CS 247 – Scientific Visualization
Lecture 13: Scalar Fields, Pt.9

Markus Hadwiger, KAUST

King Abdullah University of Science and Technology KAUST

Reading Assignment #7 (until Mar 14)

Read (required):

- Real-Time Volume Graphics, Chapter 1 (*Theoretical Background and Basic Approaches*), from beginning to 1.4.4 (inclusive)

Read (optional):

- Paper:
  Nelson Max, Optical Models for Direct Volume Rendering, IEEE Transactions on Visualization and Computer Graphics, 1995
  http://dx.doi.org/10.1109/2945.468400

Interlude: Tensor Calculus

In tensor calculus, first-order tensors can be

- Contravariant $\mathbf{v} = v^i \, \mathbf{e}_i$
- Covariant $\boldsymbol{\omega} = v_i \, \boldsymbol{\omega}^i$

The gradient vector is a contravariant vector $\mathbf{v} = v^i \, \boldsymbol{\partial}_i$

The gradient 1-form is a covariant vector (a covector) $df = \frac{\partial f}{\partial x^i} \, dx^i$

Very powerful; necessary for non-Cartesian coordinate systems

On (intrinsically) curved manifolds (sphere, ...):
Cartesian coordinates not even possible

Interlude: Tensor Calculus

In tensor calculus, first-order tensors can be

- Contravariant $\quad \mathbf{v} = v^i \, \mathbf{e}_i$
- Covariant $\quad \boldsymbol{\omega} = v_i \, \boldsymbol{\omega}^i$

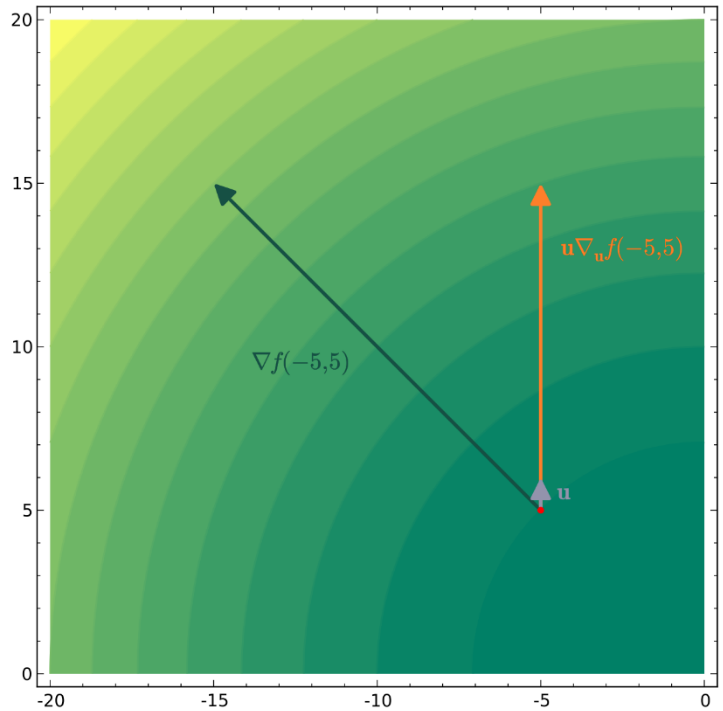
The gradient vector is a contravariant vector $\quad \mathbf{v} = v^i \, \boldsymbol{\partial}_i$

The gradient 1-form is a covariant vector (a covector) $\quad df = \dfrac{\partial f}{\partial x^i} dx^i$

This is also the fundamental reason why in graphics a normal vector transforms differently: as a covector, not as a vector!

(typical graphics rule: **n** transforms with transpose of inverse matrix)

Gradient Vectors and Differential 1-Forms

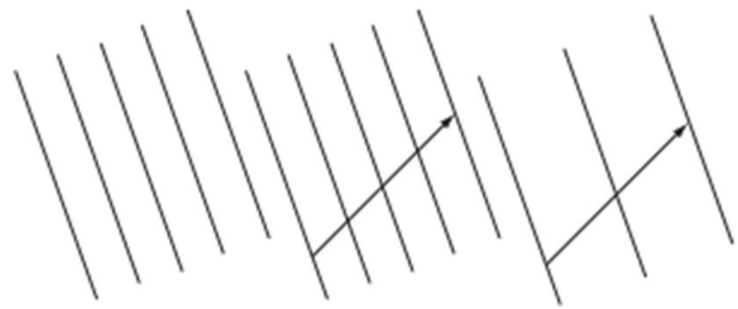


from Wikipedia (for **u** a unit vector),
the function here is $f(r,\theta) = r^2$

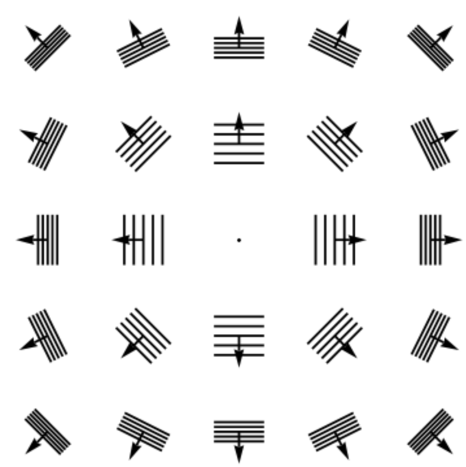
$$\nabla f(r,\theta) = 2r\,\mathbf{e}_r + 0\frac{1}{r^2}\,\mathbf{e}_\theta = 2r\,\mathbf{e}_r$$

$$df(r,\theta) = 2r\,dr + 0\,d\theta = 2r\,dr$$

different 1-forms
evaluated in some direction



1-form (field) df



Gradient Vector from Differential 1-Form

The metric (and inverse metric) *lower* or *raise* indices

(i.e., convert between covariant and contravariant tensors)

$$\boxed{\begin{aligned} v^i &= g^{ij} v_j \\ v_i &= g_{ij} v^j \end{aligned}}$$

$$\begin{aligned} v^i \mathbf{e}_i &= g^{ij} v_j \mathbf{e}_i \\ v_i \boldsymbol{\omega}^i &= g_{ij} v^j \boldsymbol{\omega}^i \end{aligned}$$

Inverse metric (contravariant)

$$\boxed{[g^{ij}] = [g_{ij}]^{-1}}$$

$$g^{ik} g_{kj} = \delta^i_j$$

Kronecker delta behaves like identity matrix

Gradient Vector from Differential 1-Form

So the gradient vector is

$$\boxed{\nabla f = \left(g^{ij} \frac{\partial f}{\partial x^j} \right) \mathbf{e}_i}$$

Vector-valued 1-form

$$d\mathbf{r} = dx^i \, \mathbf{e}_i$$

$$d\mathbf{r}(\cdot) = dx^i(\cdot) \, \mathbf{e}_i$$

Directional derivative via inner product:

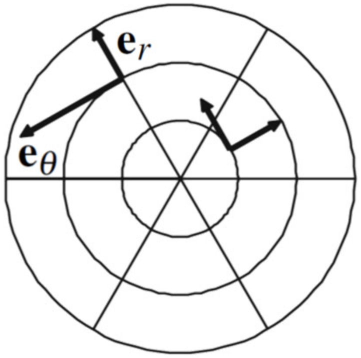
$$\begin{aligned} \langle \nabla f, \cdot \rangle &= g_{kj} g^{ik} \frac{\partial f}{\partial x^i} dx^j(\cdot) \\ &= \delta^i{}_j \frac{\partial f}{\partial x^i} dx^j(\cdot) \\ &= \frac{\partial f}{\partial x^i} dx^i(\cdot) \end{aligned}$$

$$\begin{aligned} \nabla f \cdot d\mathbf{r} &= g_{kj} g^{ik} \frac{\partial f}{\partial x^i} dx^j \\ &= \delta^i{}_j \frac{\partial f}{\partial x^i} dx^j \\ &= \frac{\partial f}{\partial x^i} dx^i \end{aligned}$$

Example: Polar Coordinates

Metric tensor and inverse metric for polar coordinates

$$g_{ij} = \begin{bmatrix} 1 & 0 \\ 0 & r^2 \end{bmatrix} \qquad g^{ij} = \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{r^2} \end{bmatrix}$$



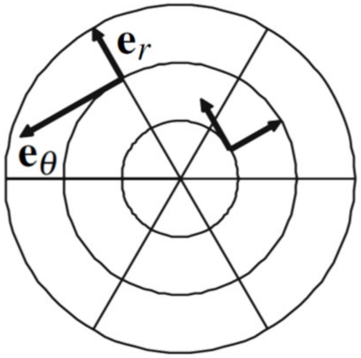
Gradient vector from 1-form: raise index with inverse metric

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial r} & \frac{\partial f}{\partial \theta} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{r^2} \end{bmatrix} \begin{bmatrix} \mathbf{e}_r \\ \mathbf{e}_\theta \end{bmatrix} = \frac{\partial f}{\partial r}\mathbf{e}_r + \frac{1}{r^2}\frac{\partial f}{\partial \theta}\mathbf{e}_\theta$$

Example: Polar Coordinates

Metric tensor and inverse metric for polar coordinates

$$g_{ij} = \begin{bmatrix} 1 & 0 \\ 0 & r^2 \end{bmatrix} \qquad g^{ij} = \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{r^2} \end{bmatrix}$$



Gradient vector from 1-form: raise index with inverse metric

$$\nabla f(r,\theta) = \frac{\partial f(r,\theta)}{\partial r}\,\mathbf{e}_r(r,\theta) + \frac{1}{r^2}\frac{\partial f(r,\theta)}{\partial \theta}\,\mathbf{e}_\theta(r,\theta)$$

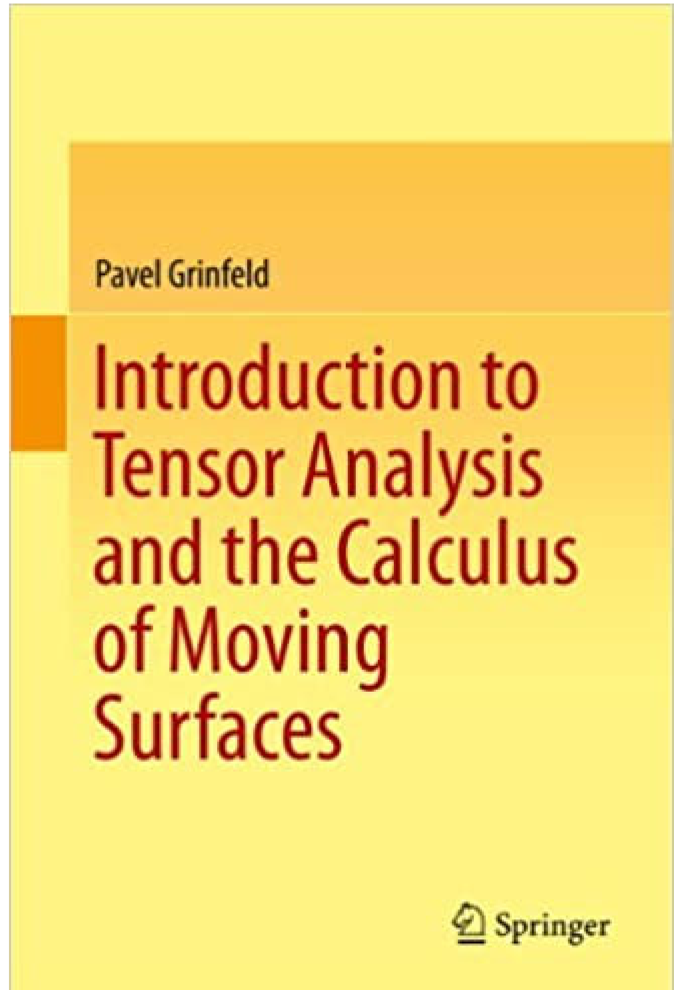
don’t forget that all of this is position-dependent!

Tensor Calculus

Highly recommended:

Very nice book,

complete lecture on Youtube!



Multi-Linear Interpolation

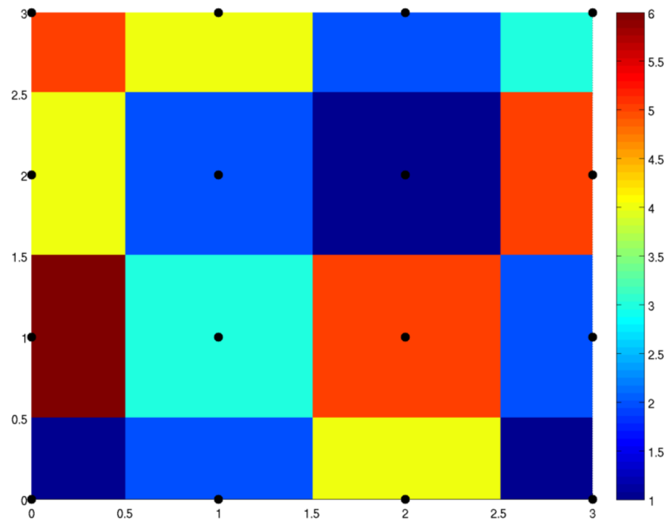
Bi-linear Filtering Example (Magnification)

