Computational Cardiology Lecture 11

# Image coordinates, pixel-cell meshes and geometric sensitivity

Lecture notes

A medical image supplies an array together with a spatial coordinate system. Thresholding, coordinate transformation and mesh construction are separate operations. Their effects can be derived explicitly for a two-dimensional pixel-cell mesh, allowing geometric scale and segmentation decisions to be distinguished.

## 1 Intensity arrays and spatial metadata

Slides for this section

An image-based model begins with a chain of decisions. Our question is whether a change in a reported chamber area comes from the segmentation, the physical scale, or the mesh construction. Those are different causes, so we keep separate objects for intensity values, the binary mask, spatial metadata and mesh connectivity. The companion makes every object from a synthetic generator; the ellipse-like foreground is an instructional shape, not an anatomical reconstruction. That controlled origin gives us a reference mask without pretending that expert clinical annotation is perfect.

The stored image has 28 rows and 36 columns. Its values lie between zero and one and represent arbitrary synthetic intensity. They are not millimeters, pressure or a probability of disease. The generator combines overlapping ellipses, blurs the ideal mask, and adds a small reproducible noise field. Blurring means that a sharp reference boundary becomes a band of intermediate intensities. A threshold converts that band into a discrete choice. The choice should be recorded even if the resulting outline looks convincing.

The pipeline figure deliberately shows indices on the left and physical coordinates on the right. A screenshot alone cannot establish that two images occupy the same place. SimpleITK's Images section identifies size, spacing, origin and direction as the spatial description; it also emphasizes consistent units. In this lesson we adopt millimeters explicitly. We retain the array and metadata together rather than infer physical scale from display width.

The intensity function $I_{r,c}$ and the physical point $x_{r,c}$ are distinct objects. A threshold $t$ defines a mask $M_t = \{(r, c) : I_{r,c} \geq t\}$, but it supplies no length scale. Area becomes meaningful only after a spatial map and physical unit are attached.

## 2 The affine index-to-physical map

Slides for this section

NumPy indexing uses row first and column second. Our geometric point vector instead lists column then row, corresponding to local x then local y. Let the column spacing be 1.25 mm and row spacing be 2 mm. Multiply these spacings by the respective indices, rotate the resulting local displacement, then add

the physical origin. Matrix multiplication order matters: rotating an index vector and then applying unequal spacings generally produces a different answer. The direction matrix in this companion is a proper rotation; it preserves lengths and signed area after spacing has been applied.

The origin is the center of pixel zero, not its lower-left corner. A pixel cell extends half a spacing on either side of its center. Therefore the lower-left corner of pixel at column c, row r uses indices c minus one half and r minus one half. Confusing center and corner conventions translates the boundary. Translation leaves area unchanged, so an area check alone cannot catch that error. We additionally test the location of index zero and inspect coordinate orientation.

For our rotation of 20 degrees, positive columns and positive rows point along rotated axes. The implementation rejects nonpositive spacing and unsupported direction matrices instead of silently producing an ambiguous mesh. This is a deliberate scope restriction of the teaching implementation, not a claim that all imaging formats use this particular orientation convention.

$$\mathbf{x} = \mathbf{o} + D\,\mathrm{diag}(s_x, s_y)\begin{bmatrix} c \\ r \end{bmatrix}$$

Origin and position are in mm; spacings are mm per index increment; D is a dimensionless proper rotation.

Let $A = D\,\mathrm{diag}(s_x, s_y)$. Translation changes locations but not edge differences. The signed area scale is

$$\det A = \det D\, s_x s_y = s_x s_y$$

for the proper rotations accepted here. Applying unequal spacing after rotation would use a different matrix, since in general $D\,\mathrm{diag}(s_x, s_y) \neq \mathrm{diag}(s_x, s_y)D$.

**Calculation.** With no rotation, origin (−20,−10) mm, spacing (1.25,2) mm, locate row 3, column 4.

Write the geometric index vector as (4,3), not (3,4).

The displacement is (1.25×4, 2×3) = (5,6) mm.

Add the origin: (−20,−10)+(5,6) = (−15,−4) mm.

