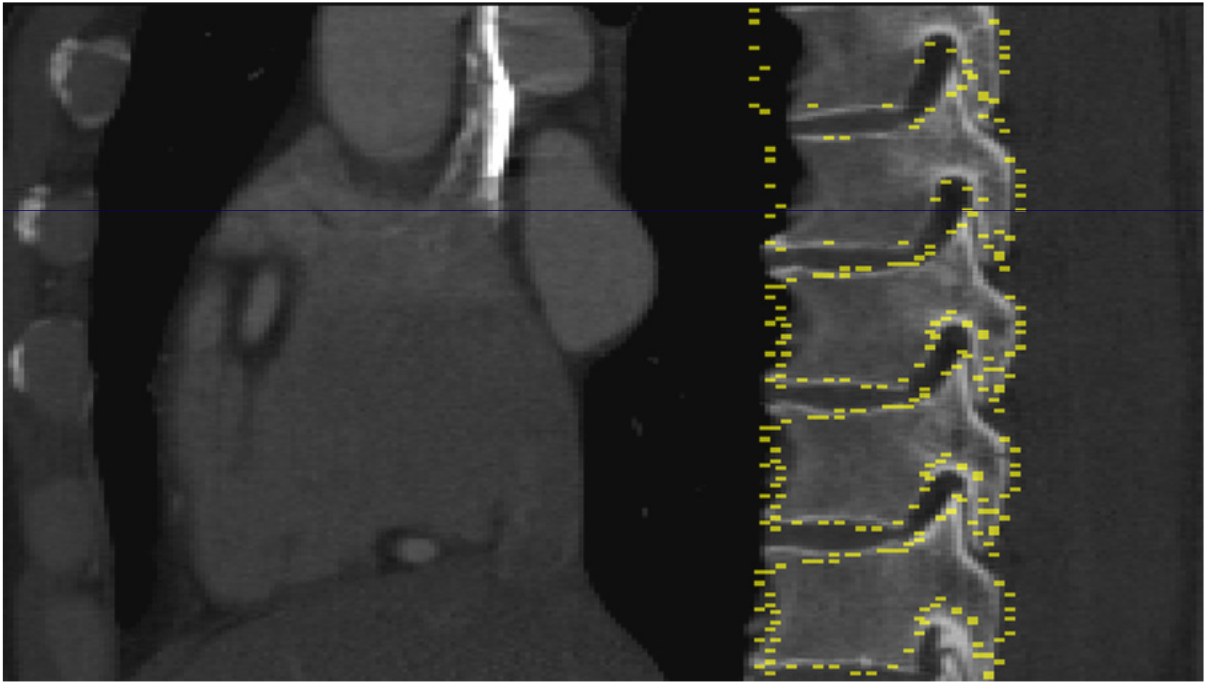
- Determination of 4 transformation parameters: 3D translation + scaling
- Quantization of translation parameters according to voxel size

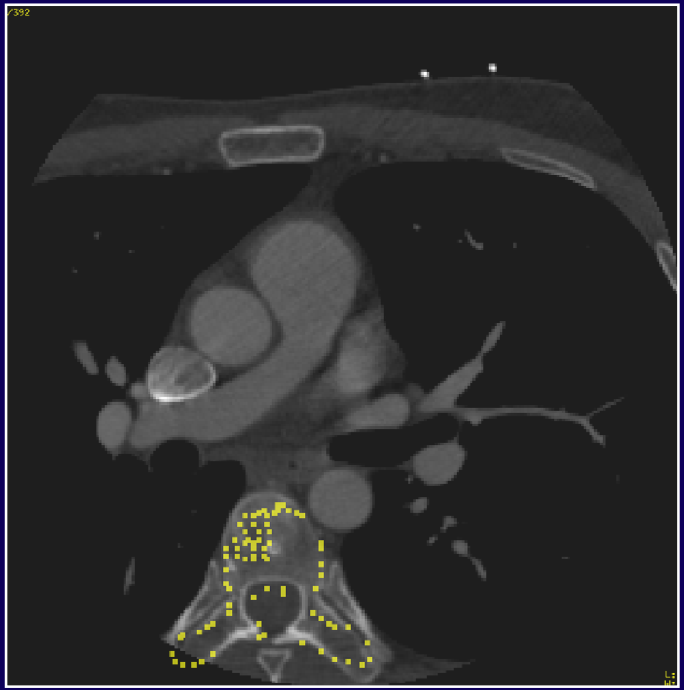
  ⇒ Hough space size = Image size • # Scaling steps
- Applied scaling factors: s = {0.8, 0.9, 1.0, 1.1, 1.2}
- Total memory requirement below 1GByte ⇒ standard workstation
- "Successful" detection ⇔ Sufficient to initialize segmentation procedure

	Femur	Vertebra	Heart
Shape model points	1018	1396	14771
Active model points	100	140	290
Average detection time	10s	20s	50s
# Target objects	9	25	39
Detection rate	100%	100%	95%

Examples – Vertebra Detection

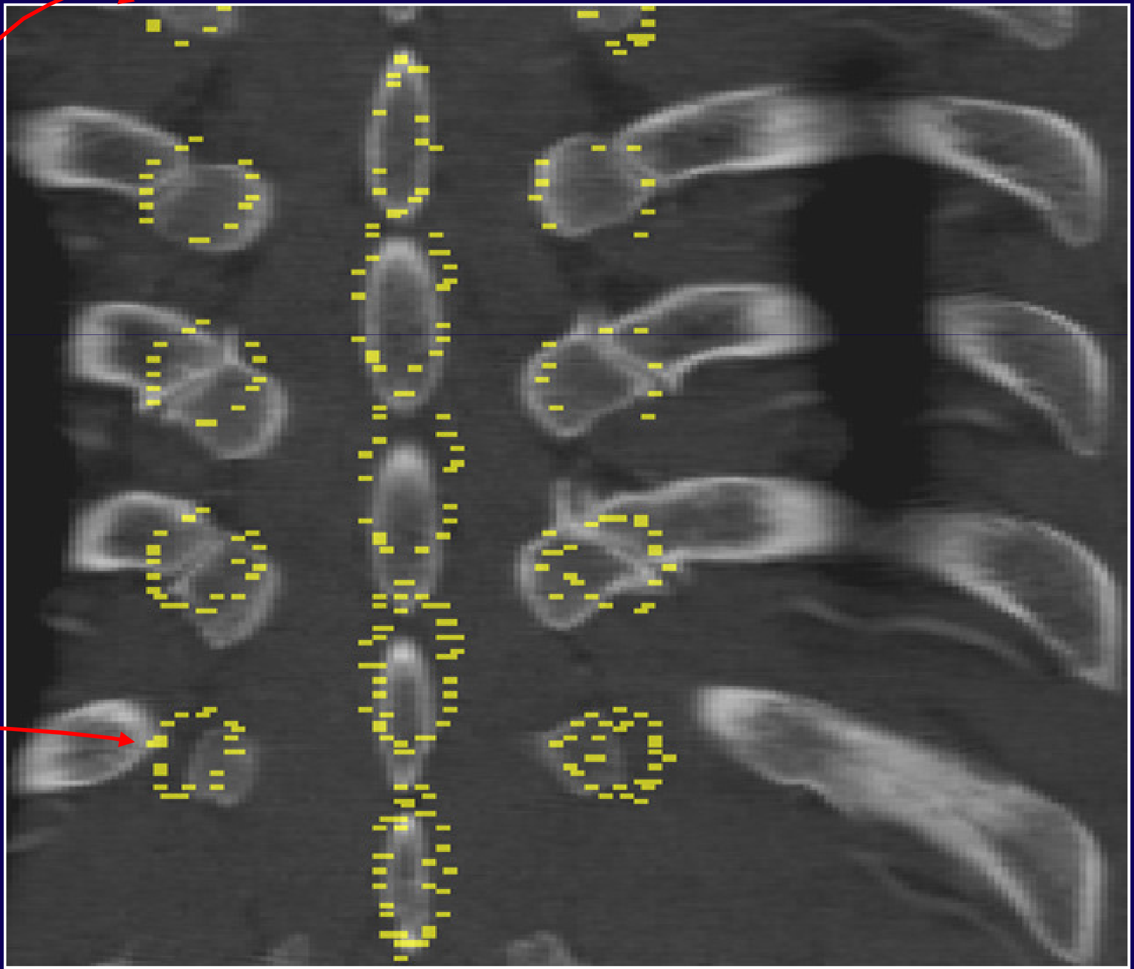
Unknown individual





Constraint: z-distance of solutions > threshold

Examples – Vertebra Detection

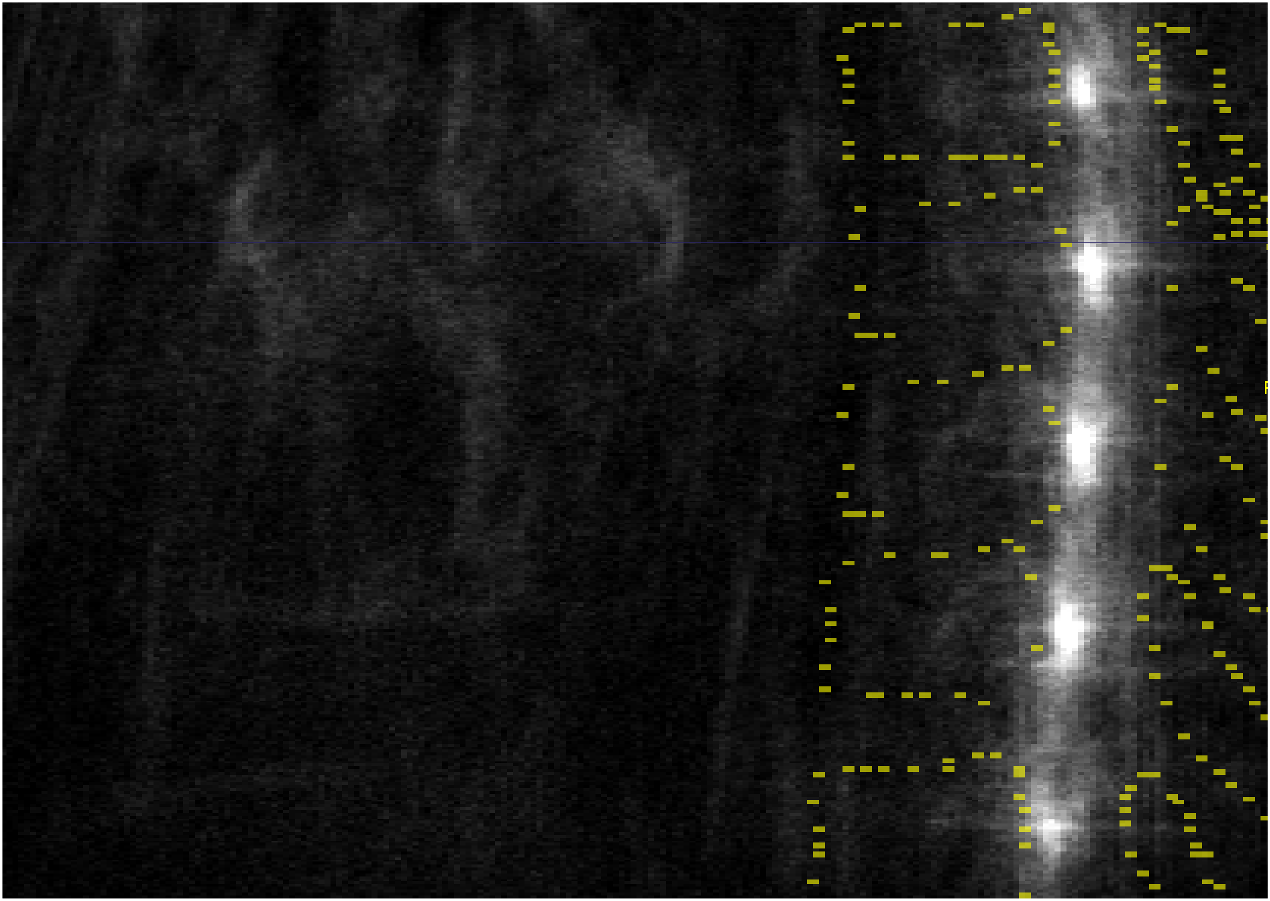


Partially cropped objects

No need for an exact match!