Original image

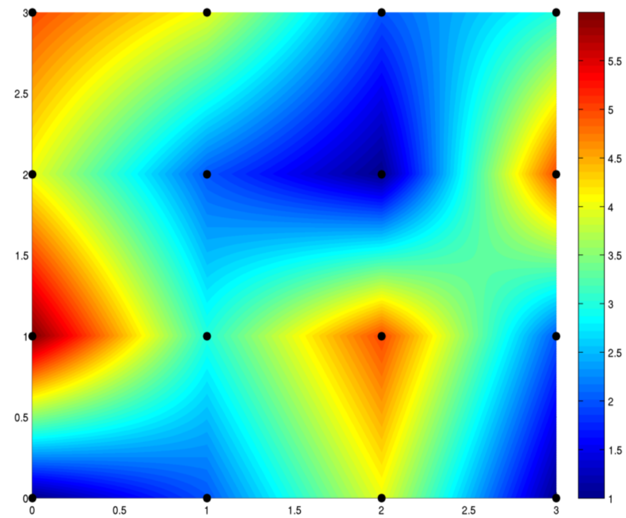
Nearest neighbor

Bi-linear filtering

Bi-Linear Interpolation vs. Nearest Neighbor



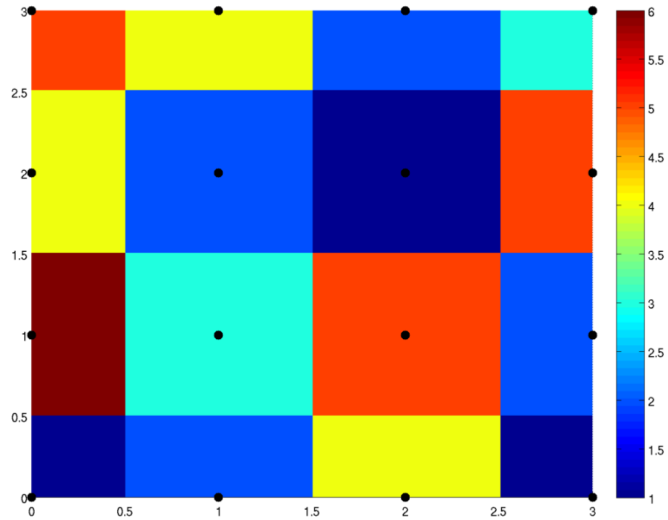
nearest-neighbor



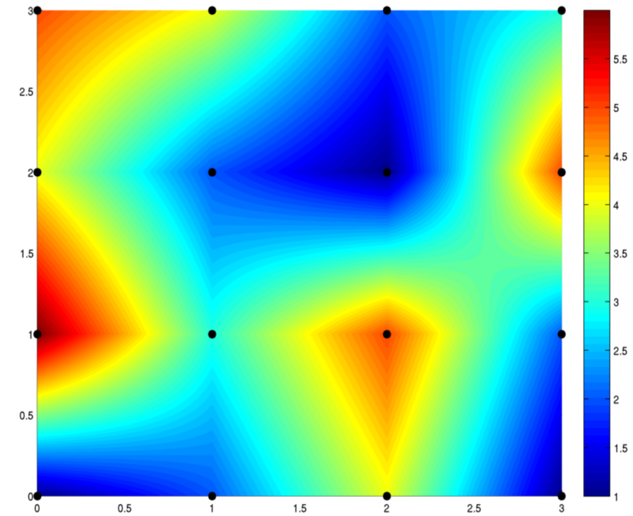
bi-linear

wikipedia

Bi-Linear Interpolation vs. Nearest Neighbor



nearest-neighbor



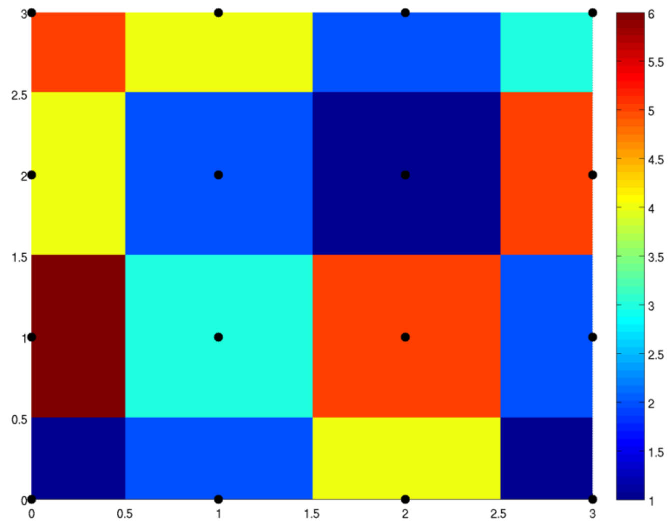
bi-linear

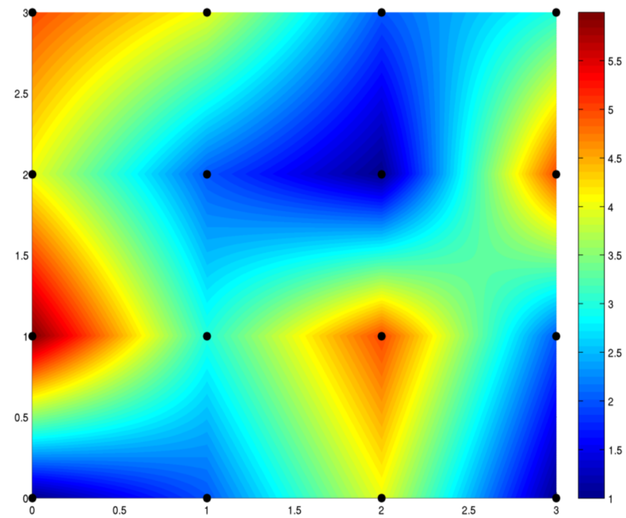
wikipedia

for surfaces,
height interpolation:

Bilinear patch (courtesy J. Han)

Bi-Linear Interpolation vs. Nearest Neighbor



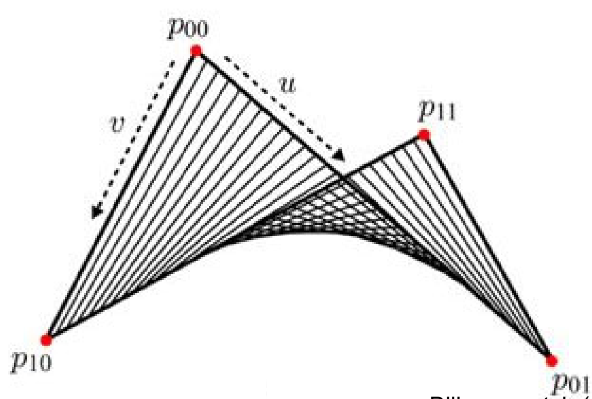


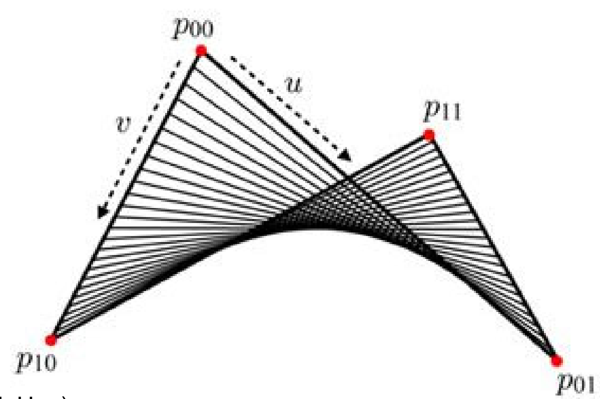
wikipedia

nearest-neighbor

bi-linear

for surfaces,
height interpolation:



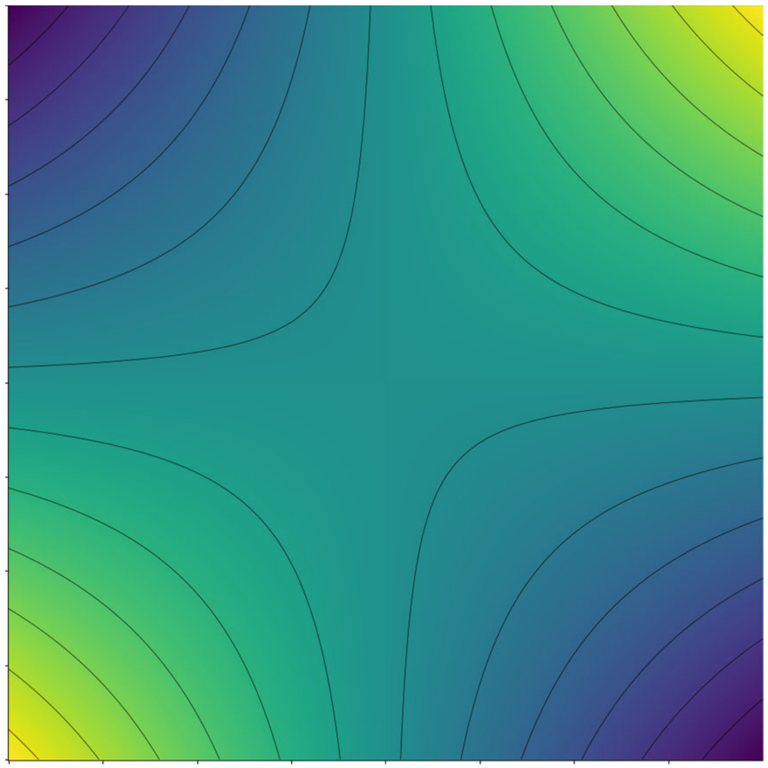


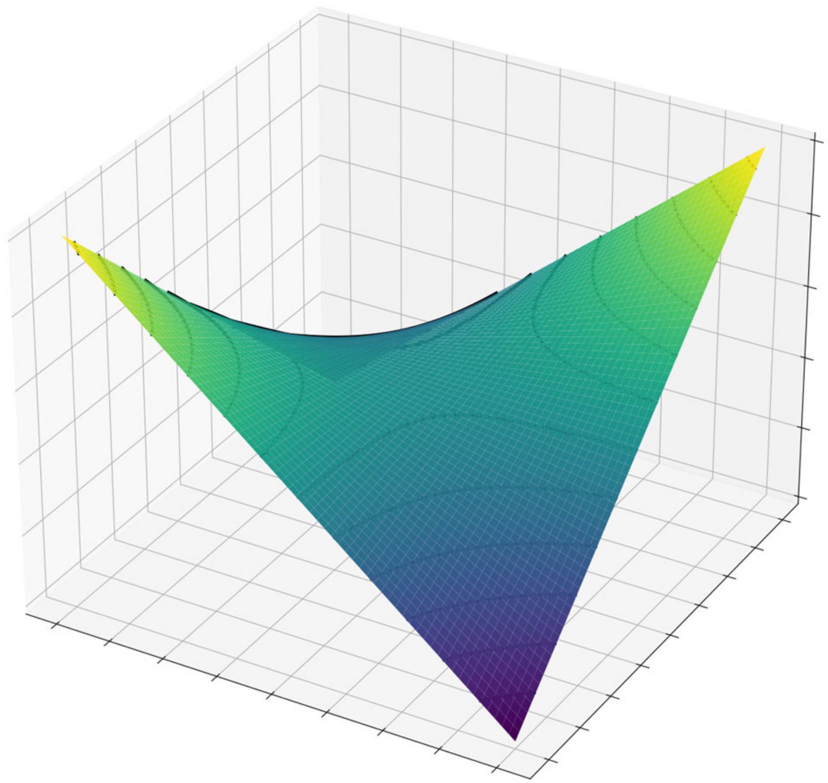
Bilinear patch (courtesy J. Han)

Bi-Linear Interpolation

Consider area between 2x2 adjacent samples (e.g., pixel centers)

Example #1: 1 at bottom-left and top-right, 0 at top-left and bottom-right

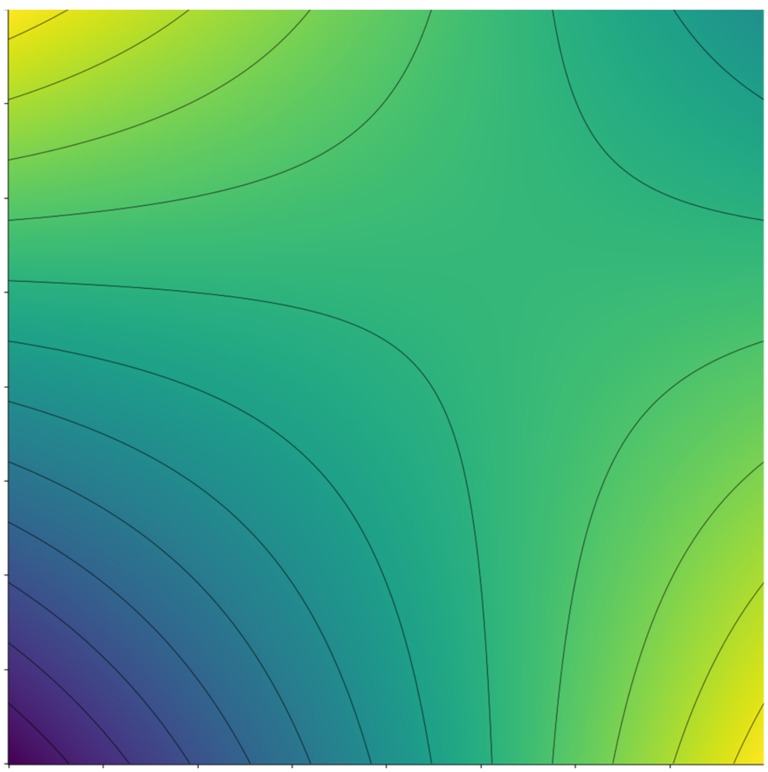


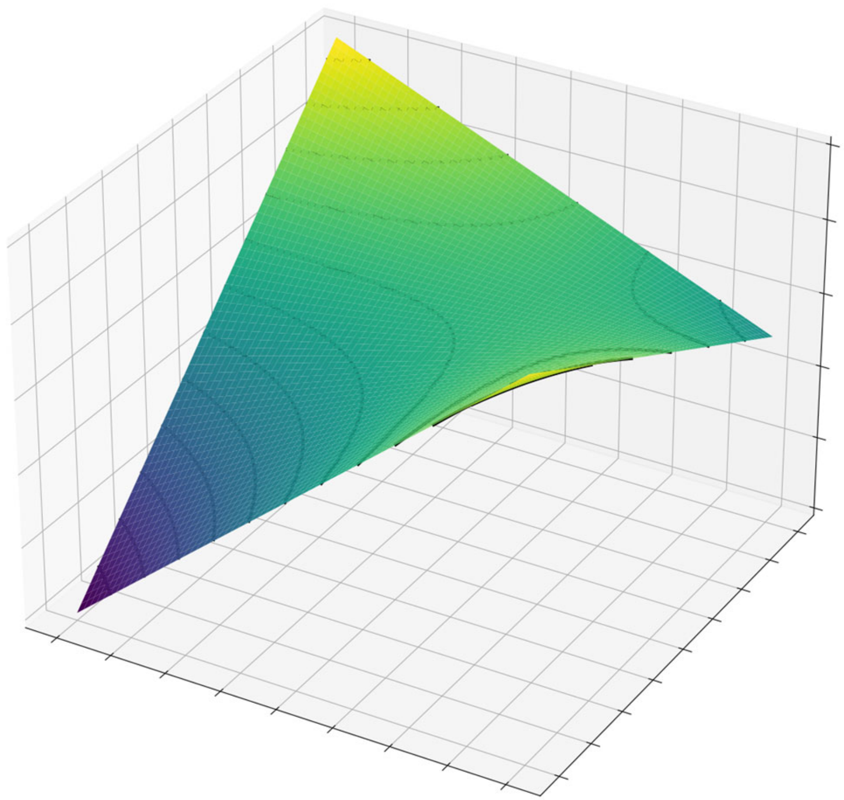


Bi-Linear Interpolation

Consider area between 2x2 adjacent samples (e.g., pixel centers)

