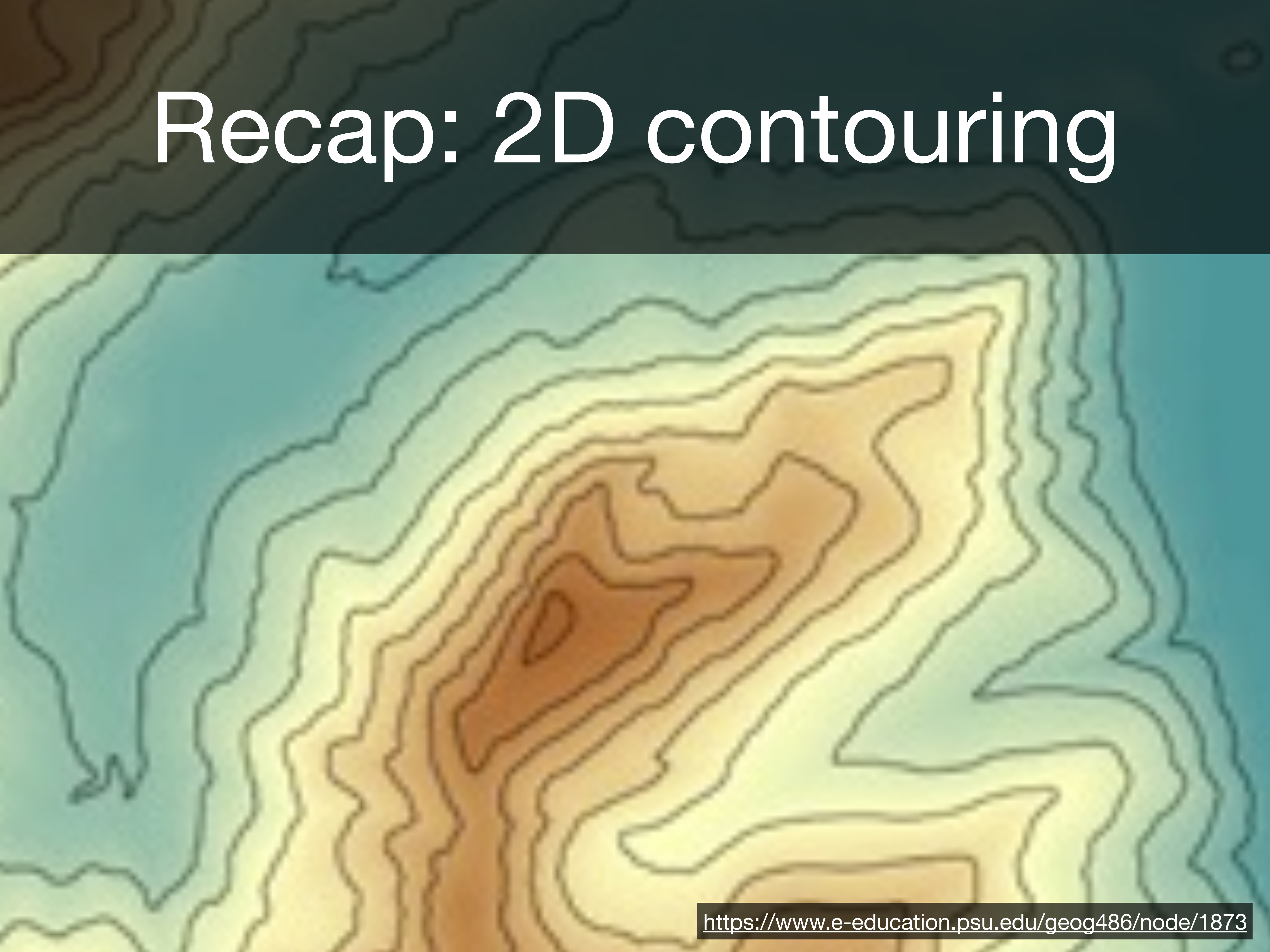
Spatial Data: 3D Scalar Fields

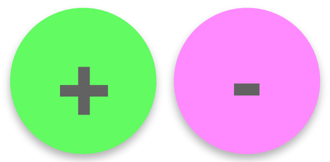
CSC444

Recap: 2D contouring

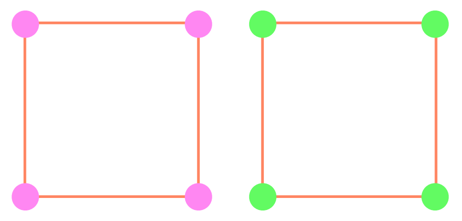


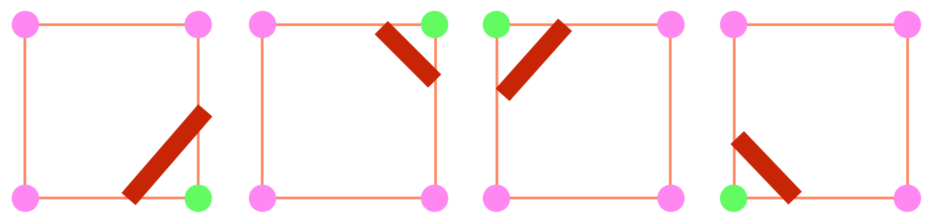
https://www.e-education.psu.edu/geog486/node/1873

Recap: 2D contouring

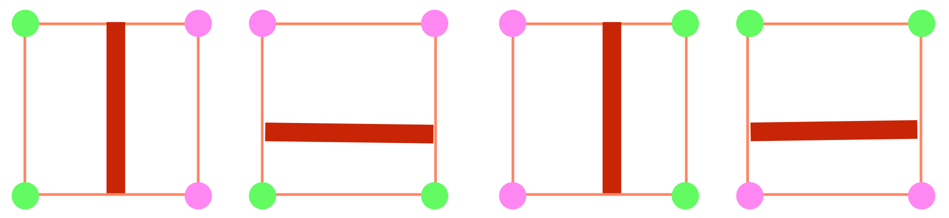


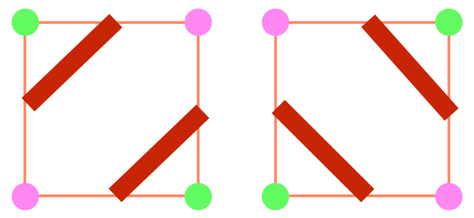
Case	Polarity	Rotation	Total
No Crossings	x2		2
Singlet	x2	x4	8
Double adjacent	x2	x2 (4)	4
Double Opposite	x2	x1 (2)	2
			$16 = 2^4$



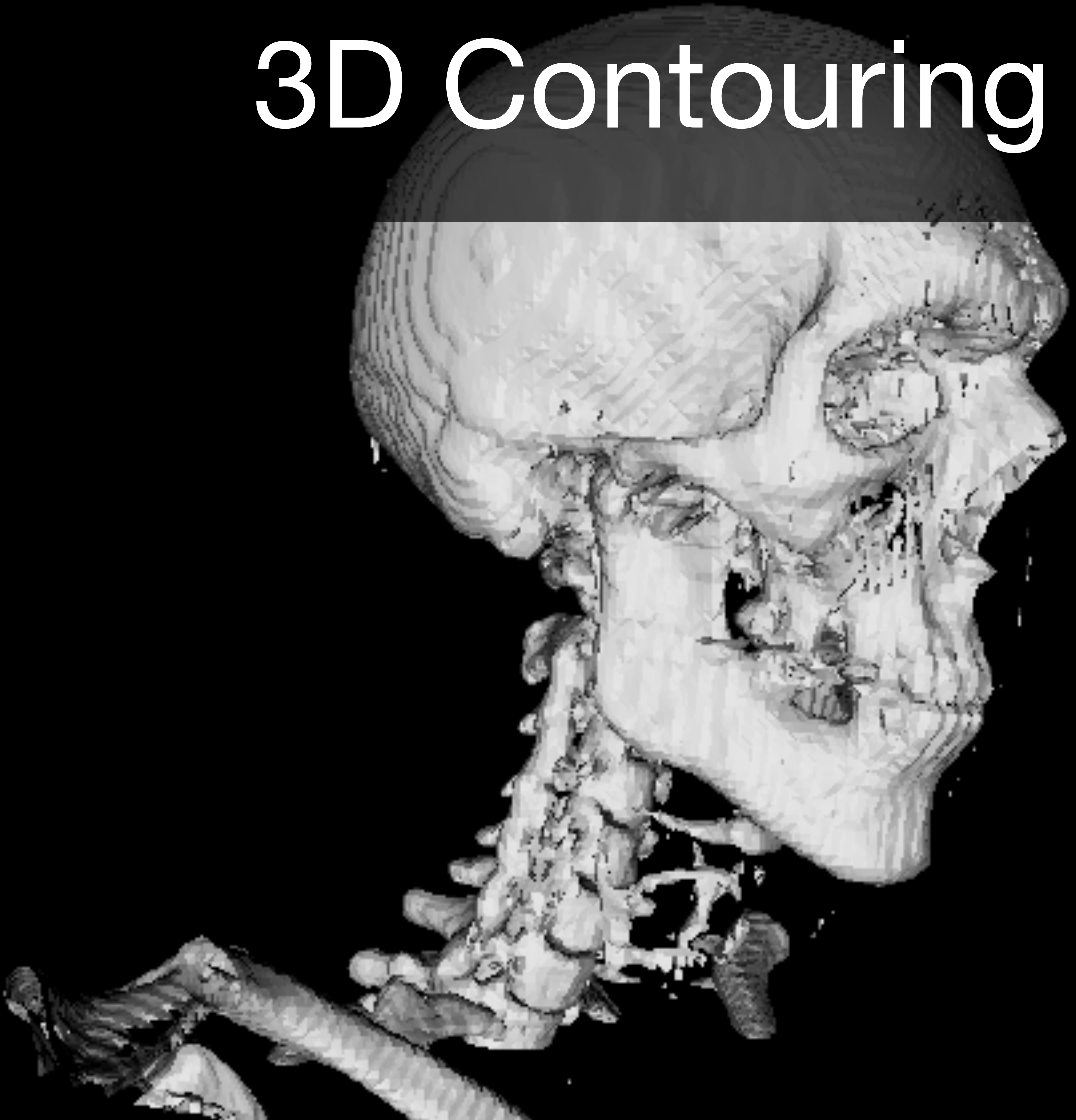


(x2 for polarity)