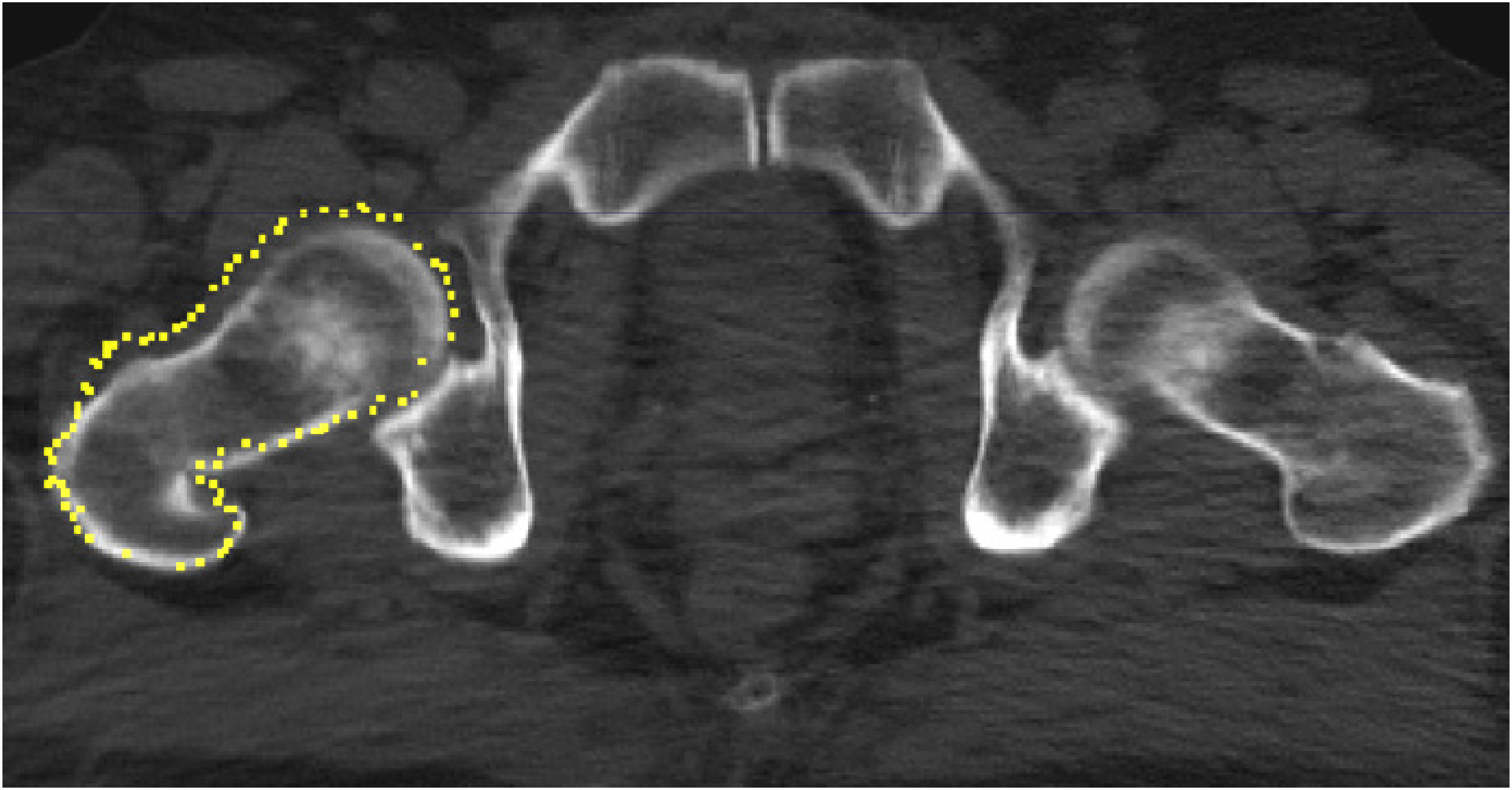
Examples – Vertebra Detection

Hough Space:

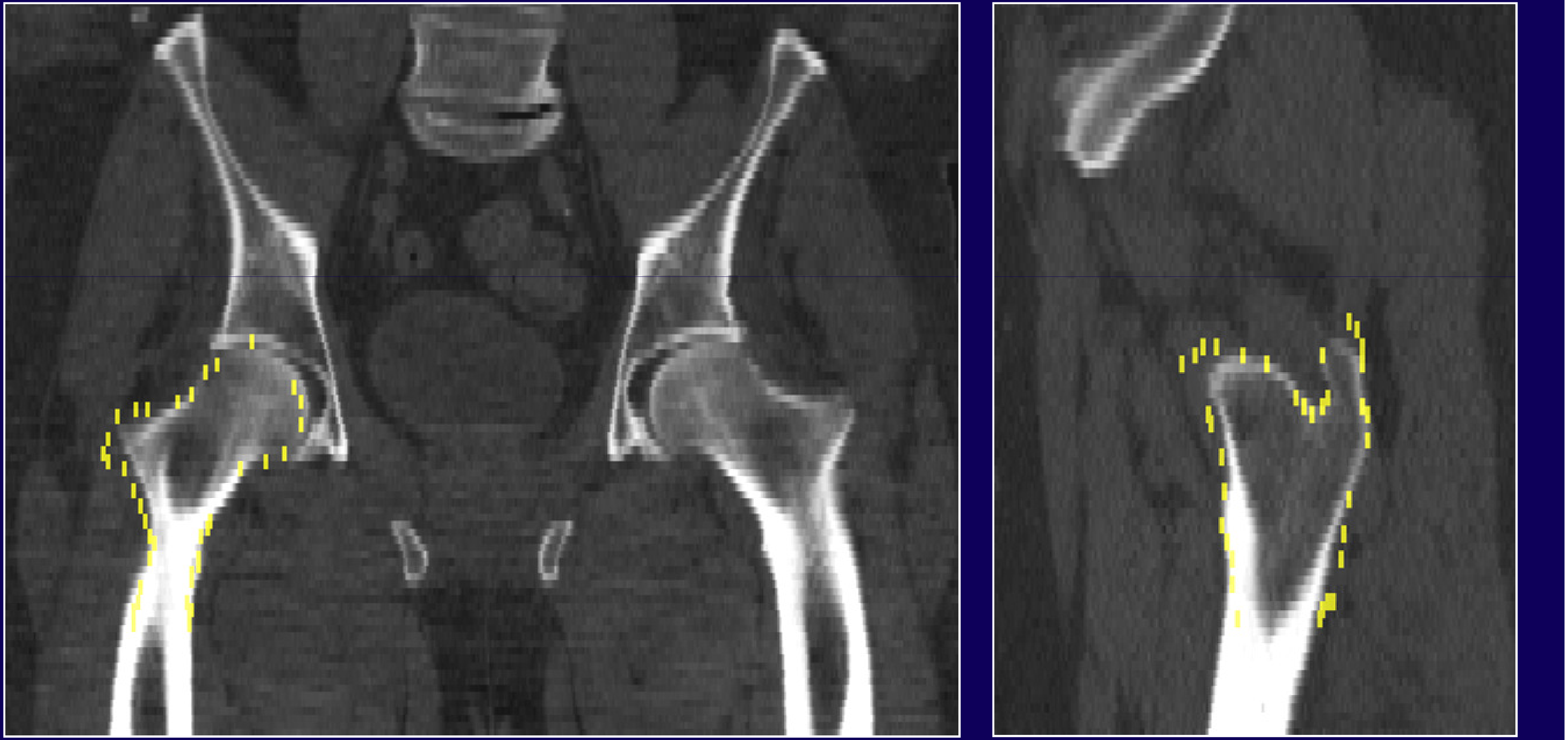


Examples – Femur Detection

Unknown individual



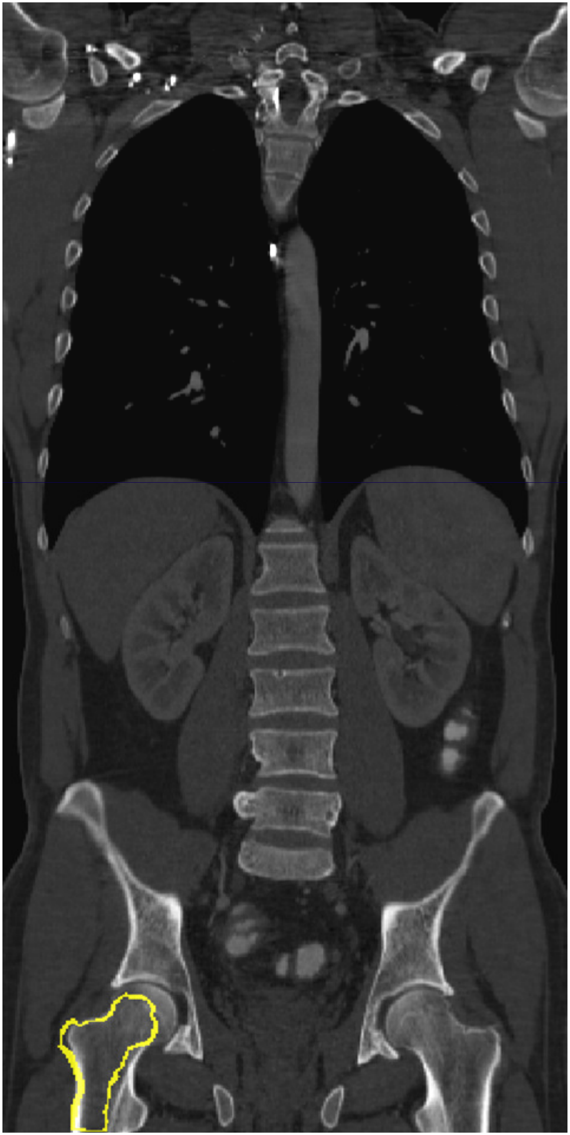
Examples – Femur Detection

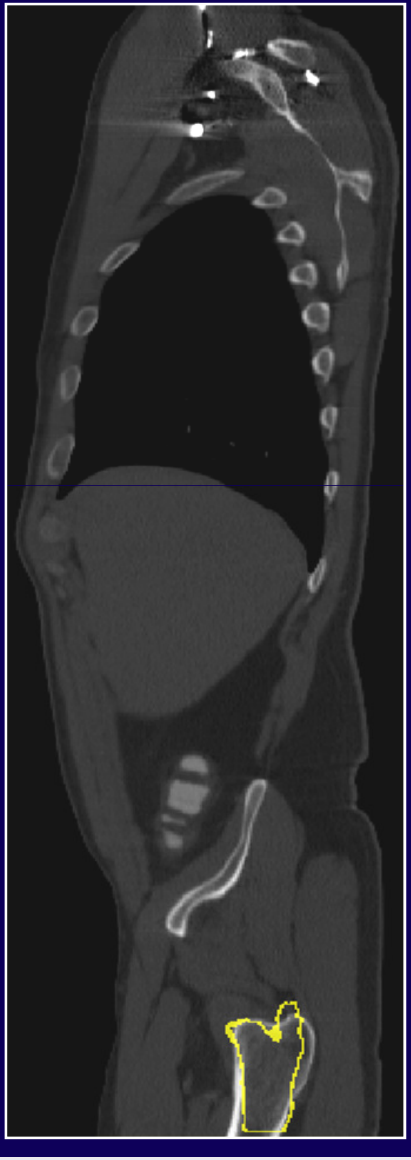


Examples – Femur Detection

Larger image

Processing time ~ 1 min.