Example #2: 1 at top-left and bottom-right, 0 at bottom-left, 0.5 at top-right





Bi-Linear Interpolation

Consider area between 2x2 adjacent samples (e.g., pixel centers):

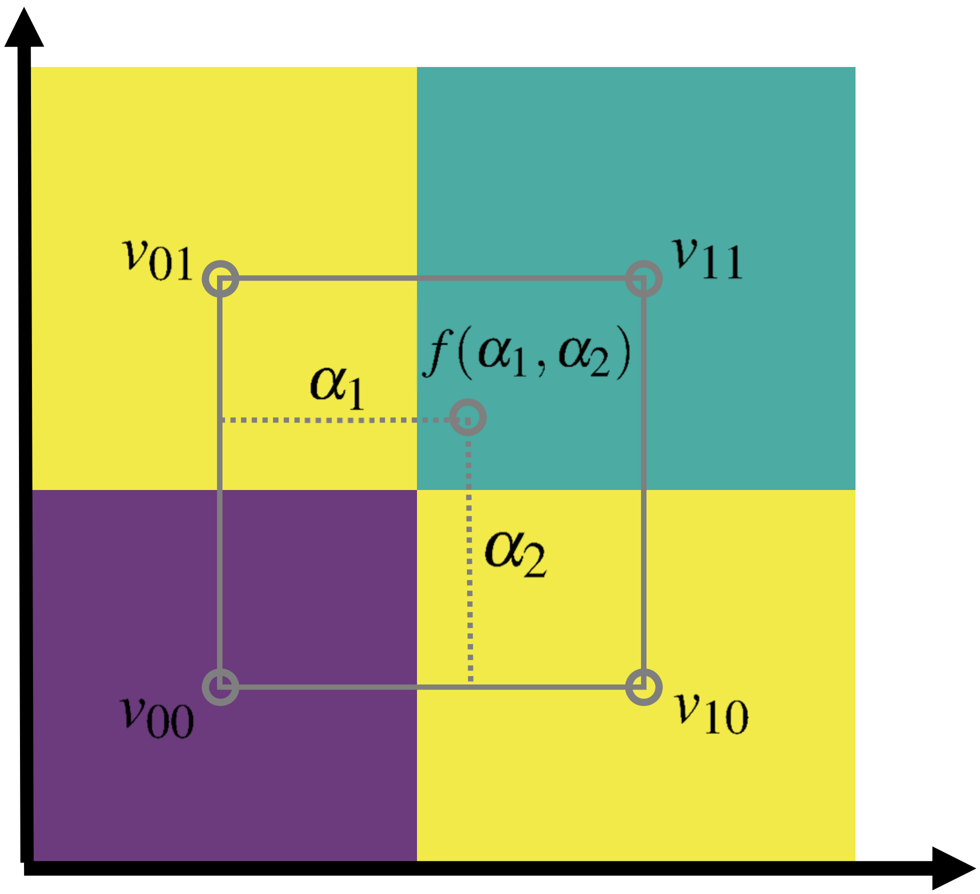
Given any (fractional) position

$$\alpha_1 := x_1 - \lfloor x_1 \rfloor \qquad \alpha_1 \in [0.0, 1.0)$$
$$\alpha_2 := x_2 - \lfloor x_2 \rfloor \qquad \alpha_2 \in [0.0, 1.0)$$

and 2x2 sample values

$$\begin{bmatrix} v_{01} & v_{11} \\ v_{00} & v_{10} \end{bmatrix}$$

Compute: $f(\alpha_1, \alpha_2)$



Bi-Linear Interpolation

Consider area between 2x2 adjacent samples (e.g., pixel centers):

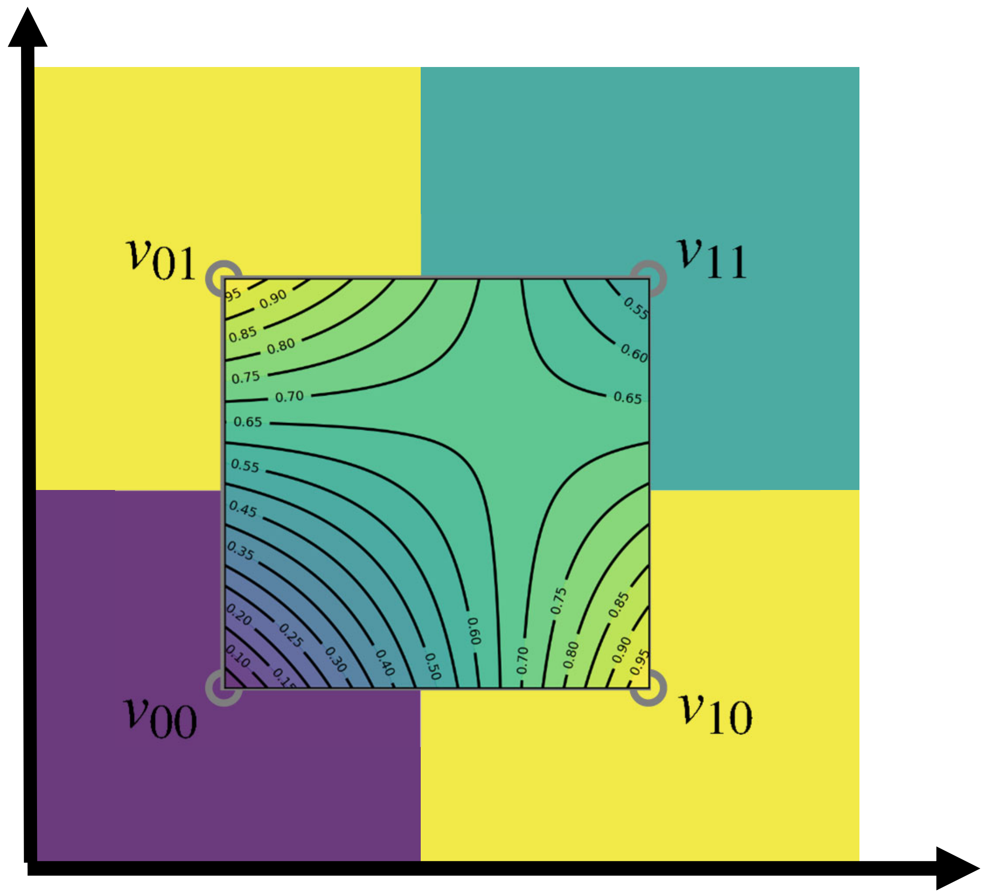
Given any (fractional) position

$$\alpha_1 := x_1 - \lfloor x_1 \rfloor \qquad \alpha_1 \in [0.0, 1.0)$$
$$\alpha_2 := x_2 - \lfloor x_2 \rfloor \qquad \alpha_2 \in [0.0, 1.0)$$

and 2x2 sample values

$$\begin{bmatrix} v_{01} & v_{11} \\ v_{00} & v_{10} \end{bmatrix}$$

Compute: $f(\alpha_1, \alpha_2)$



Bi-Linear Interpolation

Interpolate function at (fractional) position (α_1, α_2) :

$$f(\alpha_1, \alpha_2) = \begin{bmatrix} \alpha_2 & (1-\alpha_2) \end{bmatrix} \begin{bmatrix} v_{01} & v_{11} \\ v_{00} & v_{10} \end{bmatrix} \begin{bmatrix} (1-\alpha_1) \\ \alpha_1 \end{bmatrix}$$

$$= \begin{bmatrix} \alpha_2 & (1-\alpha_2) \end{bmatrix} \begin{bmatrix} (1-\alpha_1)v_{01} + \alpha_1 v_{11} \\ (1-\alpha_1)v_{00} + \alpha_1 v_{10} \end{bmatrix}$$

$$= \begin{bmatrix} \alpha_2 v_{01} + (1-\alpha_2)v_{00} & \alpha_2 v_{11} + (1-\alpha_2)v_{10} \end{bmatrix} \begin{bmatrix} (1-\alpha_1) \\ \alpha_1 \end{bmatrix}$$

Bi-Linear Interpolation

Interpolate function at (fractional) position (α_1, α_2) :

$$f(\alpha_1, \alpha_2) = \begin{bmatrix} \alpha_2 & (1-\alpha_2) \end{bmatrix} \begin{bmatrix} v_{01} & v_{11} \\ v_{00} & v_{10} \end{bmatrix} \begin{bmatrix} (1-\alpha_1) \\ \alpha_1 \end{bmatrix}$$

$$= (1-\alpha_1)(1-\alpha_2)v_{00} + \alpha_1(1-\alpha_2)v_{10} + (1-\alpha_1)\alpha_2 v_{01} + \alpha_1\alpha_2 v_{11}$$

$$= v_{00} + \alpha_1(v_{10} - v_{00}) + \alpha_2(v_{01} - v_{00}) + \alpha_1\alpha_2(v_{00} + v_{11} - v_{10} - v_{01})$$

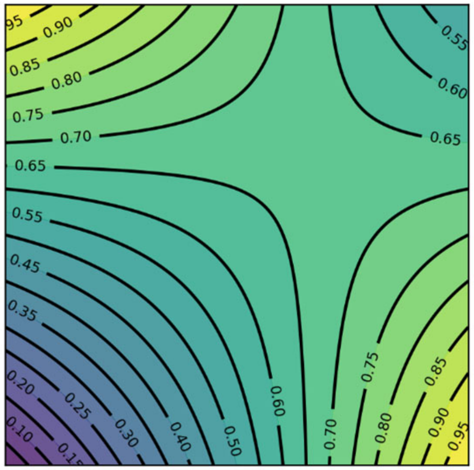
Bi-Linear Interpolation: Contours

Find one specific iso-contour (can of course do this for any/all isovalues):

$$f(\alpha_1, \alpha_2) = c$$

Find all (α_1, α_2) where:

$$v_{00} + \alpha_1 (v_{10} - v_{00}) + \alpha_2 (v_{01} - v_{00}) + \alpha_1 \alpha_2 (v_{00} + v_{11} - v_{10} - v_{01}) = c$$



Bi-Linear Interpolation: Critical Points

Compute gradient (critical points are where gradient is zero vector):

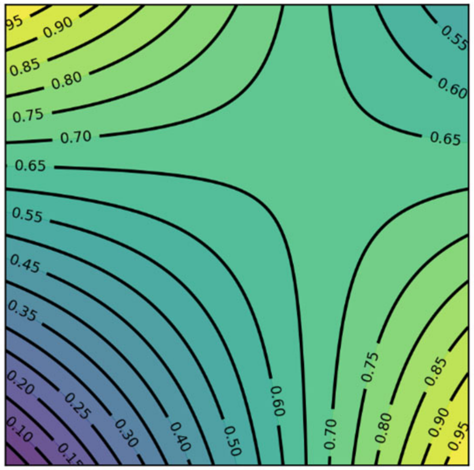
$$\frac{\partial f(\alpha_1, \alpha_2)}{\partial \alpha_1} = (v_{10} - v_{00}) + \alpha_2 (v_{00} + v_{11} - v_{10} - v_{01})$$

$$\frac{\partial f(\alpha_1, \alpha_2)}{\partial \alpha_2} = (v_{01} - v_{00}) + \alpha_1 (v_{00} + v_{11} - v_{10} - v_{01})$$

Where are lines of constant value / critical points?

$$\frac{\partial f(\alpha_1, \alpha_2)}{\partial \alpha_1} = 0: \qquad \alpha_2 = \frac{v_{00} - v_{10}}{v_{00} + v_{11} - v_{10} - v_{01}}$$

$$\frac{\partial f(\alpha_1, \alpha_2)}{\partial \alpha_2} = 0: \qquad \alpha_1 = \frac{v_{00} - v_{01}}{v_{00} + v_{11} - v_{10} - v_{01}}$$



Bi-Linear Interpolation: Critical Points

Compute gradient (critical points are where gradient is zero vector):

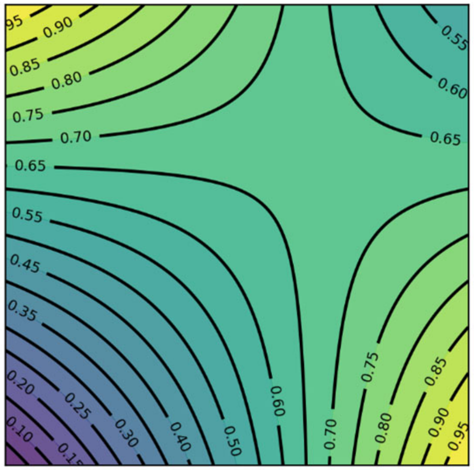
$$\frac{\partial f(\alpha_1,\alpha_2)}{\partial \alpha_1} = (v_{10} - v_{00}) + \alpha_2 (v_{00} + v_{11} - v_{10} - v_{01})$$

$$\frac{\partial f(\alpha_1,\alpha_2)}{\partial \alpha_2} = (v_{01} - v_{00}) + \alpha_1 (v_{00} + v_{11} - v_{10} - v_{01})$$

Where are lines of constant value / critical points?

$$\frac{\partial f(\alpha_1,\alpha_2)}{\partial \alpha_1} = 0: \qquad \alpha_2 = \frac{v_{00} - v_{10}}{v_{00} + v_{11} - v_{10} - v_{01}}$$

$$\frac{\partial f(\alpha_1,\alpha_2)}{\partial \alpha_2} = 0: \qquad \alpha_1 = \frac{v_{00} - v_{01}}{v_{00} + v_{11} - v_{10} - v_{01}}$$



if denominator is zero, bi-linear interpolation has degenerated to linear interpolation (or const)! (also means: no isolated critical points!)