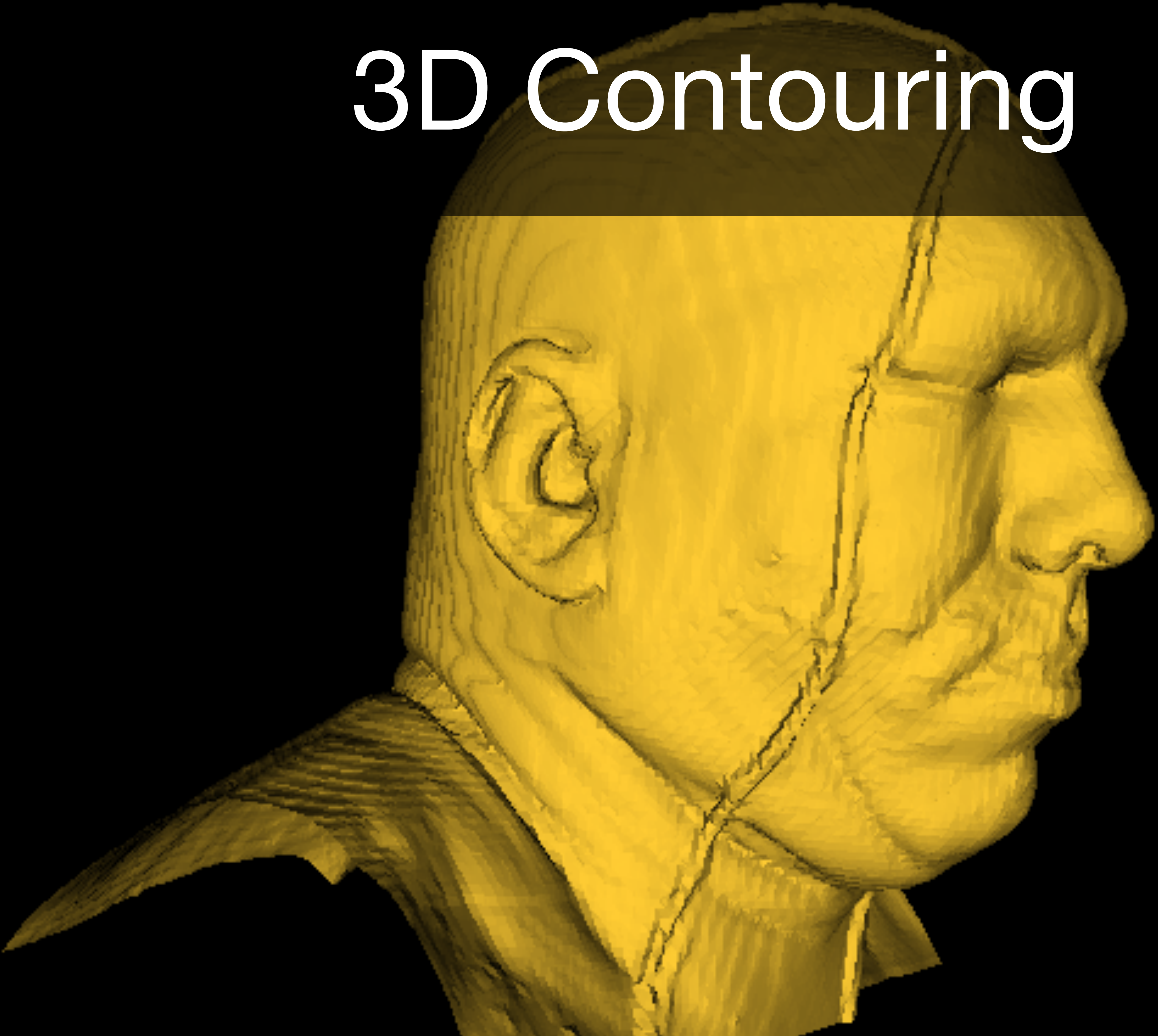




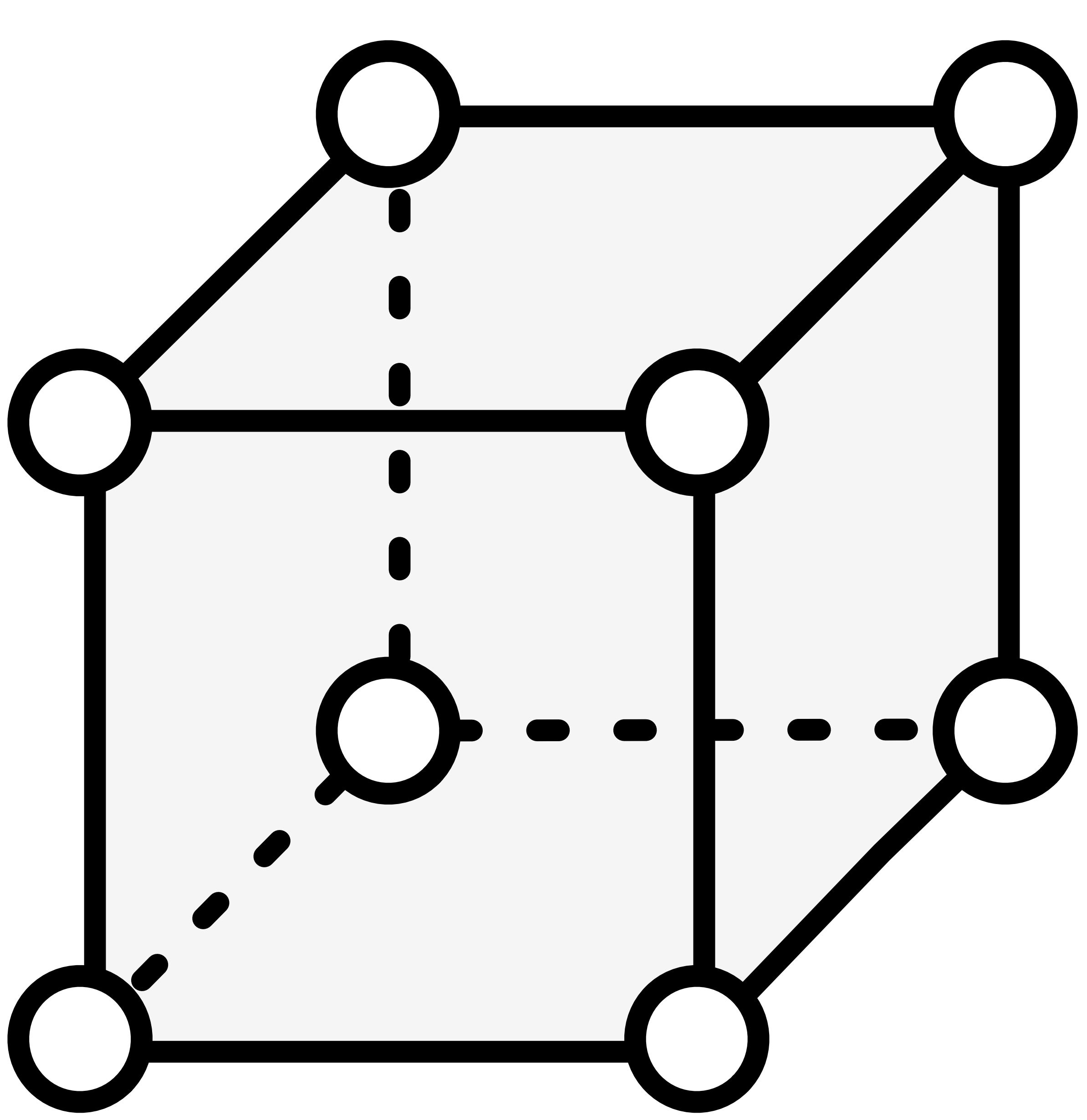
3D Contouring

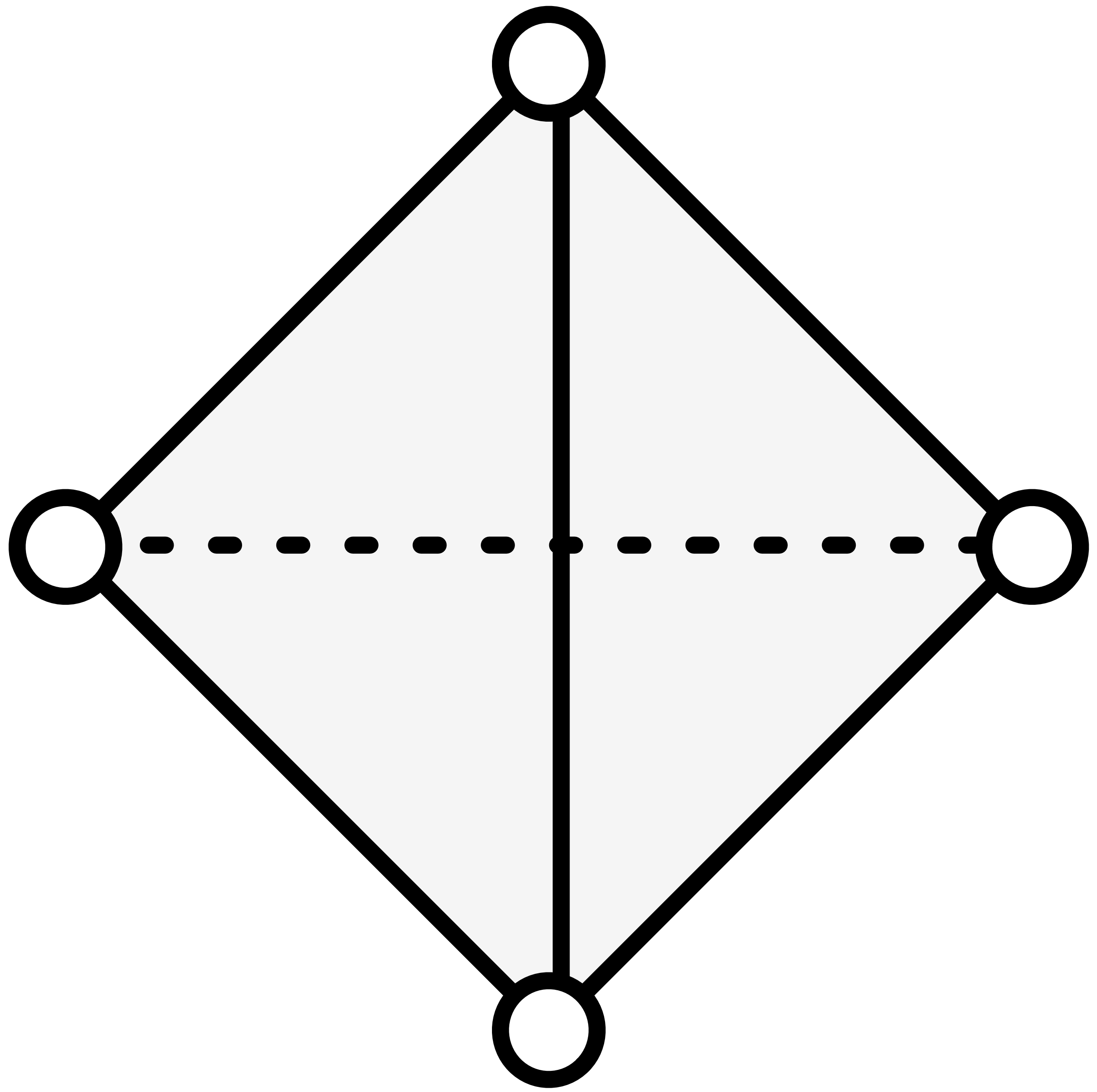


3D Contouring

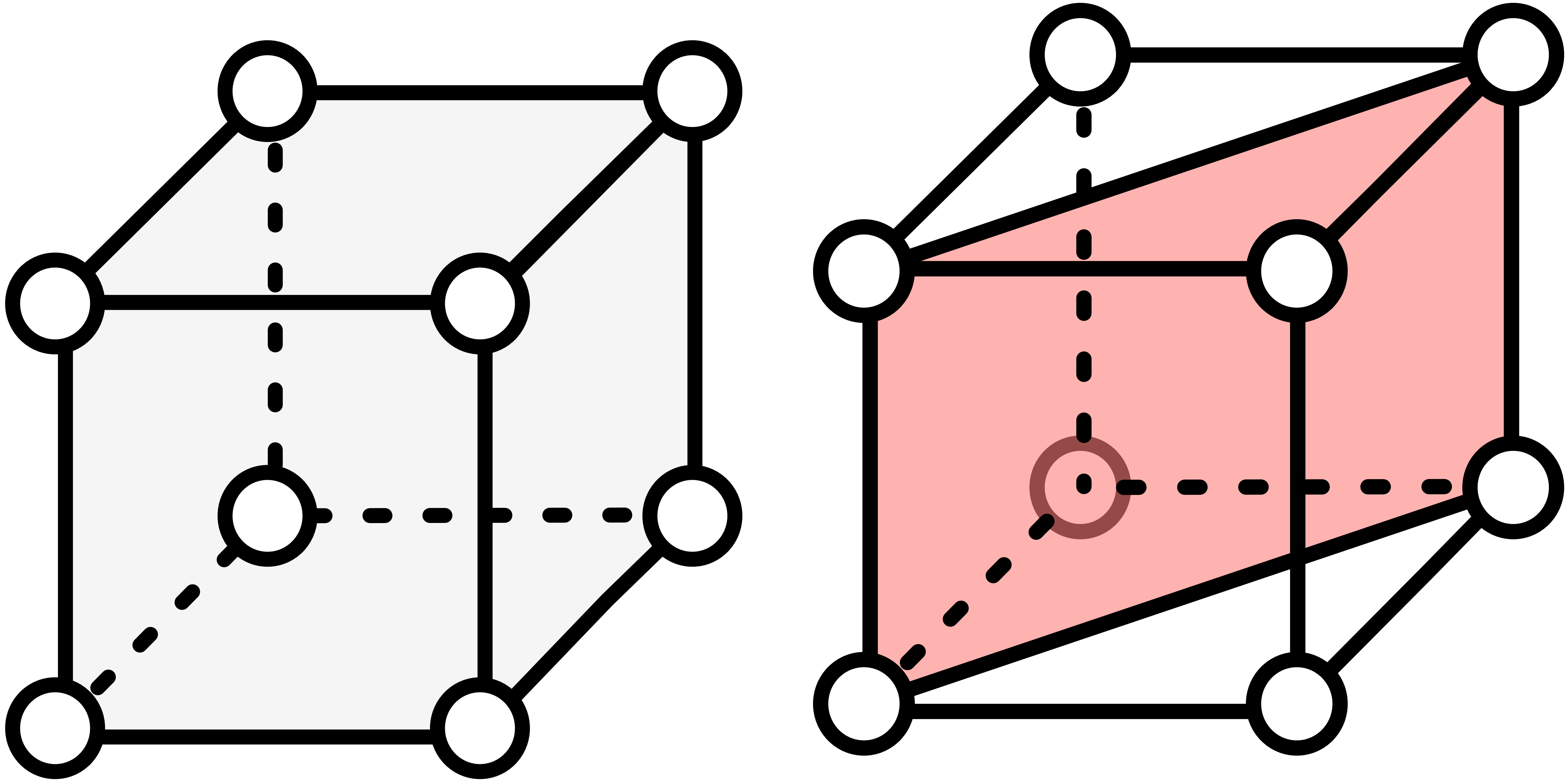


Splitting 3D space into simple shapes

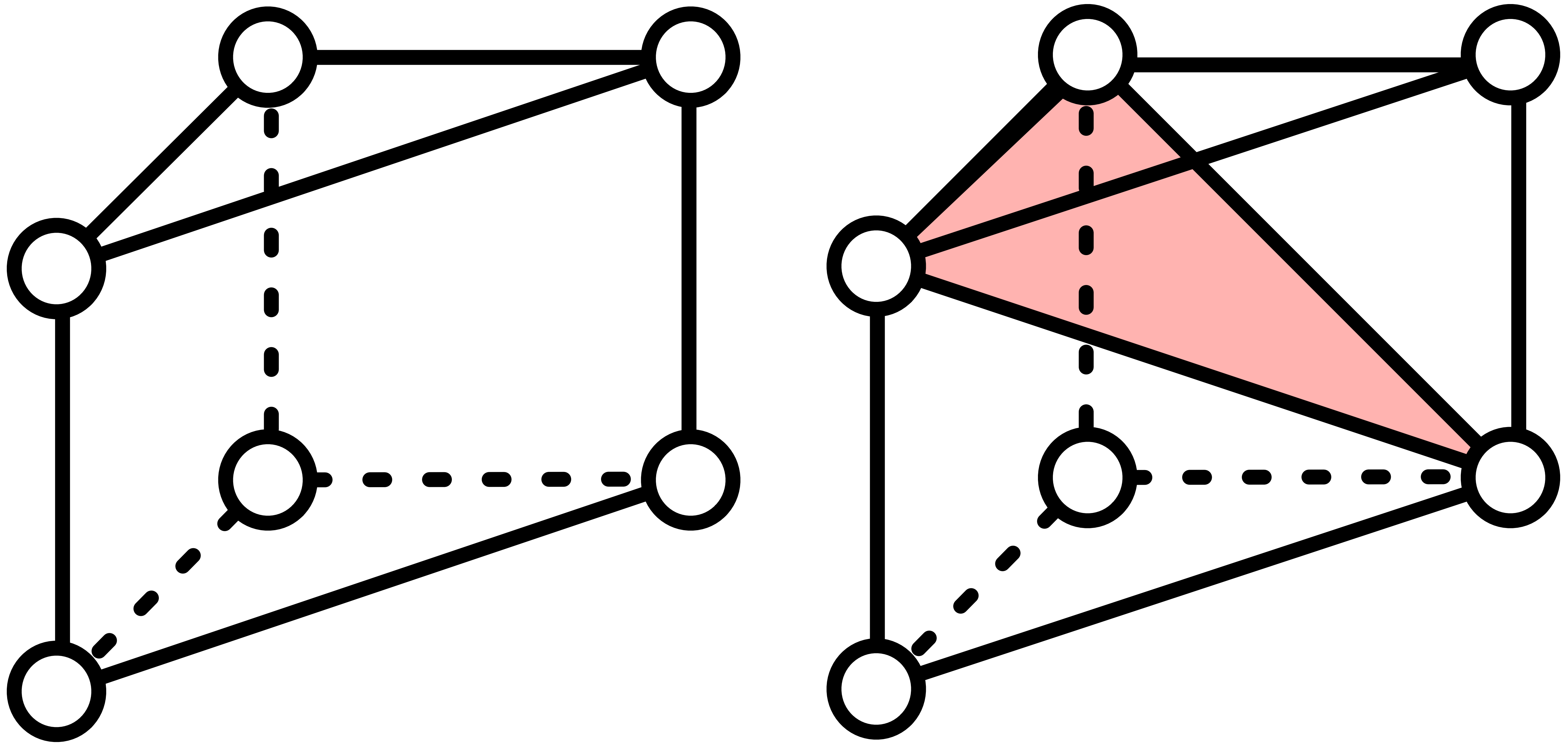




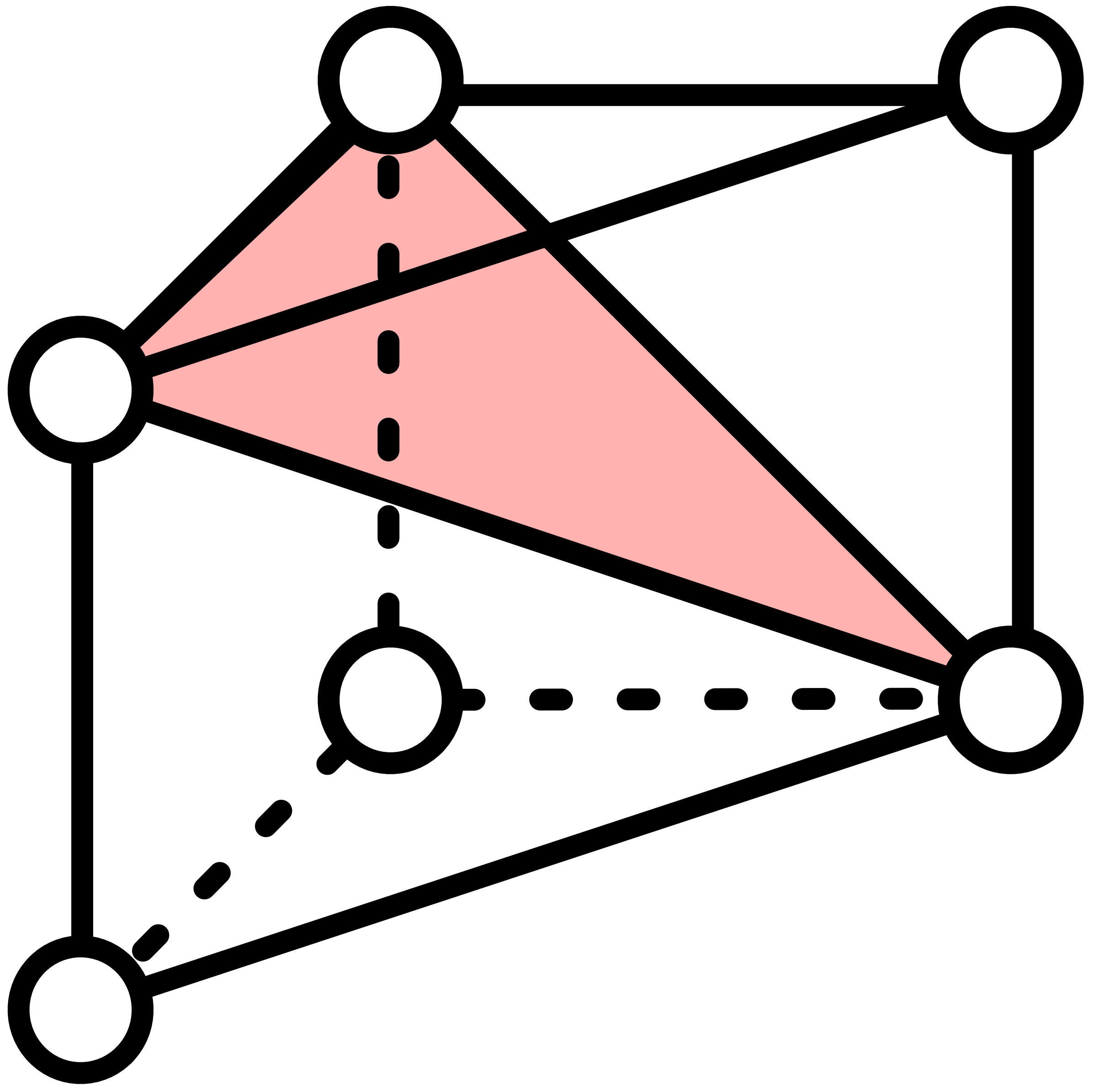
Cube into tetrahedra

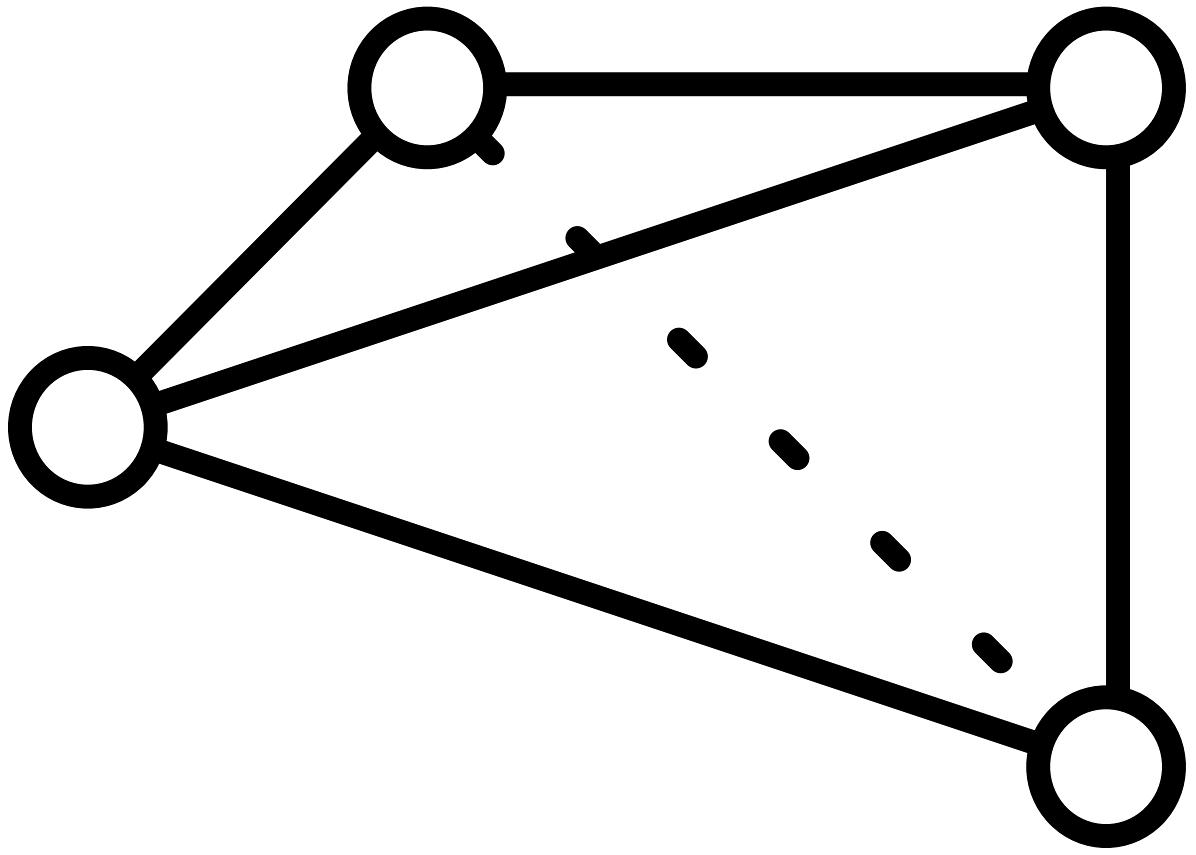


Cube into tetrahedra



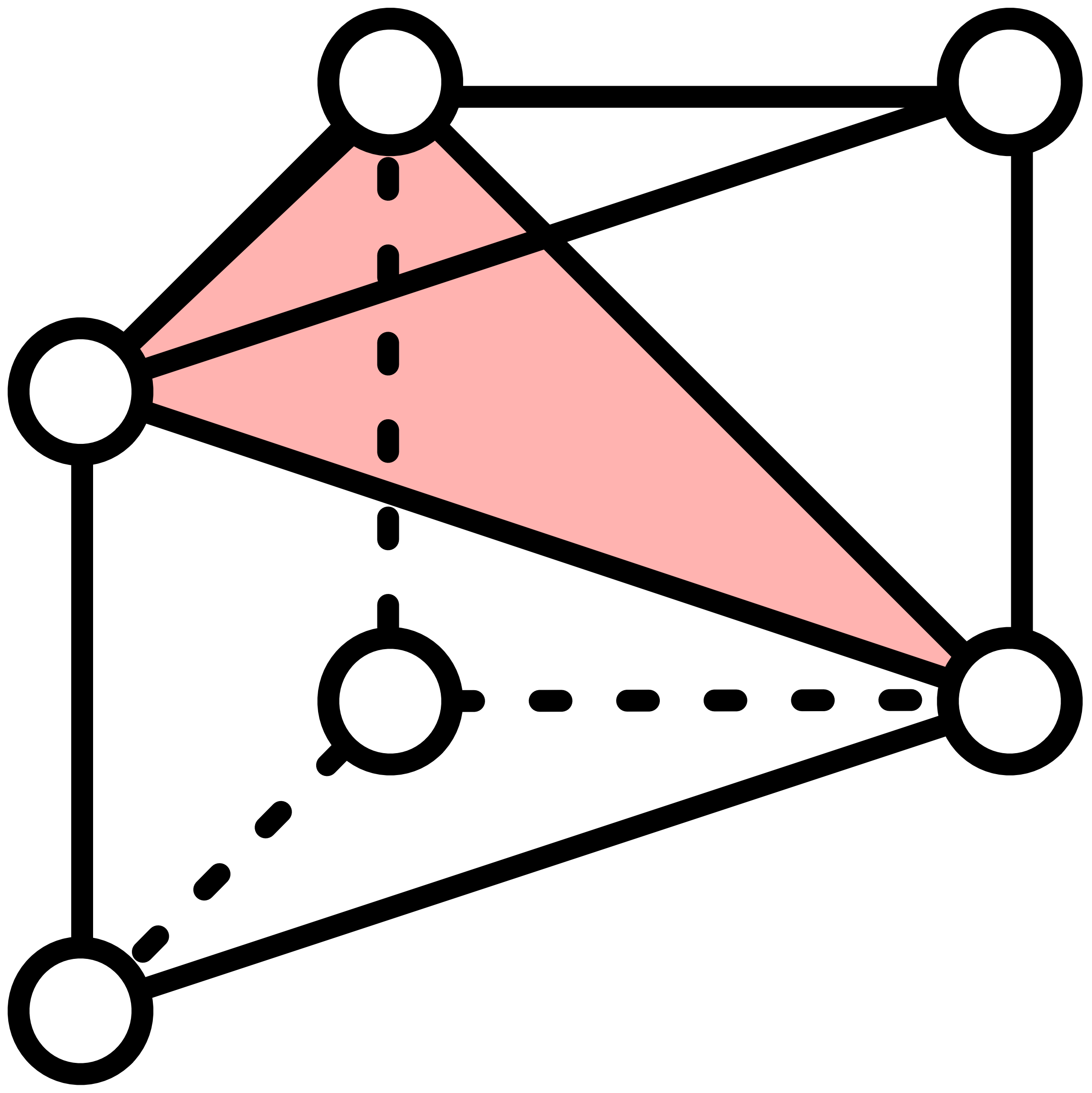
Cube into tetrahedra

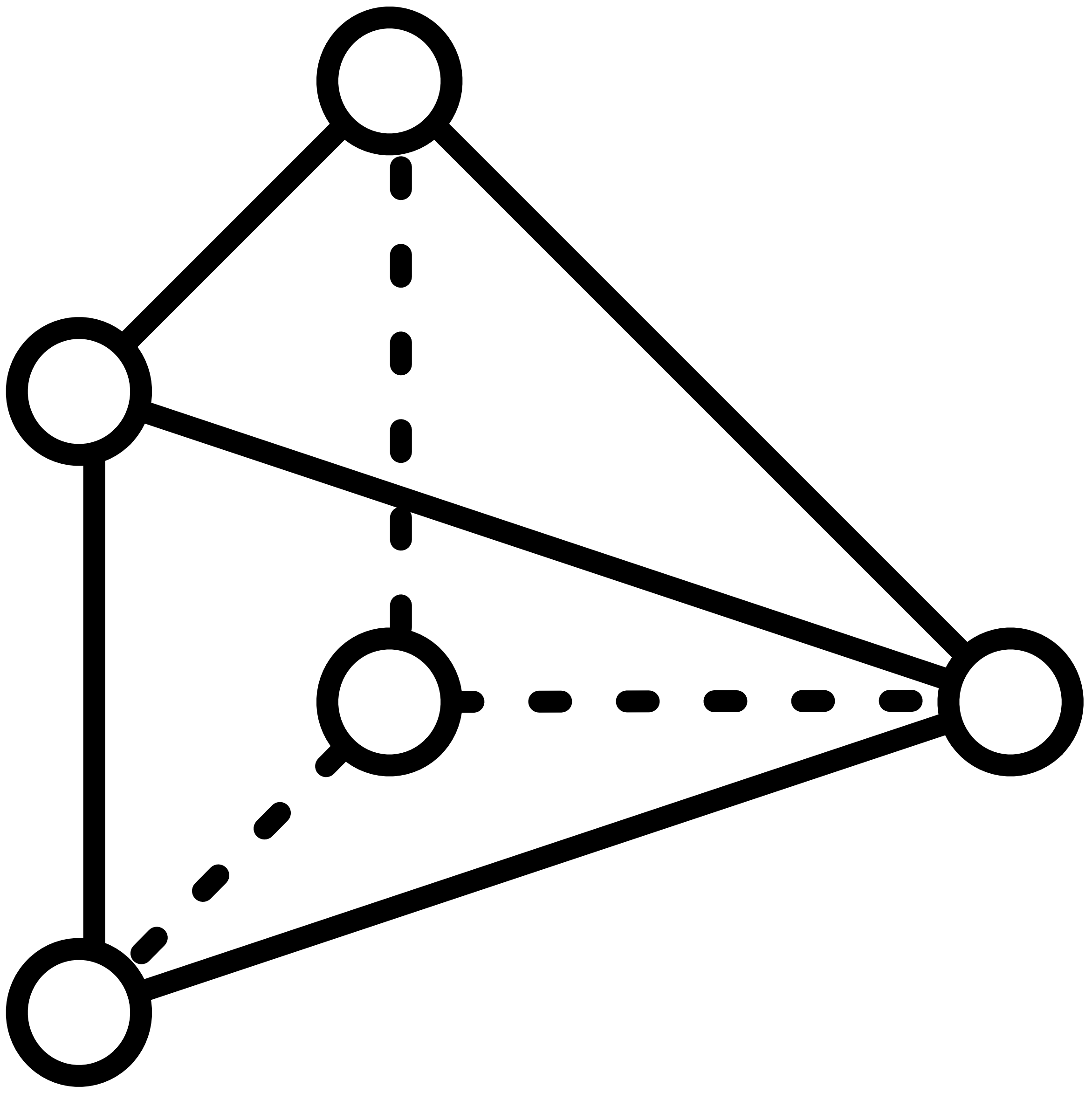




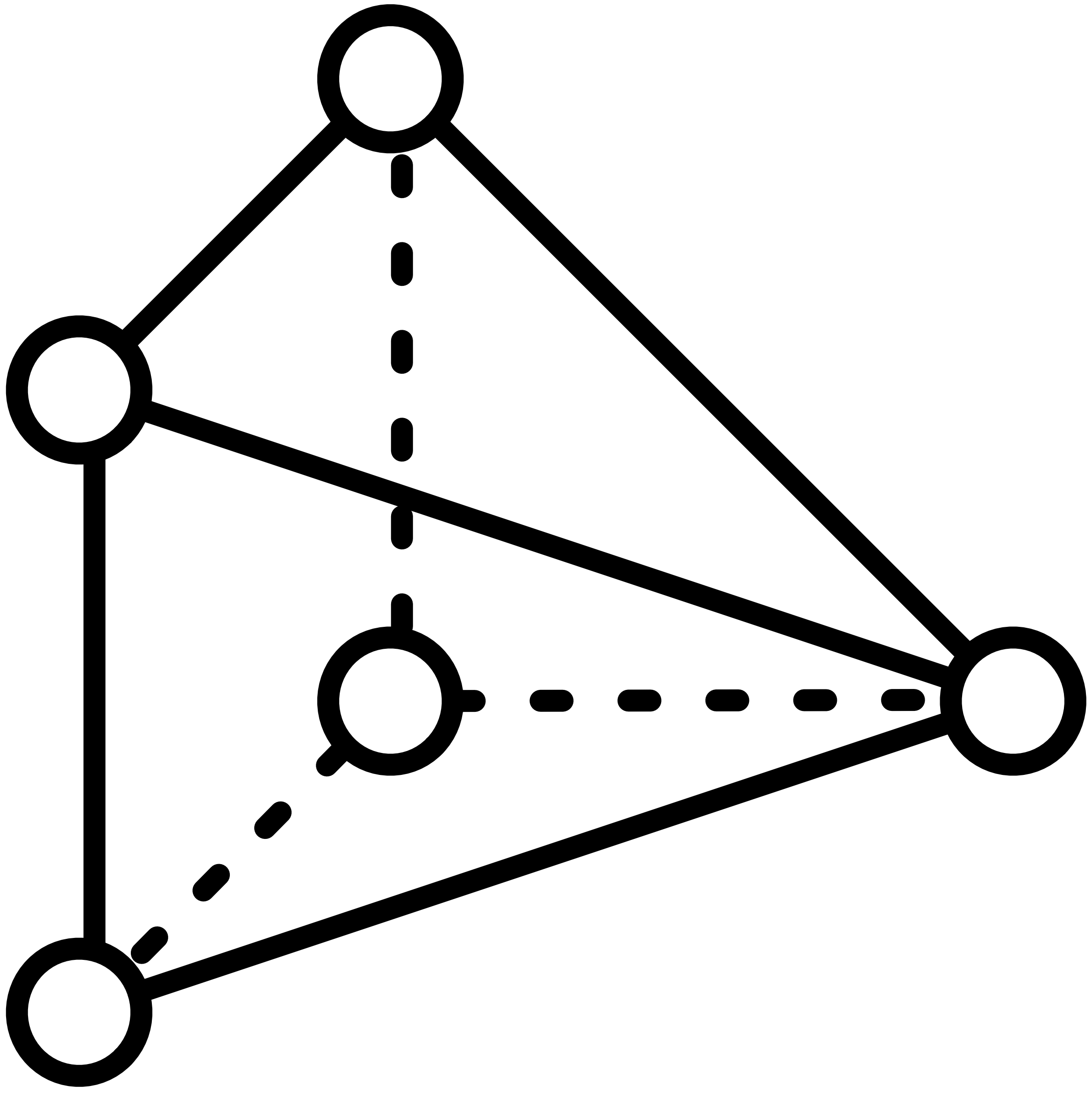
1 tetrahedron,