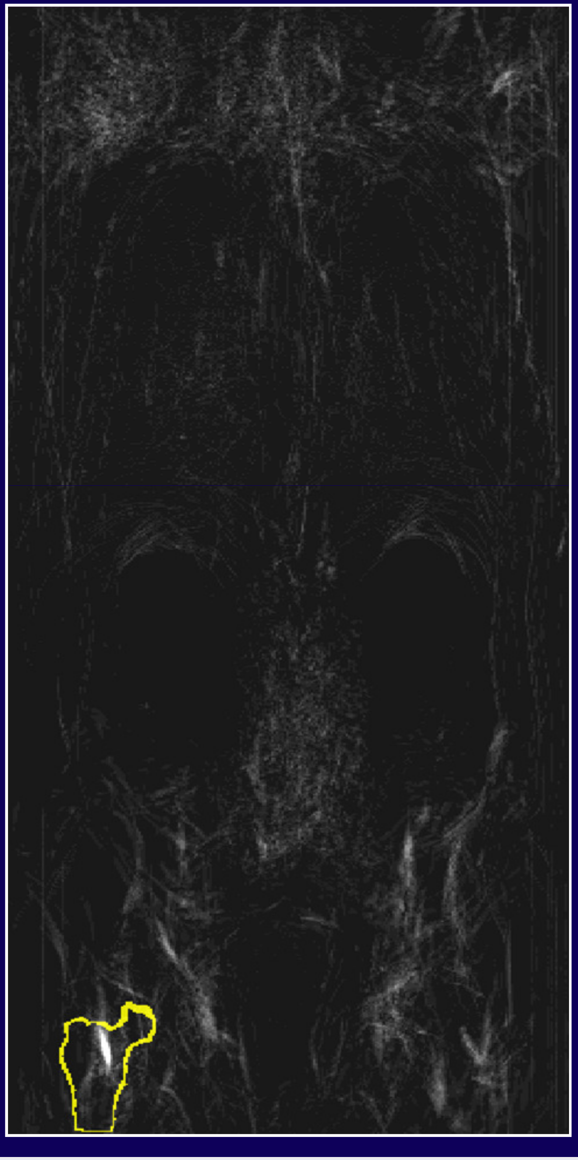


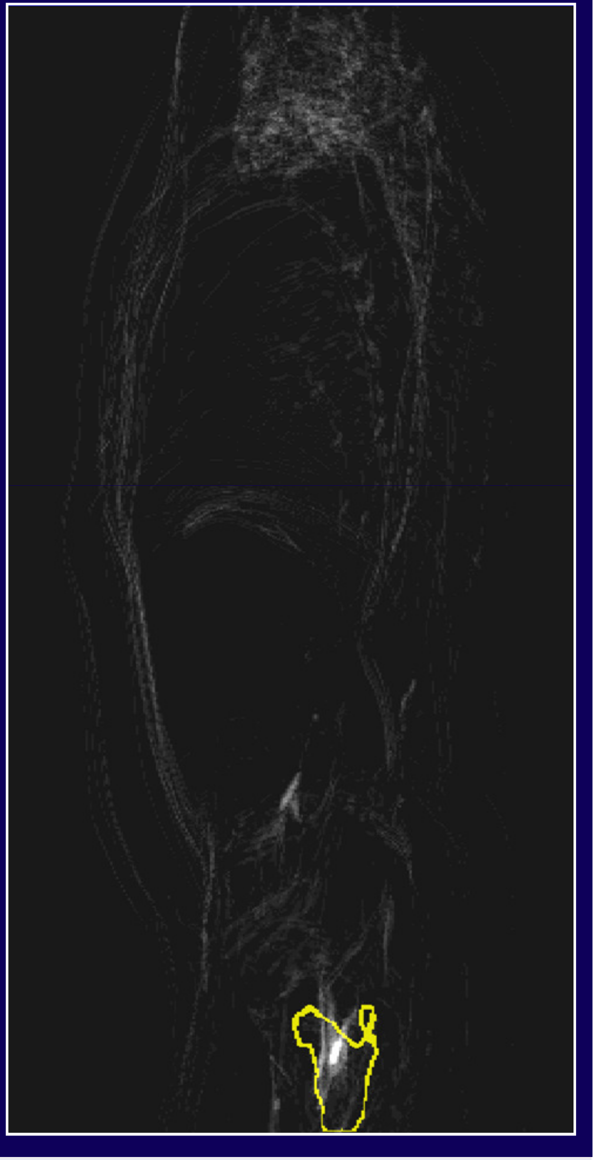


Examples – Femur Detection

Larger image

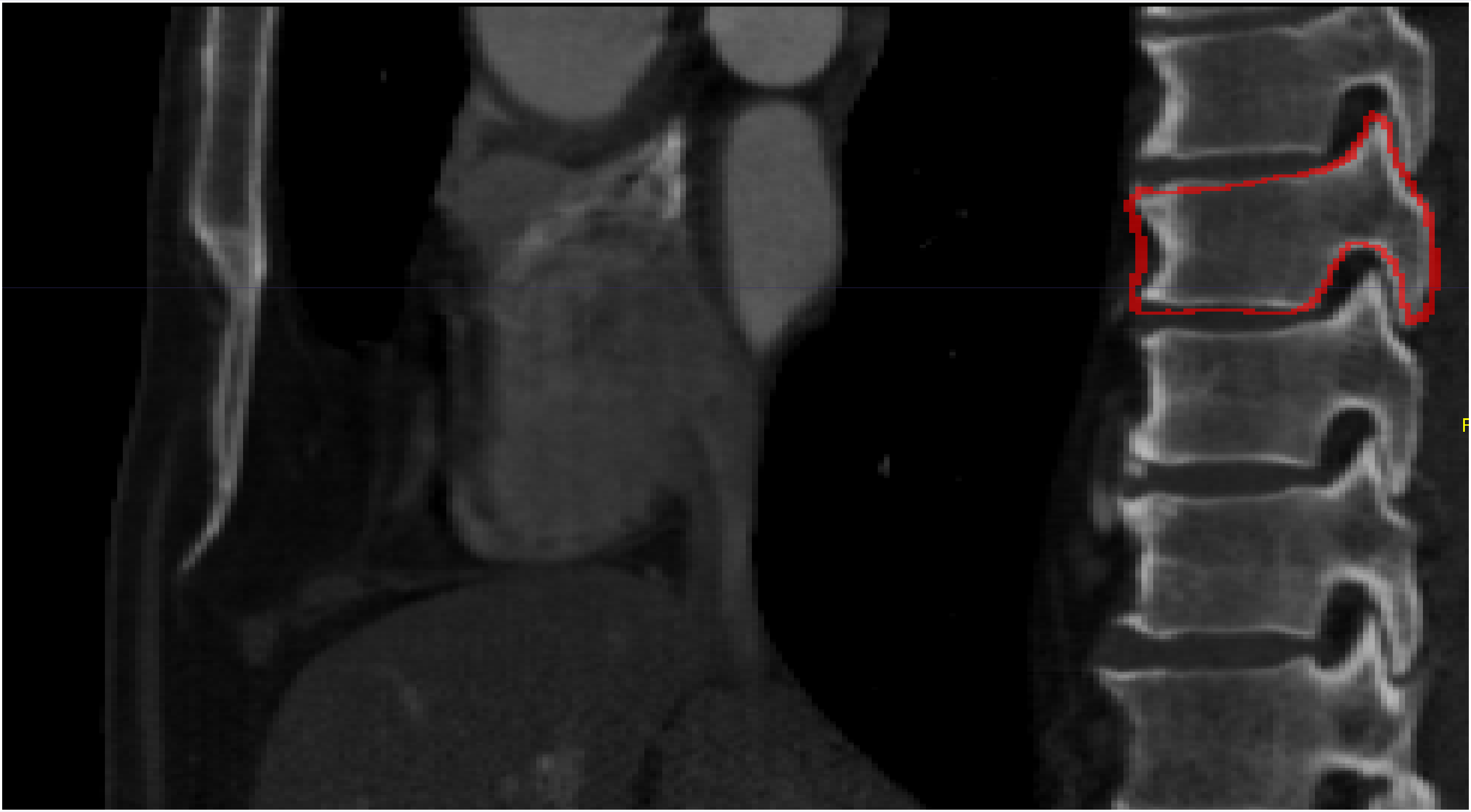
Hough space (s = 0.8)





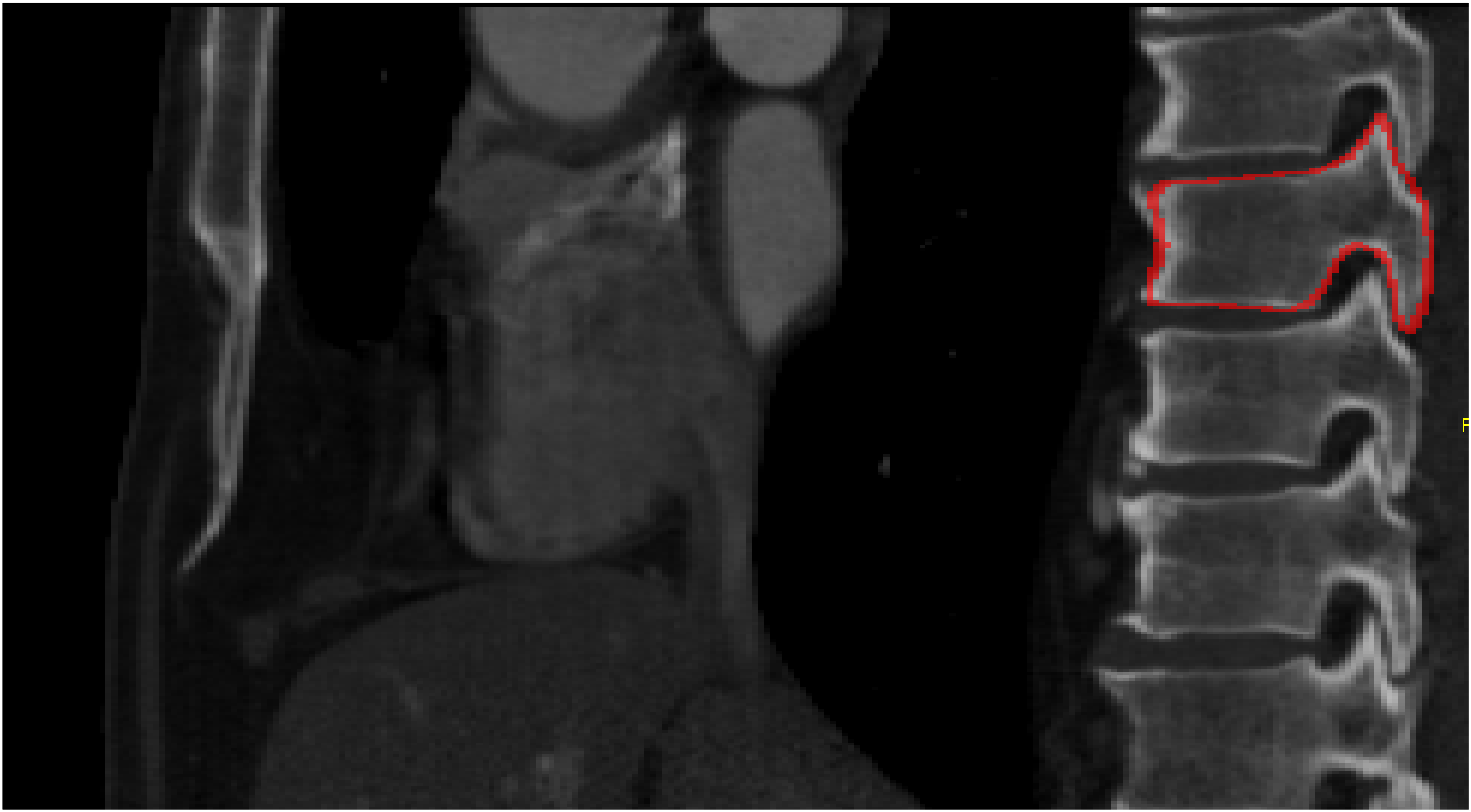
Vertebra Detection and Segmentation

Vertebra detection 1st best



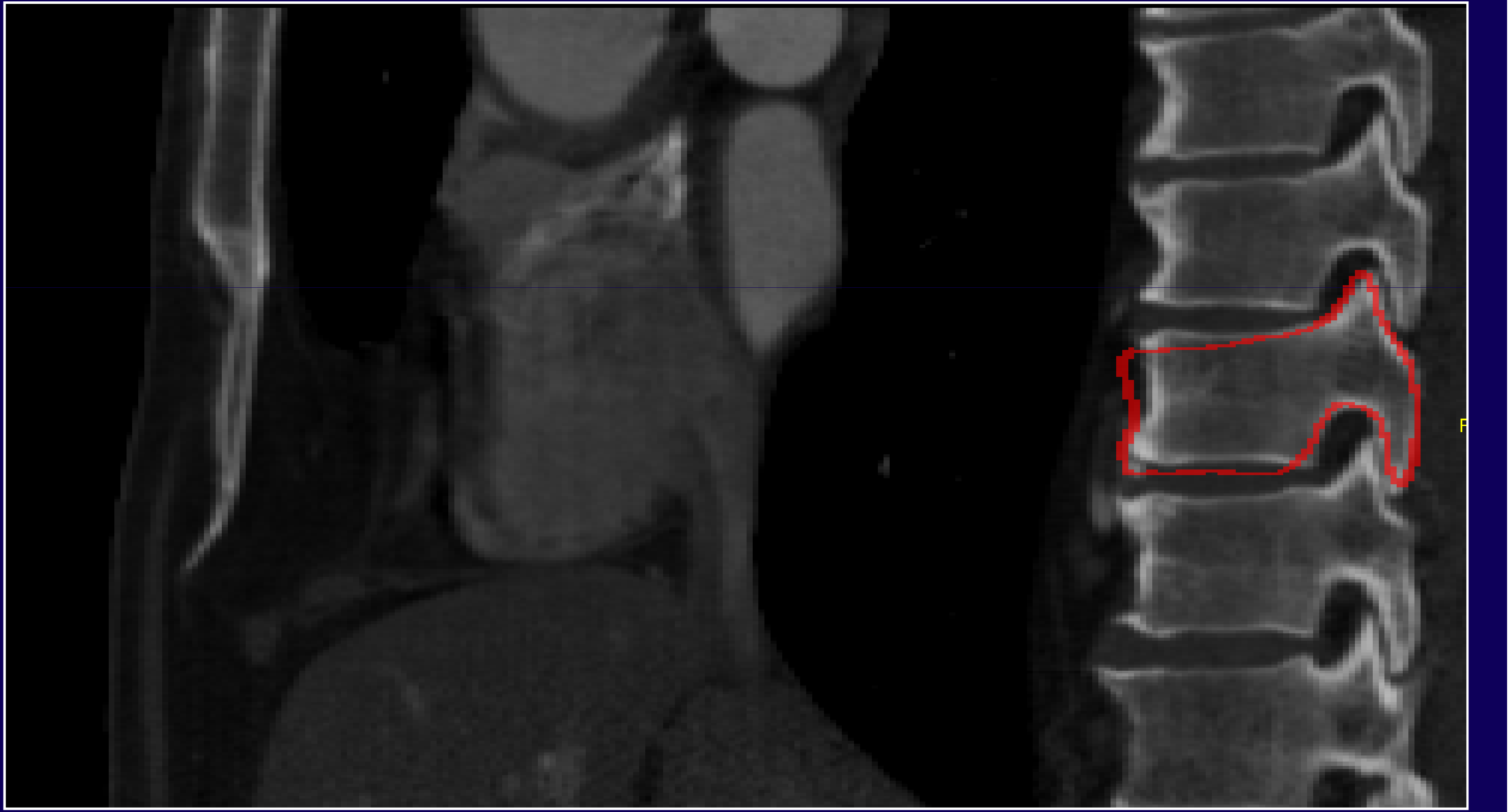
Vertebra Detection and Segmentation

+ segm.