Bi-Linear Interpolation: Critical Points

Compute gradient (critical points are where gradient is zero vector):

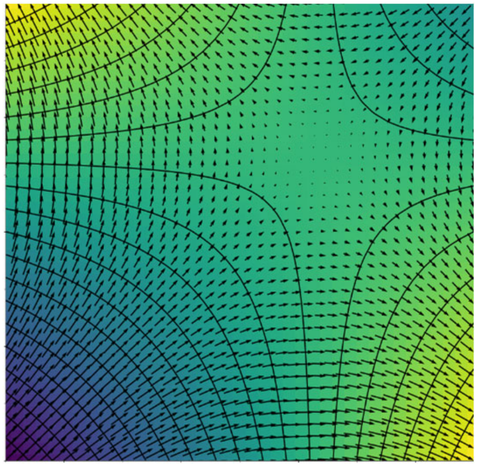
$$\frac{\partial f(\alpha_1, \alpha_2)}{\partial \alpha_1} = (v_{10} - v_{00}) + \alpha_2 (v_{00} + v_{11} - v_{10} - v_{01})$$

$$\frac{\partial f(\alpha_1, \alpha_2)}{\partial \alpha_2} = (v_{01} - v_{00}) + \alpha_1 (v_{00} + v_{11} - v_{10} - v_{01})$$

Where are lines of constant value / critical points?

$$\frac{\partial f(\alpha_1, \alpha_2)}{\partial \alpha_1} = 0: \qquad \alpha_2 = \frac{v_{00} - v_{10}}{v_{00} + v_{11} - v_{10} - v_{01}}$$

$$\frac{\partial f(\alpha_1, \alpha_2)}{\partial \alpha_2} = 0: \qquad \alpha_1 = \frac{v_{00} - v_{01}}{v_{00} + v_{11} - v_{10} - v_{01}}$$



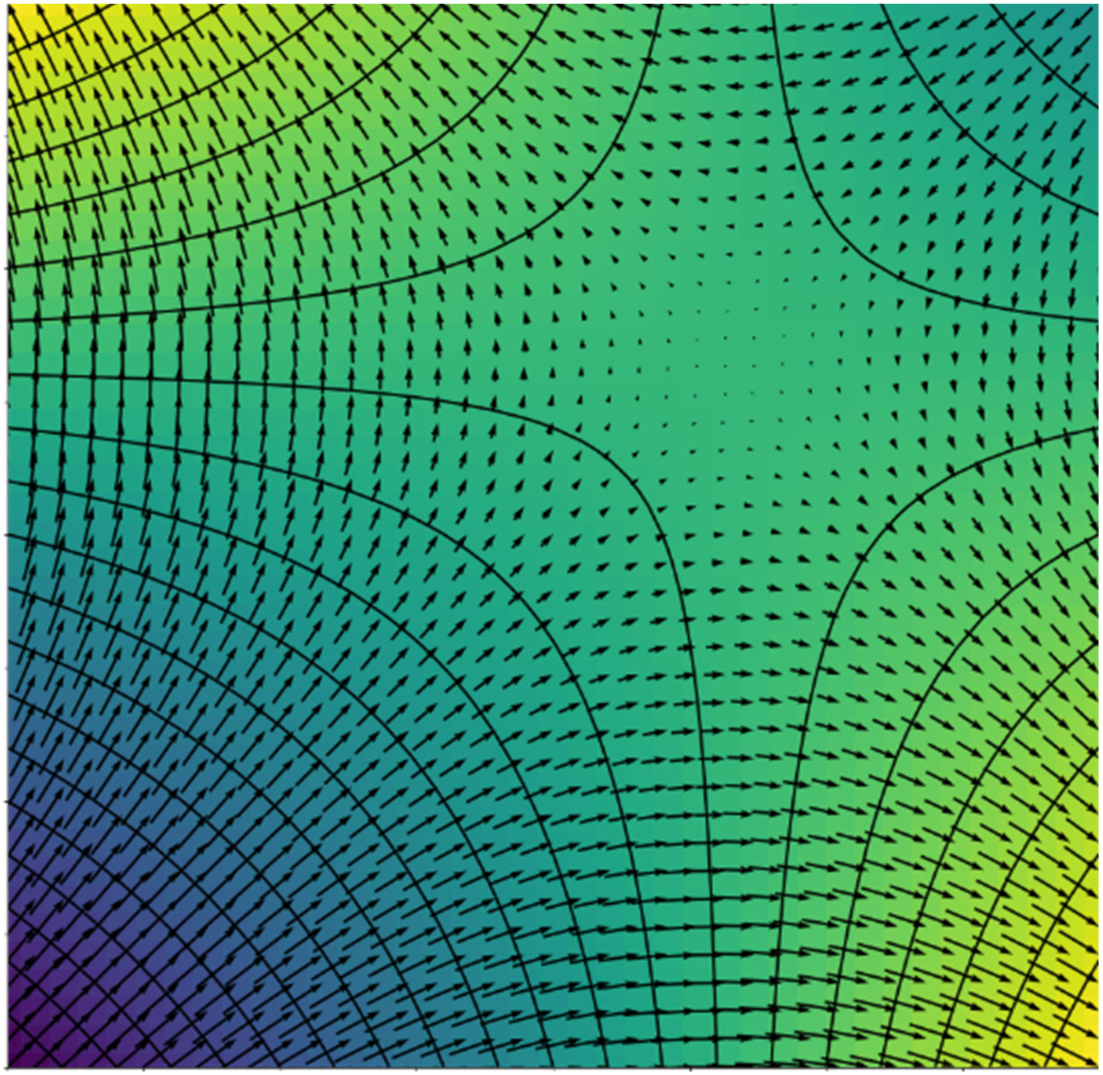
Bi-Linear Interpolation: Critical Points

Compute gradient

Note that isolines are
farther apart where
gradient is smaller

Note the horizontal and
vertical lines where
gradient becomes
vertical/horizontal

Note the critical point



Bi-Linear Interpolation: Critical Points

Compute gradient (critical points are where gradient is zero vector):

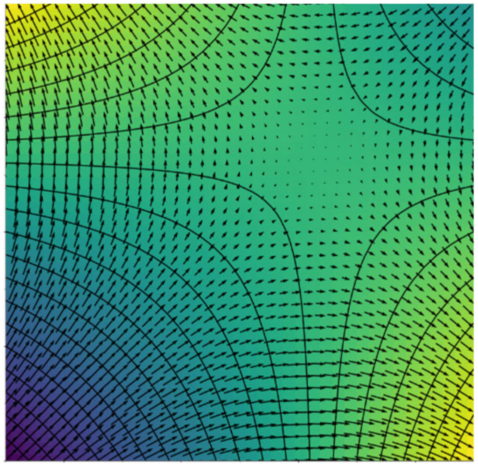
$$\frac{\partial f(\alpha_1,\alpha_2)}{\partial \alpha_1} = (v_{10} - v_{00}) + \alpha_2 (v_{00} + v_{11} - v_{10} - v_{01})$$

$$\frac{\partial f(\alpha_1,\alpha_2)}{\partial \alpha_2} = (v_{01} - v_{00}) + \alpha_1 (v_{00} + v_{11} - v_{10} - v_{01})$$

Where are lines of constant value / critical points?

$$\frac{\partial f(\alpha_1,\alpha_2)}{\partial \alpha_1} = 0: \qquad \alpha_2 = \frac{v_{00} - v_{10}}{v_{00} + v_{11} - v_{10} - v_{01}}$$

$$\frac{\partial f(\alpha_1,\alpha_2)}{\partial \alpha_2} = 0: \qquad \alpha_1 = \frac{v_{00} - v_{01}}{v_{00} + v_{11} - v_{10} - v_{01}}$$



Bi-Linear Interpolation: Critical Points

Compute gradient (critical points are where gradient is zero vector):

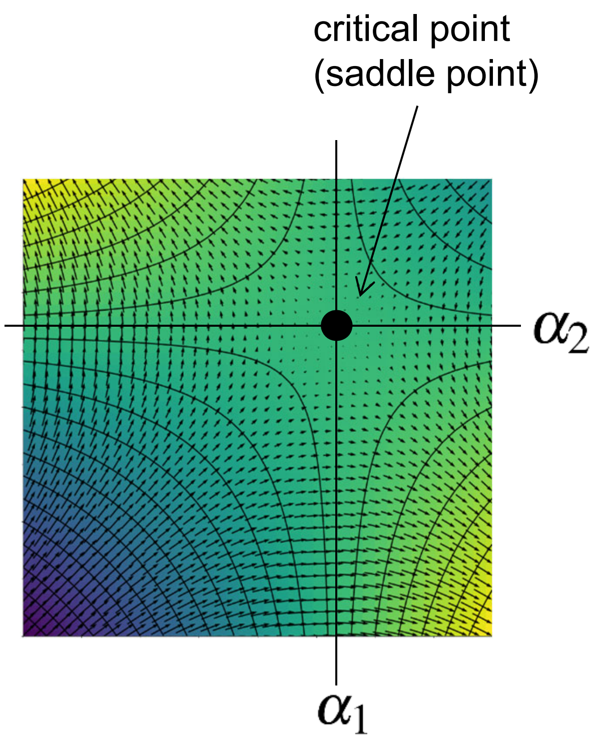
$$\frac{\partial f(\alpha_1,\alpha_2)}{\partial \alpha_1} = (v_{10} - v_{00}) + \alpha_2(v_{00} + v_{11} - v_{10} - v_{01})$$

$$\frac{\partial f(\alpha_1,\alpha_2)}{\partial \alpha_2} = (v_{01} - v_{00}) + \alpha_1(v_{00} + v_{11} - v_{10} - v_{01})$$

Where are lines of constant value / critical points?

$$\frac{\partial f(\alpha_1,\alpha_2)}{\partial \alpha_1} = 0: \qquad \alpha_2 = \frac{v_{00} - v_{10}}{v_{00} + v_{11} - v_{10} - v_{01}}$$

$$\frac{\partial f(\alpha_1,\alpha_2)}{\partial \alpha_2} = 0: \qquad \alpha_1 = \frac{v_{00} - v_{01}}{v_{00} + v_{11} - v_{10} - v_{01}}$$



Bi-Linear Interpolation: Critical Points

Compute gradient (critical points are where gradient is zero vector):

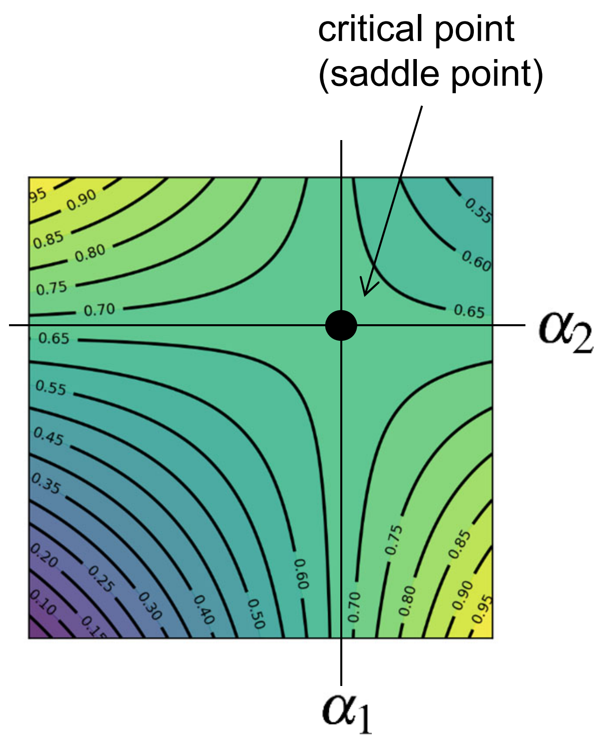
$$\frac{\partial f(\alpha_1,\alpha_2)}{\partial \alpha_1} = (v_{10} - v_{00}) + \alpha_2 (v_{00} + v_{11} - v_{10} - v_{01})$$

$$\frac{\partial f(\alpha_1,\alpha_2)}{\partial \alpha_2} = (v_{01} - v_{00}) + \alpha_1 (v_{00} + v_{11} - v_{10} - v_{01})$$

Where are lines of constant value / critical points?

$$\frac{\partial f(\alpha_1,\alpha_2)}{\partial \alpha_1} = 0: \qquad \alpha_2 = \frac{v_{00} - v_{10}}{v_{00} + v_{11} - v_{10} - v_{01}}$$

$$\frac{\partial f(\alpha_1,\alpha_2)}{\partial \alpha_2} = 0: \qquad \alpha_1 = \frac{v_{00} - v_{01}}{v_{00} + v_{11} - v_{10} - v_{01}}$$



Bi-Linear Interpolation: Critical Points

Examine Hessian matrix at critical point (non-degenerate critical p.?, ...)

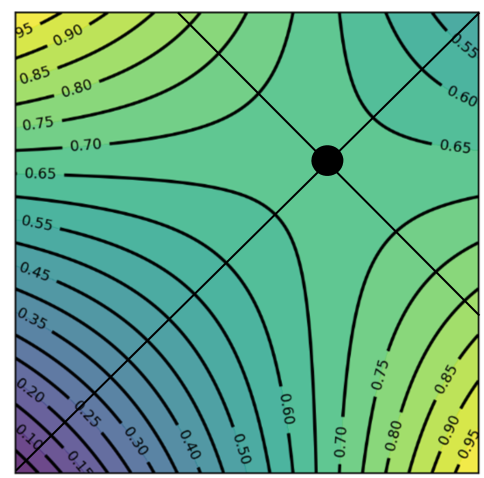
$$\begin{bmatrix} \frac{\partial^2 f}{\partial \alpha_1^2} & \frac{\partial^2 f}{\partial \alpha_1 \partial \alpha_2} \\ \frac{\partial^2 f}{\partial \alpha_2 \partial \alpha_1} & \frac{\partial^2 f}{\partial \alpha_2^2} \end{bmatrix} = \begin{bmatrix} 0 & a \\ a & 0 \end{bmatrix}$$

$$a = v_{00} + v_{11} - v_{10} - v_{01}$$

Eigenvalues and eigenvectors (Hessian is symmetric: always real)

$$\lambda_1 = -a \text{ and } \lambda_2 = a$$

$$v_1 = \begin{bmatrix} -1 \\ 1 \end{bmatrix}, v_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$



(here also: principal curvature magnitudes and directions
of this function's graph == surface embedded in 3D)

Bi-Linear Interpolation: Critical Points

Examine Hessian matrix at critical point (non-degenerate critical p.?, ...)