The point is (−15,−4) mm. An index-order swap would produce (−16.25,−2) mm.

## 3 Oriented triangles and mask area

Slides for this section

Thresholding at 0.5 selects 295 pixels in the default image. Each foreground pixel becomes a physical rectangle, split along a consistent diagonal into two triangles. Adjacent cells share the same vertex indices. This produces a conforming mesh of the pixel-cell union, rather than a set of disconnected triangles that merely overlap in a plot. The companion removes unused grid vertices and saves coordinates and connectivity in geometry-data.npz. There are 590 triangles and 337 used vertices in the default construction.

The signed determinant formula gives each triangle's area. Positive signs check the chosen counterclockwise ordering, while the sum checks agreement with the pixel-cell area. Every triangle has area 1.25 square millimeters within floating-point tolerance. Shared edges occur twice and boundary edges once.

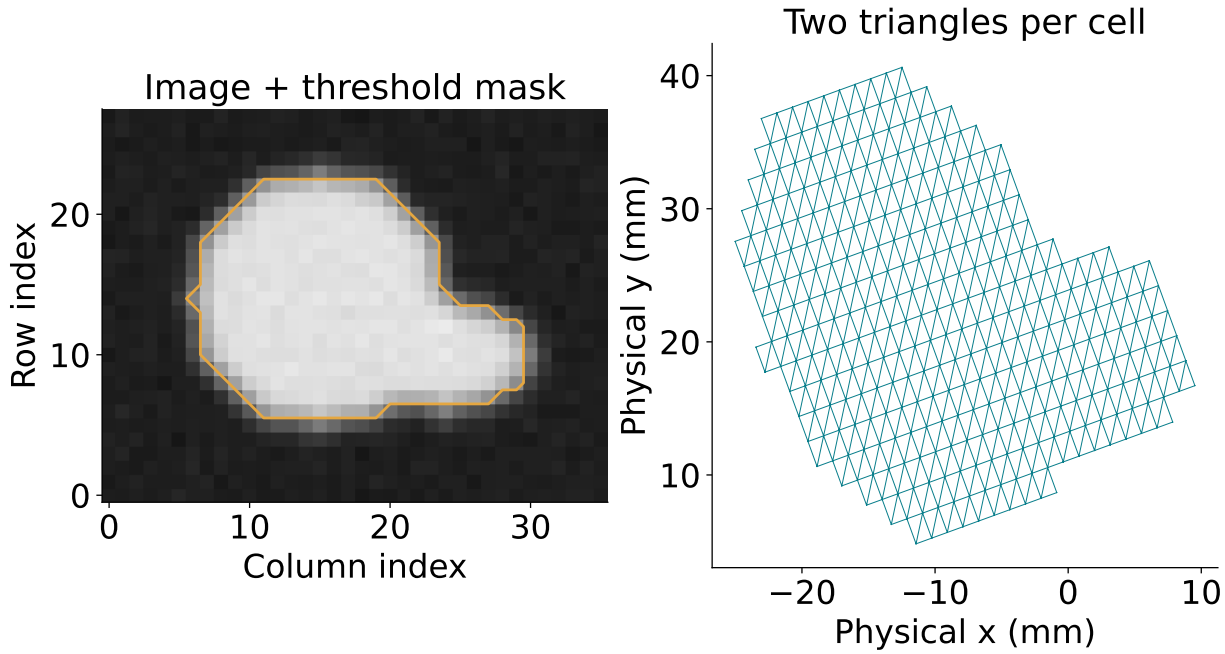
These checks would reveal reversed ordering or duplicate elements, but they do not prove that a mesh is suitable for a fluid solver. A three-dimensional simulation would additionally need a valid volume mesh, appropriately labeled boundaries, physical models and solver verification.

An exact area match here means the representation of this particular mask is consistent. It does not establish that the mask is anatomically correct. Our staircase boundary remains a consequence of pixel resolution. Smoothing it would be a new geometry operation whose effect on area, connectivity and narrow features must be measured separately.

$$A_T = \tfrac{1}{2} \det \begin{bmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{bmatrix}, \qquad A_M = N_f s_x s_y$$

Triangle vertex coordinates are in mm; both areas are in $\mathrm{mm}^2$; N_f is the foreground pixel count.