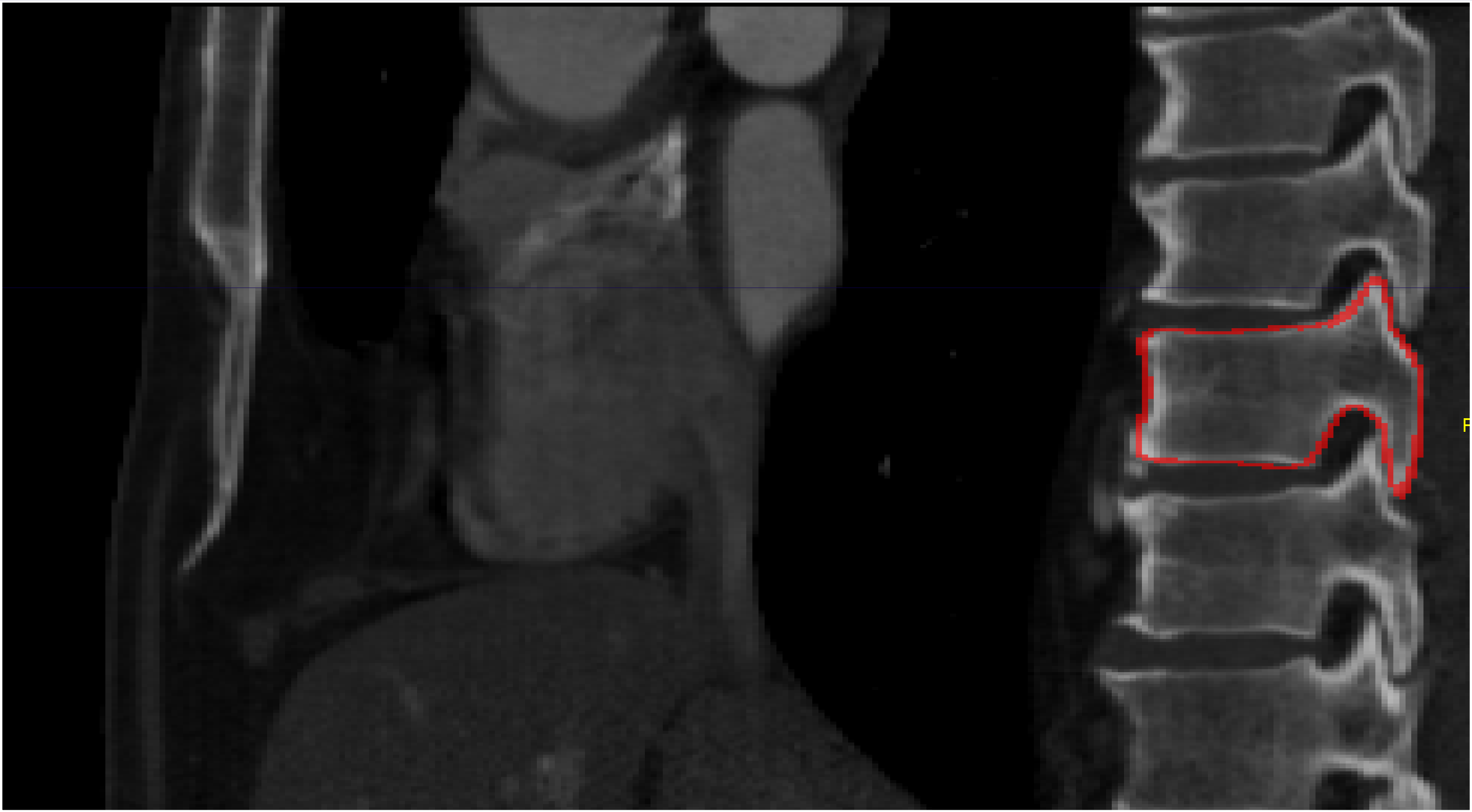
Vertebra Detection and Segmentation

Vertebra detection 2nd best



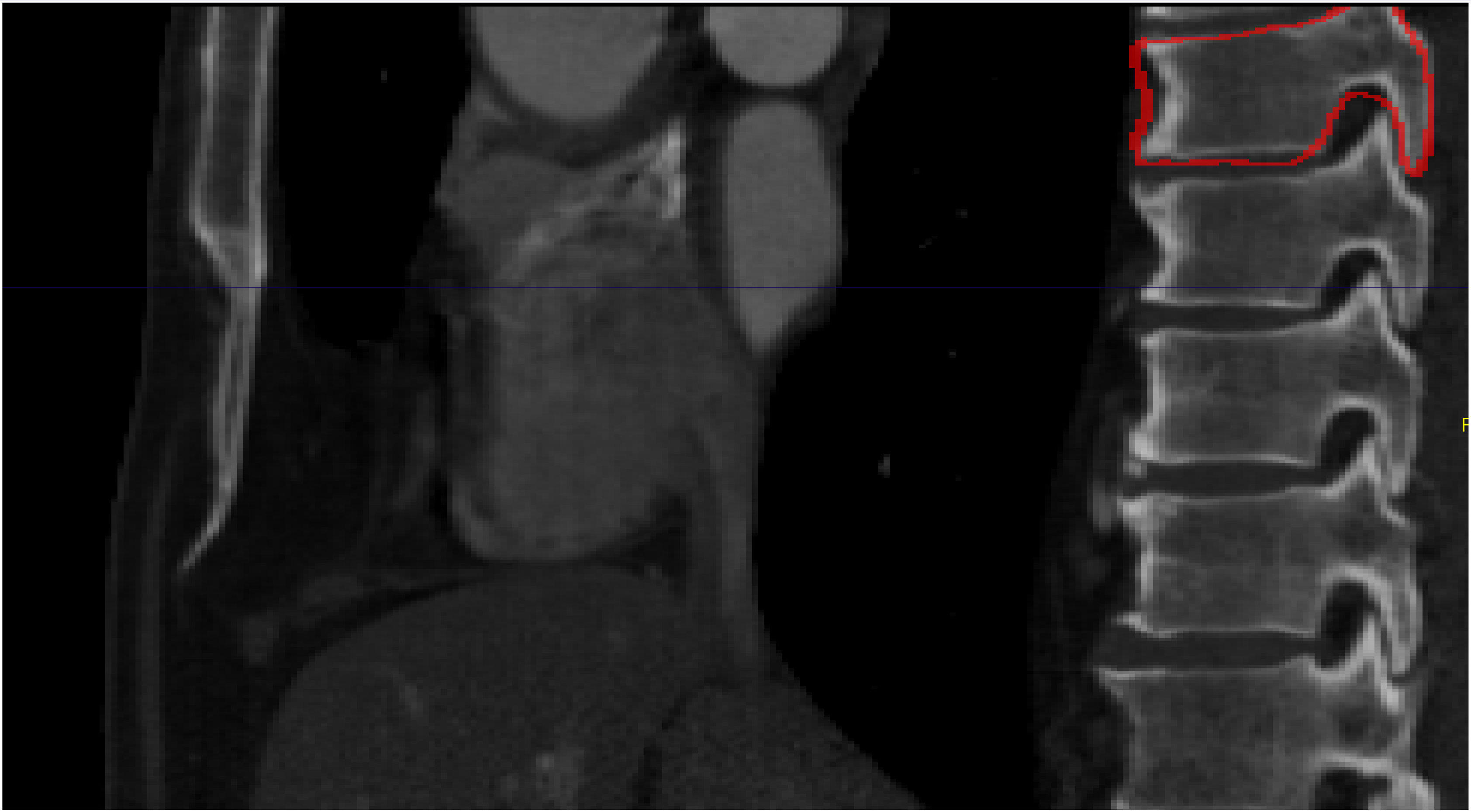
Vertebra Detection and Segmentation

+ segm.



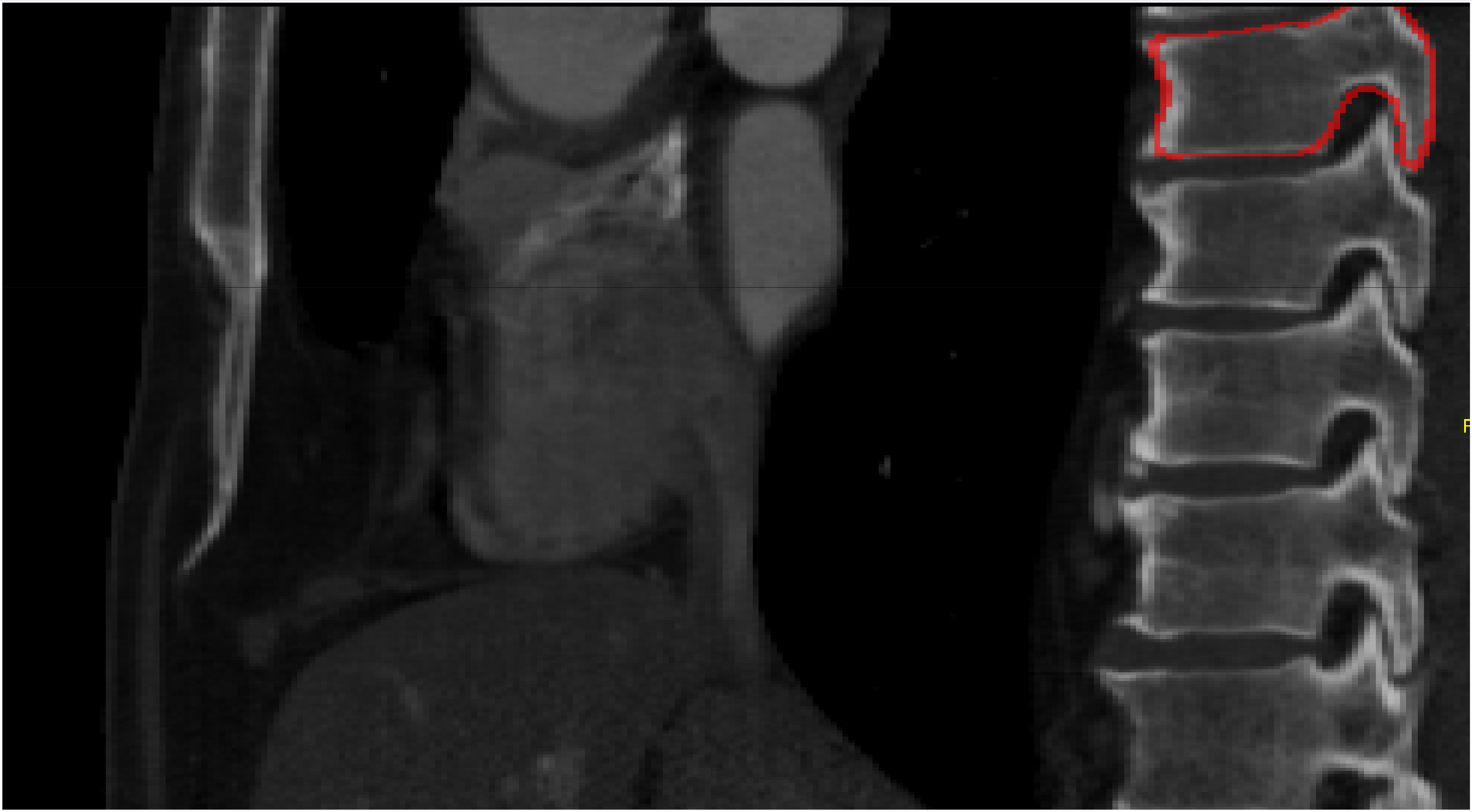
Vertebra Detection and Segmentation

Vertebra detection 3rd best



Vertebra Detection and Segmentation

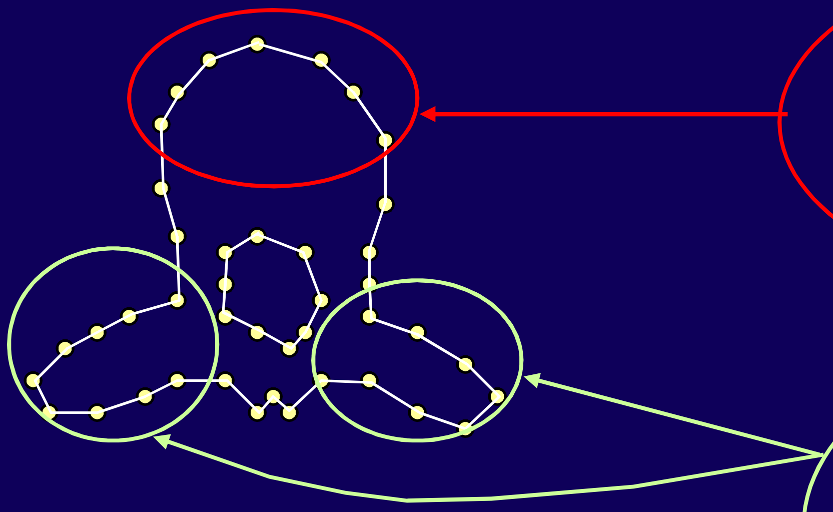
+ segm.



3. Discriminative Shape Model Optimization

Motivation 1: Learn to assess the **importance** of shape model parts

Some shape model parts are more important for detection than others



Decrease influence of unspecific regions

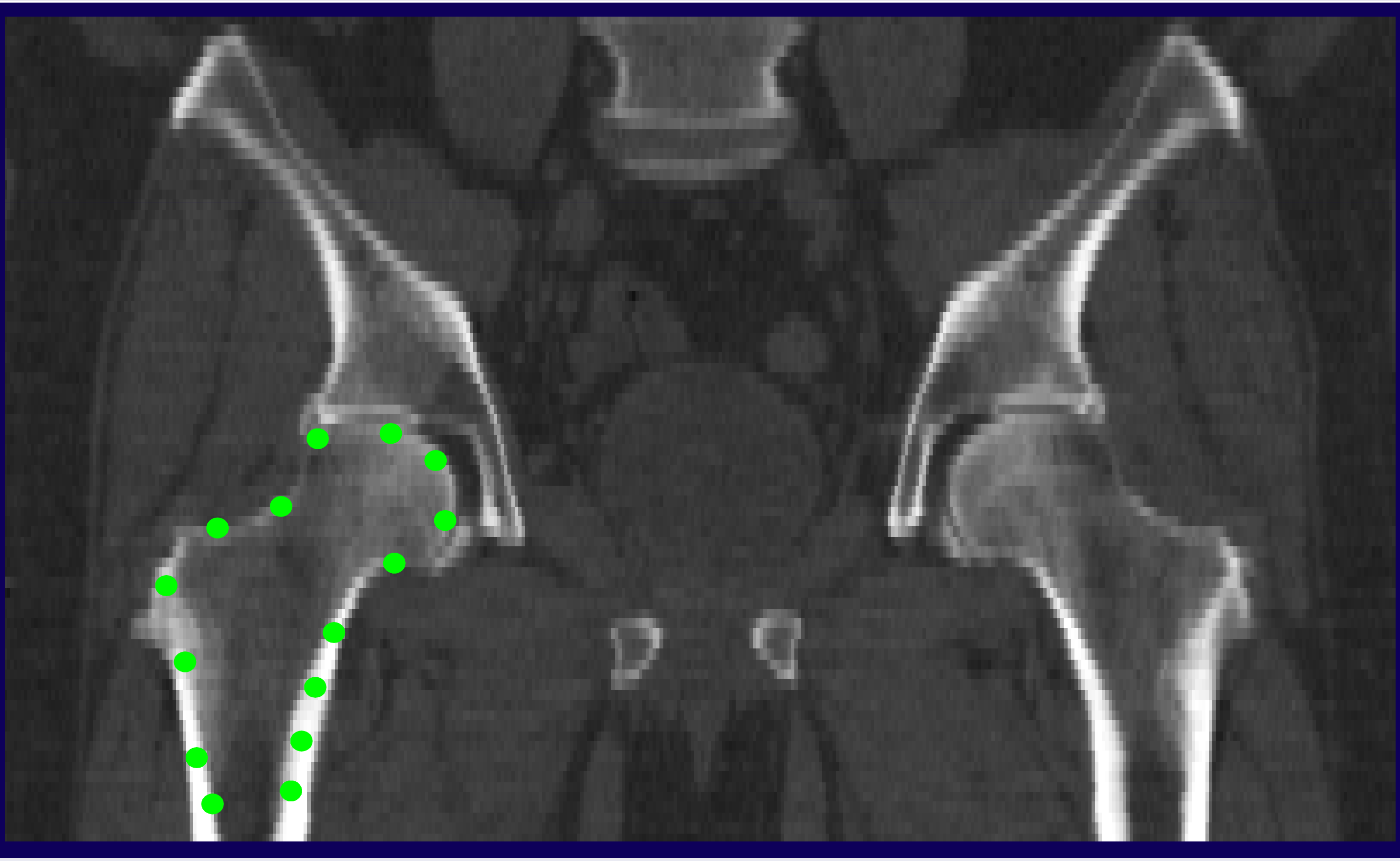
Increase influence of regions which are very object-specific