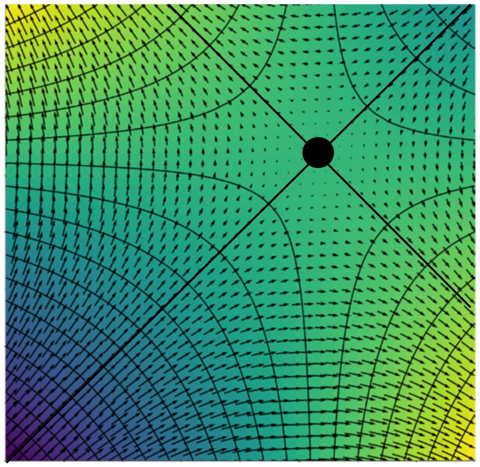
$$\begin{bmatrix} \frac{\partial^2 f}{\partial \alpha_1^2} & \frac{\partial^2 f}{\partial \alpha_1 \partial \alpha_2} \\ \frac{\partial^2 f}{\partial \alpha_2 \partial \alpha_1} & \frac{\partial^2 f}{\partial \alpha_2^2} \end{bmatrix} = \begin{bmatrix} 0 & a \\ a & 0 \end{bmatrix}$$

$$a = v_{00} + v_{11} - v_{10} - v_{01}$$

Eigenvalues and eigenvectors (Hessian is symmetric: always real)

$$\lambda_1 = -a \text{ and } \lambda_2 = a$$

$$v_1 = \begin{bmatrix} -1 \\ 1 \end{bmatrix}, v_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$



(here also: principal curvature magnitudes and directions
of this function's graph == surface embedded in 3D)

Bi-Linear Interpolation: Critical Points

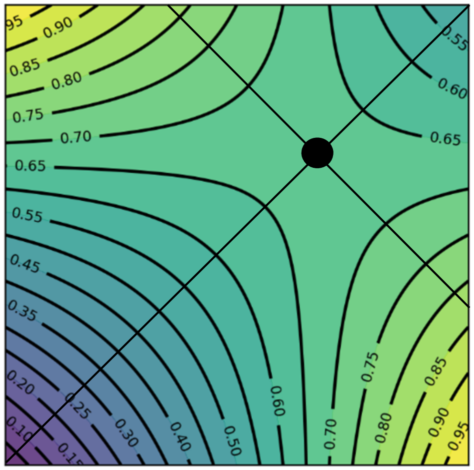
Examine Hessian matrix at critical point (non-degenerate critical p.?, ...)

$$\begin{bmatrix} \frac{\partial^2 f}{\partial \alpha_1^2} & \frac{\partial^2 f}{\partial \alpha_1 \partial \alpha_2} \\ \frac{\partial^2 f}{\partial \alpha_2 \partial \alpha_1} & \frac{\partial^2 f}{\partial \alpha_2^2} \end{bmatrix} = \begin{bmatrix} 0 & a \\ a & 0 \end{bmatrix} \qquad a = v_{00} + v_{11} - v_{10} - v_{01}$$

Eigenvalues and eigenvectors (Hessian is symmetric: always real)

$$\lambda_1 = -a \text{ and } \lambda_2 = a$$

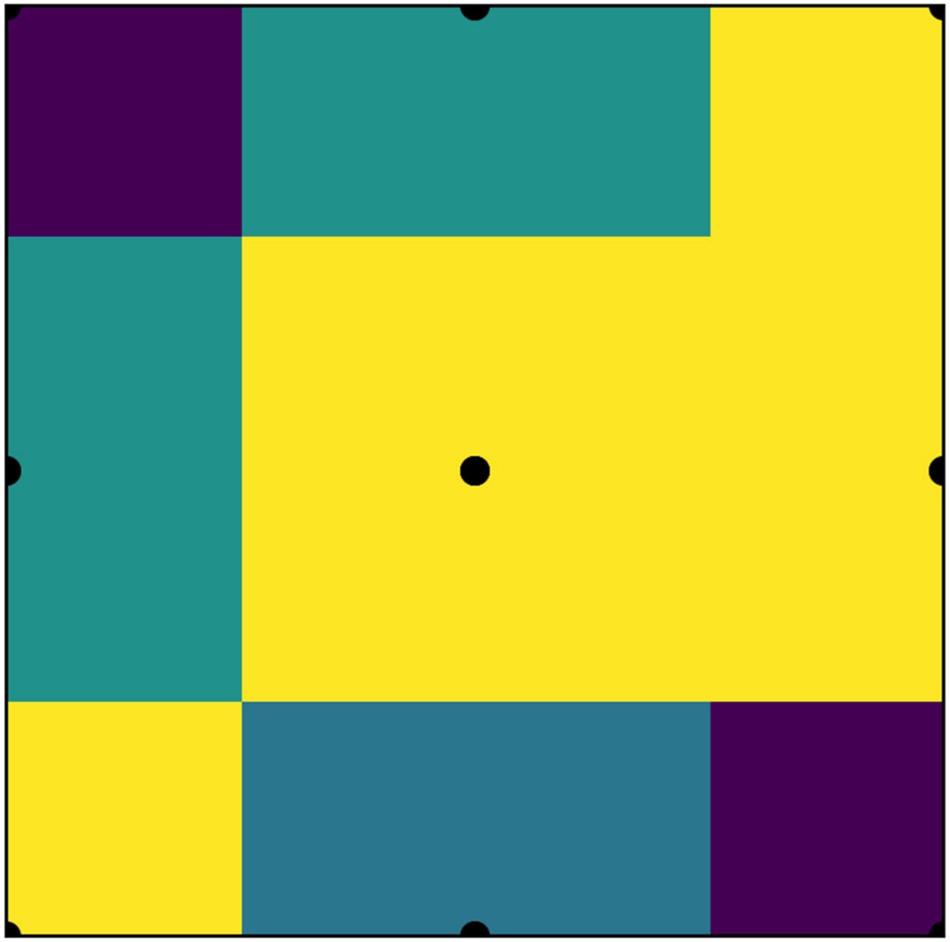
$$v_1 = \begin{bmatrix} -1 \\ 1 \end{bmatrix}, v_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$



degenerate means determinant = 0 (at least one eigenvalue = 0);
bi-linear is simple: a = 0 means degenerated to
linear anyway: no critical point at all! (except constant function)
(but with more than one cell: can have max or min at vertices)

Bi-Linear Interpolation: Comparisons

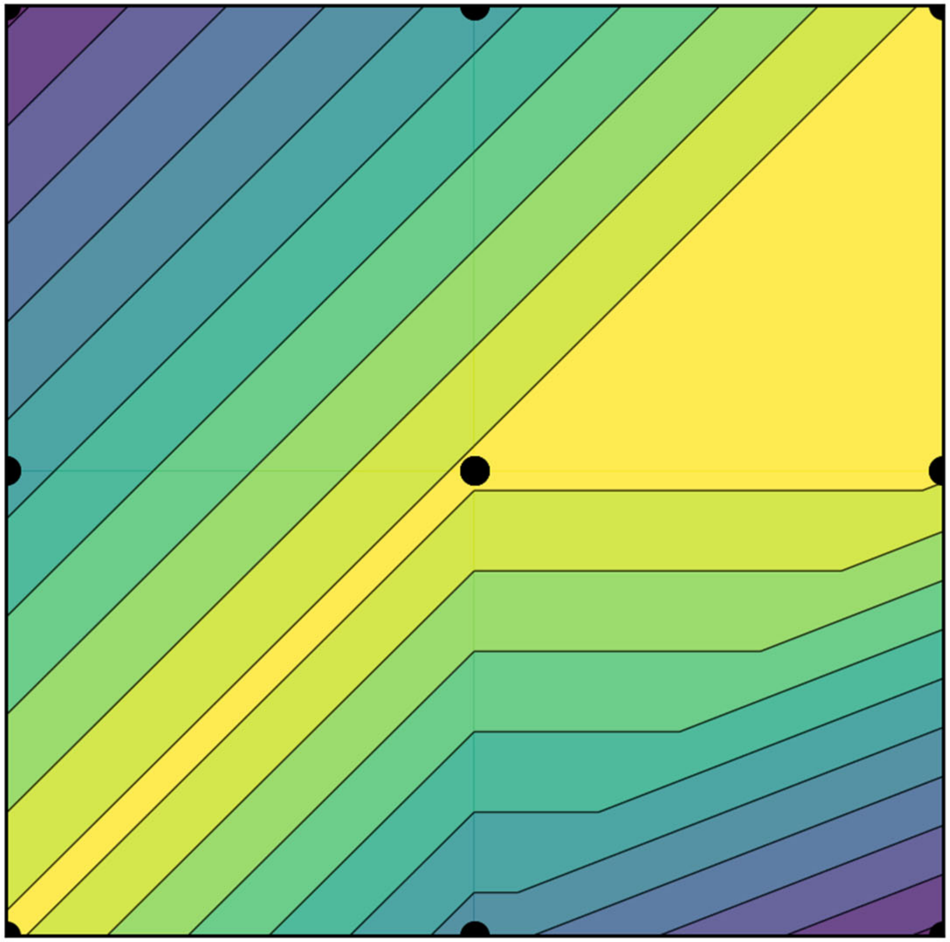
nearest-neighbor



Bi-Linear Interpolation: Comparisons

linear

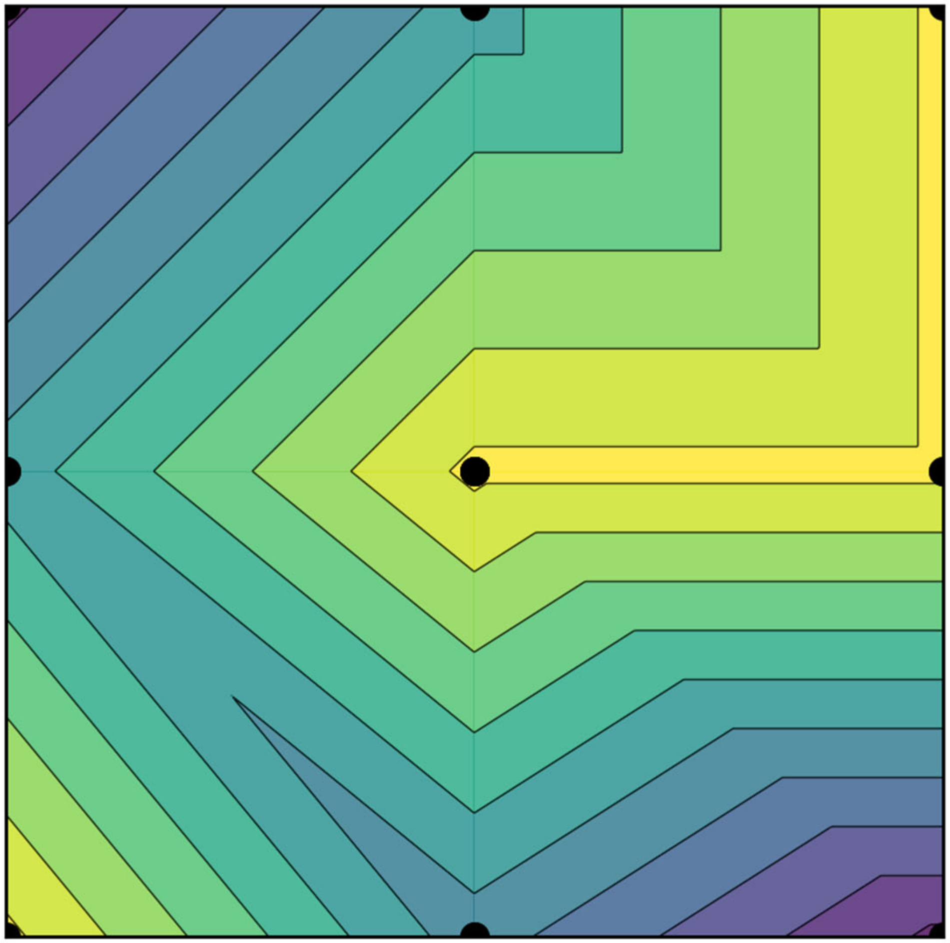
(2 triangles per quad;
diagonal:
bottom-left,
top-right)



Bi-Linear Interpolation: Comparisons

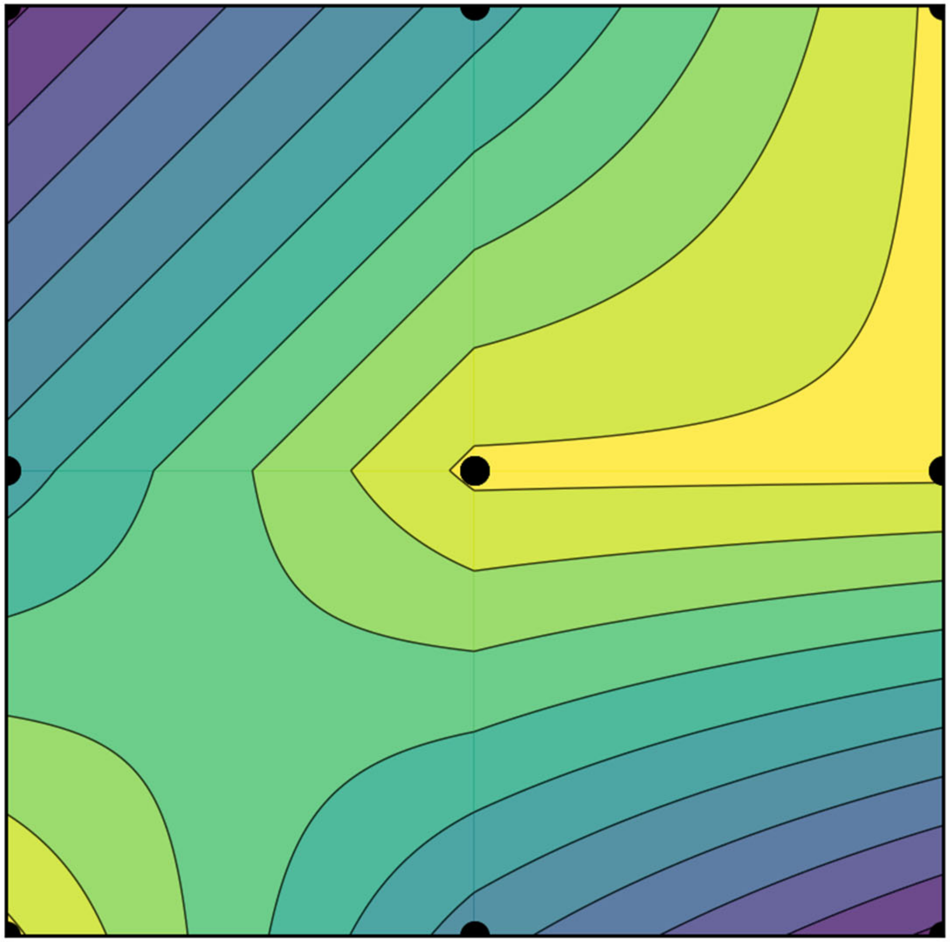
linear

(2 triangles per quad;
diagonal:
top-left,
bottom-right)