Each foreground pixel has four corner coordinates obtained from $(c \pm \frac{1}{2}, r \pm \frac{1}{2})$. A consistent diagonal divides its area $s_x s_y$ into two triangles of area $s_x s_y/2$. Adjacent pixels share indexed vertices and edges, so the mesh represents the union of selected cells without internal overlaps.




Original synthetic image and actual 2D mesh. The 0.5 mask has 295 cells and 590 triangles; area is 737.5 $\mathrm{mm}^2$. Index axes and physical axes are intentionally different.

## 4 Threshold sets and geometric summaries

Slides for this section

The threshold sweep changes one decision while holding the image and spatial metadata fixed. Higher thresholds cannot add foreground under the greater-than-or-equal rule, so the masks must be nested. The default sweep from 0.35 to 0.65 decreases area from 815 to 682.5 square millimeters. All seven masks remain one component with four-neighbor connectivity. That last fact is a result for this image, not a guarantee that other images preserve a narrow connection.

We also report the diameter of a circle with the same area. This is a scalar summary, not a fitted circle or the width of an appendage. Because diameter depends on the square root of area, its relative changes are smaller. A threshold sensitivity range is not a confidence interval: no probability distribution over plausible thresholds has been supplied. A defensible report names the perturbed parameter, the chosen range and what stayed fixed.

Refining triangles inside an unchanged pixel cell cannot improve the underlying image boundary. Likewise, doubling both physical spacings quadruples every area even though the mask is identical. These controls distinguish implementation consistency, segmentation sensitivity and physical-scale uncertainty. They should be examined before interpreting a downstream flow change.

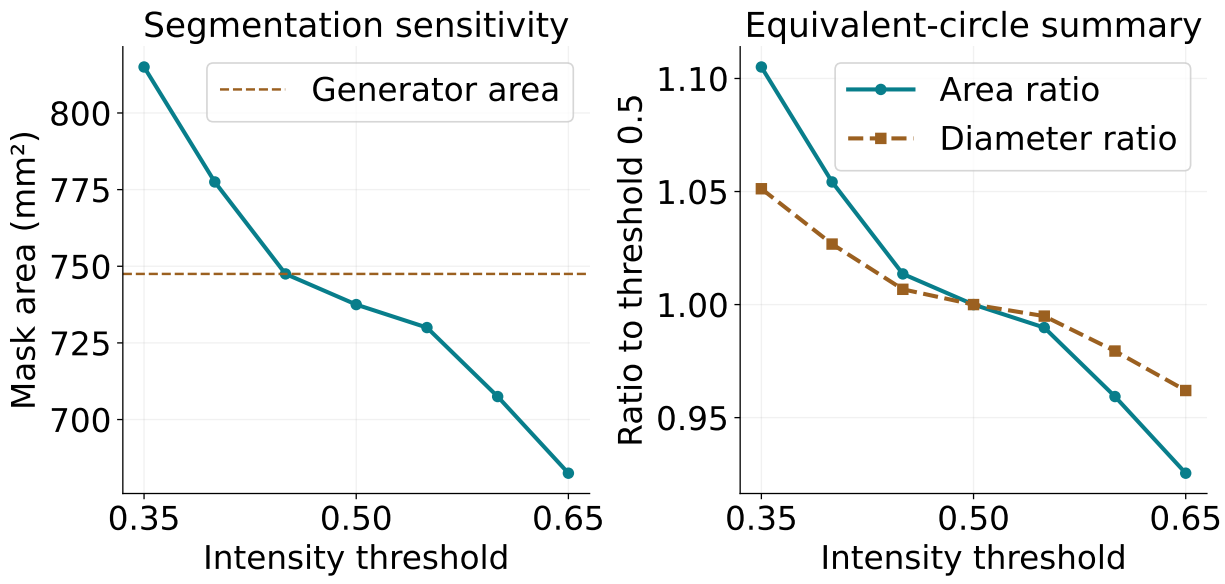
$$d_{\mathrm{eq}} = 2\sqrt{A_M/\pi}$$

Area is in $\mathrm{mm}^2$ and equivalent diameter is in mm. This defines a geometric summary, not a flow law.