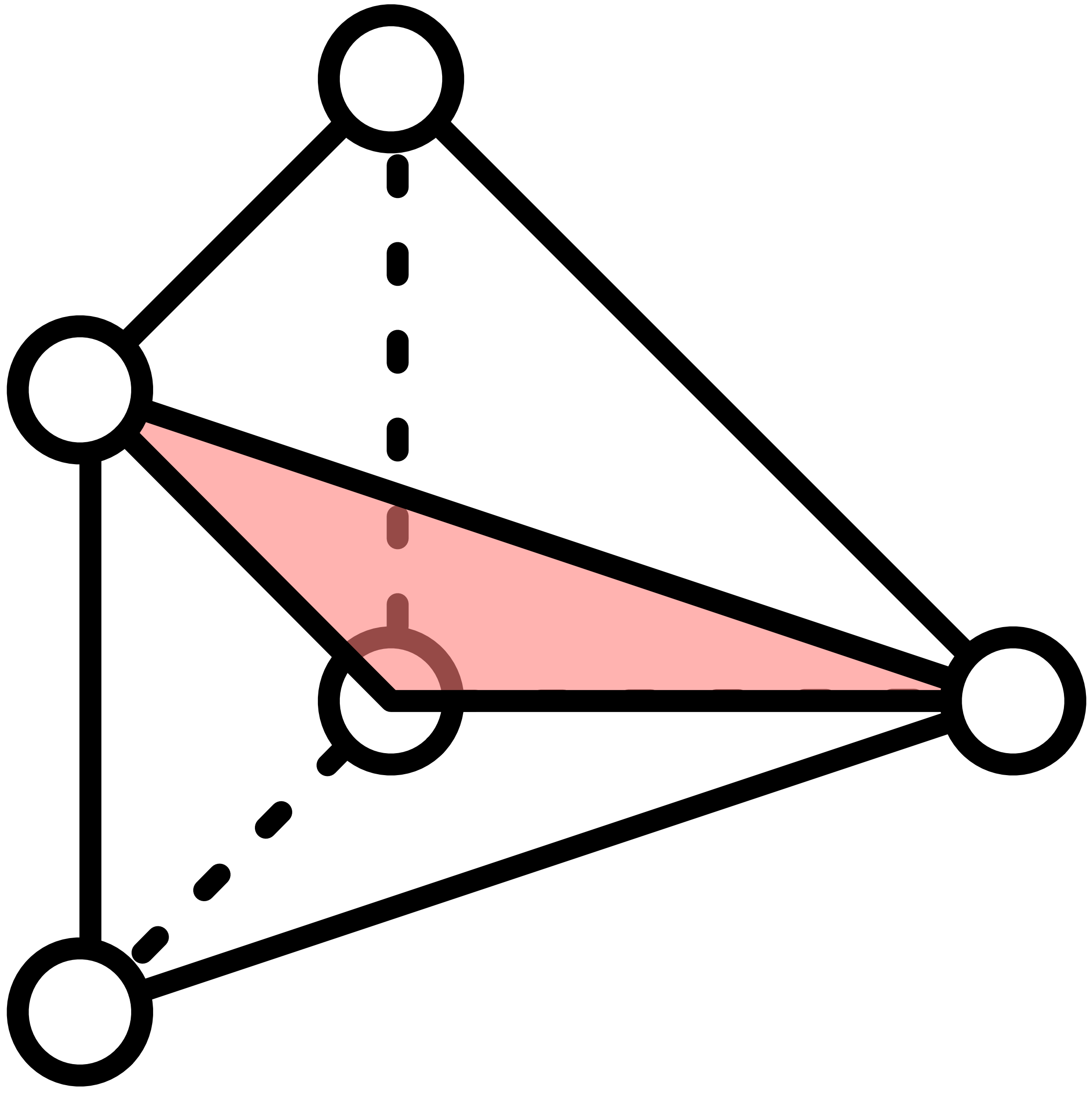
Cube into tetrahedra





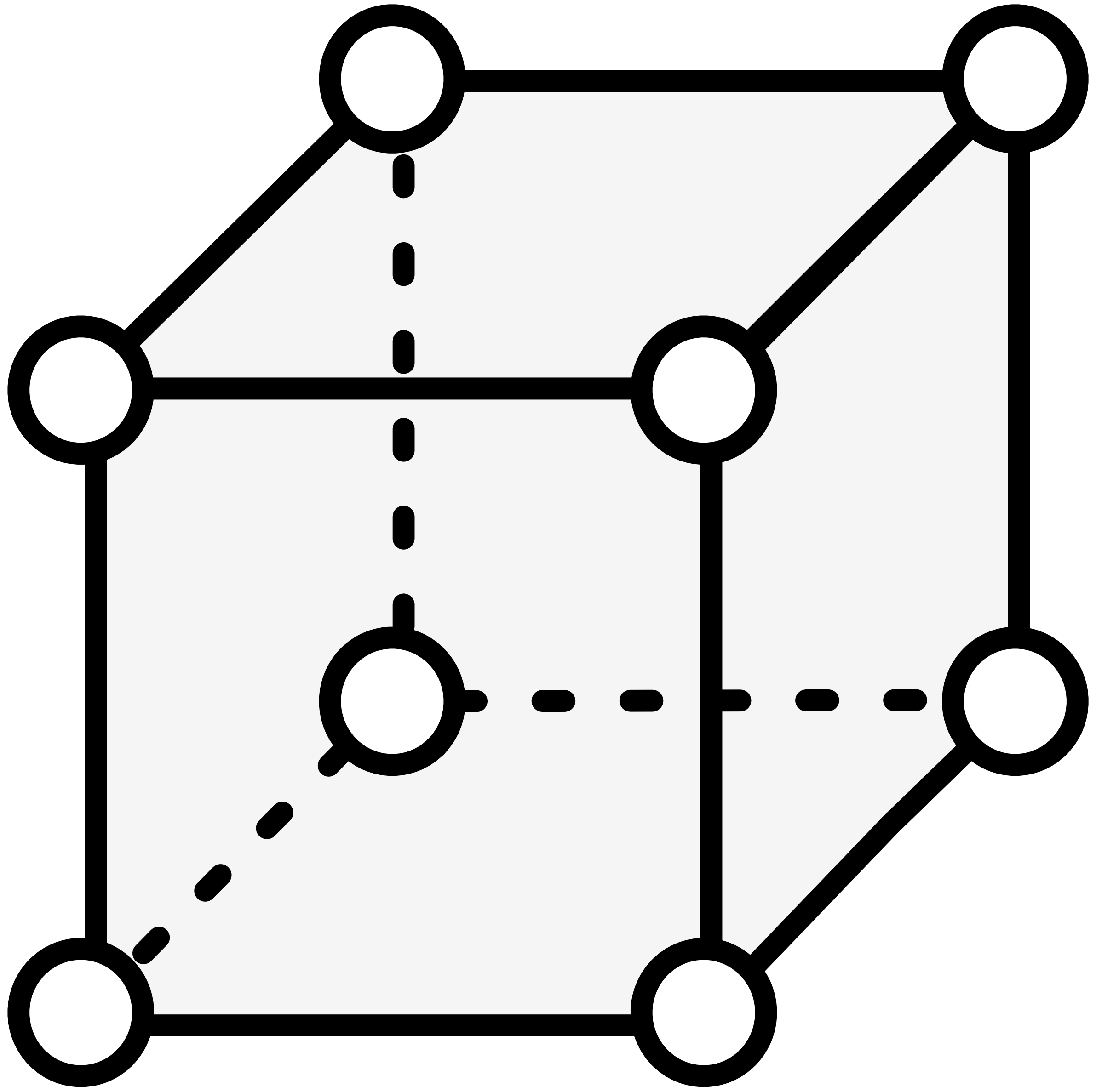
Cube into tetrahedra





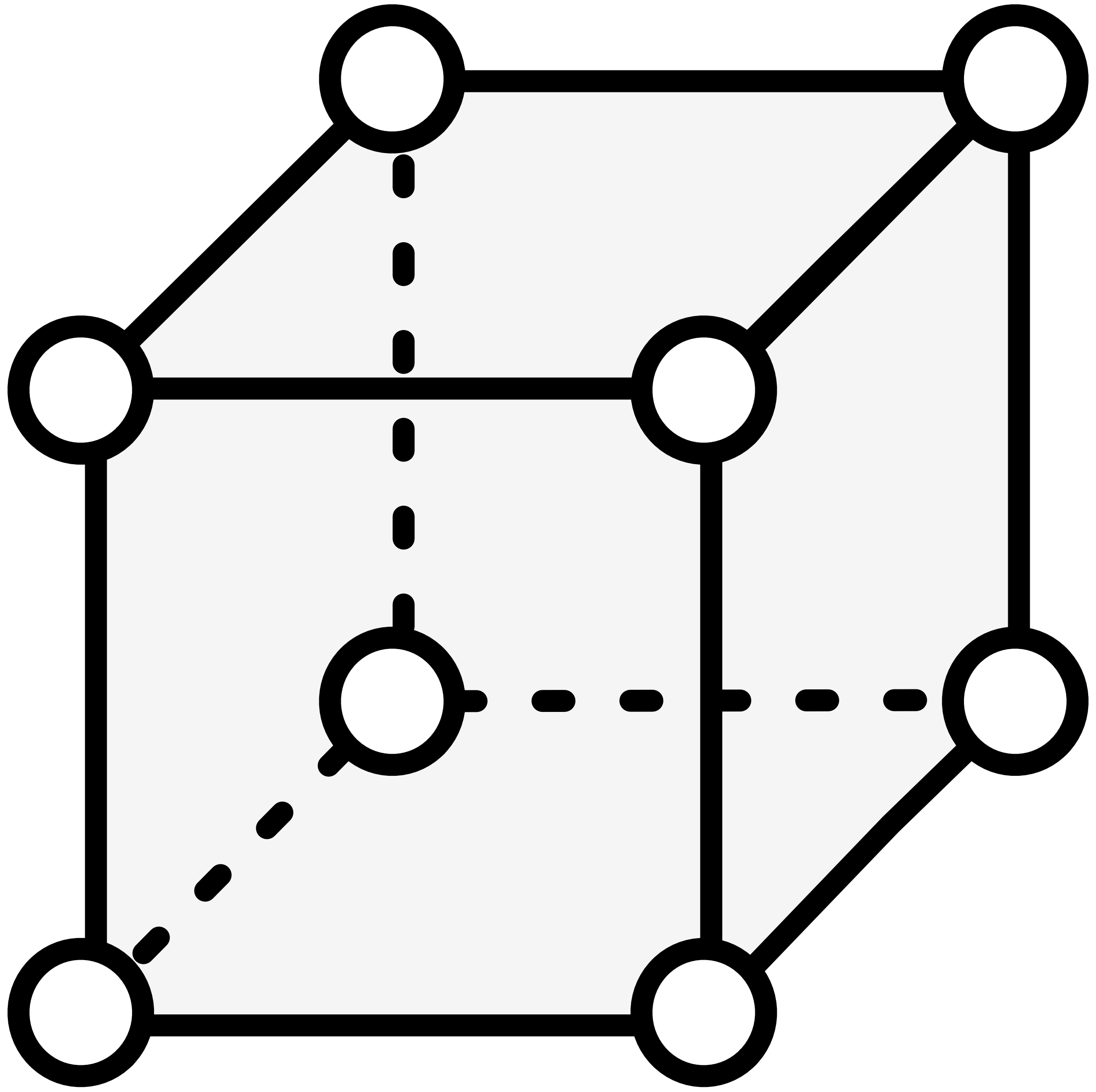
2 tetrahedra

Cube into tetrahedra



1 cube splits into
6 tetrahedra

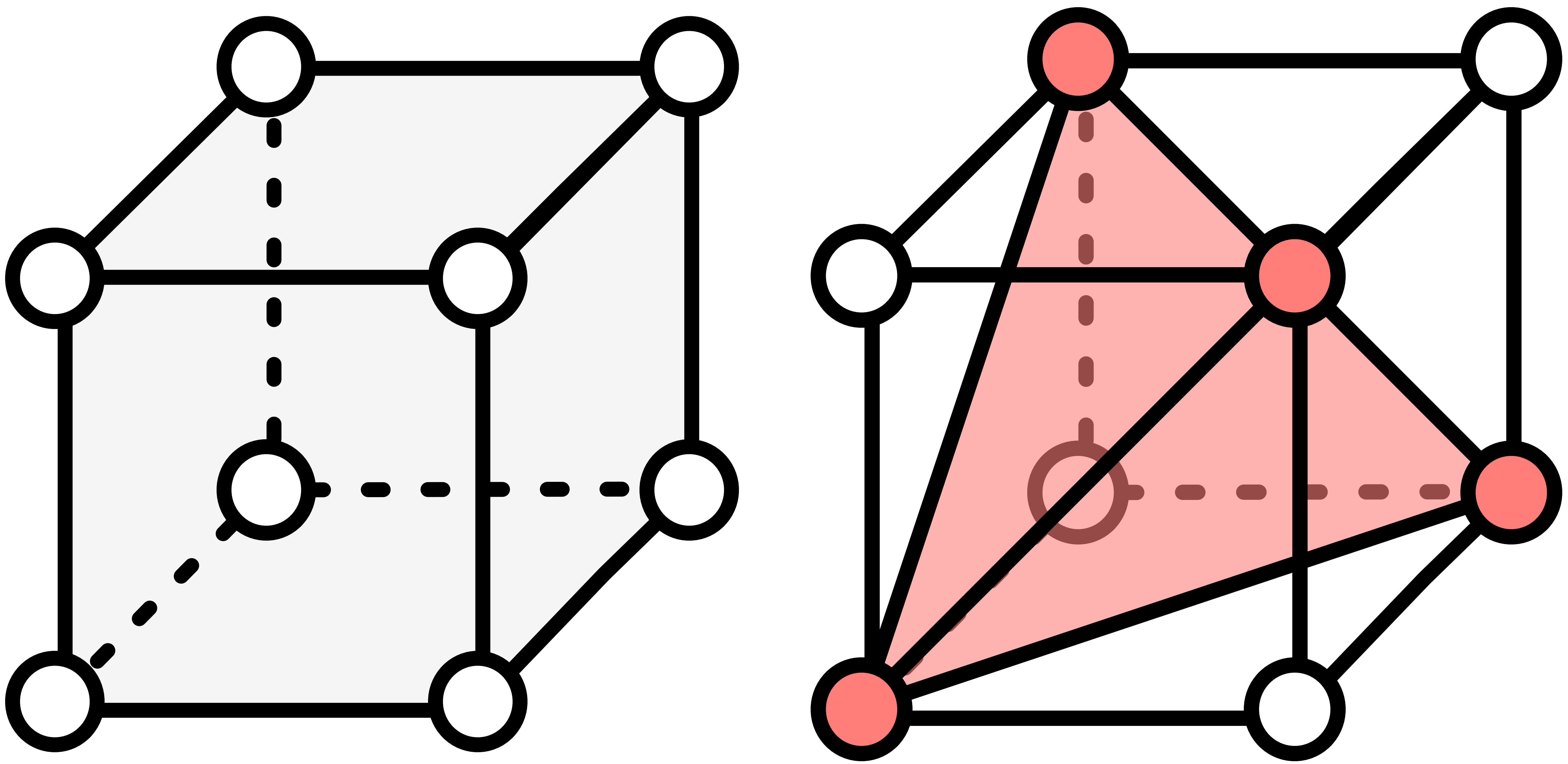
Cube into tetrahedra



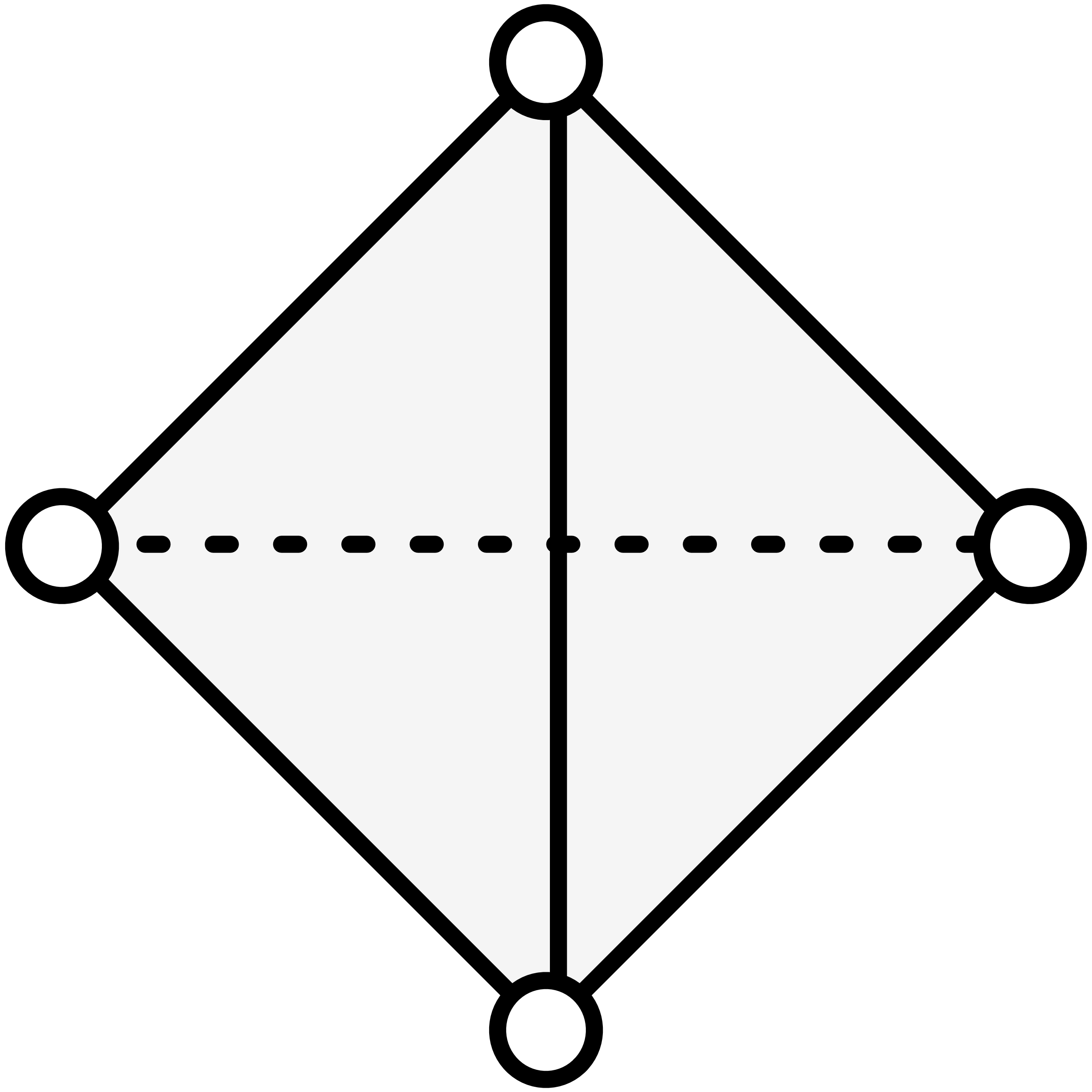
1 cube splits into
6 tetrahedra…

but also into 5 tetrahedra!

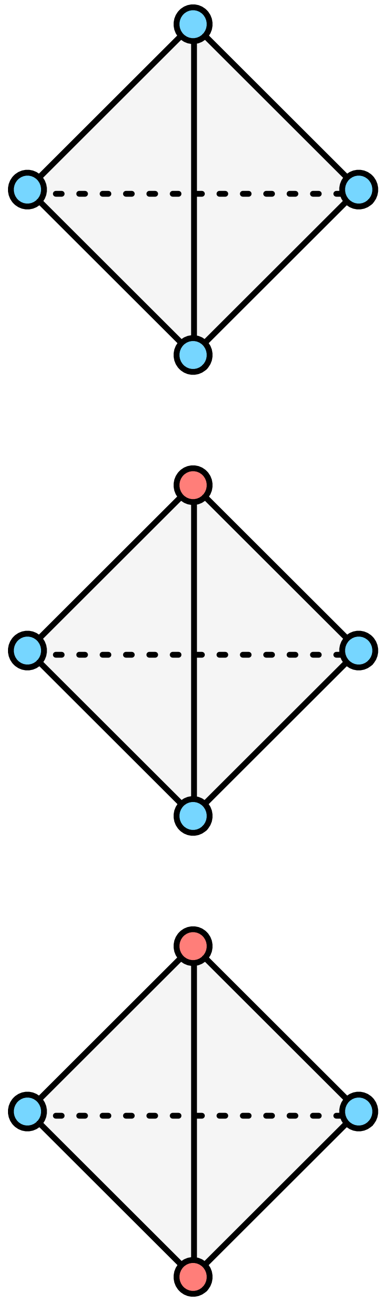
Cube into tetrahedra



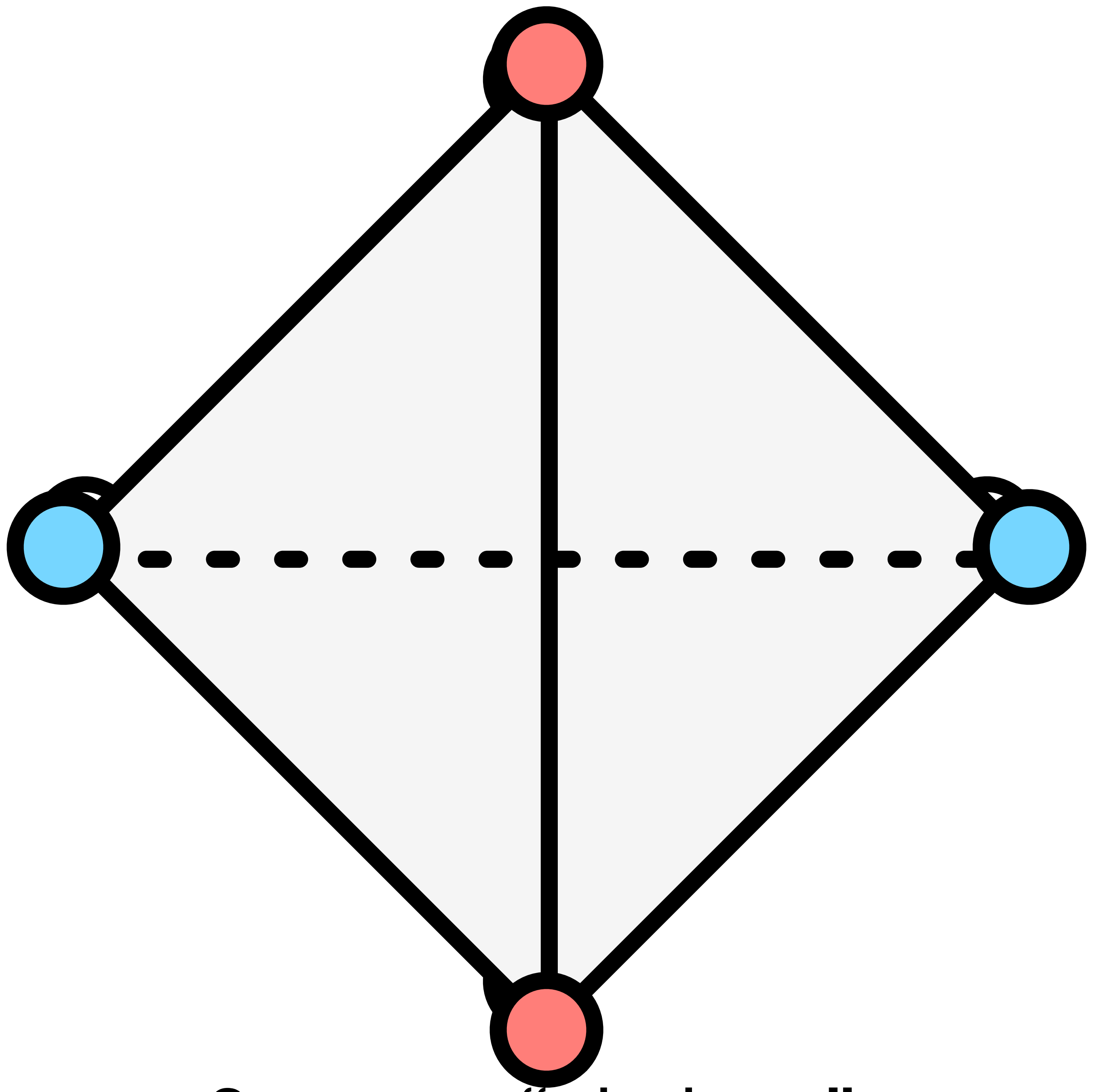
Marching Tetrahedra

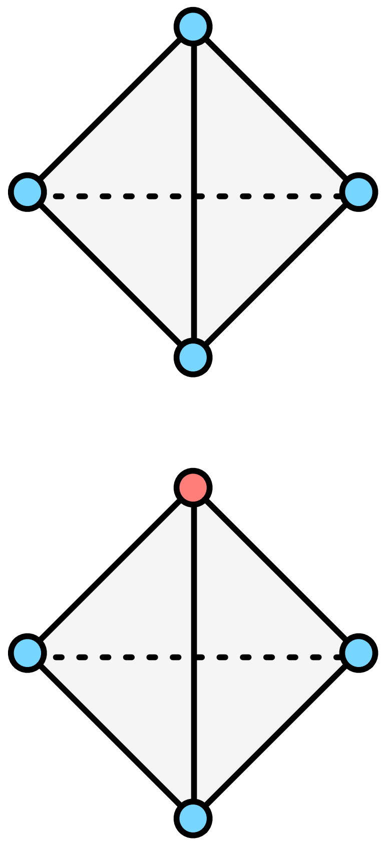


3 cases, “obvious”