Bi-Linear Interpolation: Comparisons

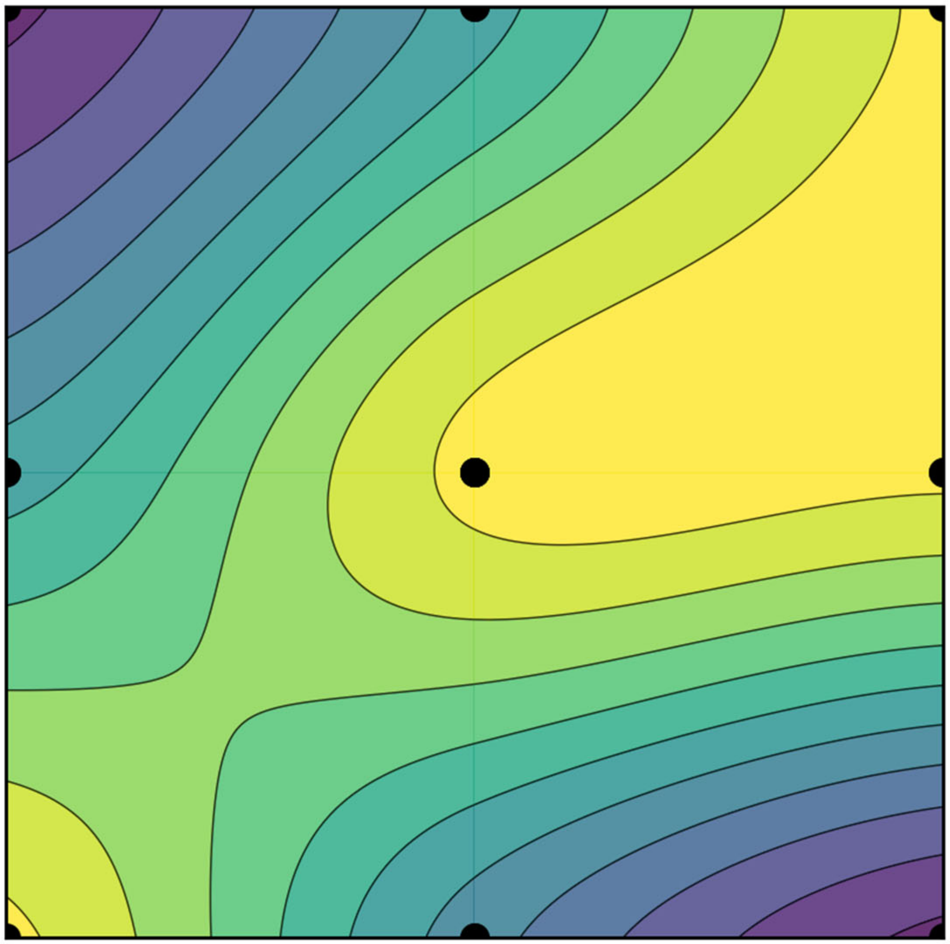
bi-linear



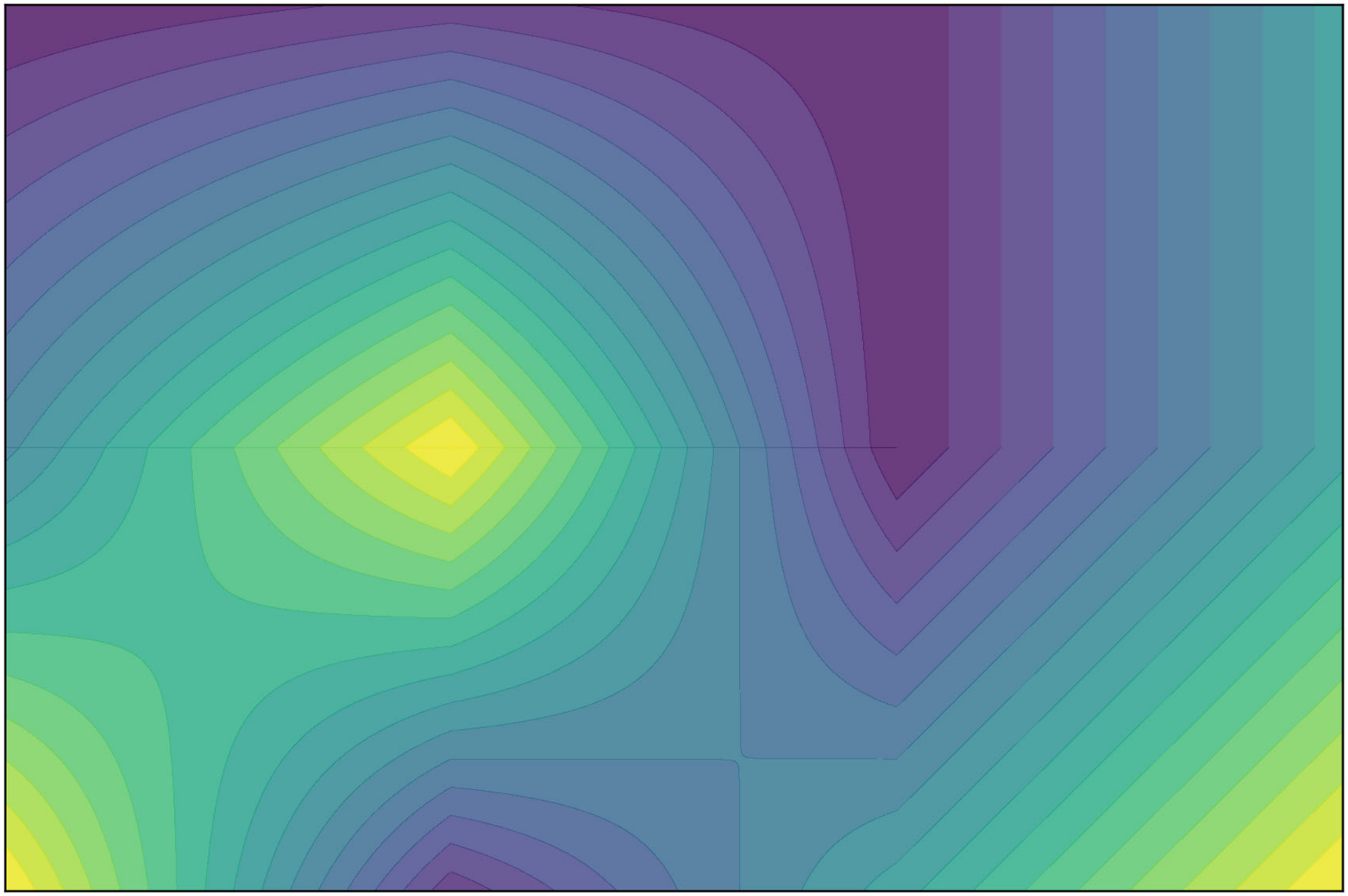
Bi-Linear Interpolation: Comparisons

bi-cubic

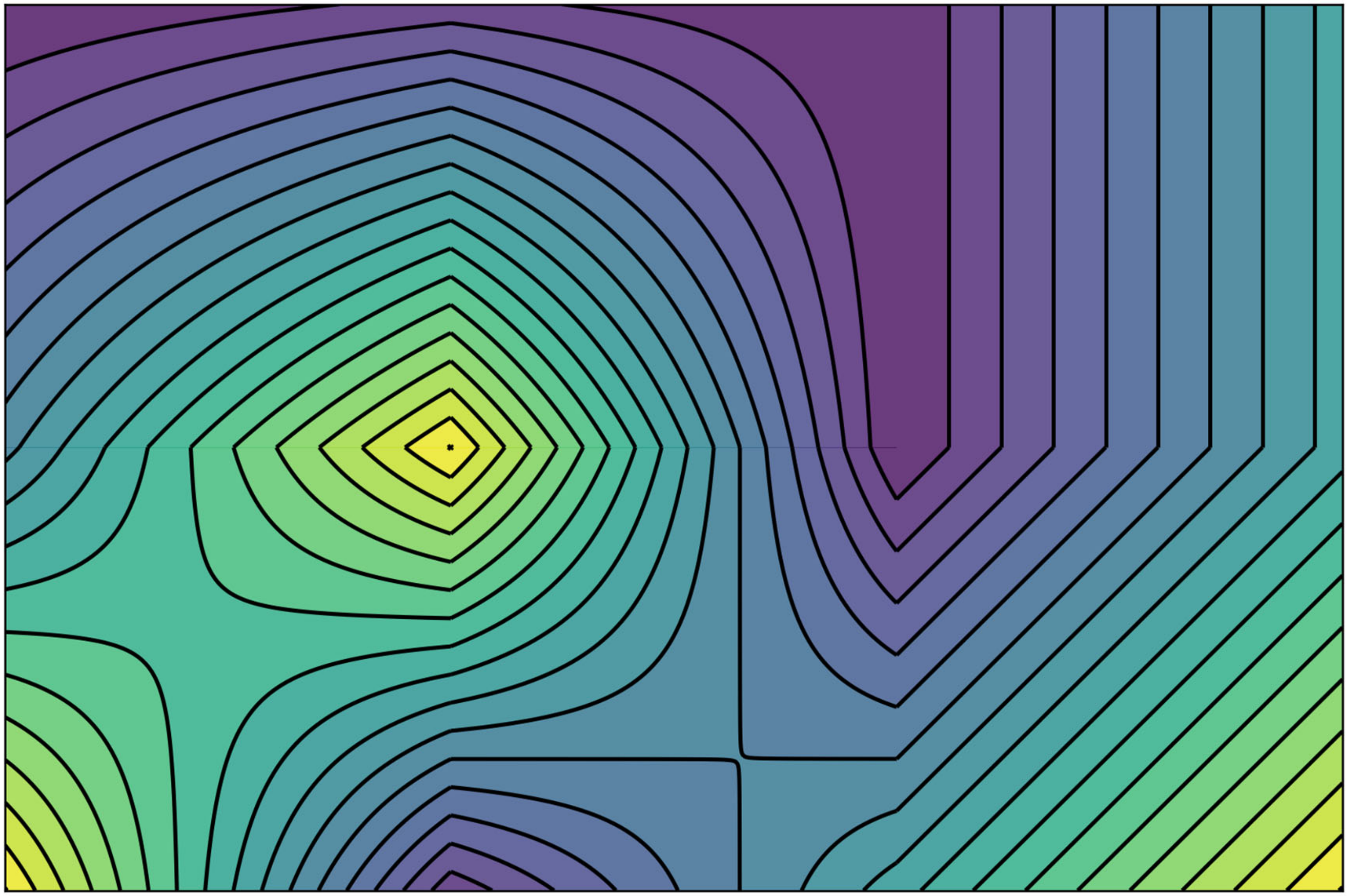
(Catmull-Rom spline)



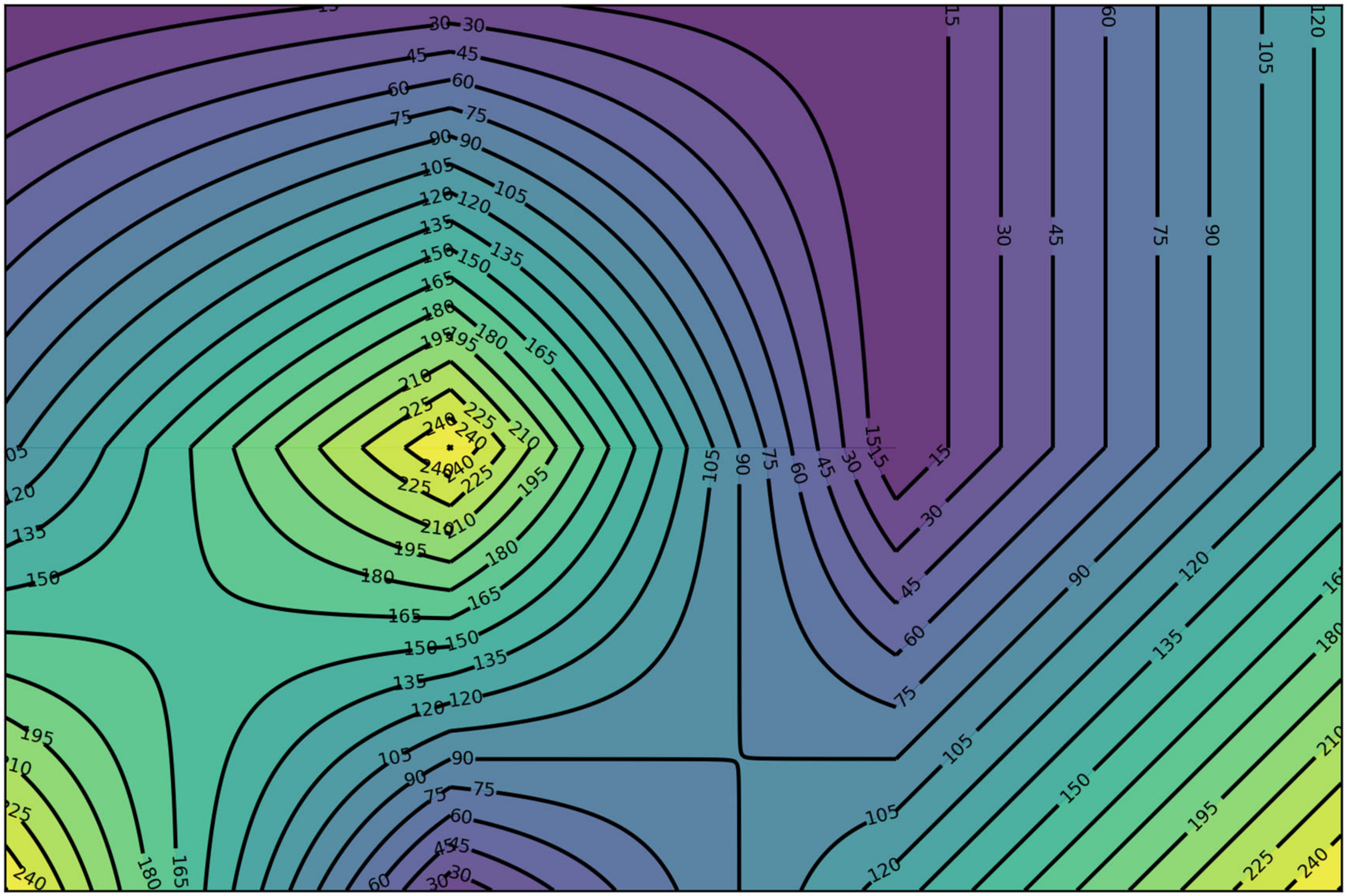
Piecewise Bi-Linear (Example: 3x2 Cells)