Remove "unimportant" model points from shape model

3. Discriminative Shape Model Optimization

Motivation 1: Learn to assess the **importance** of shape model parts

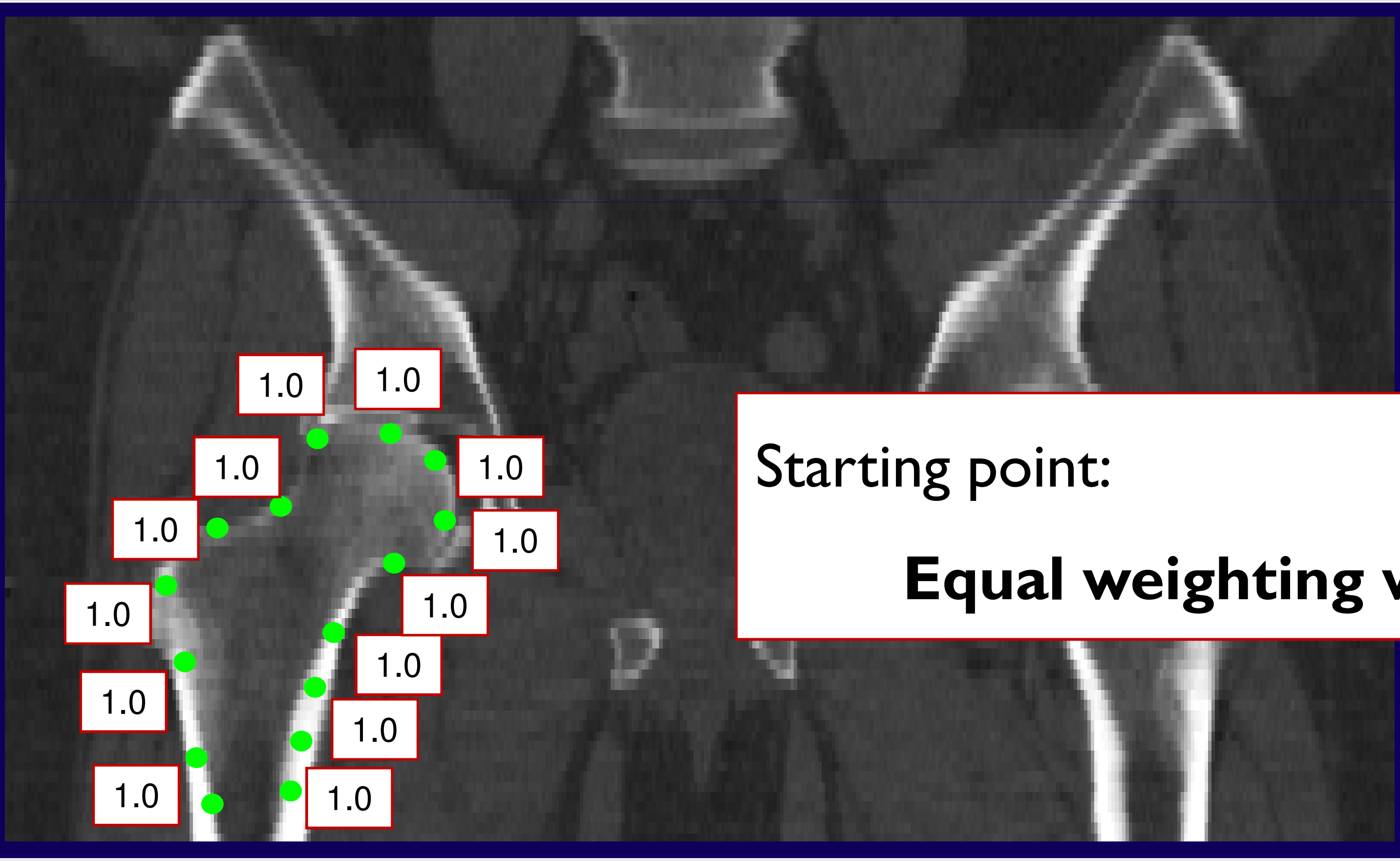
Some shape model parts are more important for detection than others



3. Discriminative Shape Model Optimization

Motivation 1: Learn to assess the **importance** of shape model parts

Some shape model parts are more important for detection than others



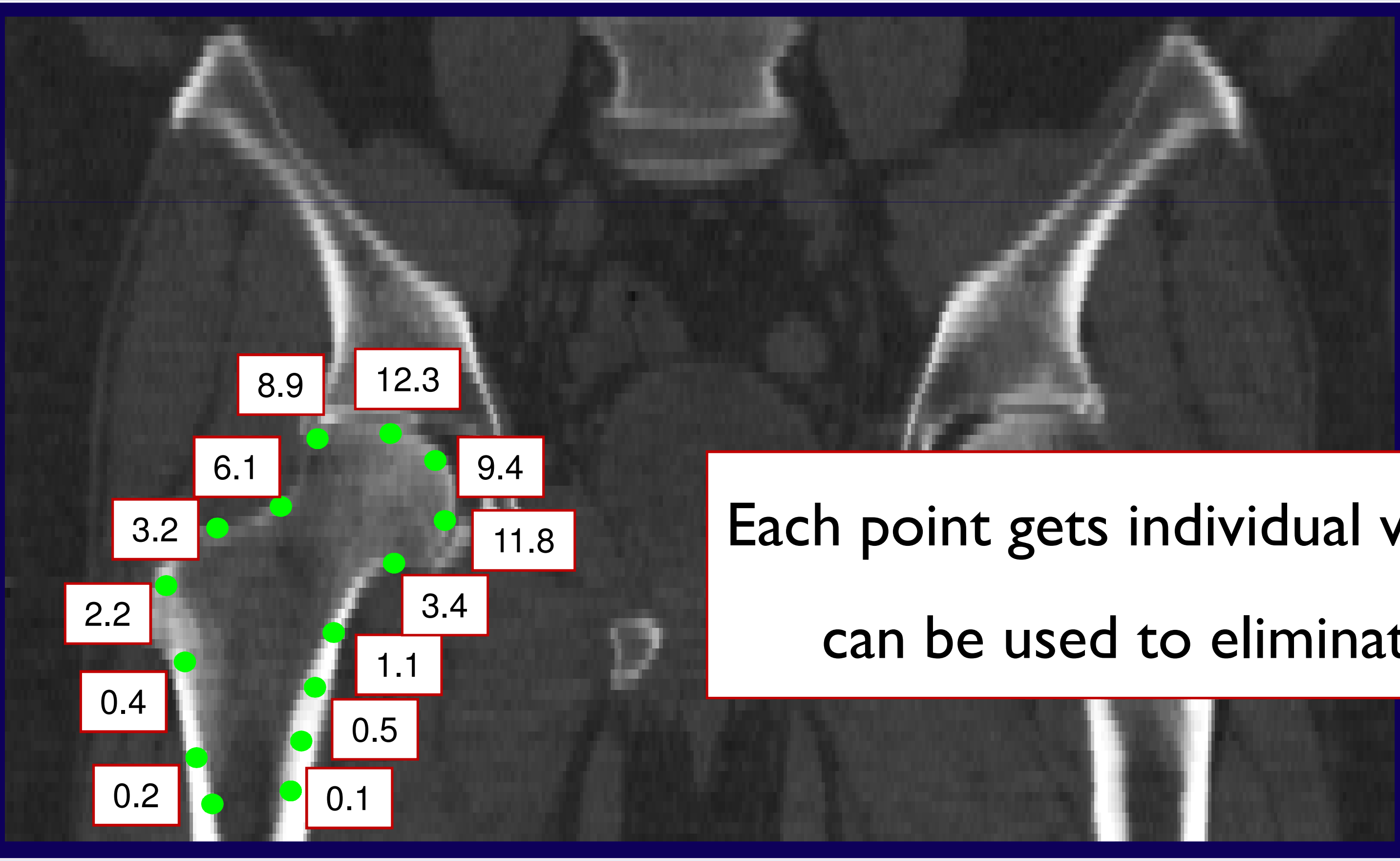
Starting point:

Equal weighting with 1.0

3. Discriminative Shape Model Optimization

Motivation I: Learn to assess the **importance** of shape model parts

Some shape model parts are more important for detection than others

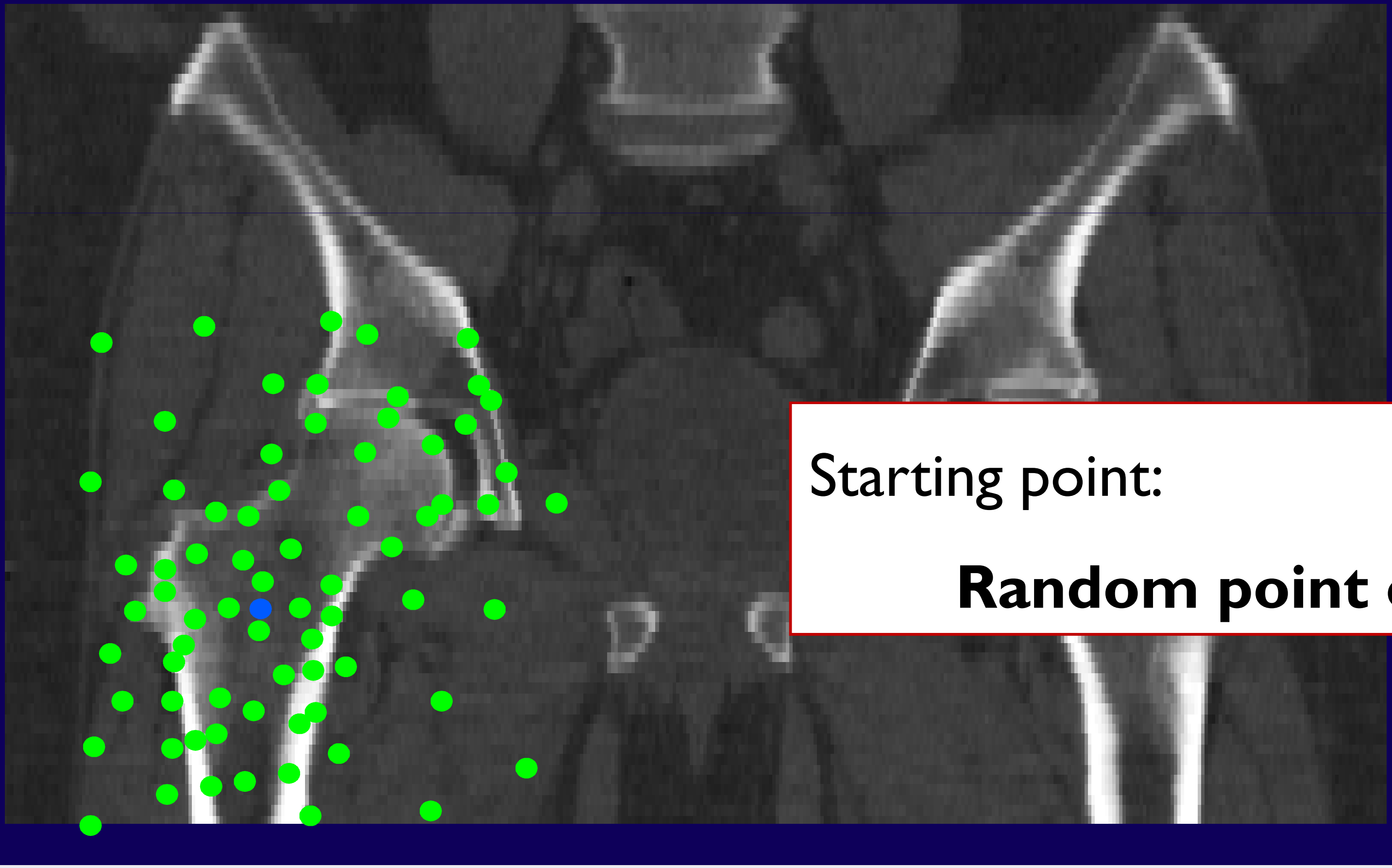


Each point gets individual weighting

can be used to eliminate points

3. Discriminative Shape Model Optimization

Motivation 2: Automatic learning instead of manual shape generation

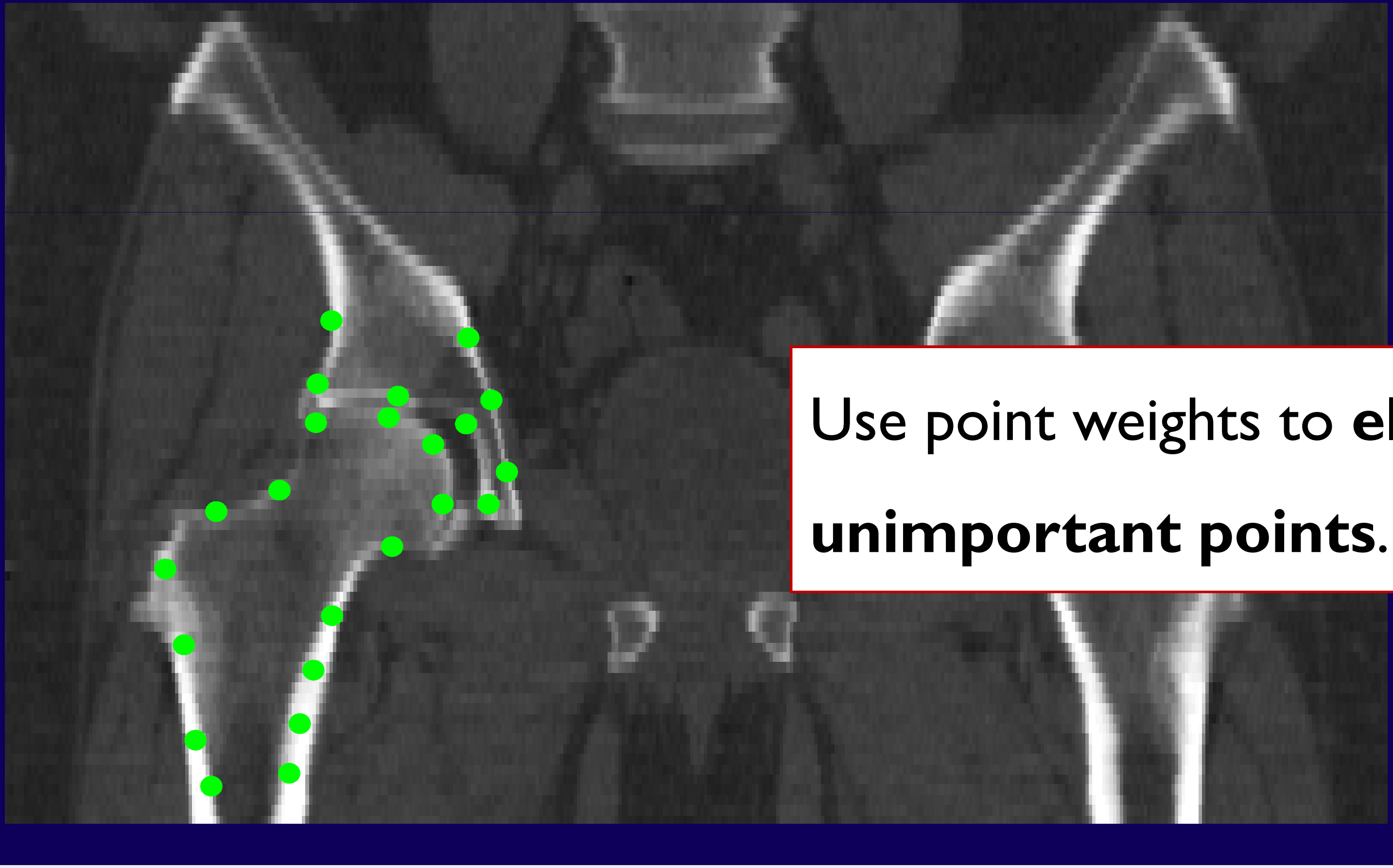


Starting point:

Random point cloud

3. Discriminative Shape Model Optimization

Motivation 2: Automatic learning instead of manual shape generation



Use point weights to **eliminate unimportant points**.