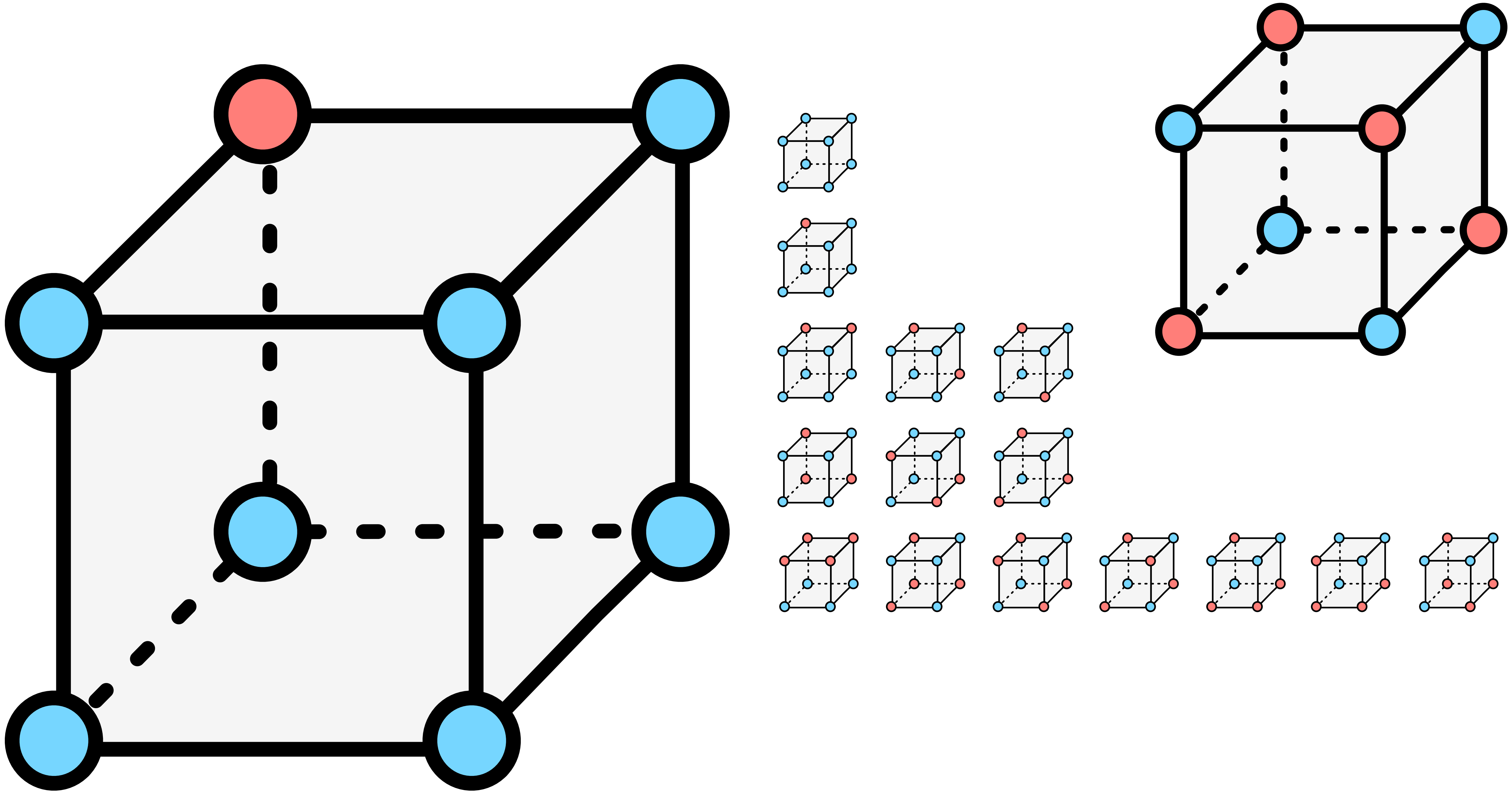
Marching Tetrahedra



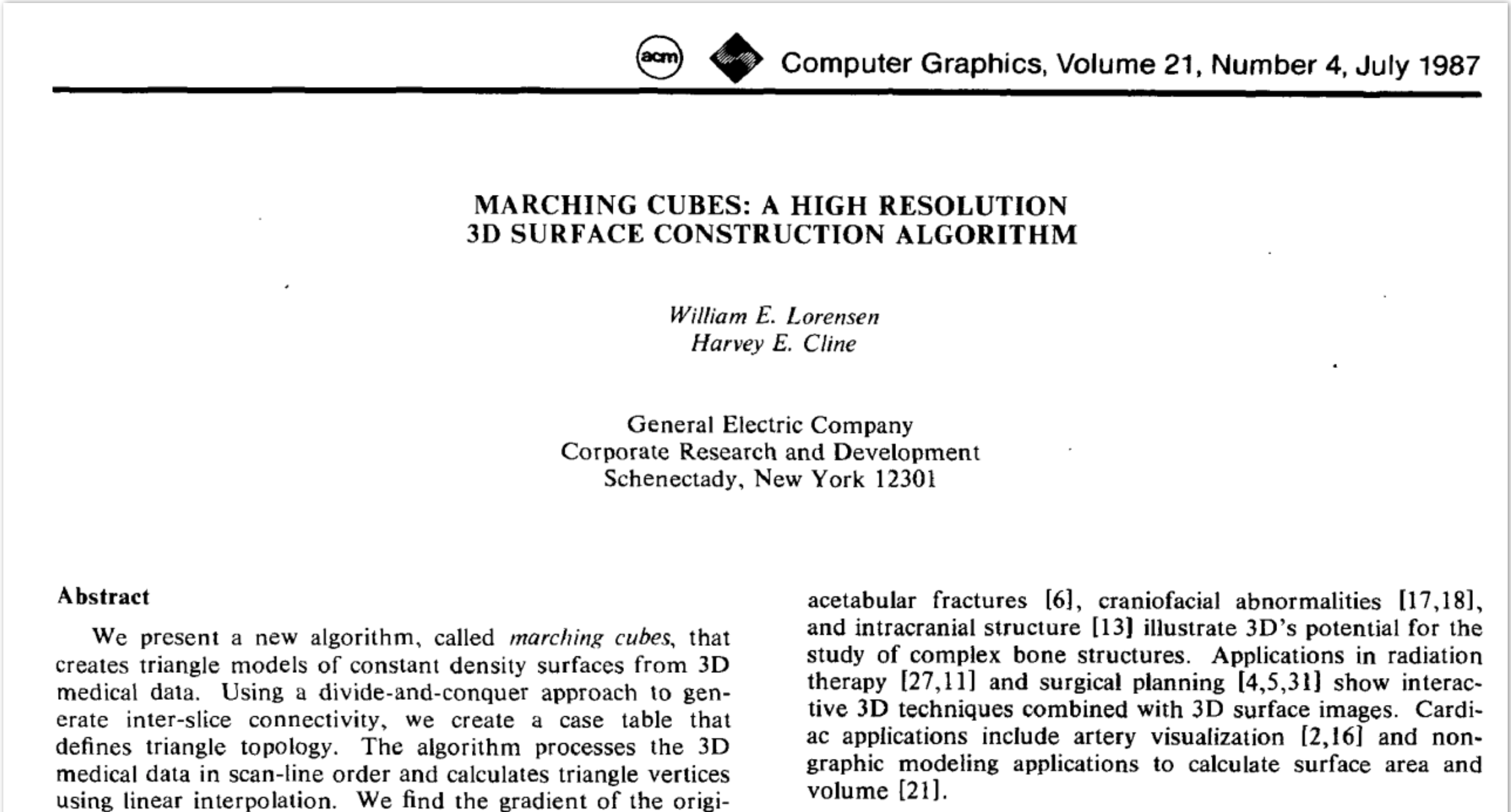


3 cases, “obvious”

3D Contouring



3D Contouring



Computer Graphics, Volume 21, Number 4, July 1987

MARCHING CUBES: A HIGH RESOLUTION 3D SURFACE CONSTRUCTION ALGORITHM

William E. Lorensen
Harvey E. Cline

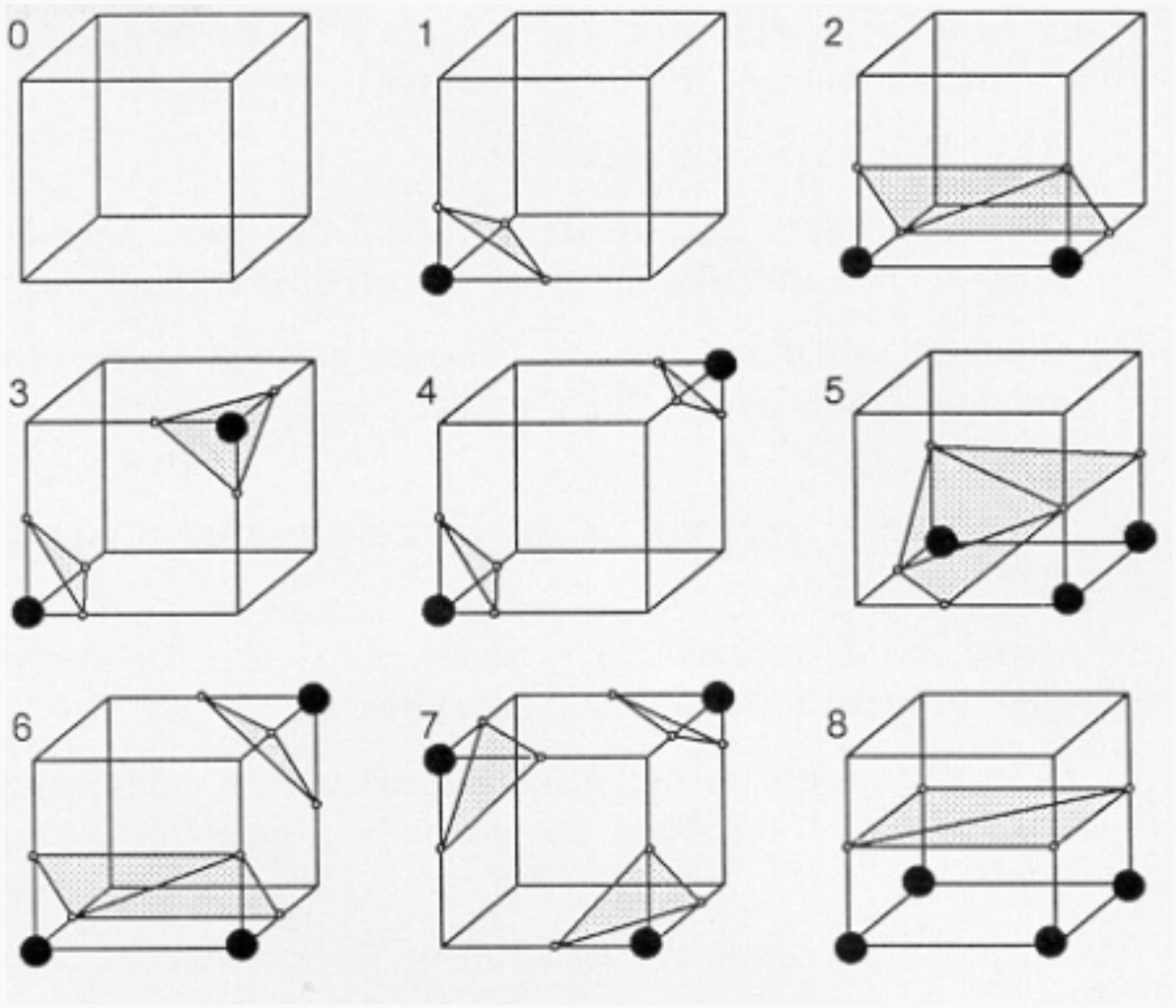
General Electric Company
Corporate Research and Development
Schenectady, New York 12301

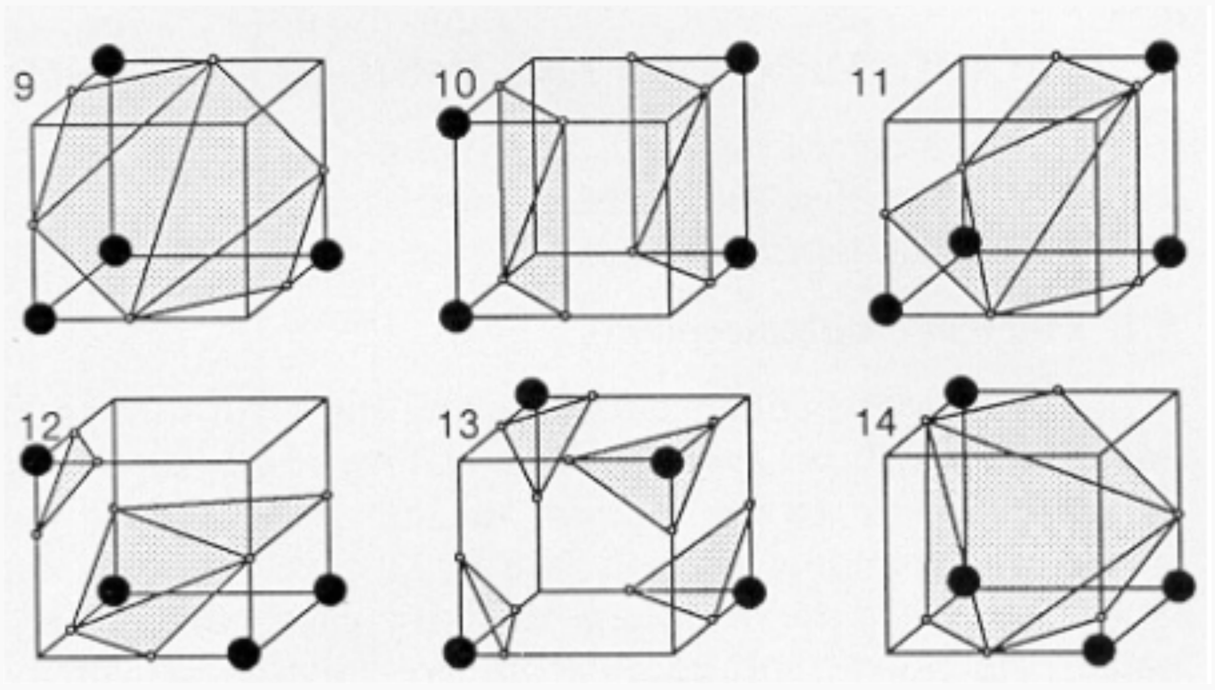
Abstract

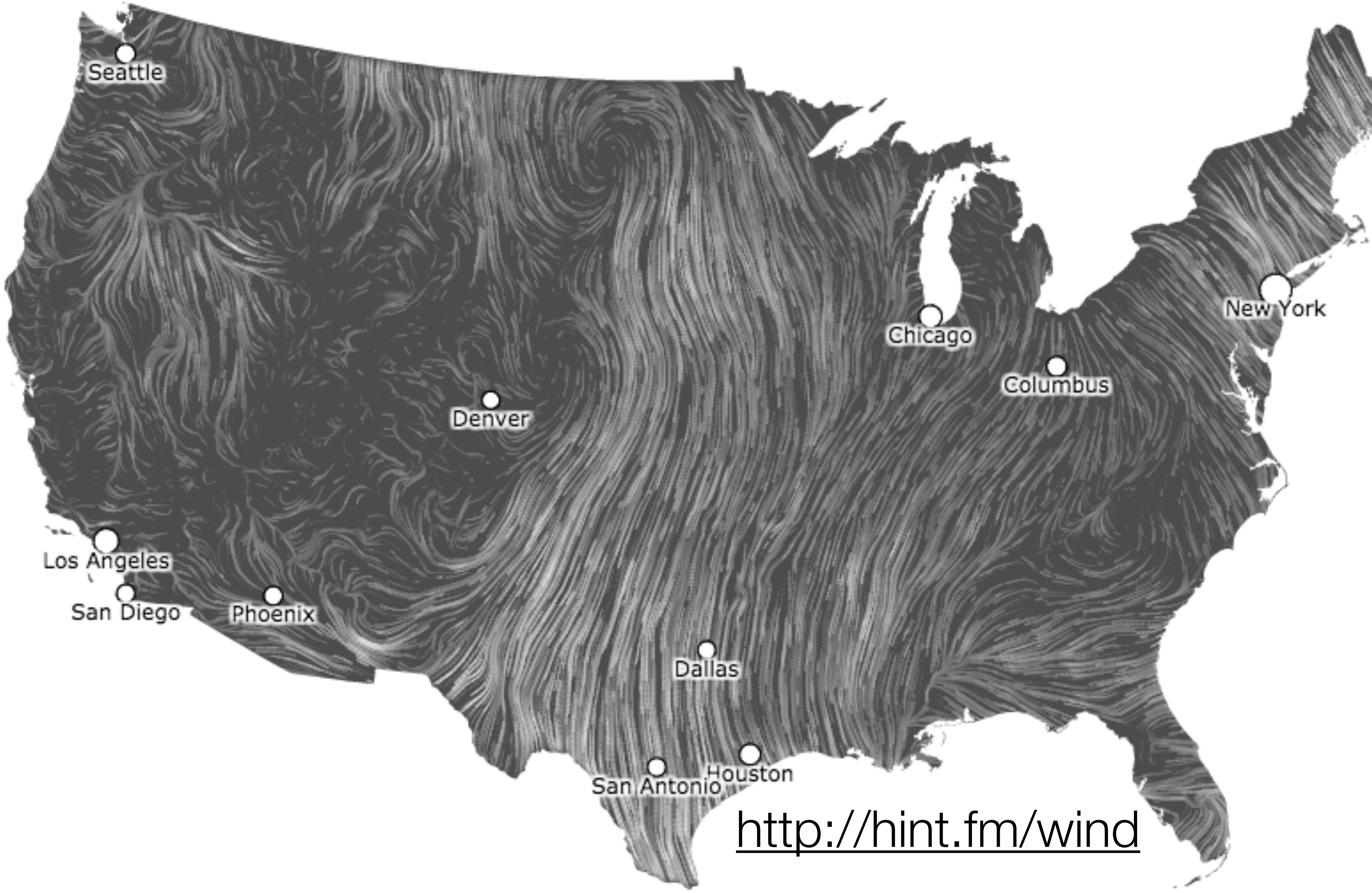
We present a new algorithm, called *marching cubes*, that creates triangle models of constant density surfaces from 3D medical data. Using a divide-and-conquer approach to generate inter-slice connectivity, we create a case table that defines triangle topology. The algorithm processes the 3D medical data in scan-line order and calculates triangle vertices using linear interpolation. We find the gradient of the origi-

acetabular fractures [6], craniofacial abnormalities [17,18], and intracranial structure [13] illustrate 3D's potential for the study of complex bone structures. Applications in radiation therapy [27,11] and surgical planning [4,5,31] show interactive 3D techniques combined with 3D surface images. Cardiac applications include artery visualization [2,16] and nongraphic modeling applications to calculate surface area and volume [21].