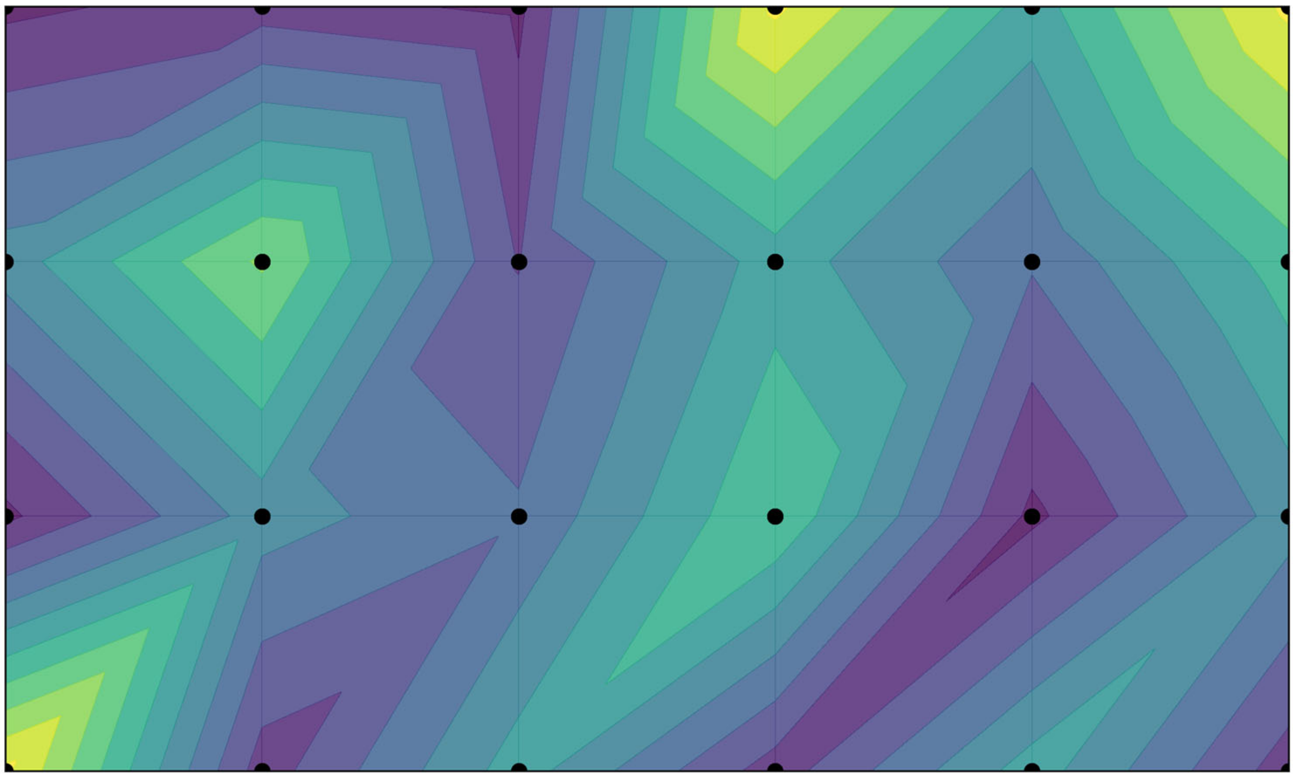
Piecewise Bi-Linear (Example: 3x2 Cells)



Piecewise Bi-Linear (Example: 3x2 Cells)

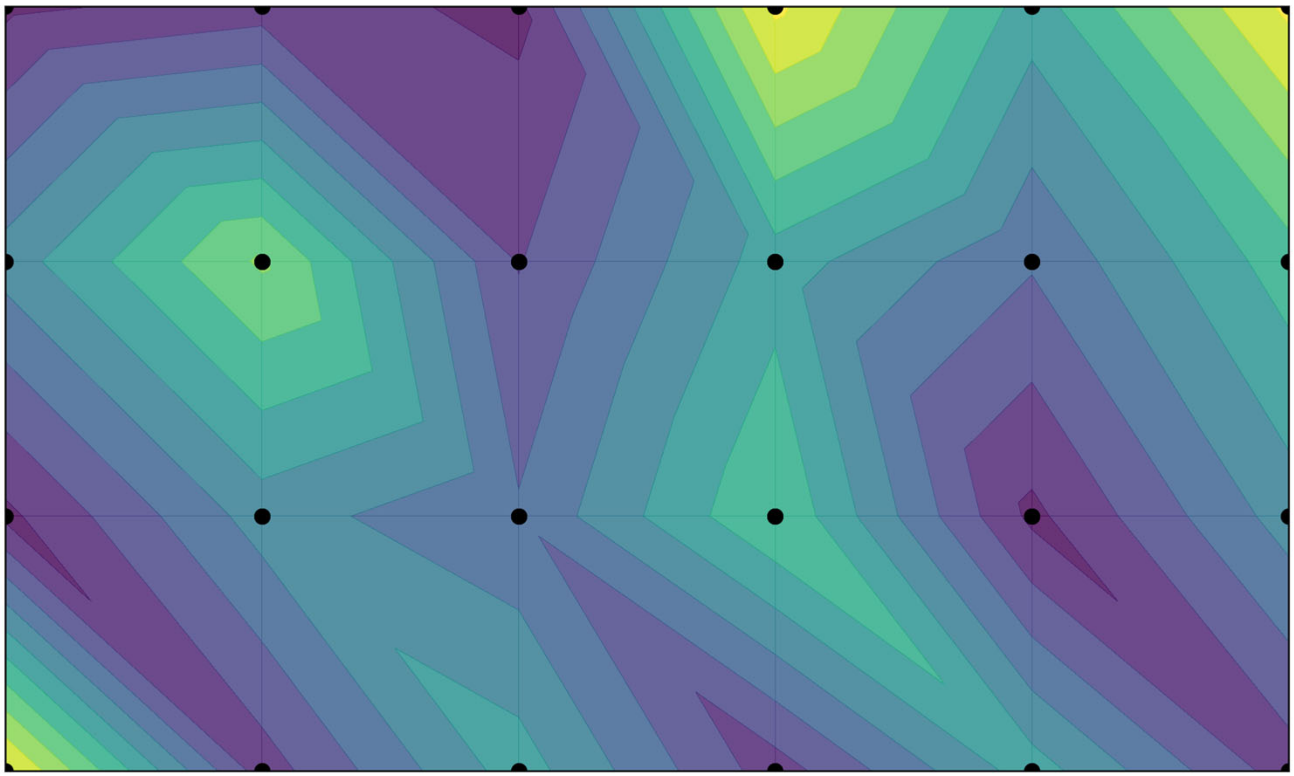


Bi-Linear Interpolation: Comparisons



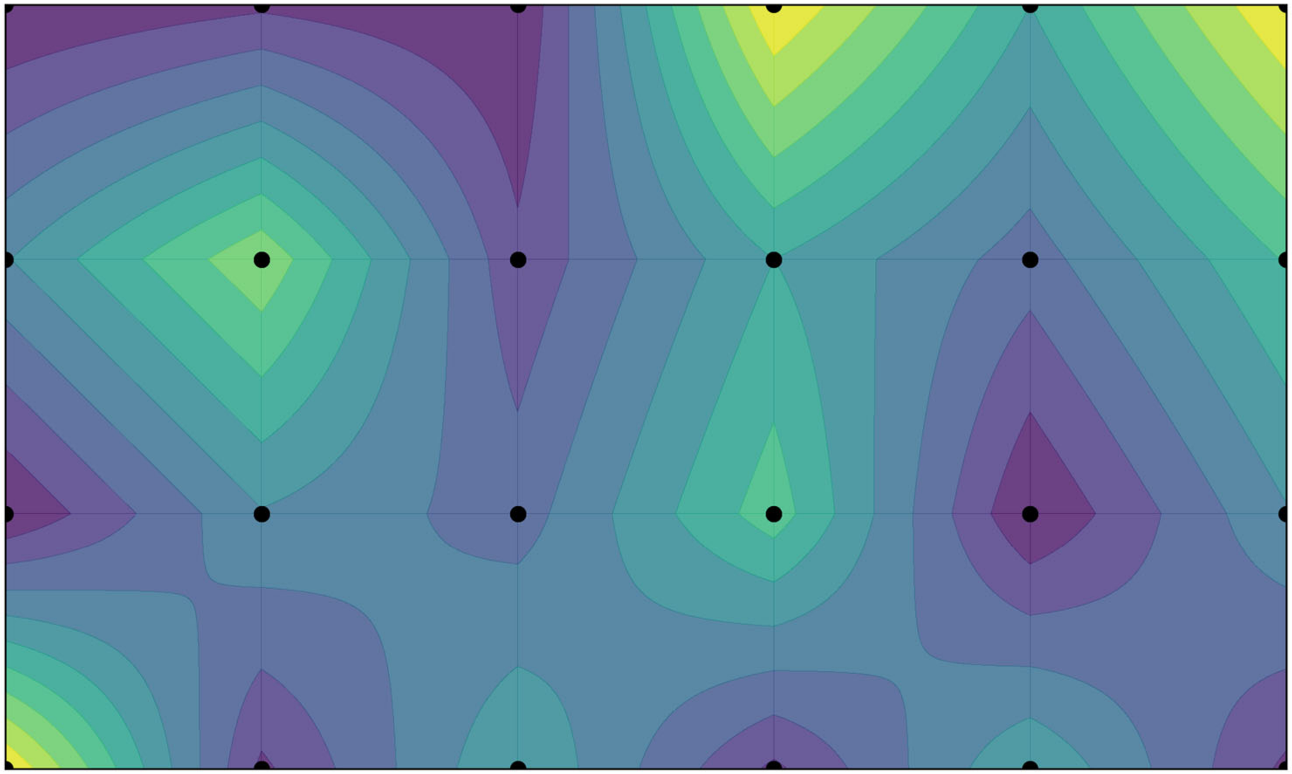
linear (diagonal 1)

Bi-Linear Interpolation: Comparisons



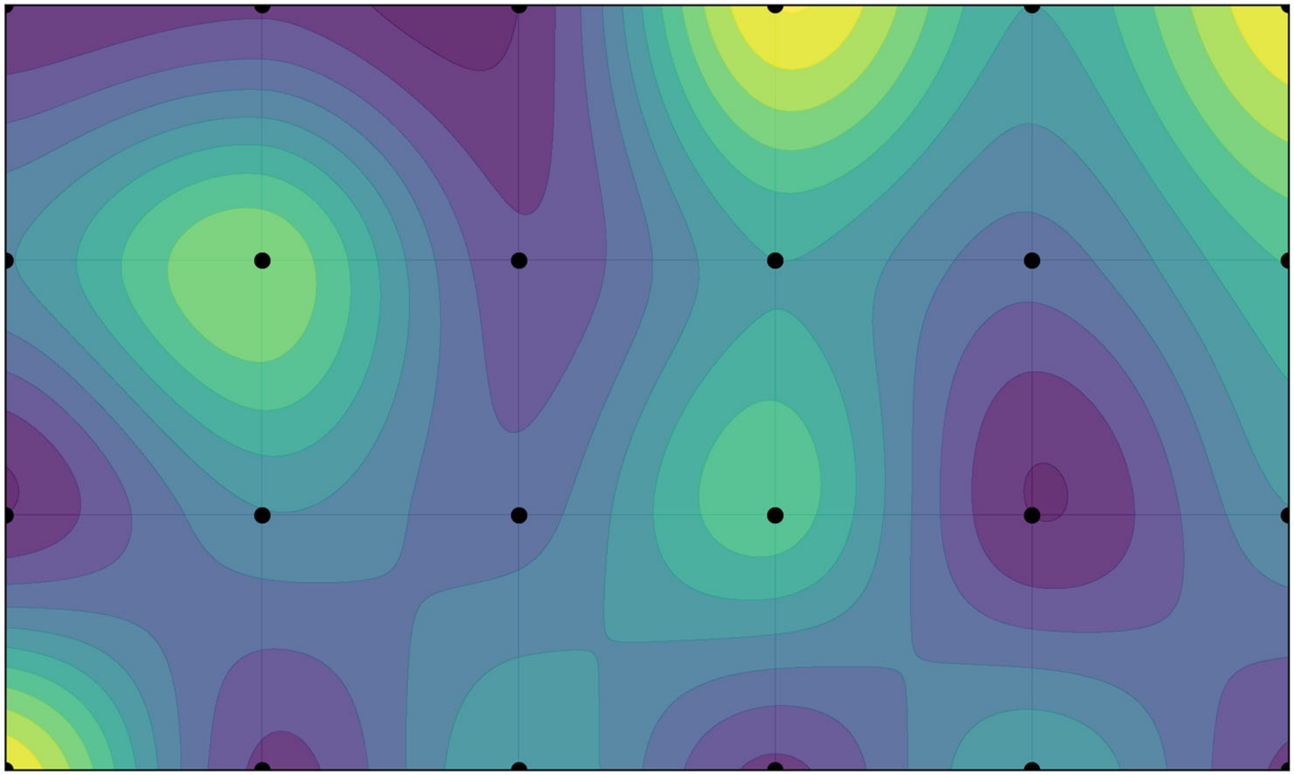
linear (diagonal 2)