3. Discriminative Shape Model Optimization

Motivation 3: Incorporate **discrimination knowledge** into the model

learn to discriminate between object and most confusable other structures

learn how object looks like and

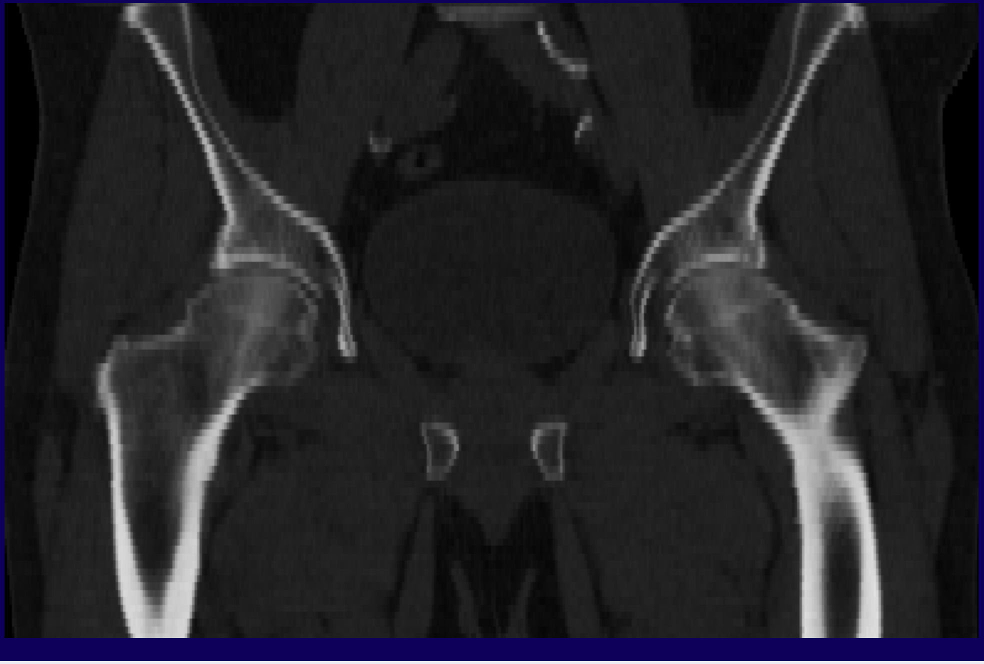
how object **does not look like**

New concept: Shapes and ***Anti-Shapes***

Realization: introduce ***negative votes*** into the GHT

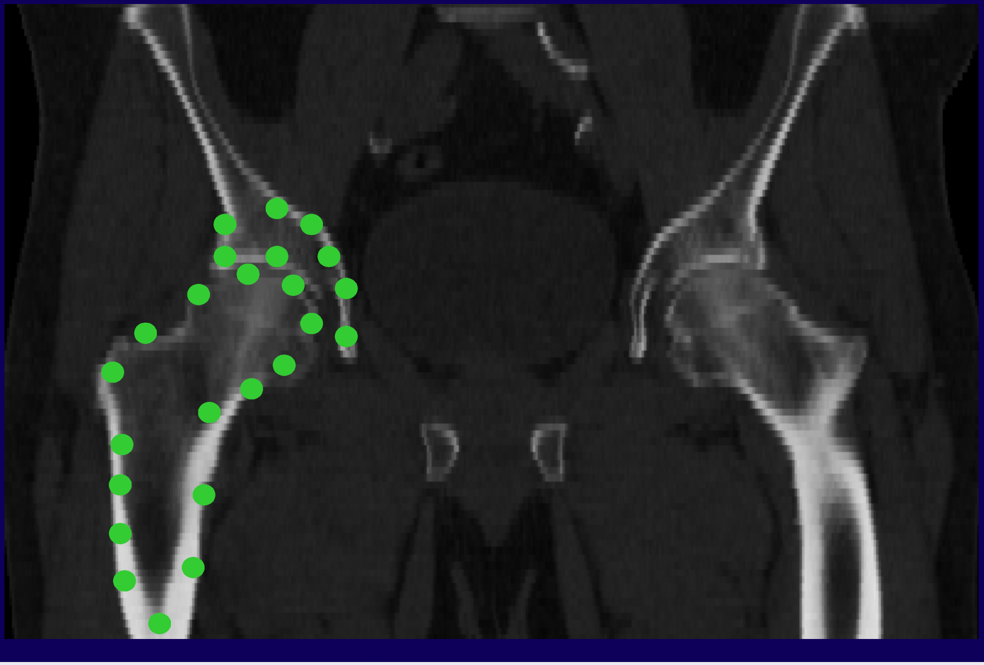
Example:

1. Learn the shape of the searched anatomical object



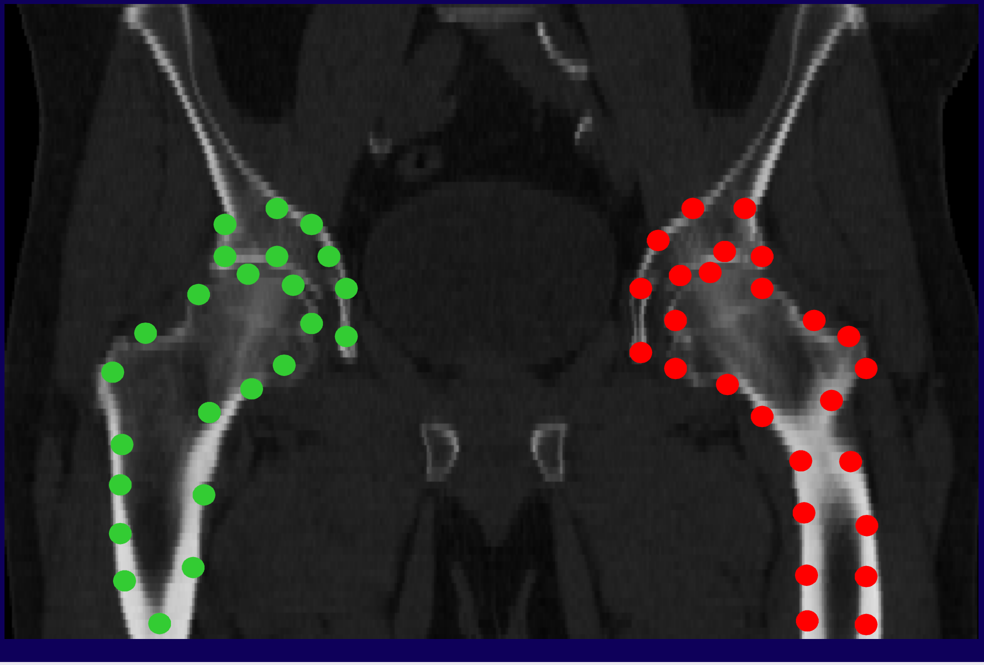
Example:

1. Learn the shape of the searched anatomical object



Example:

1. Learn the shape of the searched anatomical object
2. Learn shape of most confusable other object(s): **anti-shape**

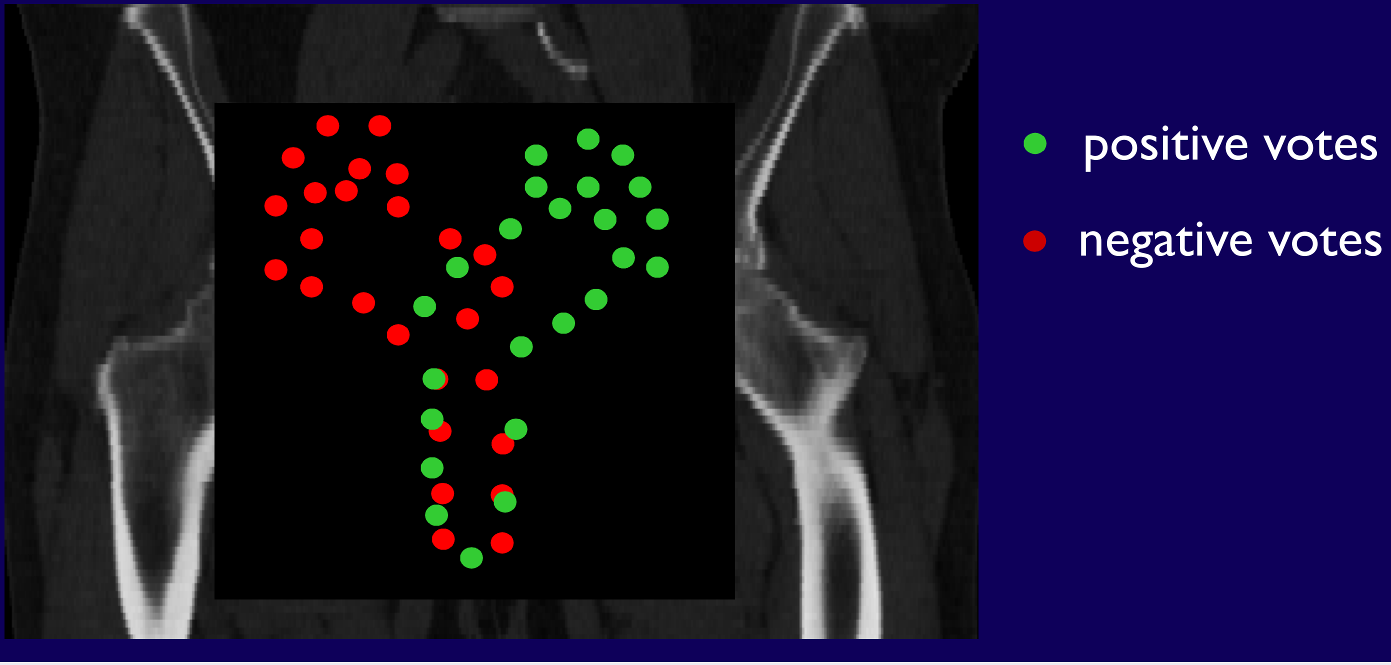


- 🟢 positive votes
- 🔴 negative votes

Example:

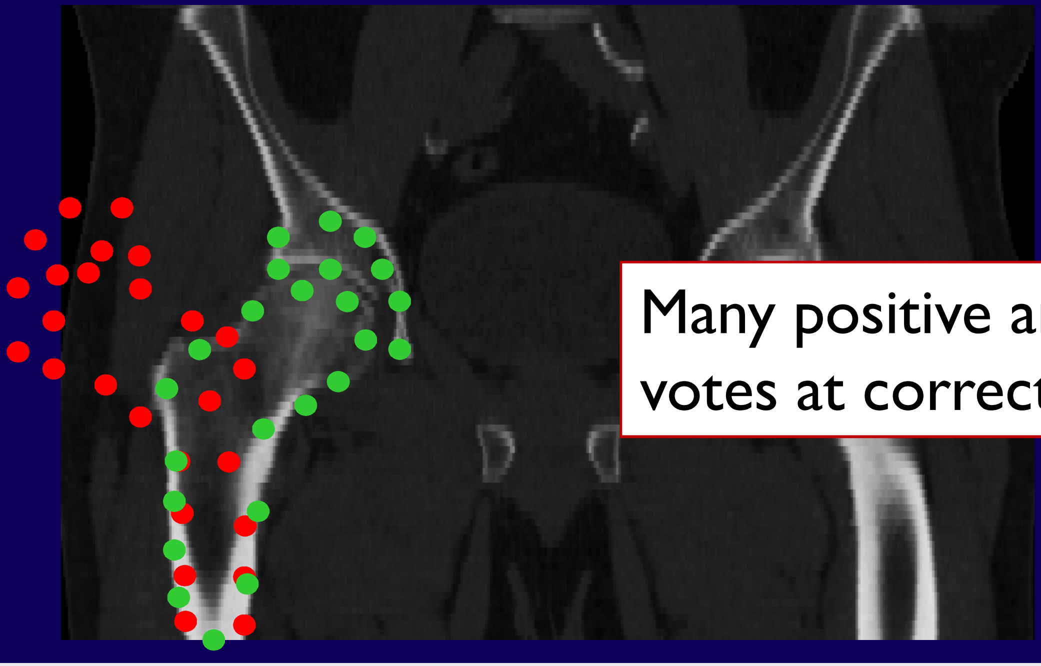
1. Learn the shape of the searched anatomical object
2. Learn shape of most confusable other object(s): **anti-shape**

Combination into
a single shape model



Example:

1. Learn the shape of the searched anatomical object
2. Learn shape of most confusable other object(s): **anti-shape**
3. Apply model