3D Contouring



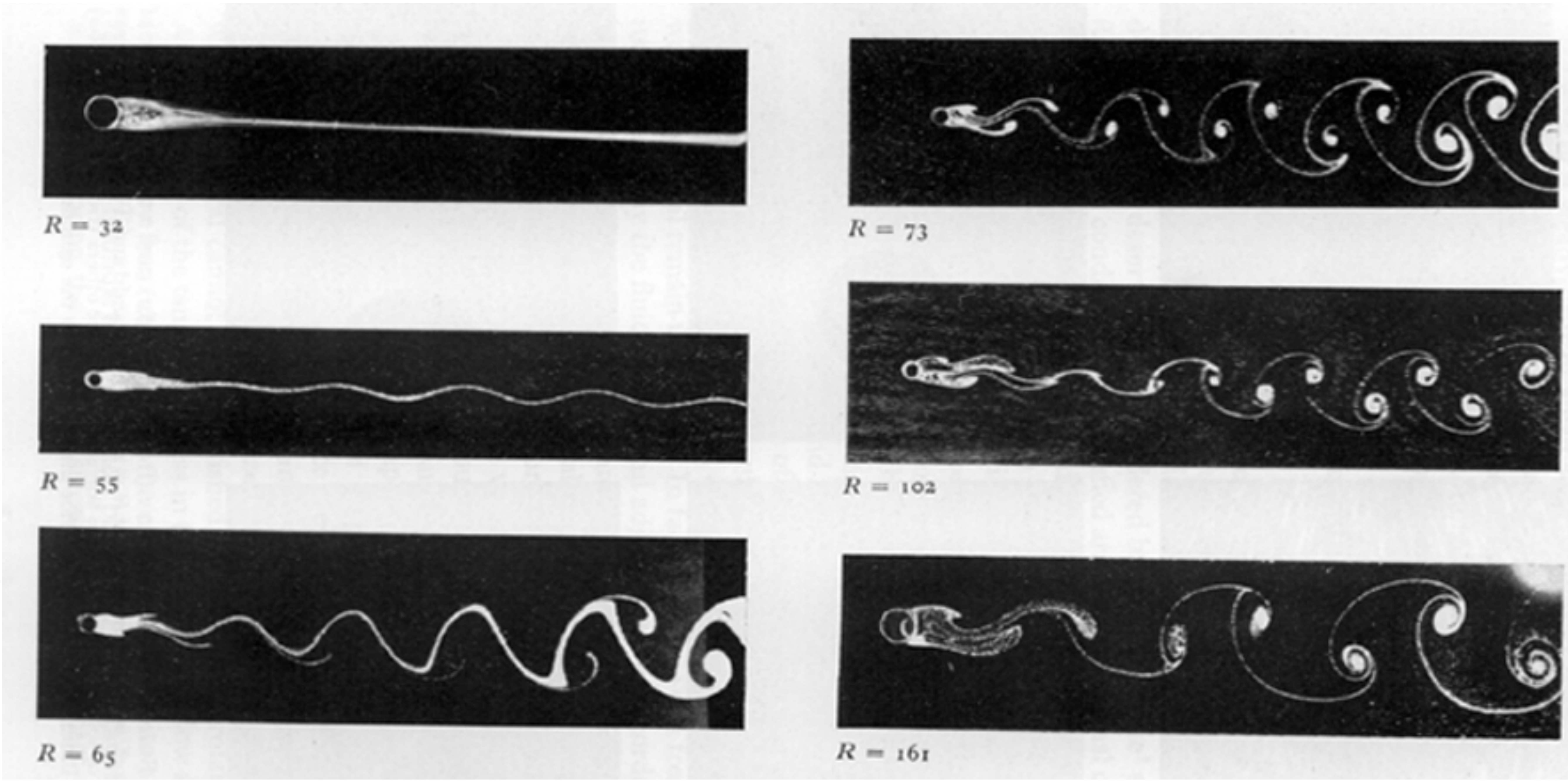




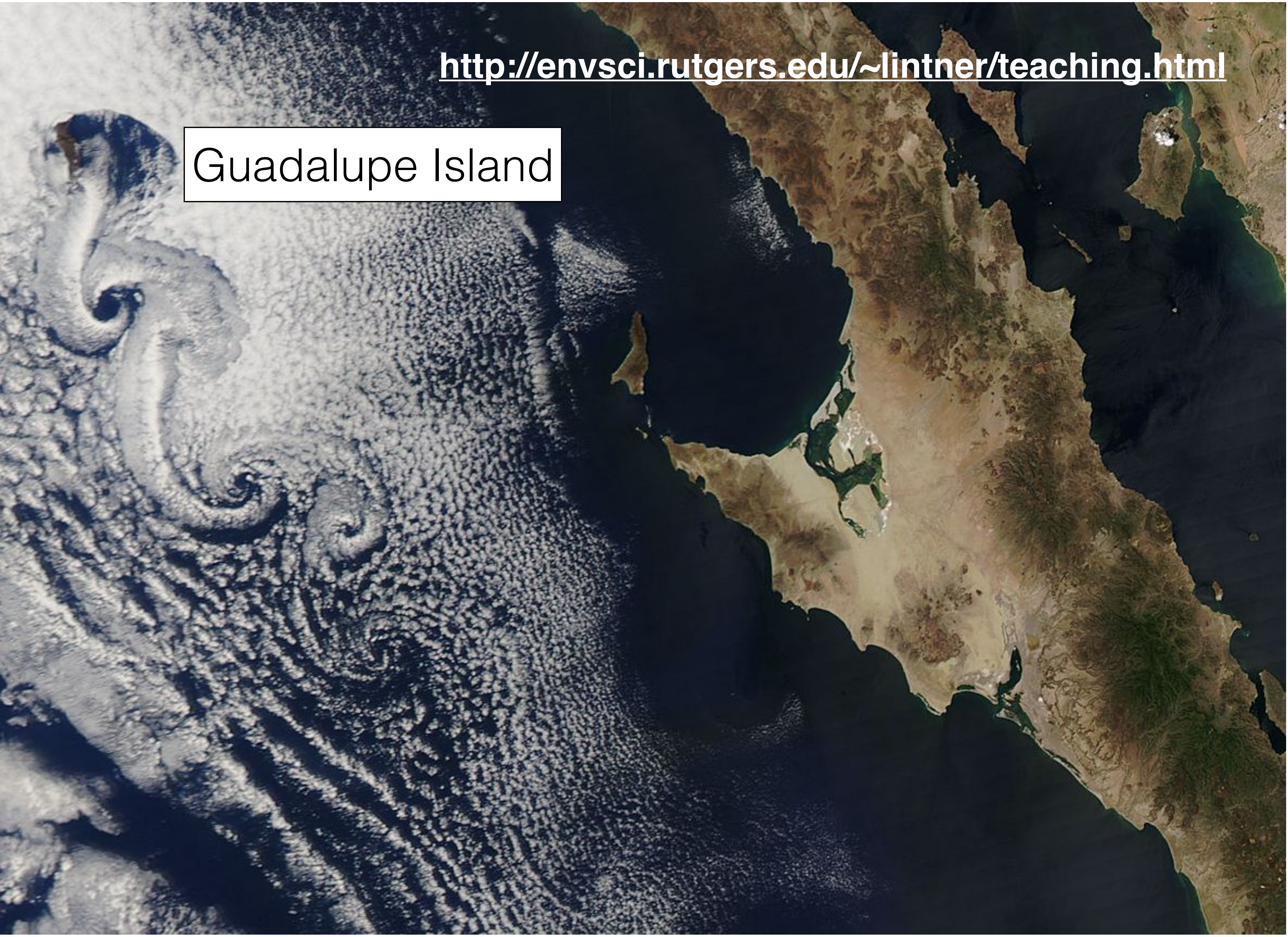
http://hint.fm/wind

Spatial Data: Vector Fields

Experimental Flow Vis



von Kármán vortex street, depending on Reynolds number

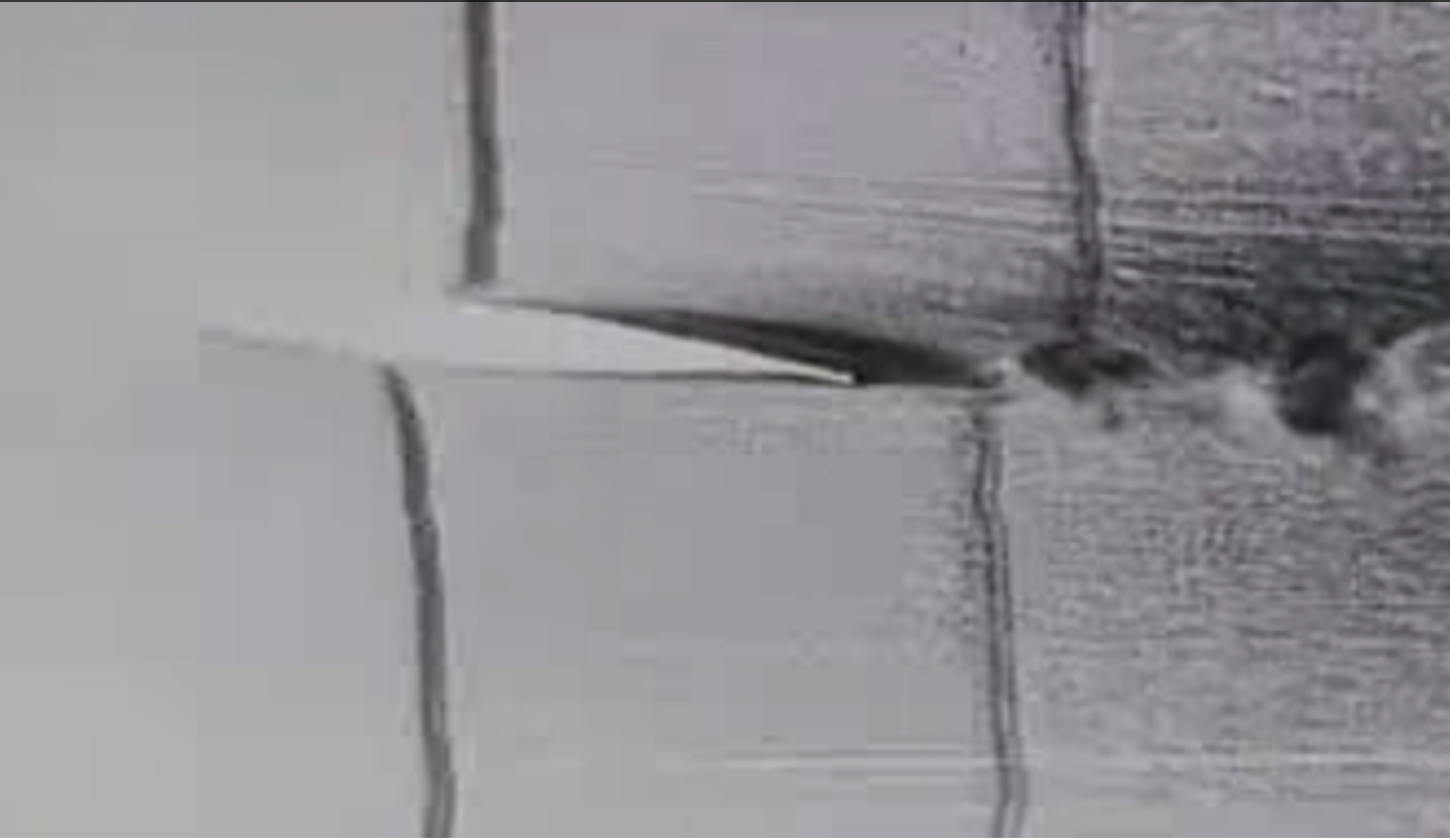

http://envsci.rutgers.edu/~lintner/teaching.html
Guadalupe Island

Mathematics of Vector Fields

$$v : R^n \rightarrow R^n$$

Function from vectors to vectors

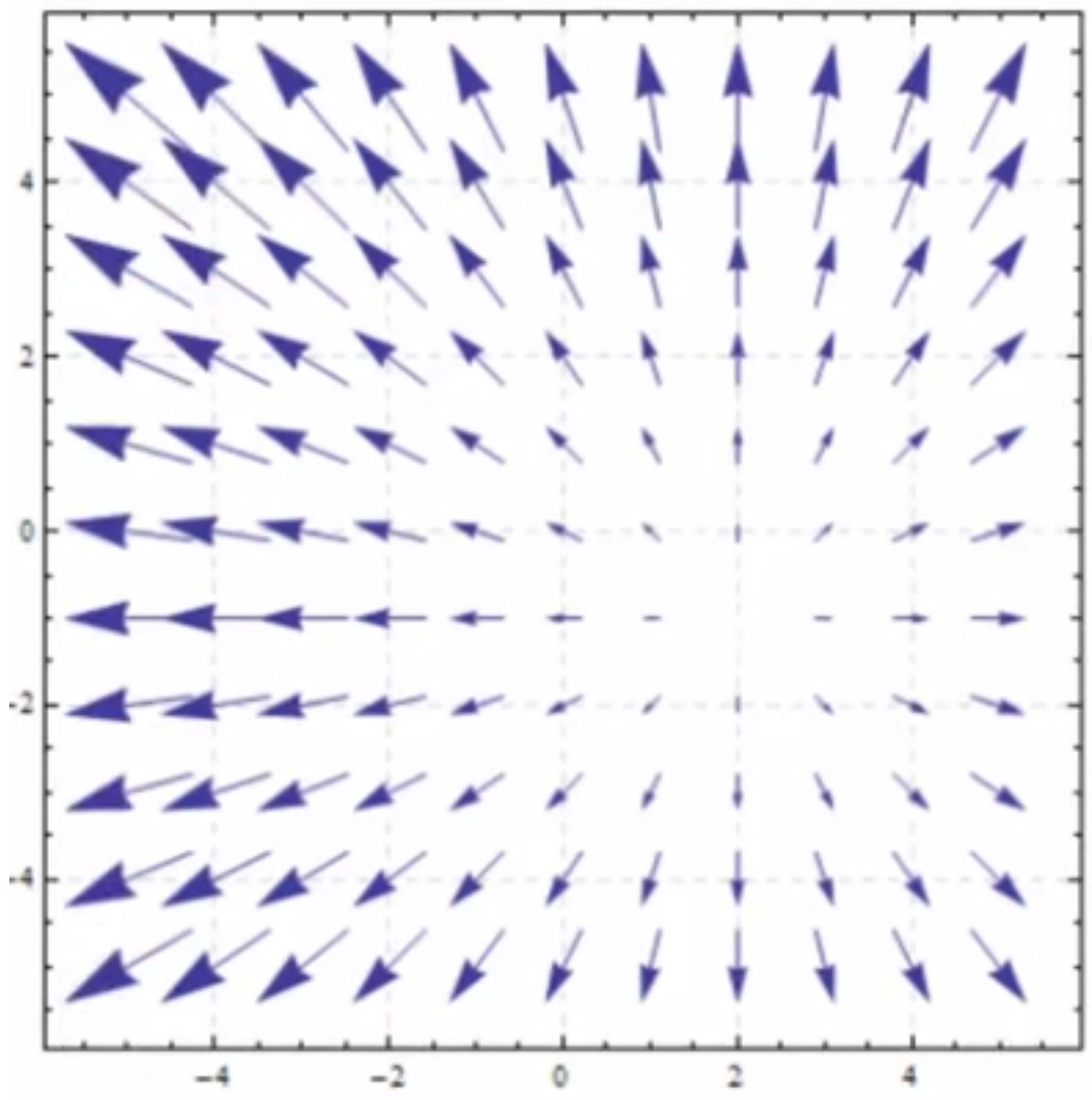
https://www.youtube.com/watch?v=nuQyKGuXJOs

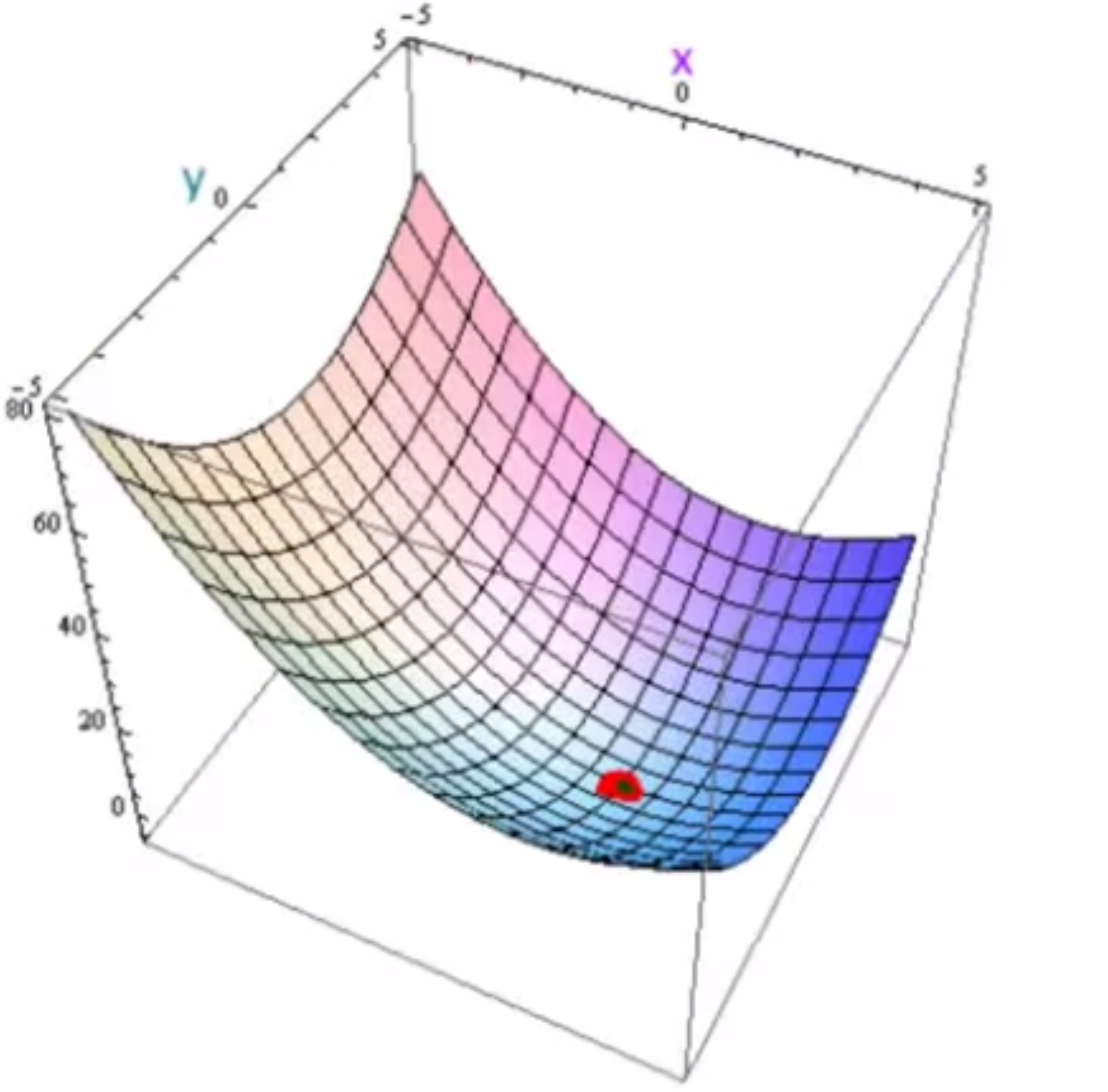


Spatial Data: Vector Fields

A simple vector field: the gradient

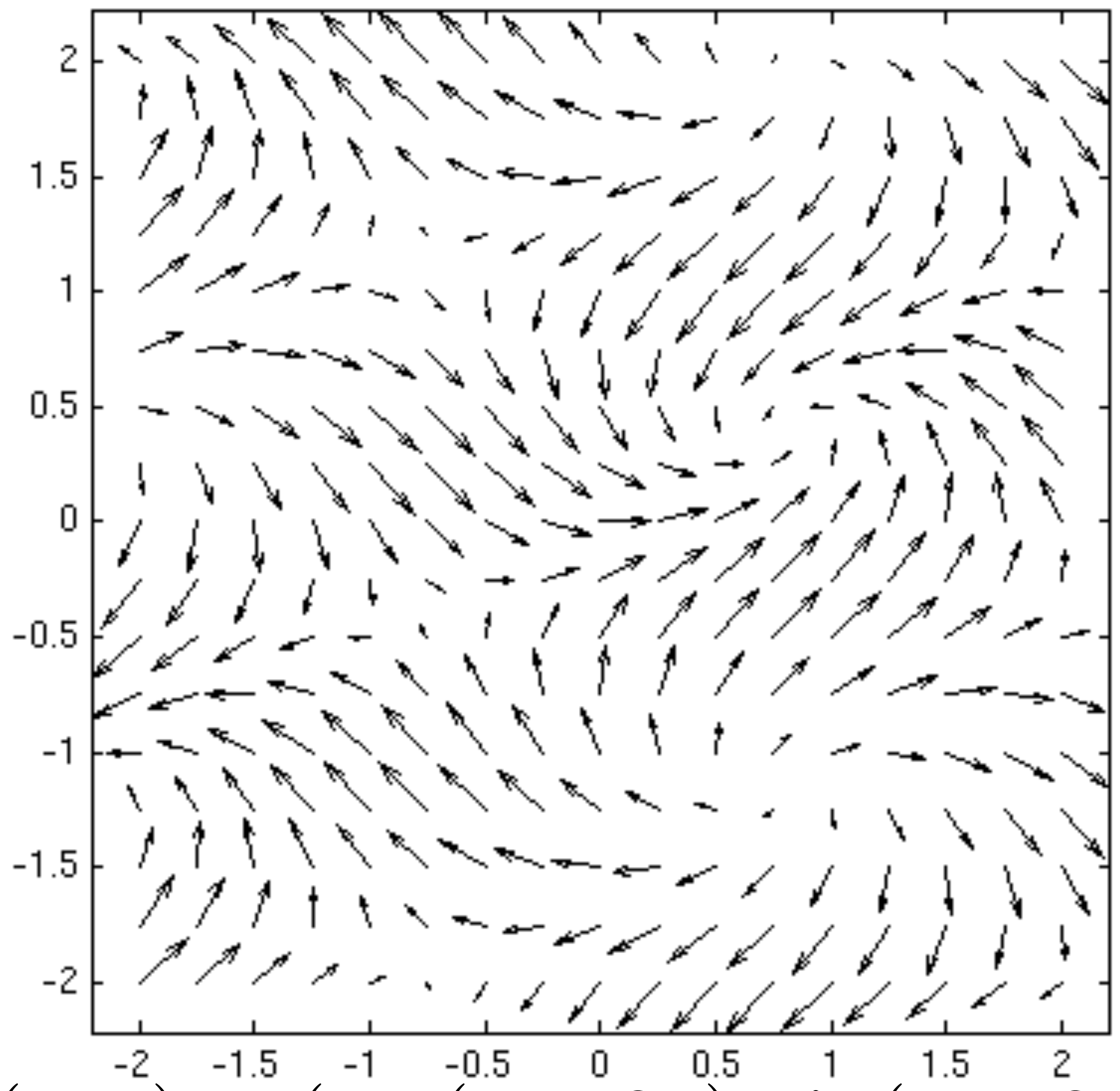
The gradient field $< 2x\text{-}4\ ,\ 2y\text{+}2 >$ of the function $f = x^2 - 4x + y^2 + 2y$.





https://www.youtube.com/watch?v=v0_LlyVquF8

Vector fields can be more complicated

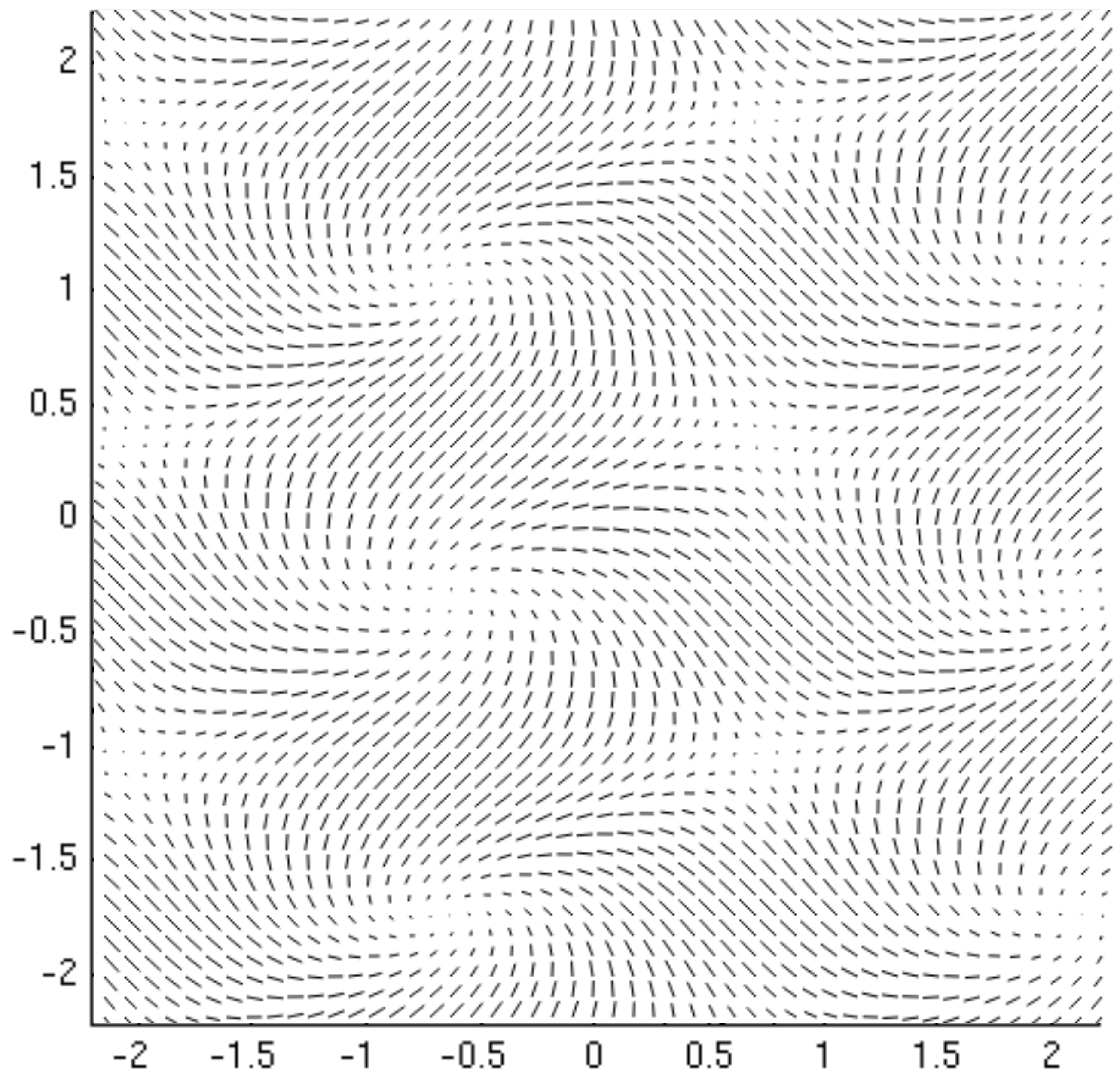


$$v(x, y) = (\cos(x + 2y), \sin(x - 2y))$$

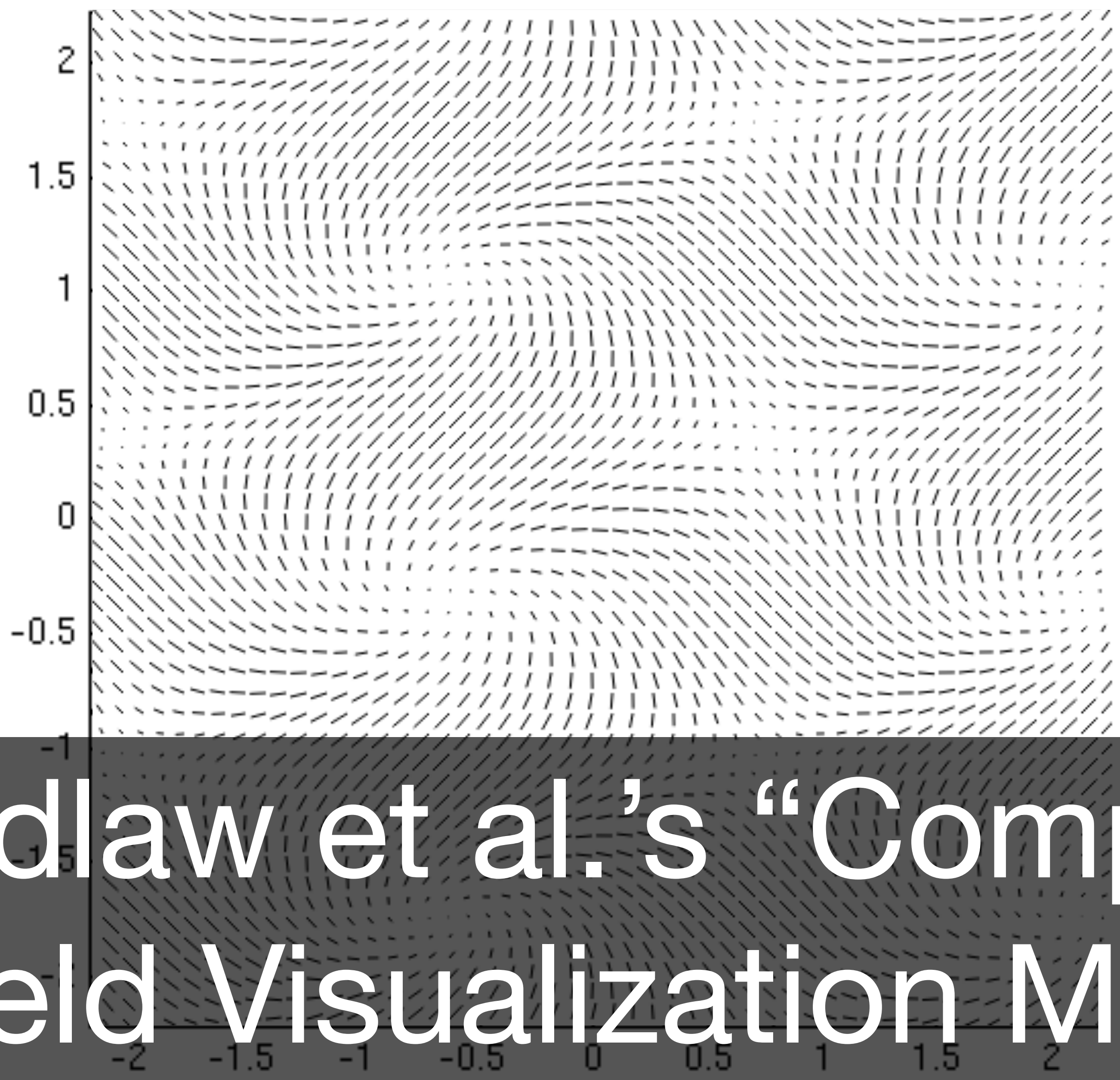
http://www.math.umd.edu/~petersd/241/html/ex27b.html