For $t_2 > t_1$, the threshold sets satisfy $M_{t_2} \subseteq M_{t_1}$, hence $A(t_2) \le A(t_1)$. No such monotonicity follows for the number of connected components: removing foreground can split a bridge or remove an isolated component. The equivalent diameter has differential

$$d_{eq} = 2\sqrt{A/\pi}, \qquad \frac{\delta d_{eq}}{d_{eq}} = \frac{1}{2}\frac{\delta A}{A}.$$

It summarizes area rather than maximum width or anatomical shape.




Same input and metadata, seven thresholds. The reference line is the generator mask area, 747.5 $\mathrm{mm}^2$. Equivalent diameter refers to an equal-area circle; ratios use threshold 0.5 as their denominator.

## 5 Resampling and geometric provenance

Slides for this section

The provenance file records the generator seed, intensity-array hash and byte convention, shape, threshold comparison, spatial metadata, unit, library versions and mesh construction rule. A hash can establish

byte identity under a stated encoding; it cannot establish correct units or biological validity. Two analysts can hash identical arrays yet disagree geometrically if one discards spacing. Conversely, tiny differences in a regenerated noise field can change a few boundary pixels, so store the actual synthetic array as well as the seed.

When resampling labels, the SimpleITK Resampling section describes nearest-neighbor interpolation as preserving the label set. That does not promise preservation of a thin feature or its area. Record the target grid and transform direction, then recheck geometry. Keep interpolation of intensities distinct from interpolation of class labels. The present construction performs no resampling, so threshold sensitivity is isolated from interpolation effects.

Changing both spacings by a factor $a$ changes area by $a^2$ and equivalent diameter by $a$, while leaving pixel counts unchanged. A coordinate-origin error can leave all these scalar quantities unchanged. Therefore area consistency cannot determine whether a mesh is correctly located or oriented.

## 6 Representation error and physical interpretation

Slides for this section

Refining a triangle inside a selected pixel changes the mesh resolution but not the pixel-cell union. It cannot recover an unresolved anatomical boundary. Conversely, a change to the mask boundary changes the represented domain even if the subsequent triangulation is exact. The synthetic construction establishes internal geometric consistency; anatomical validity requires an independent image interpretation.

## Exercises

### AC11-E1. Audit a spatial map

Use origin (10,-5) mm, spacing (2,3) mm and a 90-degree counterclockwise rotation D = [[0,-1],[1,0]].
**a** Locate the center at row 2, column 4.

**b** Locate the lower-left cell corner in local index coordinates and map it physically.

**c** What is the cell area, and does changing only the origin alter it?

### AC11-E2. Check triangle orientation

A triangle has vertices A=(0,0), B=(2,0), C=(2,3), all in millimeters.
**a** Compute signed area in order A,B,C.

**b** Compute signed area after swapping B and C. What should the mesh check report?

**c** Two triangles form the 2 mm by 3 mm rectangle. Explain why an area sum of 6 $mm^2$ is necessary but insufficient for solver readiness.

### AC11-E3. Separate segmentation and scale

A mask changes from 400 to 361 pixels when its threshold rises. Spacing remains (0.5,0.8) mm.
**a** Find both areas and the signed percent area change.

**b** Find the ratio of equivalent-circle diameters without calculating pi.

**c** If both spacings are accidentally doubled, how do area and equivalent diameter change for either fixed mask?

### AC11-E4. Write a reproducibility receipt

A colleague sends only mask.png and says it is ready for cardiac CFD.
**a** Name four missing items needed to reproduce its physical geometry.

**b** Give one implementation check and one independent validation question.

**c** State three additional ingredients a 3D flow calculation needs beyond this lesson's 2D mesh.

## References

Zan Ahmad, Advanced Topics in Computational Cardiology syllabus, Fall 2024. PDF pages 1–2.

SimpleITK, Fundamental Concepts. Images; Resampling; Resampling—Common Errors. https://simpleitk.readthedocs.io/en/master/fundamentalConcepts.html