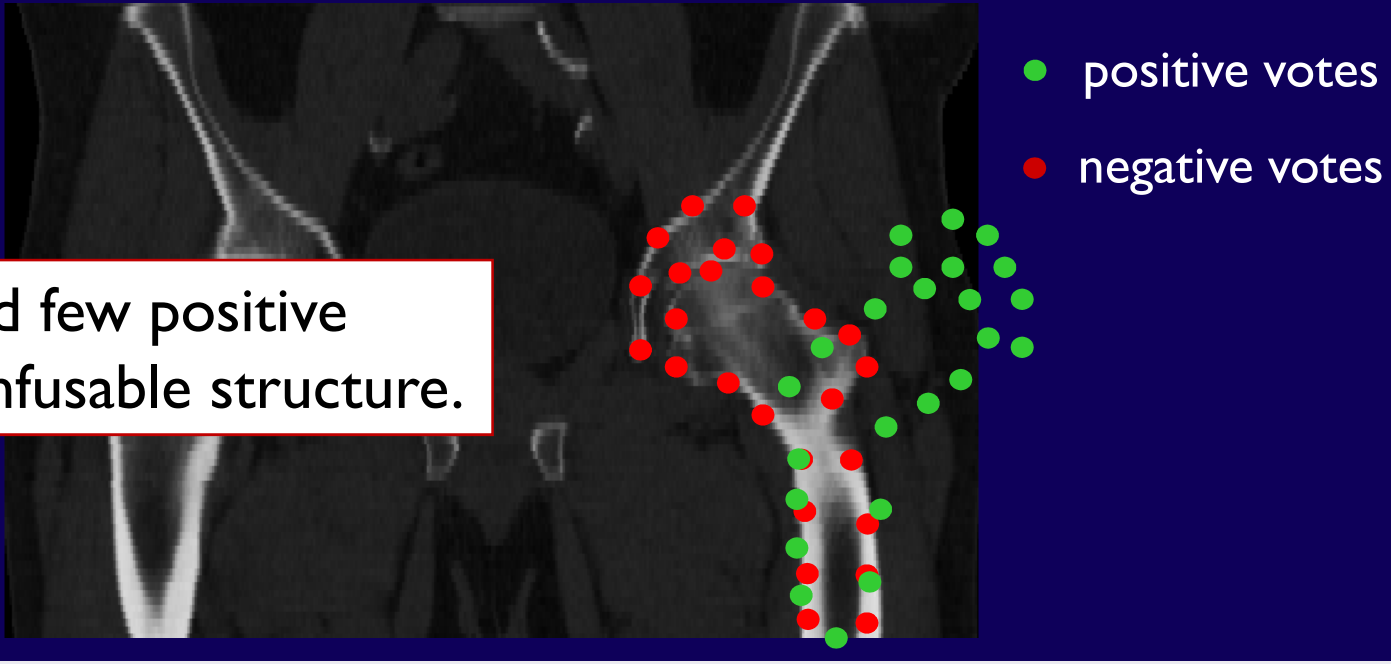


- 🟢 positive votes
- 🔴 negative votes

Many positive and few negative votes at correct position.

Example:

1. Learn the shape of the searched anatomical object
2. Learn shape of most confusable other object(s): **anti-shape**
3. Apply model



Many negative and few positive votes at most confusable structure.

Maximum Entropy Distribution & Minimum Classification Error Training

Principle

1. Split the shape model into its N individual model points
2. Interpret each point as individual knowledge source
3. For each model point:

   Learn individual classifier from Hough space votes in training images
4. Combine the individual (base) classifiers into one global classifier
5. Optimize base classifiers weights in global classifier with respect to misclassification rate **individual model point weighting**

Maximum Entropy Distribution & Minimum Classification Error Training

Theory

1. Interpret Hough space as posterior class probability $p(c_i|x)$

   c_i: class i, i.e. one specific transformation parameter setting (A_i, $\mathbf{t}_i$)

   x: observation (e.g. image features)

Posterior class probability for hypothesis c_i: $p(c_i|x) = N_i / N$

N_i : Votes for hypothesis c_i=(A_i, $\mathbf{t}_i$) in Hough space (i.e. votes for specific cell)

N : Total number of votes in Hough space

Maximum Entropy Distribution & Minimum Classification Error Training

Theory

1. Interpret Hough space as posterior class probability $p(c_i|x)$

c_i: class i, i.e. one specific transformation parameter setting (A_i, $\mathbf{t}_i$)
x: observation (e.g. image features)

Posterior class probability for hypothesis c_i: $p(c_i|x) = N_i / N$

N_i : Votes for hypothesis $c_i=(A_i, \mathbf{t}_i)$ in Hough space (i.e. votes for specific cell)
N : Total number of votes in Hough space

Object detection: Finding hypothesis ĉ with highest likelihood

Bayes Classifier: $\hat{c} = \arg\max_{c_i} p(c_i \mid x)$

Note: Just a different interpretation of GHT-based detection with identical result.

Maximum Entropy Distribution & Minimum Classification Error Training

2. Split Hough space into N model point dependent Hough spaces H_j

each H_j carries votes coming only from model point j

3. For each model point j

Learn individual classifier from its Hough space H_j

Model point dependent posterior class probability: $p_j(c_i|x) = N_{ij} / N_j$

N_{ij} : Counts for hypothesis $c_i=(A_i, \mathbf{t}_i)$ from model point j

N_j : Total number of counts from model point j

Maximum Entropy Distribution & Minimum Classification Error Training

2. Split Hough space into N model point dependent Hough spaces H_j

each H_j carries votes coming only from model point j

3. For each model point j

Learn individual classifier from its Hough space H_j

Model point dependent posterior class probability: $p_j(c_i|x) = N_{ij} / N_j$

N_{ij} : Counts for hypothesis $c_i=(A_i, \mathbf{t}_i)$ from model point j
N_j : Total number of counts from model point j

Model point specific Base Classifier: $\hat{c}_j = \arg\max_{c_i} p_j(c_i | x)$

Maximum Entropy Distribution & Minimum Classification Error Training

4. Optimal combination of the base classifiers $p_j(c_i|x)$?

"Maximum Objectivity" $\Rightarrow$ Maximum Entropy Principle

$\Rightarrow$ Log-linear combination

Maximum Entropy Distribution & Minimum Classification Error Training

4. Optimal combination of the base classifiers $p_j(c_i|x)$?

"Maximum Objectivity" ⇒ Maximum Entropy Principle

⇒ Log-linear combination

Log-linear model combination

$$p_{\Lambda}(c_i \mid x) = \frac{\exp\left\{\sum_j \lambda_j \cdot \log p_j(c_i \mid x)\right\}}{\sum_{c'} \exp\left\{\sum_j \lambda_j \cdot \log p_j(c' \mid x)\right\}}$$

Maximum Entropy Distribution & Minimum Classification Error Training

4. Optimal combination of the base classifiers $p_j(c_i|x)$?

"Maximum Objectivity" ⇒ Maximum Entropy Principle

⇒ Log-linear combination

Log-linear model combination

$$p_\Lambda(c_i \mid x) = \frac{\exp\left\{\sum_j \lambda_j \cdot \log p_j(c_i \mid x)\right\}}{\sum_{c'} \exp\left\{\sum_j \lambda_j \cdot \log p_j(c' \mid x)\right\}}$$

Global Classifier $\hat{c} = \arg\max_{c_i} p_\Lambda(c_i \mid x)$

Maximum Entropy Distribution & Minimum Classification Error Training

4. Optimal combination of the base classifiers $p_j(c_i|x)$?

“Maximum Objectivity” ⇒ Maximum Entropy Principle

⇒ Log-linear combination

Log-linear model combination

$$p_\Lambda(c_i \mid x) = \frac{\exp\left\{\sum_j \boxed{\lambda_j} \cdot \log p_j(c_i \mid x)\right\}}{\sum_{c'} \exp\left\{\sum_j \lambda_j \cdot \log p_j(c' \mid x)\right\}}$$

Optimal integration of N model point knowledge sources into one classifier!

Open: Adjustment of weights λ_j!

Maximum Entropy Distribution & Minimum Classification Error Training

5. Optimize model point specific weights λ_j w. r. t. misclassification rate

Minimum Classification Error Training (MCE):

Optimization criterion: Error rate on training data

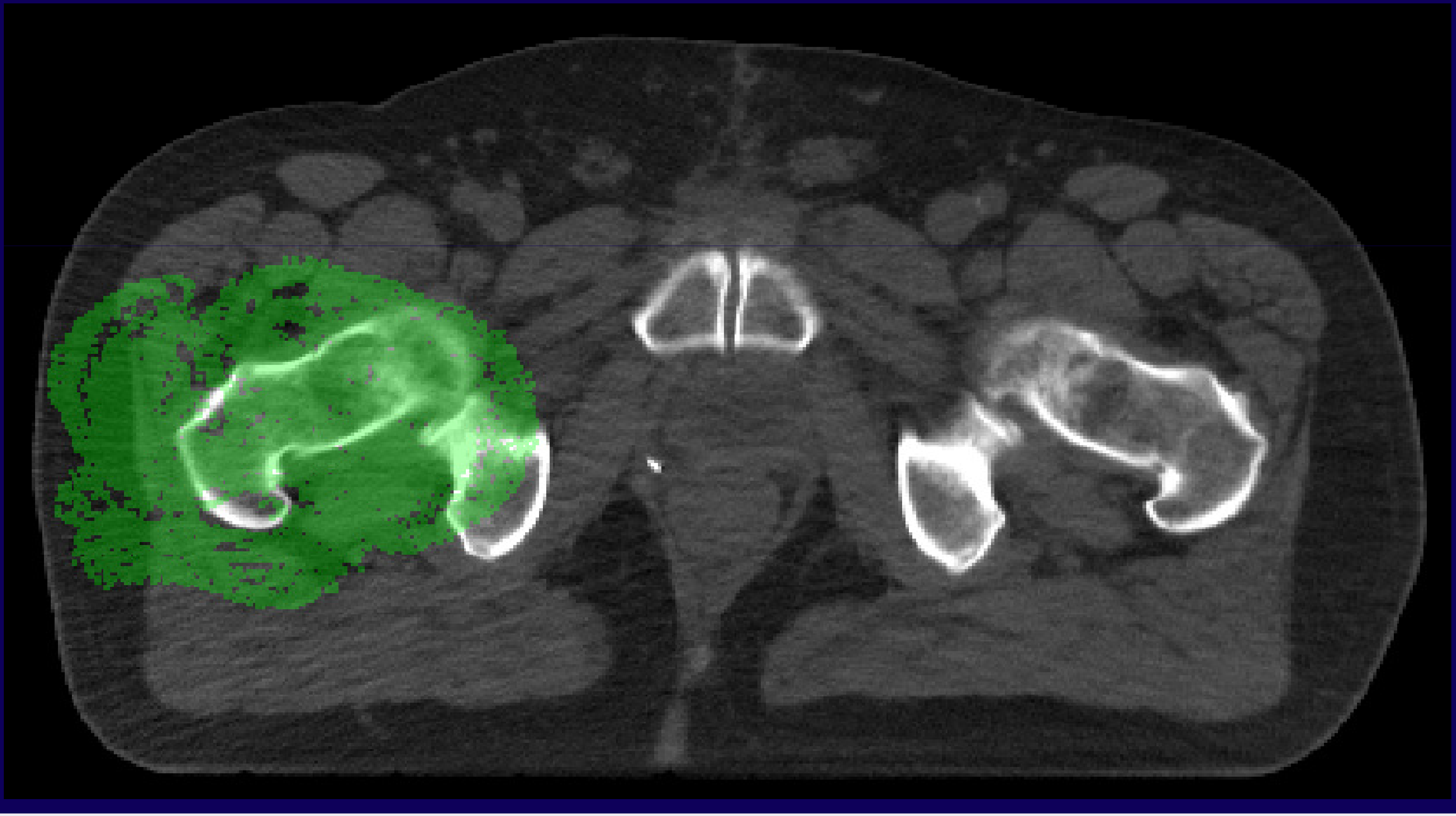
Outcome: Individual weighting of each model point in the shape model

Interpretation:

Positive weights	shape information
Negative weights	anti-shape information
Small absolute weights	unimportant model points

Experimental Results

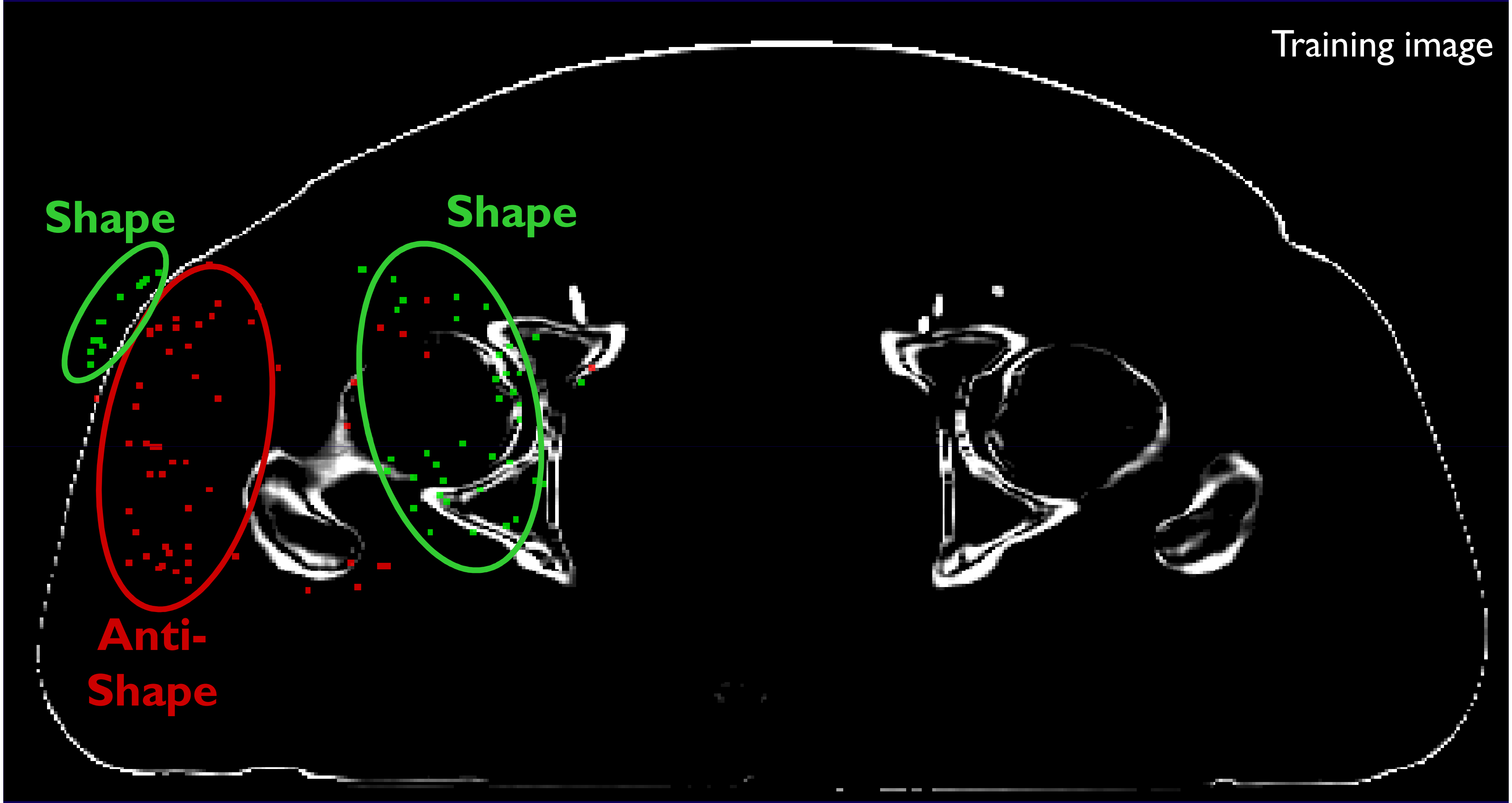
Starting point: 10k ‘random’ point cloud (taken from heart shapes)