Glyph Based Techniques

Hedgehog Plot: Not Very Good

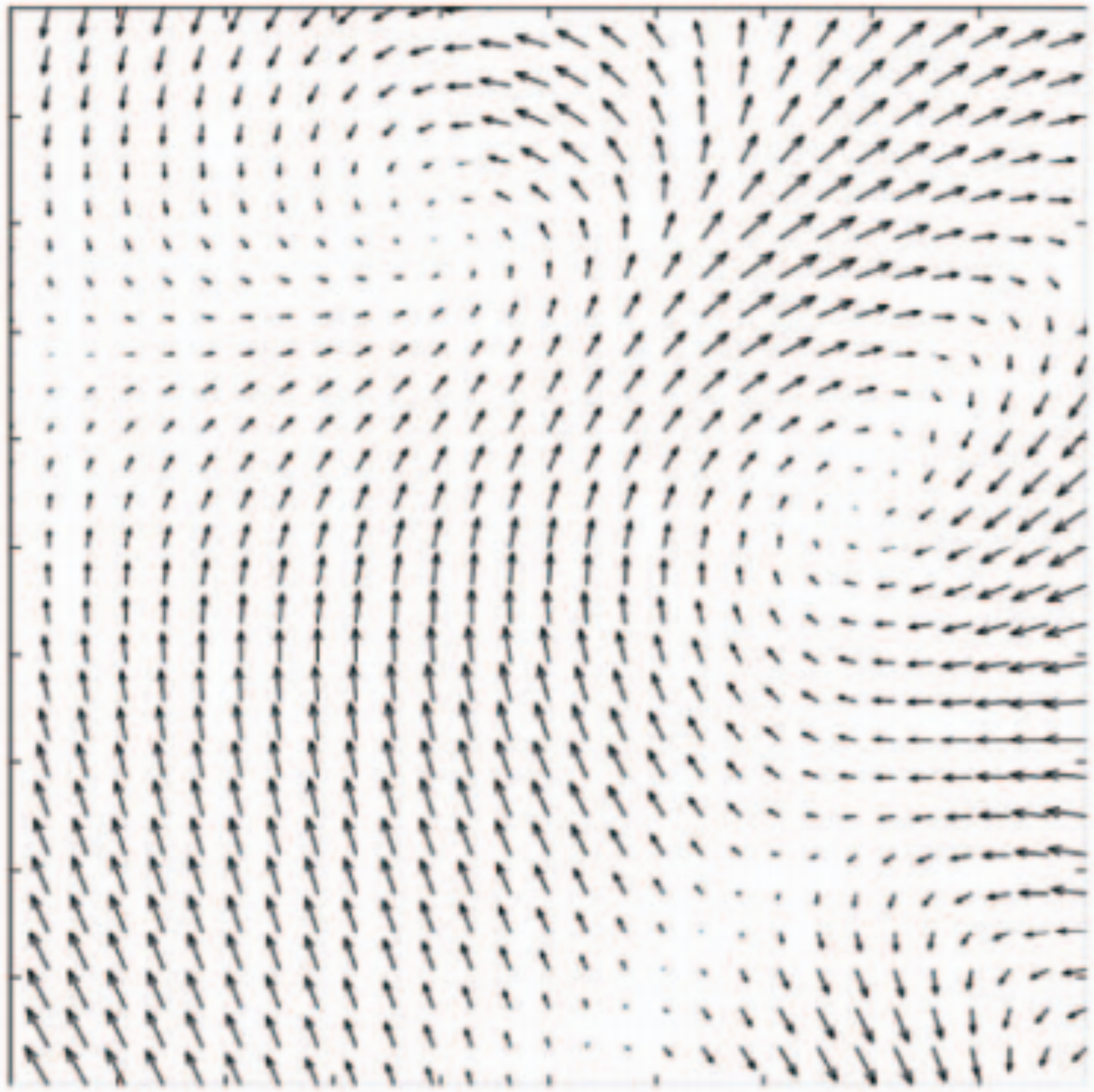


Hedgehog Plot: Not Very Good



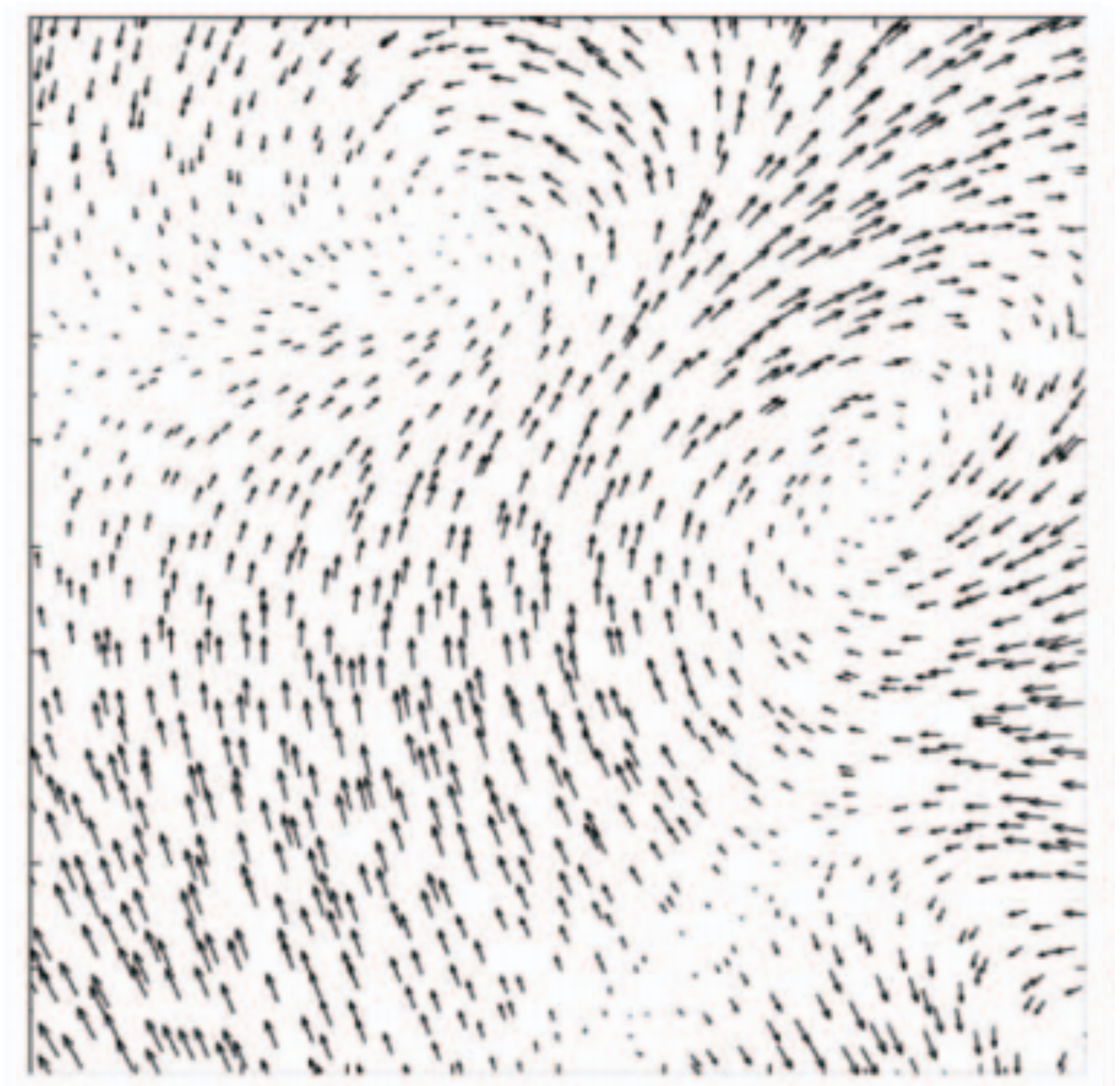
From Laidlaw et al.'s "Comparing 2D Vector Field Visualization Methods: A User Study", TVCG 2005