Bi-Linear Interpolation: Comparisons



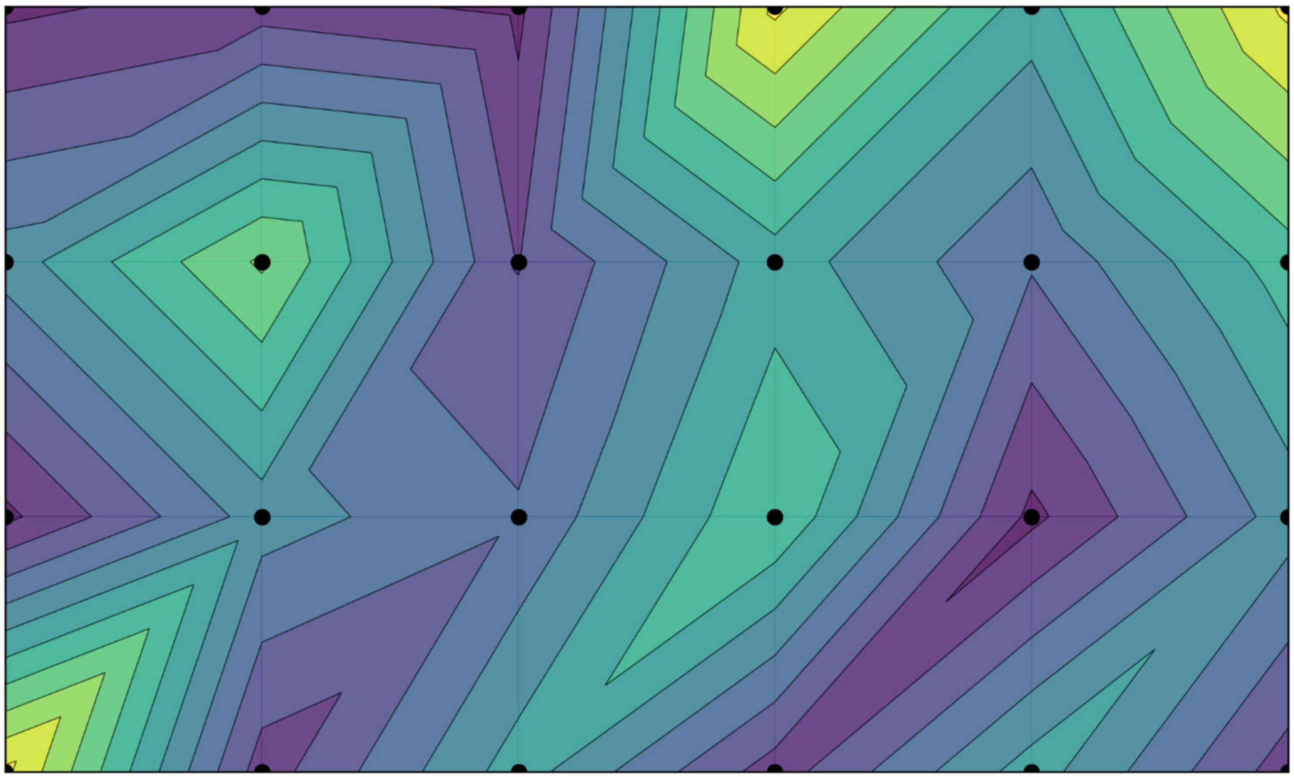
bi-linear (in 3D: tri-linear)

Bi-Linear Interpolation: Comparisons



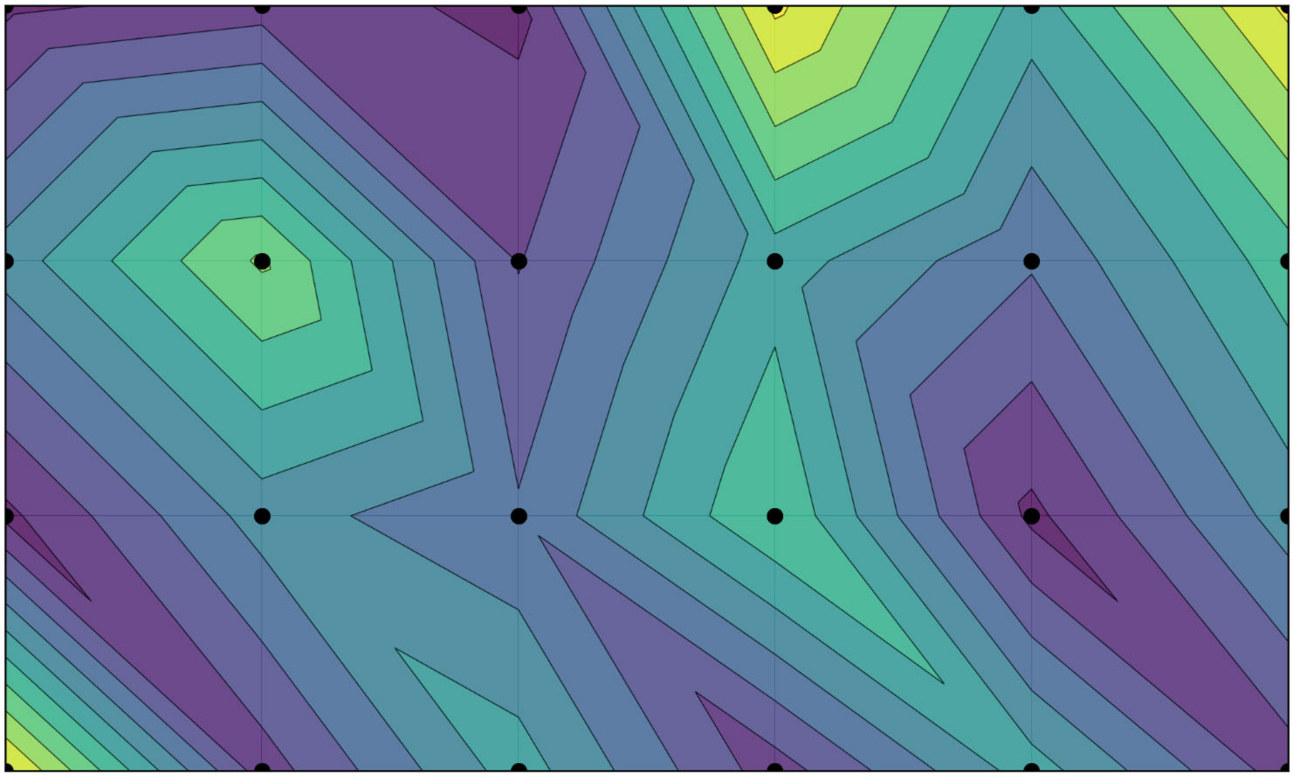
bi-cubic (in 3D: tri-cubic)

Bi-Linear Interpolation: Comparisons



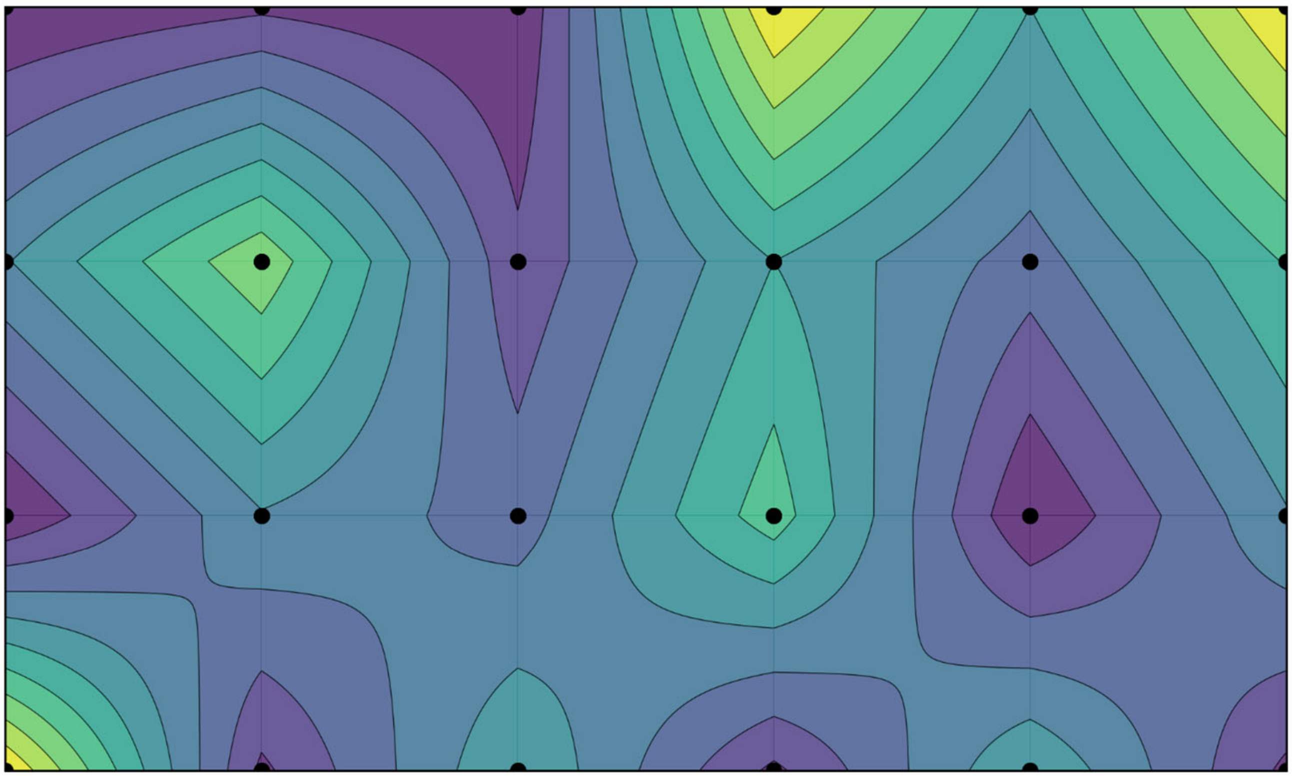
linear (diagonal 1)

Bi-Linear Interpolation: Comparisons



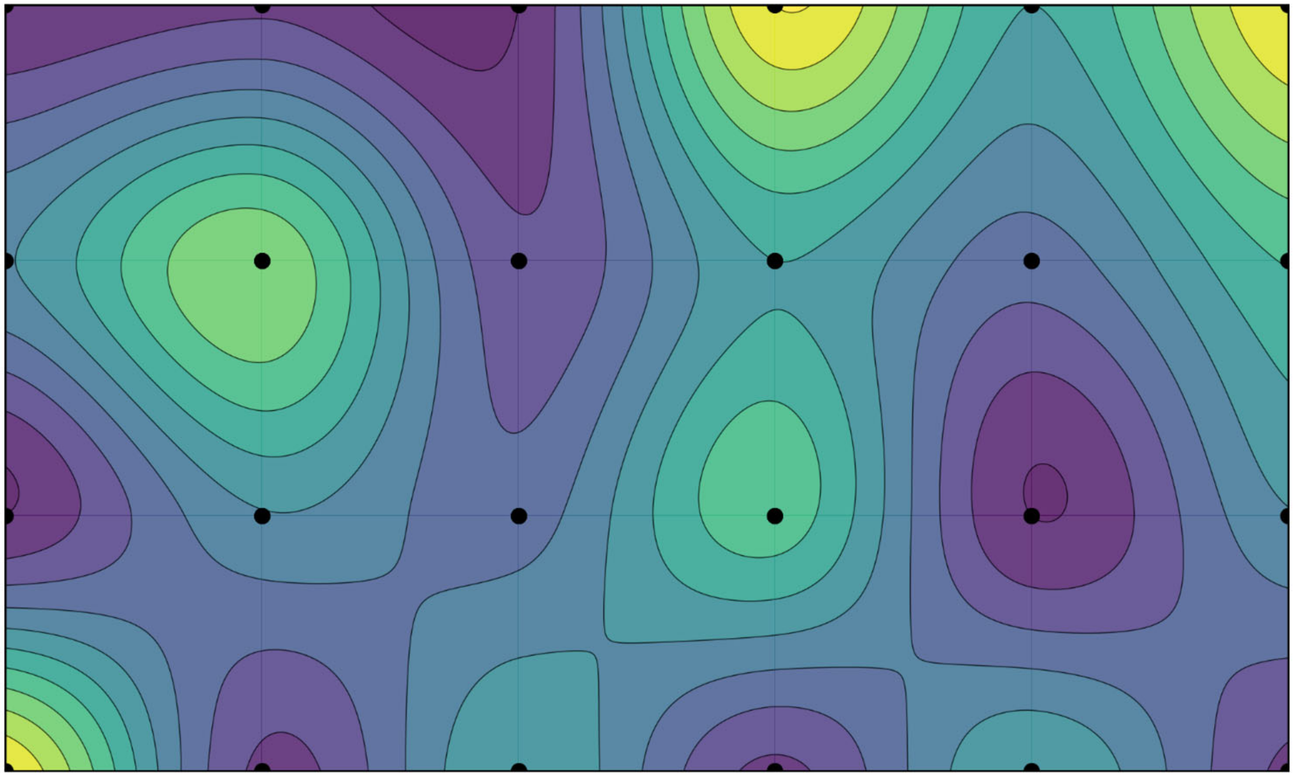
linear (diagonal 2)

Bi-Linear Interpolation: Comparisons



bi-linear (in 3D: tri-linear)

Bi-Linear Interpolation: Comparisons



bi-cubic (in 3D: tri-cubic)

Thank you.

Thanks for material

- Helwig Hauser
- Eduard Gröller
- Daniel Weiskopf
- Torsten Möller
- Ronny Peikert
- Philipp Muigg
- Christof Rezk-Salama