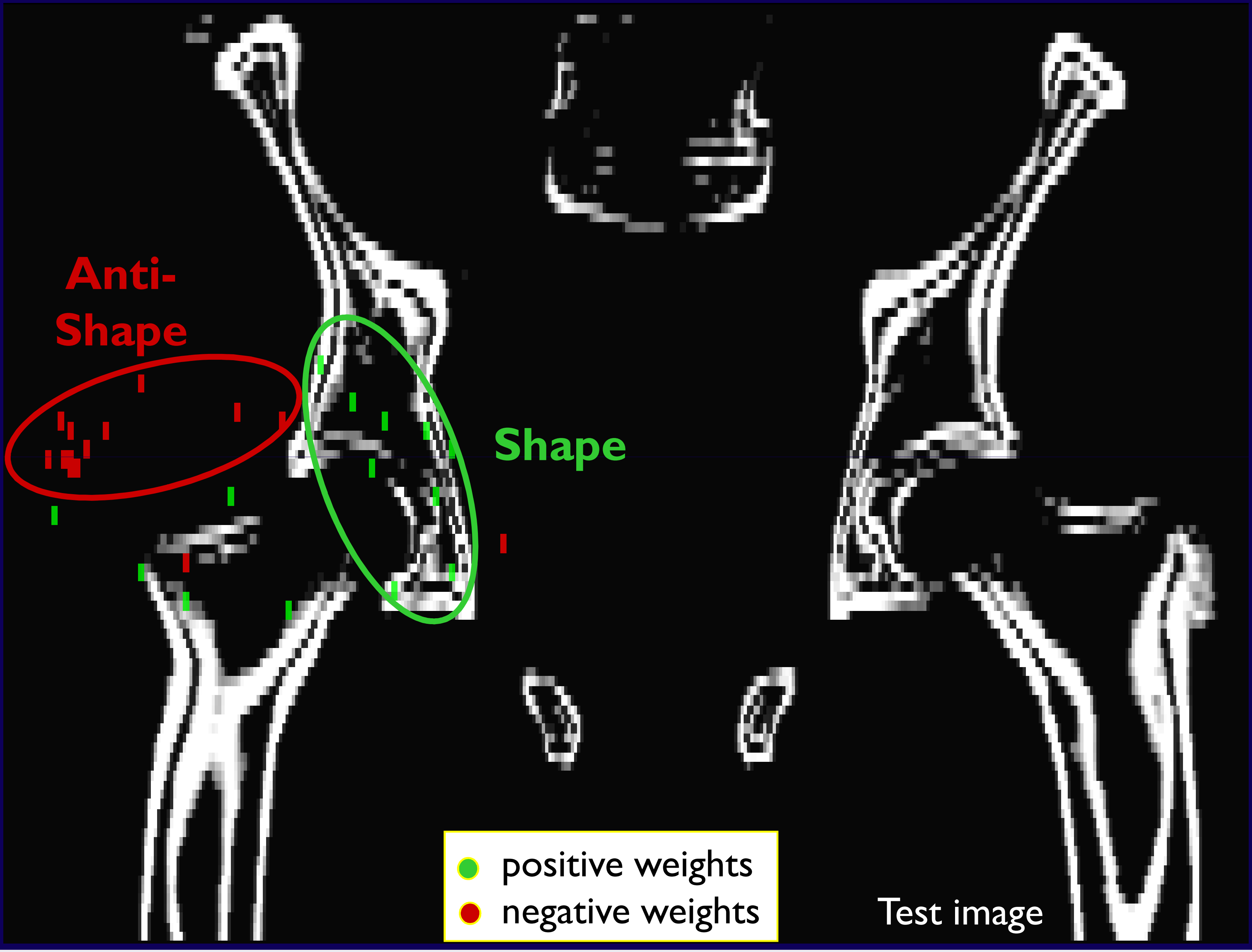
Experimental Results

Starting point: 10k 'random' point cloud (taken from heart shapes

1. Optimization of the 10k model weights on 3 training images
2. Selection of 1k points, re-optimization



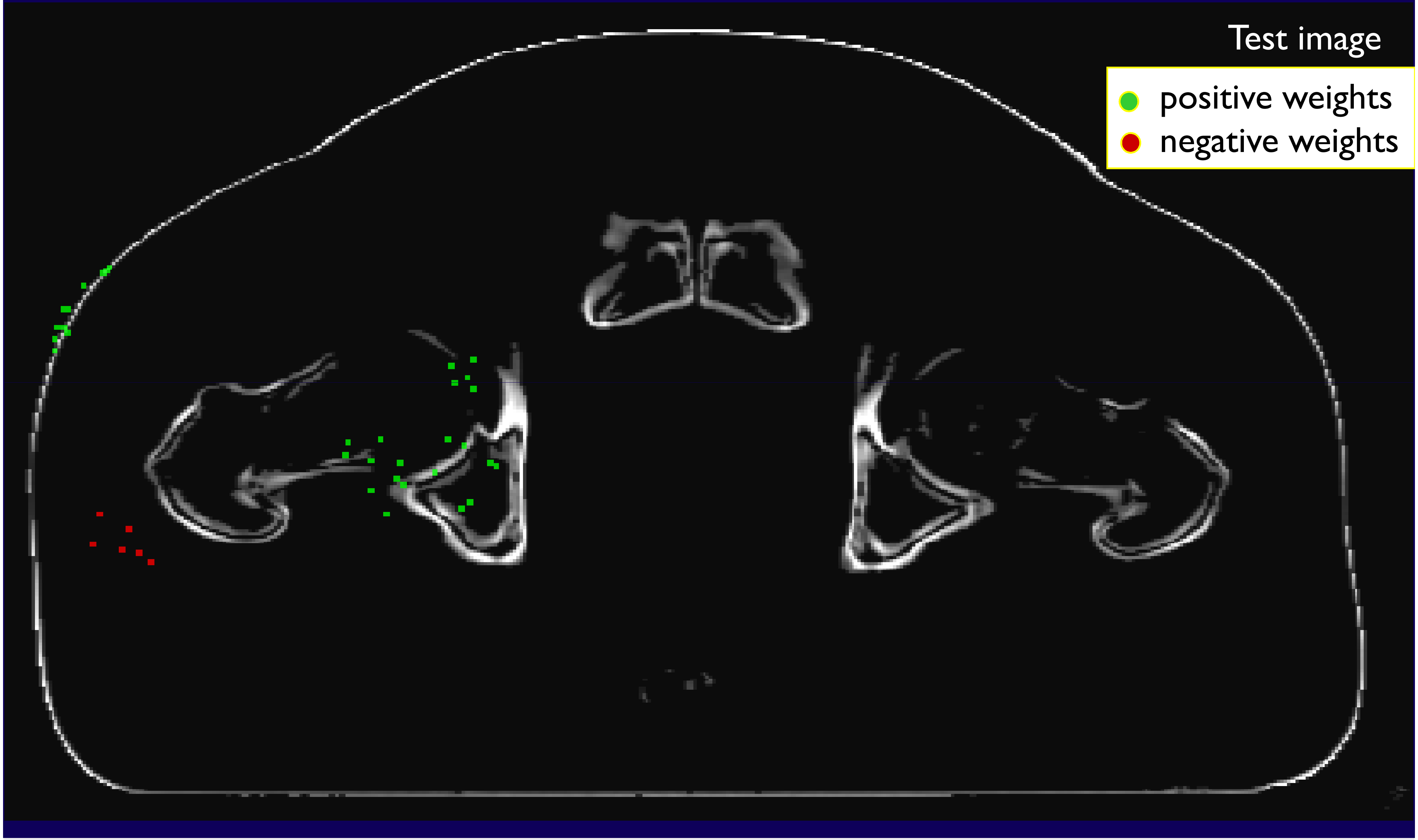
● positive weights
● negative weights


Anti-Shape
Shape
🟢 positive weights
🔴 negative weights
Test image

Experimental Results

Starting point: 10k 'random' point cloud (taken from heart shapes

1. Optimization of the 10k model weights on 3 training images
2. Selection of 1k points, re-optimization
3. Selection of 250 points (from the 1k), re-optimization



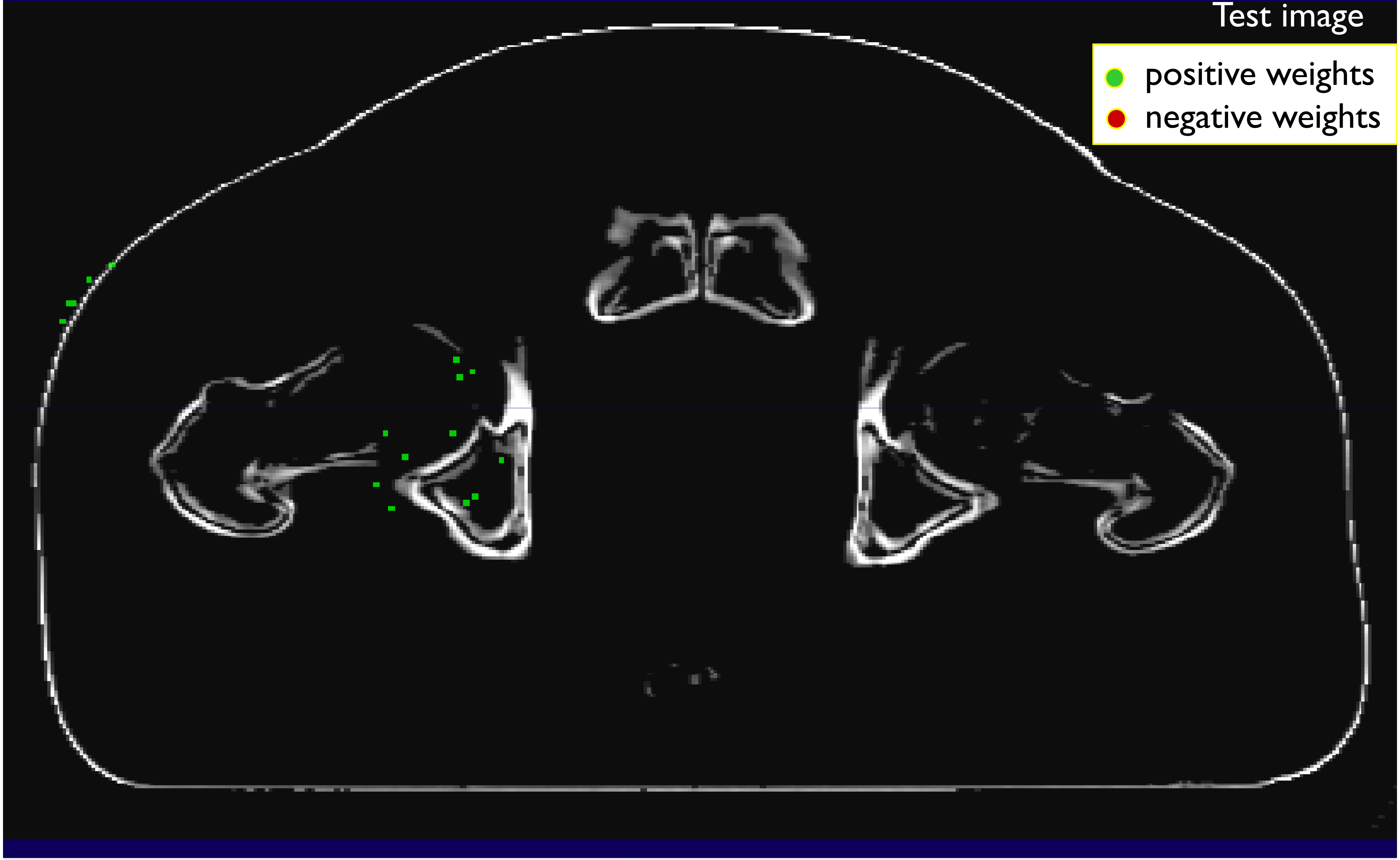
Test image

Experimental Results

Starting point: 10k ‘random’ point cloud (taken from heart shapes

1. Optimization of the 10k model weights on 3 training images
2. Selection of 1k points, re-optimization
3. Selection of 250 points (from the 1k), re-optimization
4. Selection of 100 points (from the 250), re-optimization

Test image



🟢 positive weights

🔴 negative weights

Experimental Results

Starting point: 10k ‘random’ point cloud (taken from heart shapes

1. Optimization of the 10k model weights on 3 training images
2. Selection of 1k points, re-optimization
3. Selection of 250 points (from the 1k), re-optimization
4. Selection of 100 points (from the 250), re-optimization

First validation of 100 point model on 7 unknown test images:

Maximum average distance to ground truth: ~1cell in Hough space

Hough space very focused (more than with regular shape model)

Conclusions

- Automatic 3D object detection technique presented
- Based on Generalized Hough Transform
- High detection rates despite simple shape model
- New technique developed for

  automatic generation of discriminative shape models

- Learns optimal shapes and anti-shapes w.r.t. classification error
- First experiments: femur detection
- Learning of model with 100 pts from 10k random point cloud
- Successful validation on 7 test images (traing.: 3 images)
- Very focussed Hough space ⇒ model is discriminative