Uniformly-placed arrows: Not Very Good Either



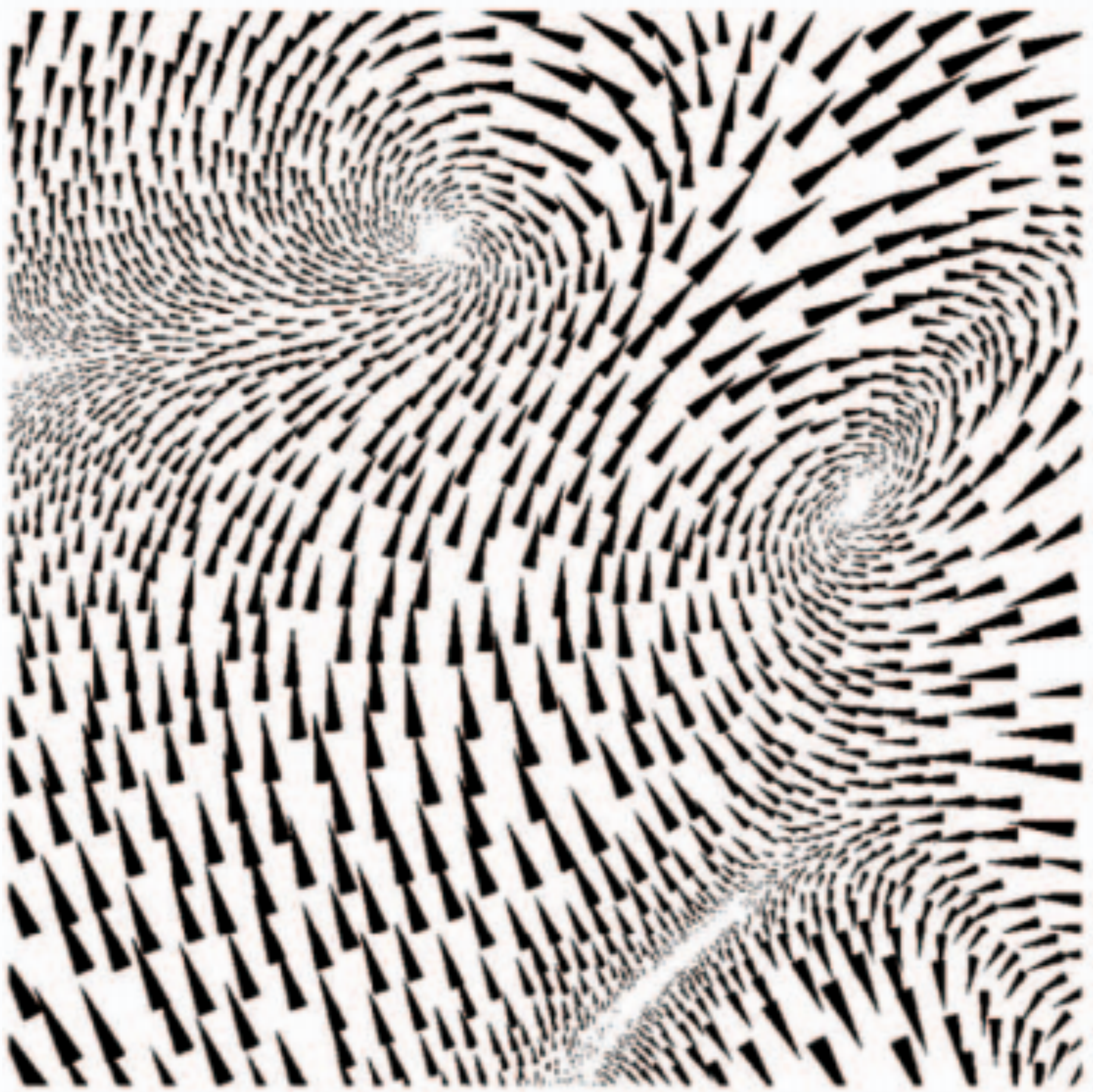
GRID

Jittered Hedgehog Plot: Better

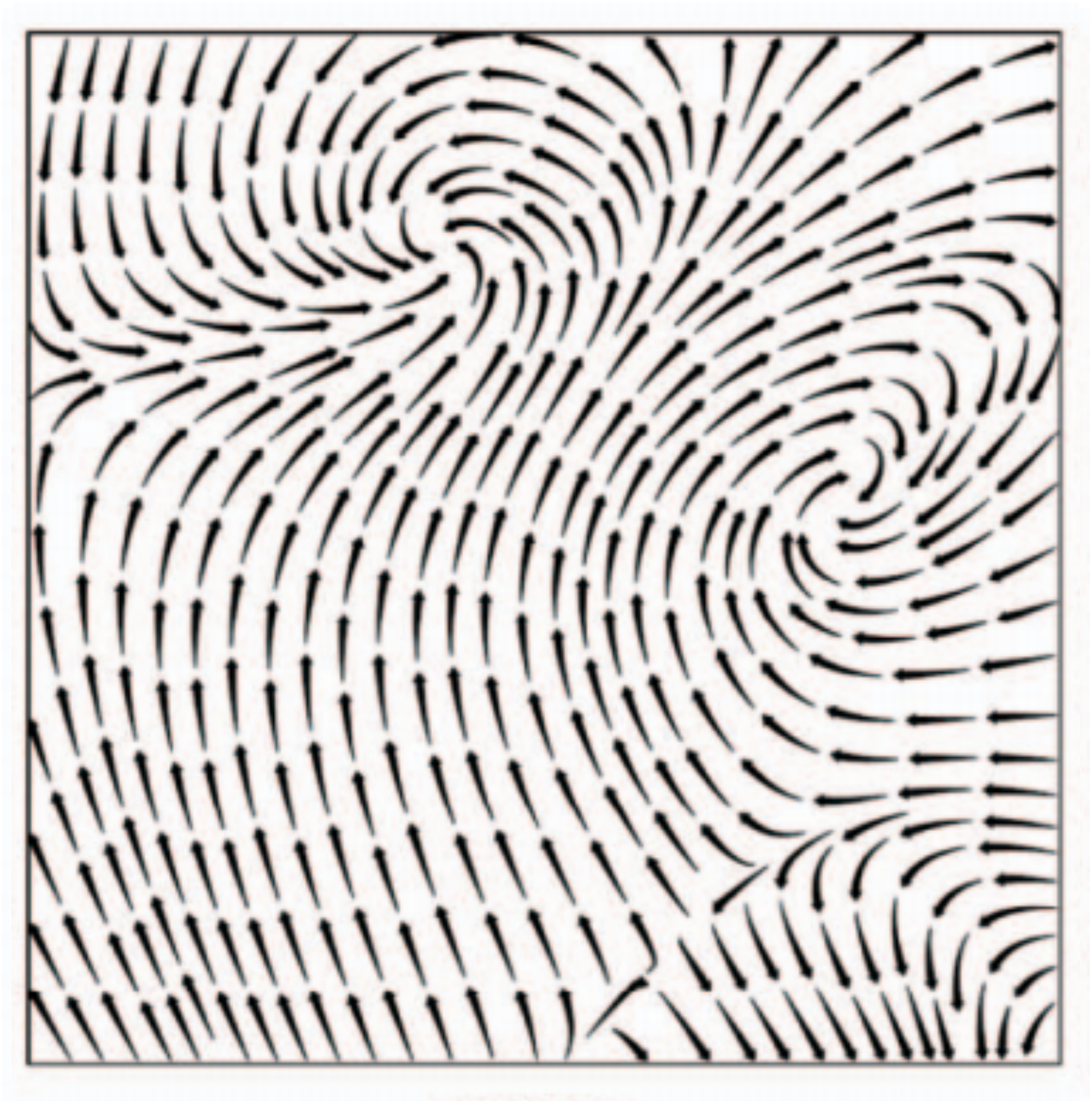


JIT

Space-filling scaled glyphs

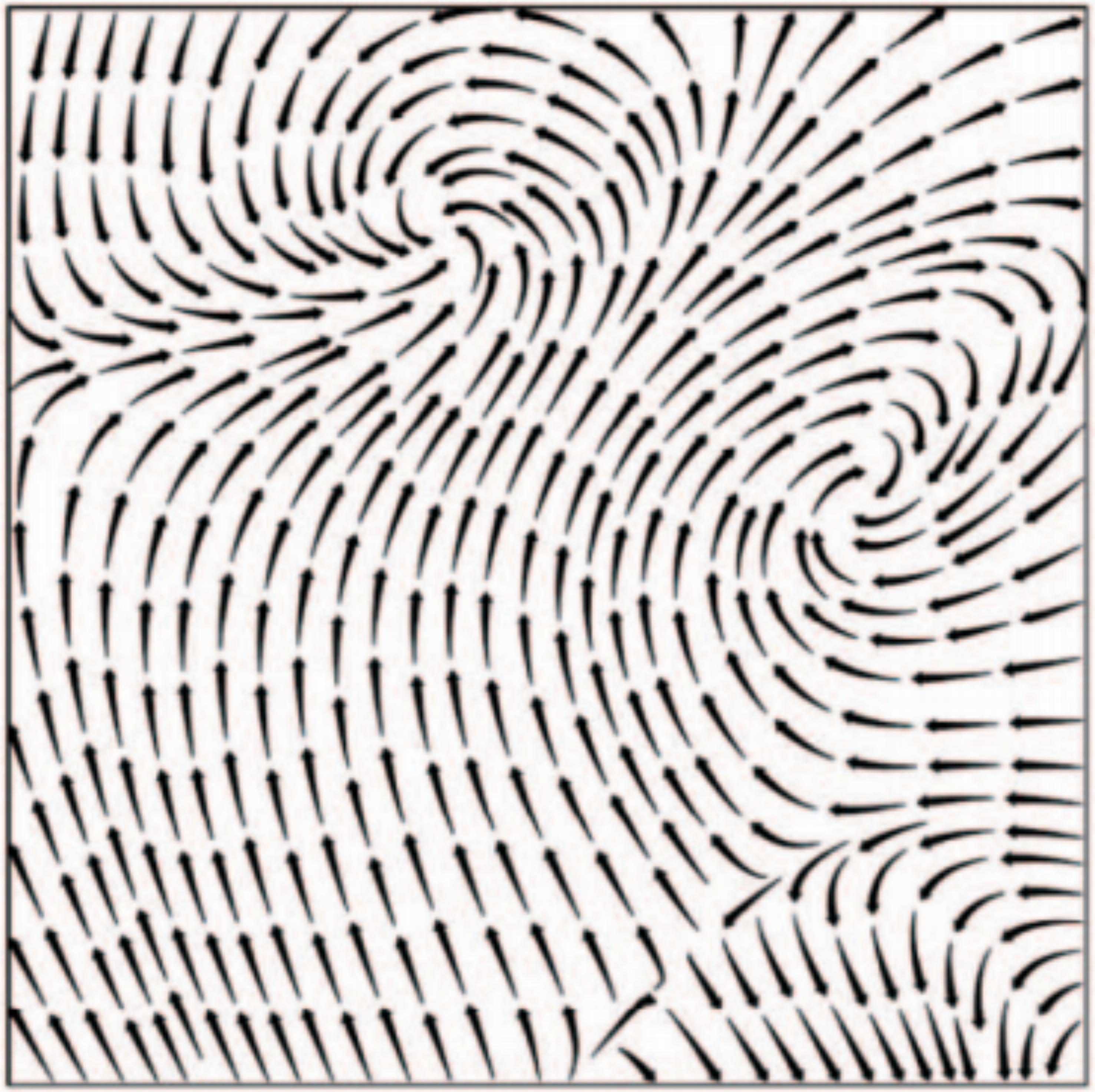


Streamline-Guided Placement



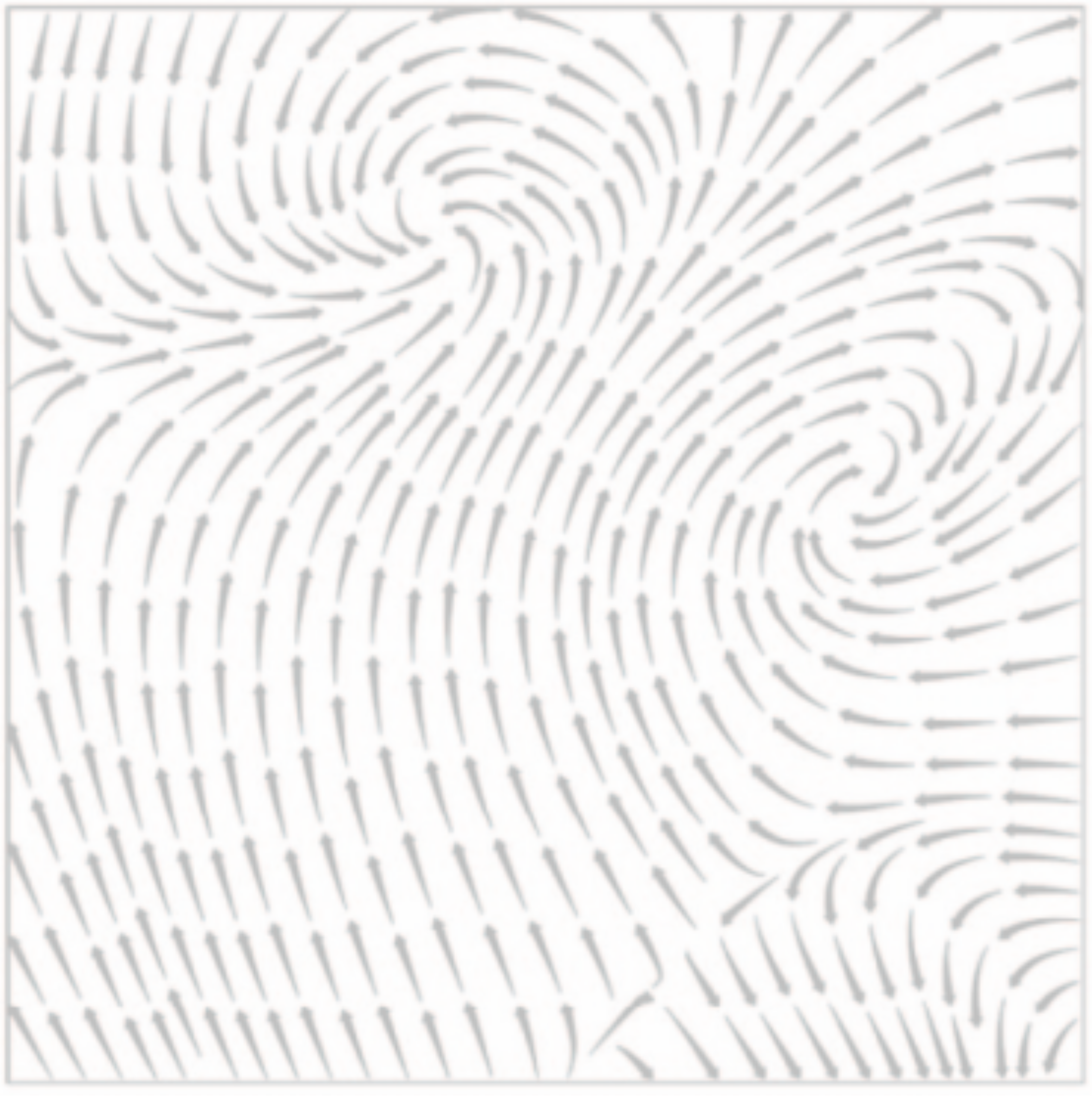
OSTR

Streamline-Guided Placement

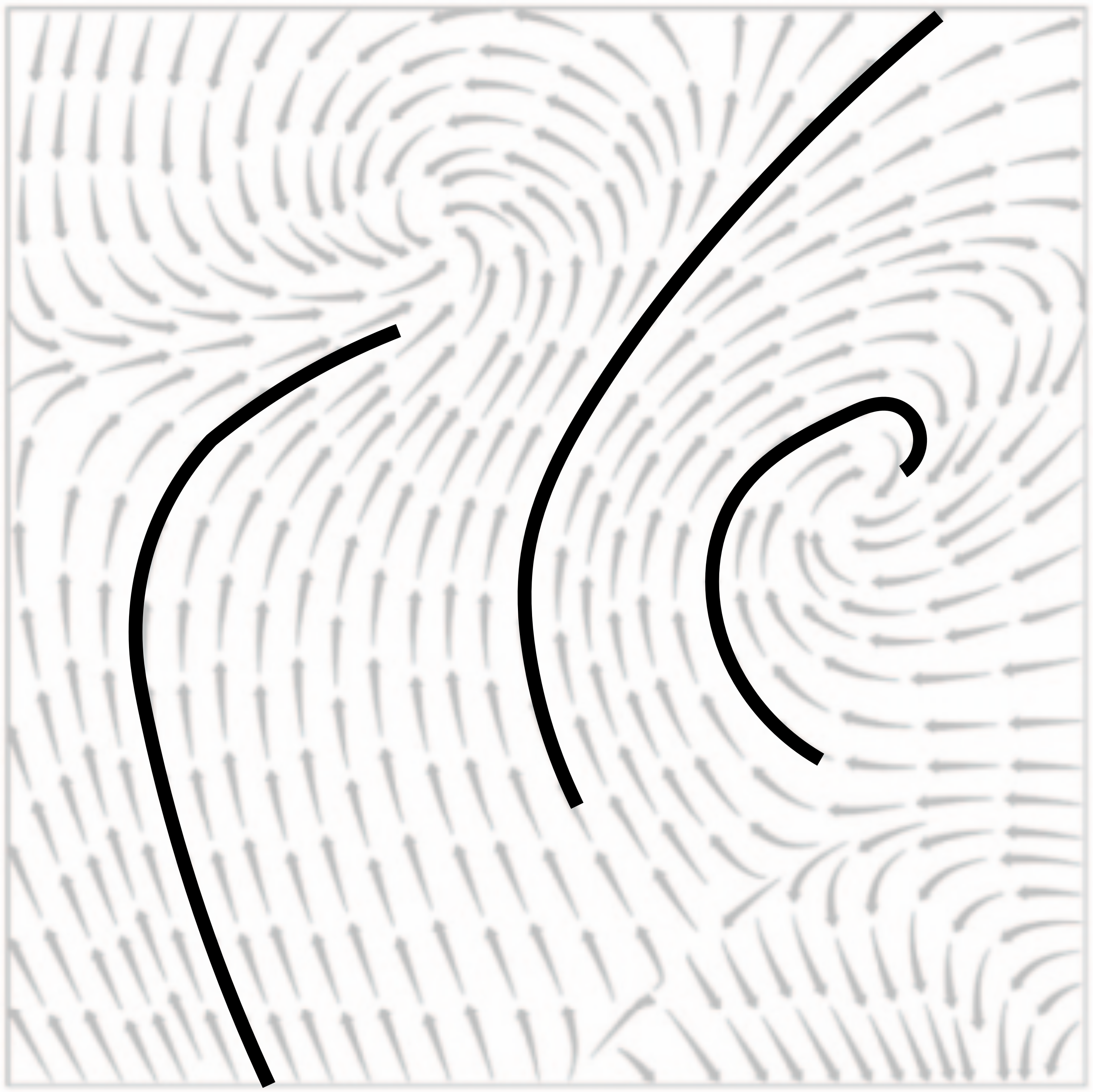

OSTR

Streamlines

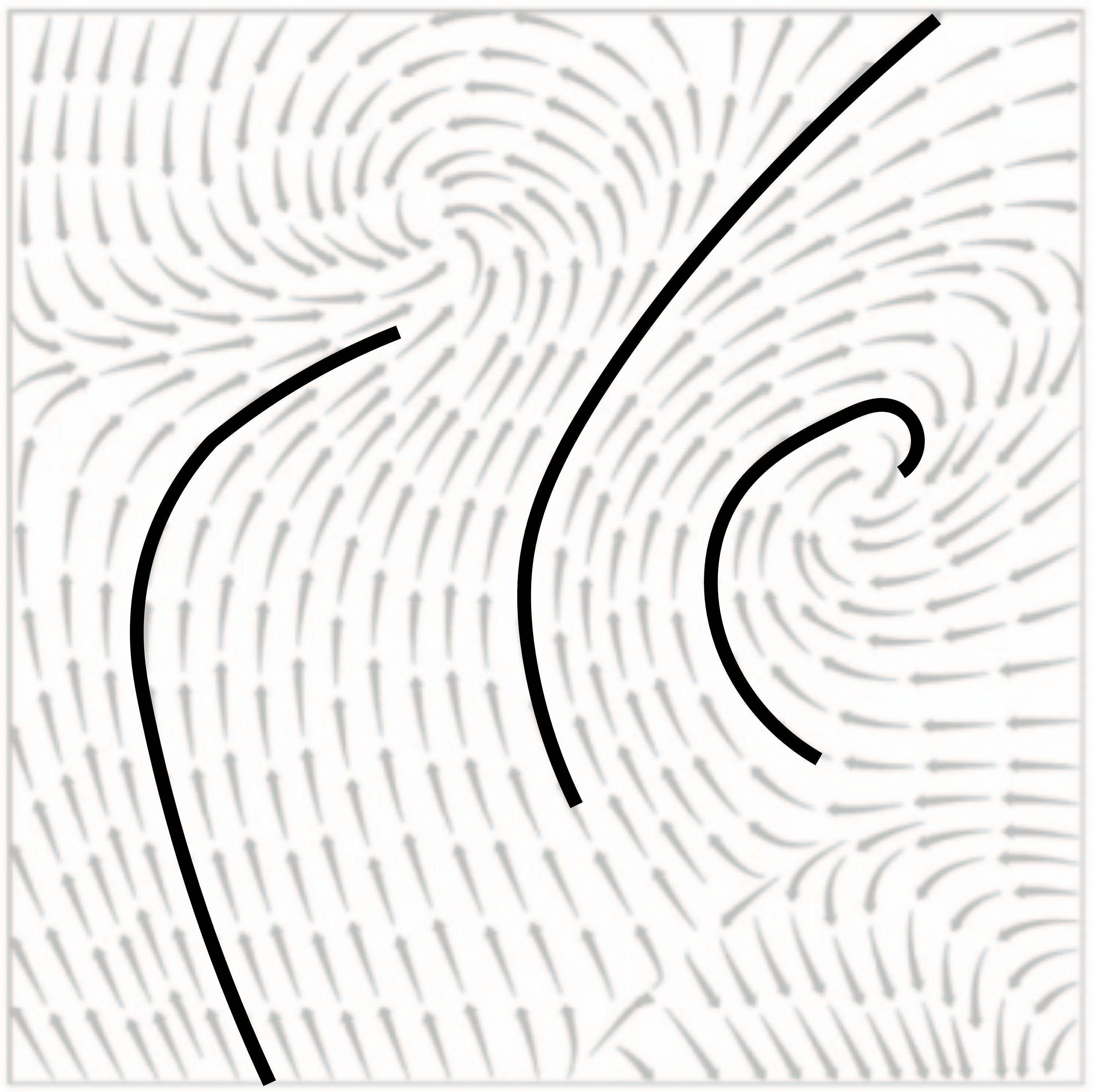
Streamlines



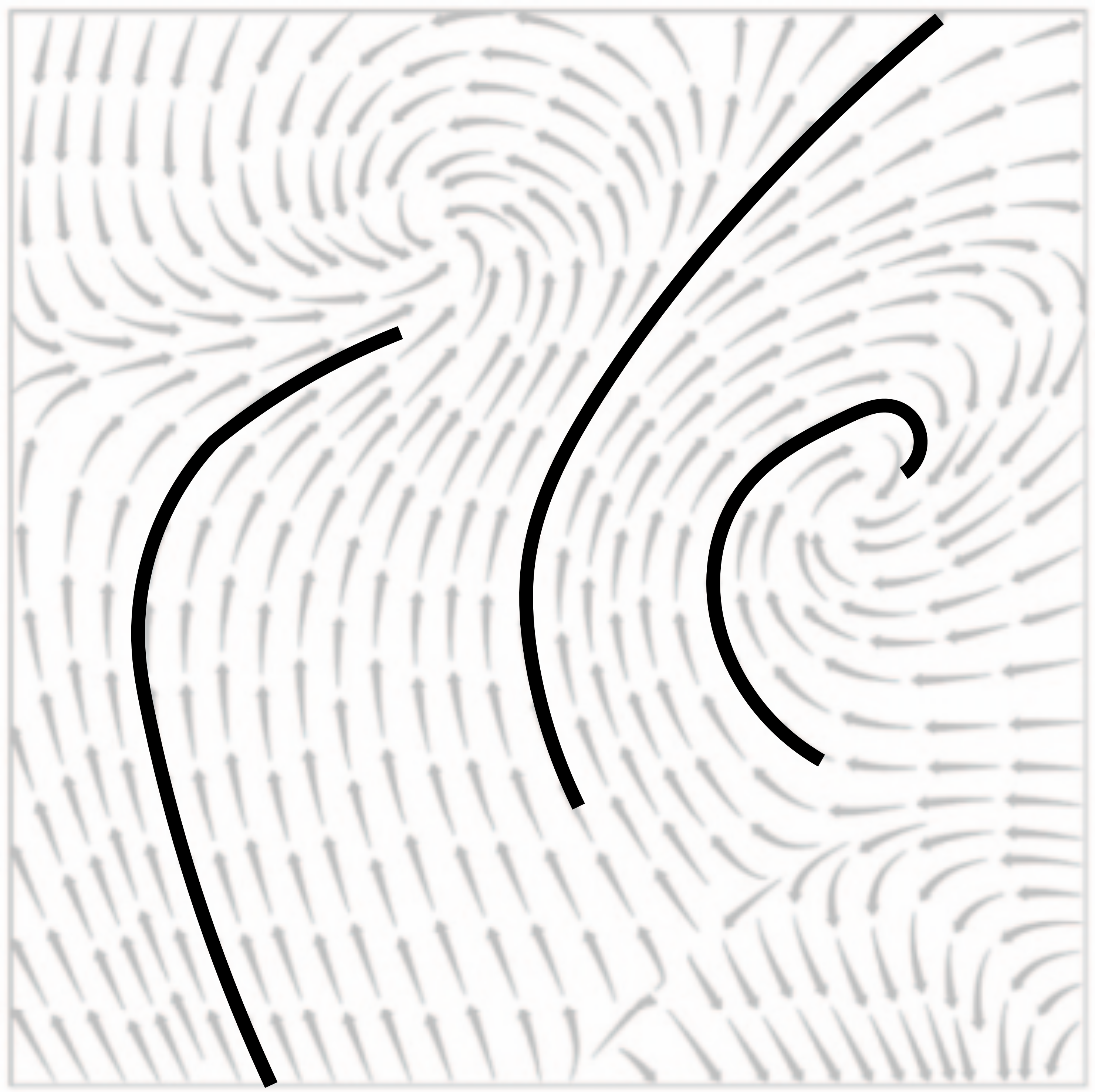
Streamlines



Curves everywhere tangent to the vector field



Curves everywhere tangent to the vector field

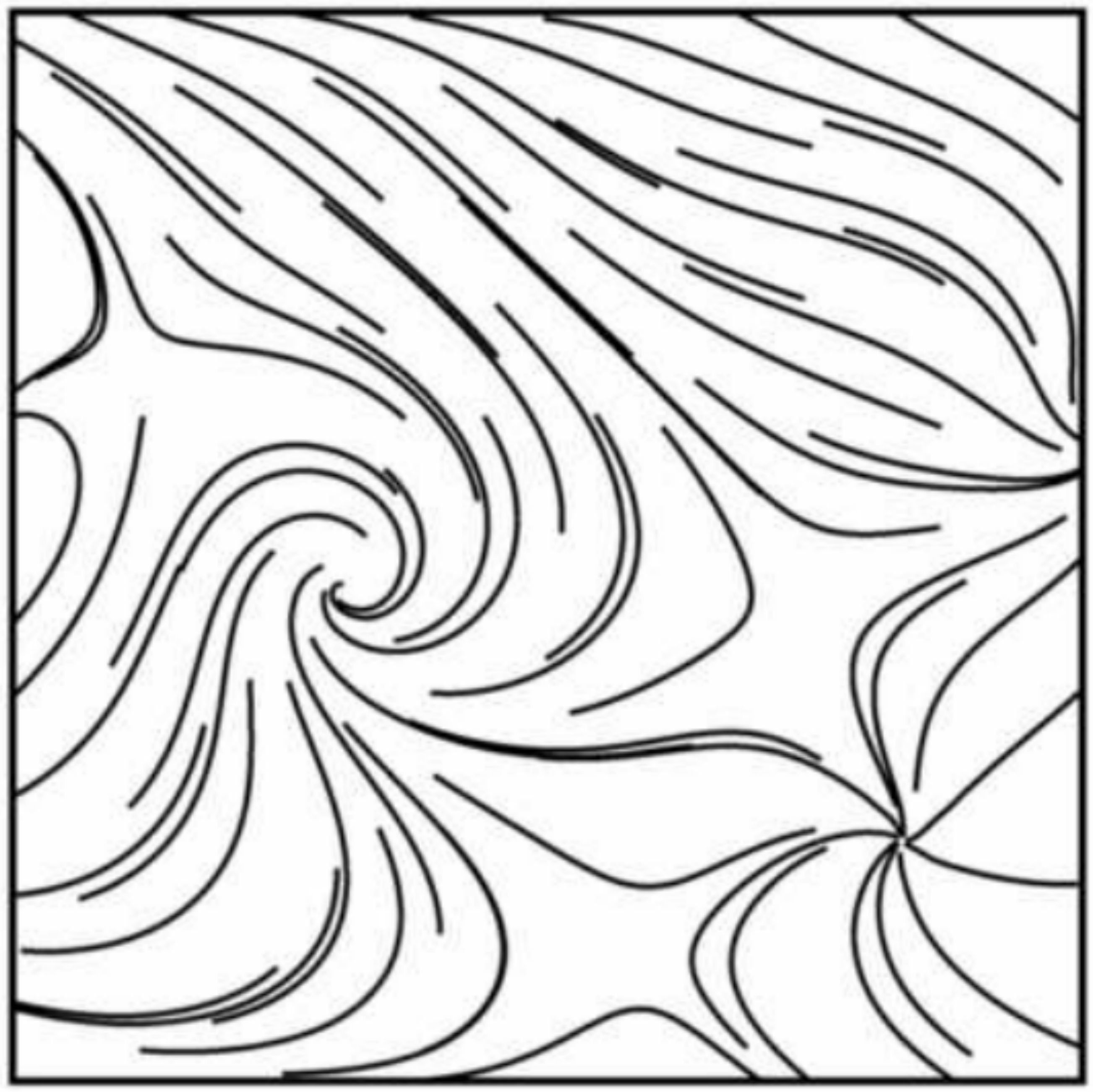


$$
\begin{aligned}
x'(t) &= v_x(x(t), y(t)) \\
y'(t) &= v_y(x(t), y(t))
\end{aligned}
$$

Visualization via streamlines

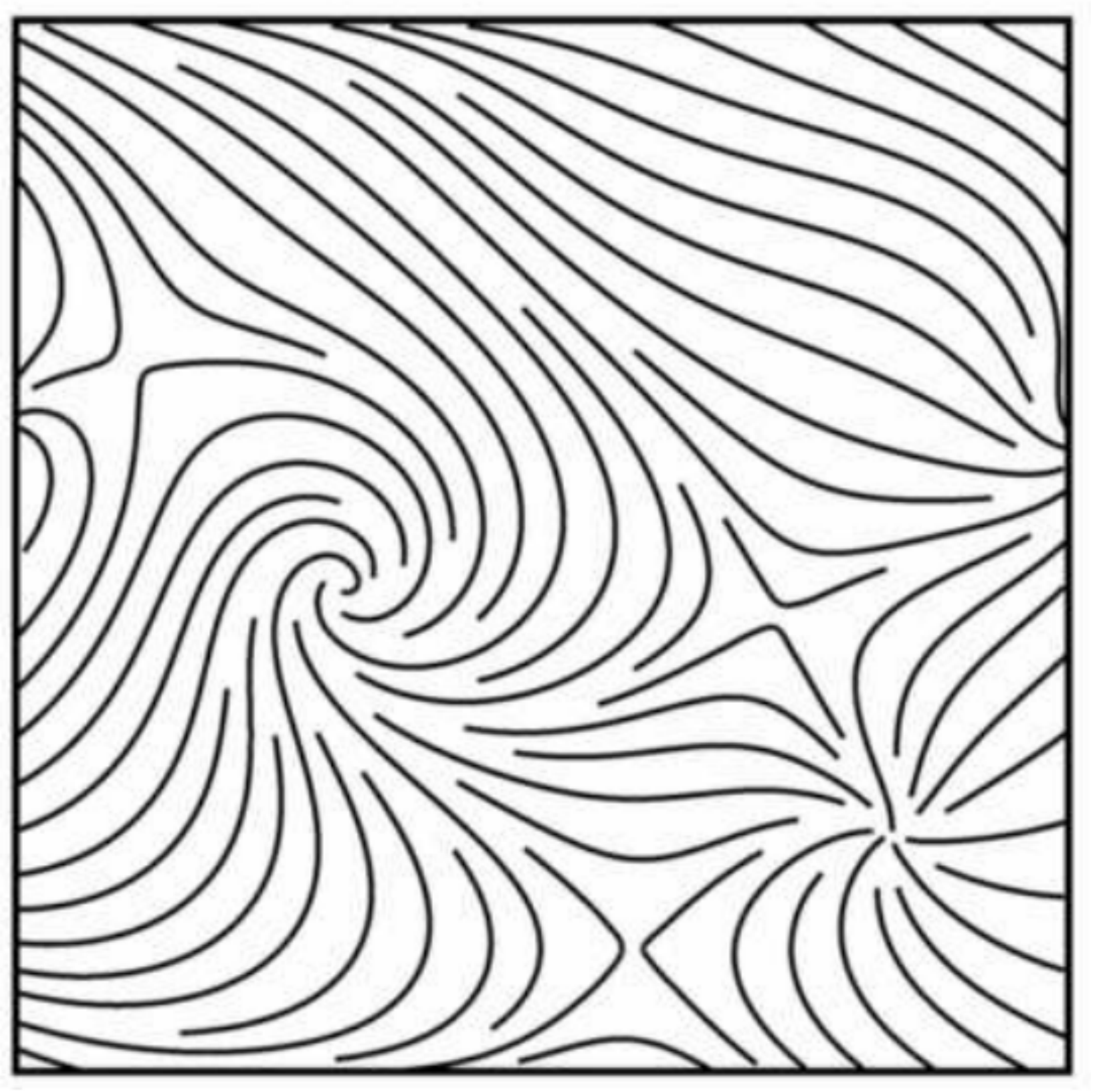
- Pick a set of seed points
- Integrate streamlines from those points
  - How do we compute this?
  - **https://cscheid.net/writing/data_science/odes/index.html**
- **Which seed points?**

Uniform placement



Turk and Banks, Image-Guided Streamline Placement
SIGGRAPH 1996

Density-optimized placement



Turk and Banks, Image-Guided Streamline Placement
SIGGRAPH 1996

Density-optimized placement

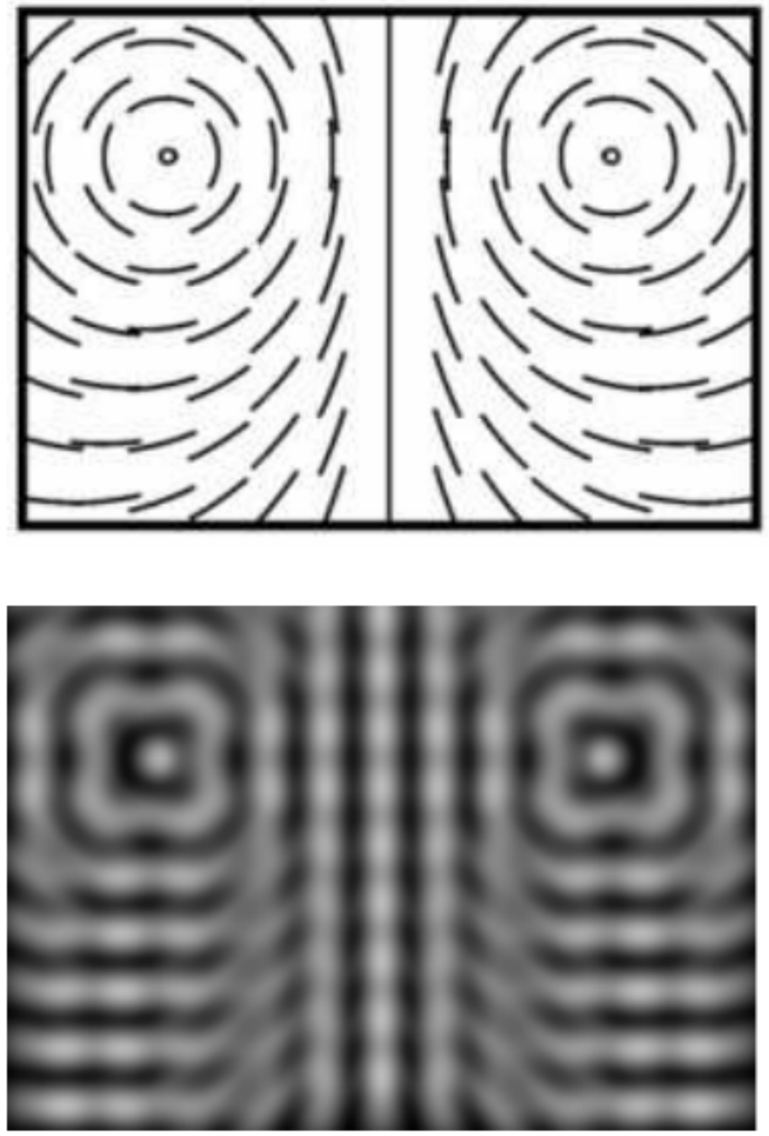


Figure 2: (a) Short streamlines with centers placed on a regular grid (top); (b) filtered version of same (bottom).

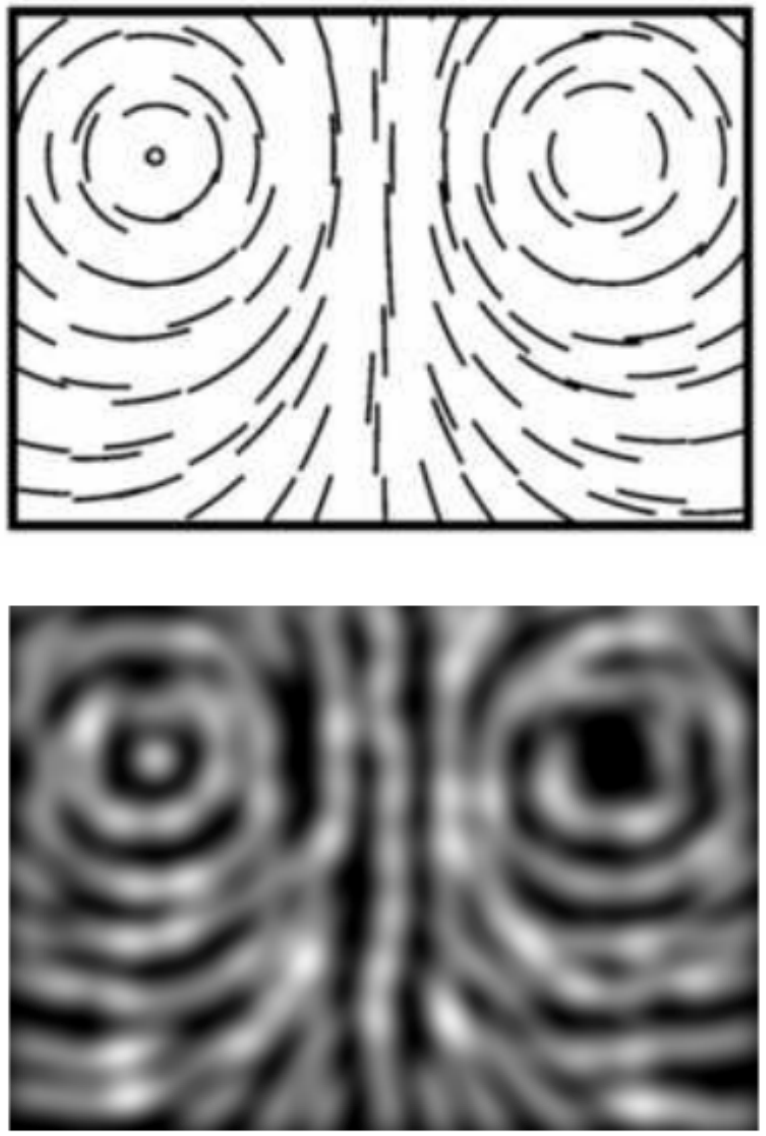


Figure 3: (a) Short streamlines with centers placed on a jittered grid (top); (b) filtered version showing bright and dark regions (bottom).

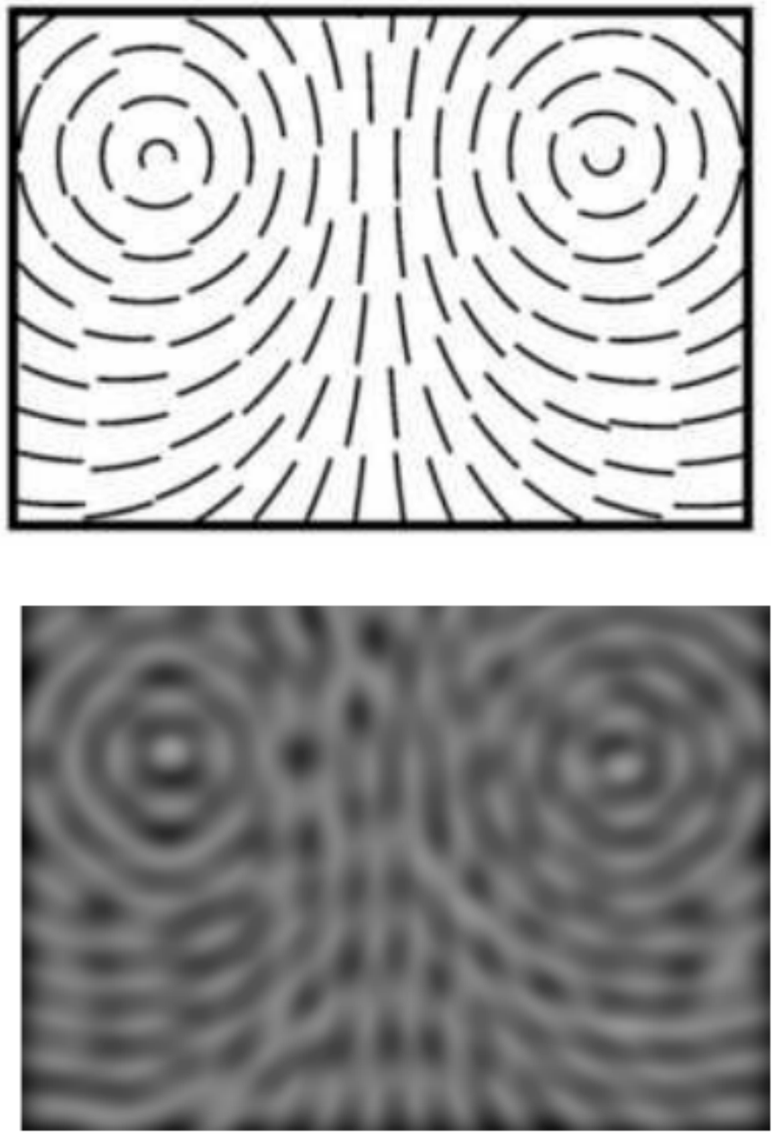
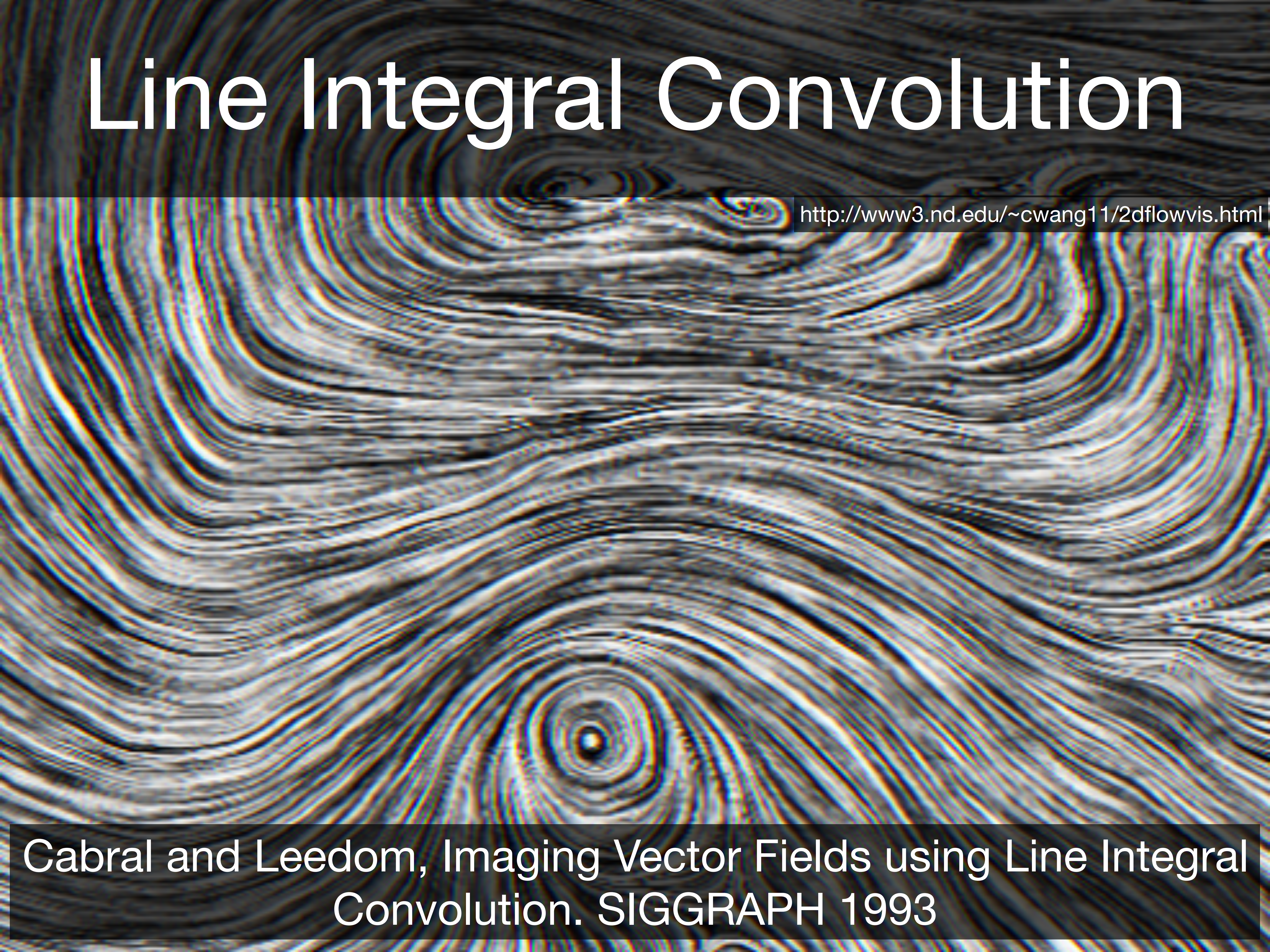


Figure 4: (a) Short streamlines placed by optimization (top); (b) filtered version showing fairly even gray value (bottom).

Turk and Banks, Image-Guided Streamline Placement
SIGGRAPH 1996

Image-Based
Vector Field Visualization

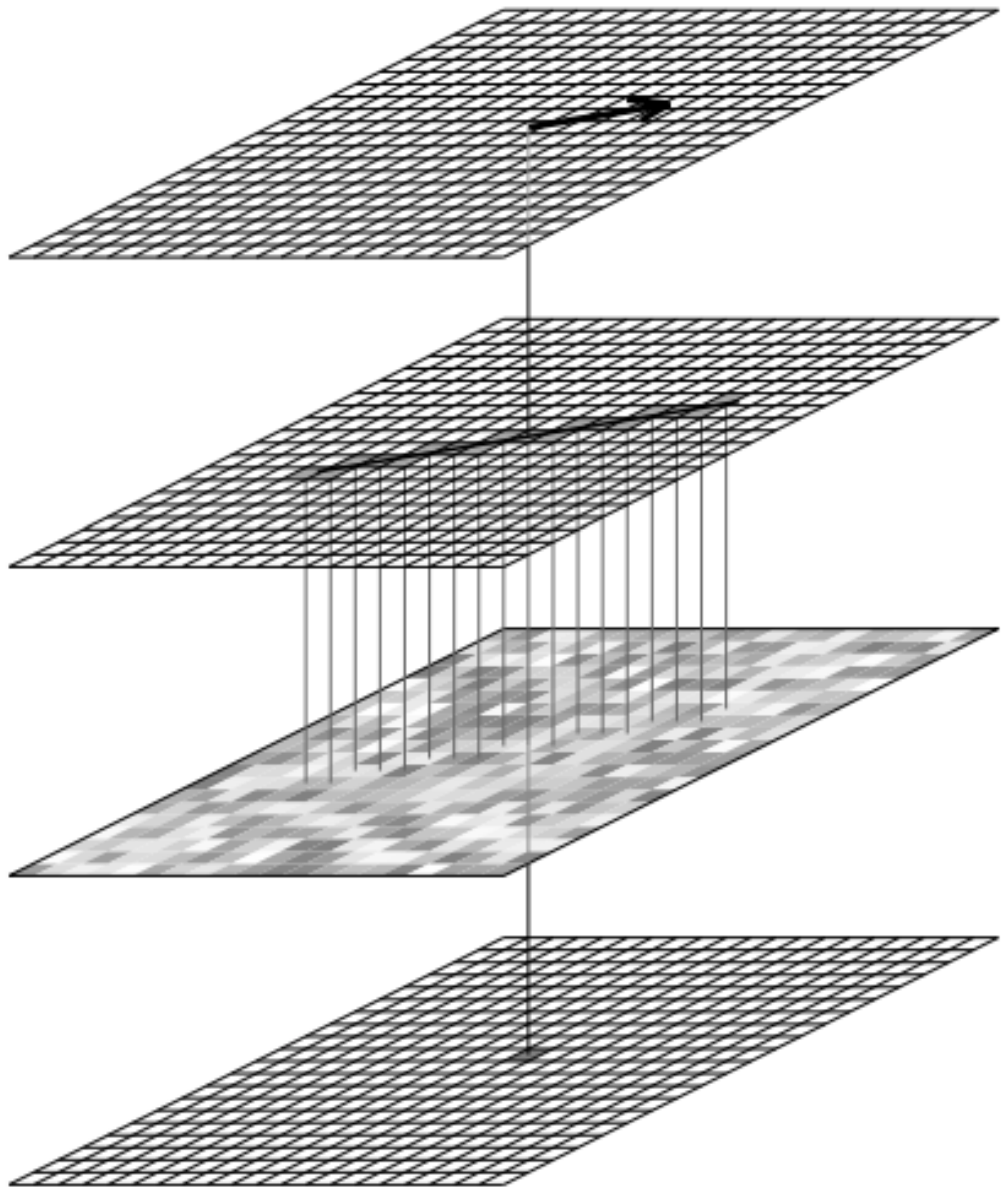
Line Integral Convolution



http://www3.nd.edu/~cwang11/2dflowvis.html

Cabral and Leedom, Imaging Vector Fields using Line Integral Convolution. SIGGRAPH 1993

Line Integral Convolution



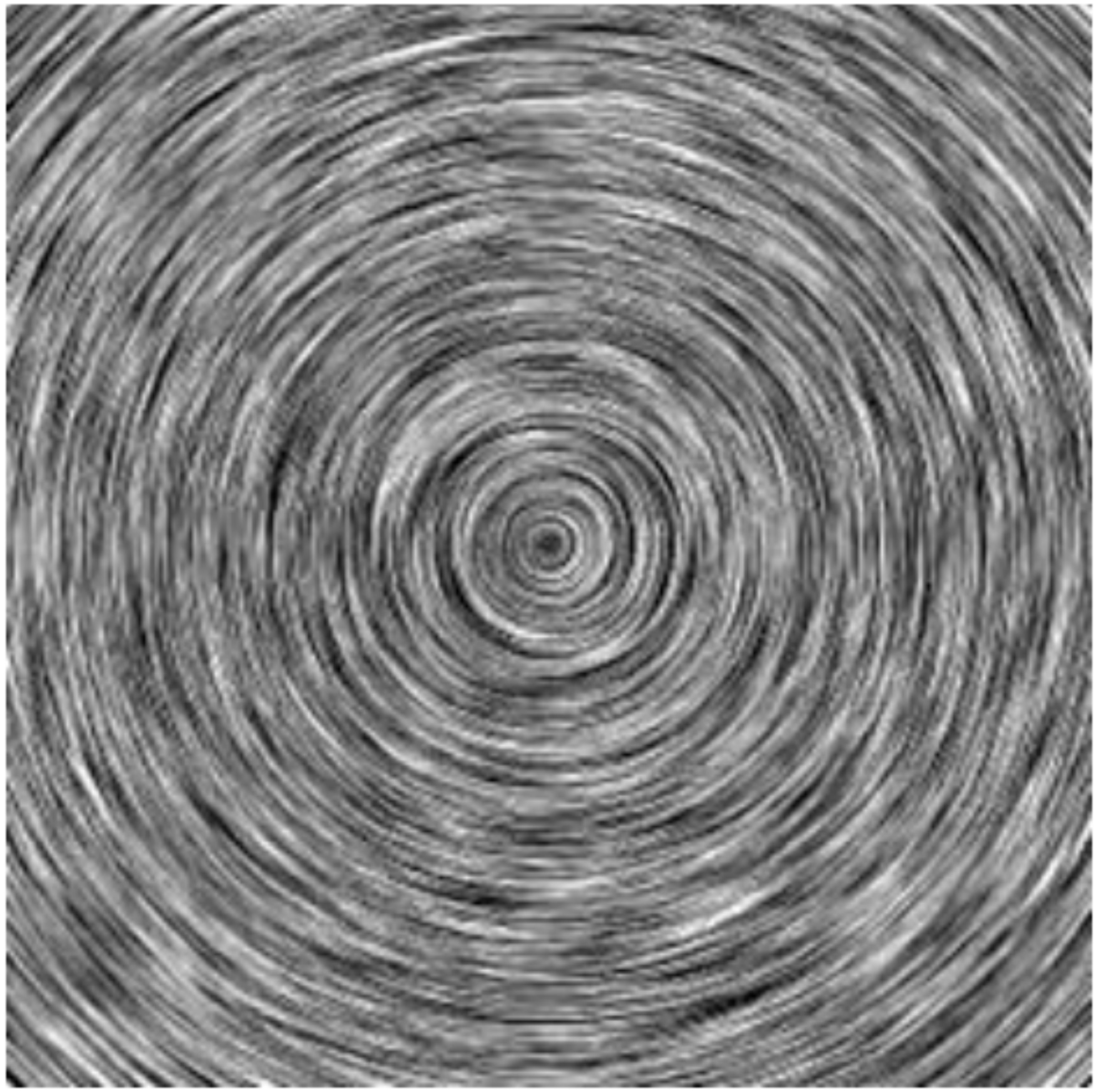
Given a
vector field

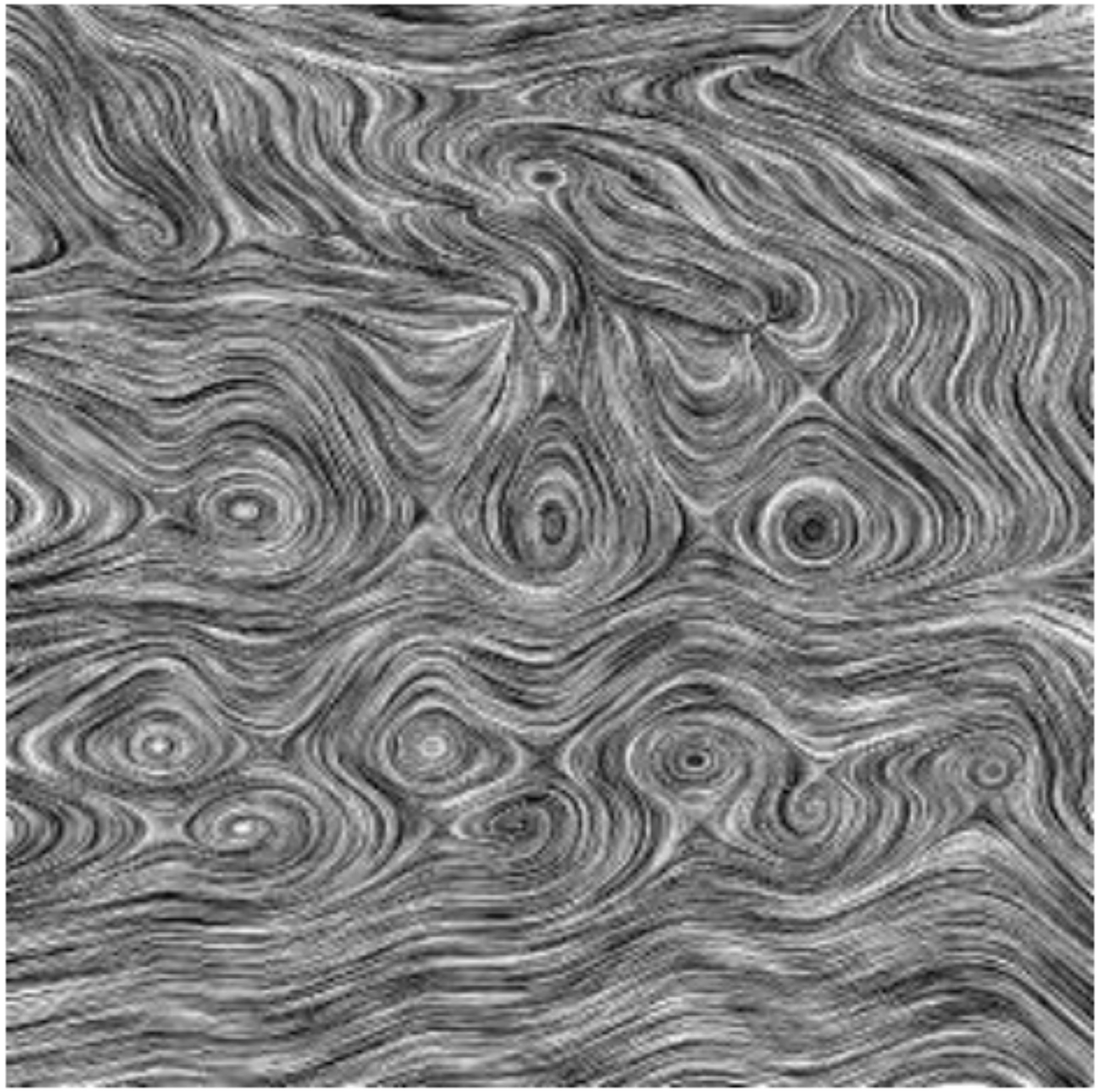
compute
streamlines

average
source of noise
along streamlines

Result

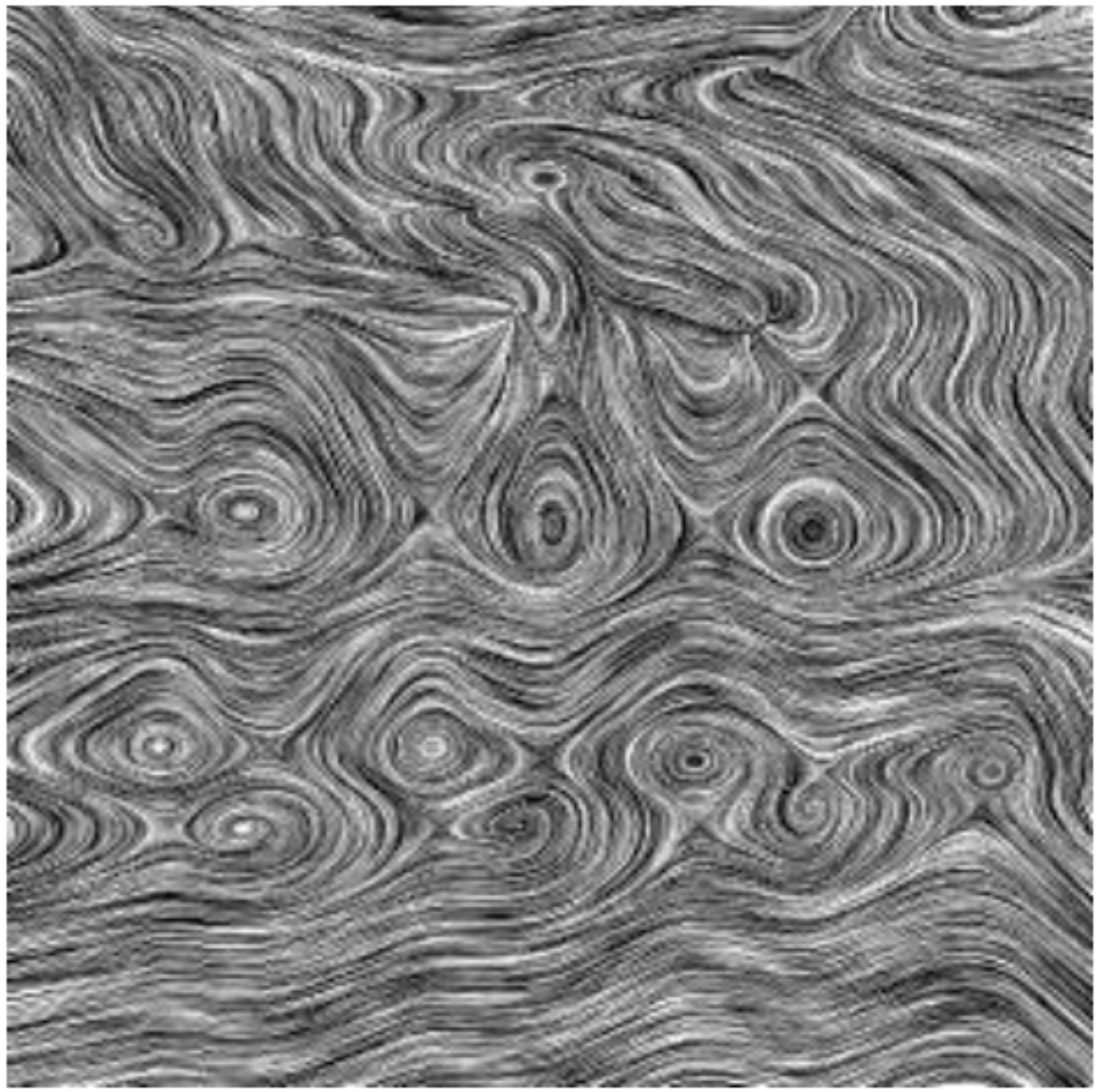
Line Integral Convolution





Advantages

- "Perfect" space usage
- Flow features are very apparent



Downsides

- No perception of velocity!
- No perception of direction!