■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

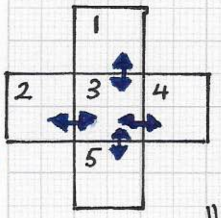
• Laplacian eigenfunctions: The 5-pixel convolution mask used in the presented examples is based on the specific indexing of the 5 pixels; the used connectivity of the 5 pixels with other pixels INSIDE and possibly OUTSIDE the mask; and the directionality of the edges between mask pixels.

For completeness, we discuss a slightly different 5-pixel mask (with mask weights that commonly occur in the literature).

(See p. 21, 9-24-2021.)

mask:

MASK HAS INSIDE "↔" CONNECTIONS ONLY!



$$\begin{array}{c} \\ 1\\2\\3\\4\\5 \end{array}\begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{bmatrix} 1 & 0 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ -1 & -1 & 4 & -1 & -1 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 0 & 1 \end{bmatrix}\end{array}$$

$$\lambda = (5,1,1,1,0)$$

$$e_1^n = (1,1,-4,1,1)^T / 2\sqrt{5}$$

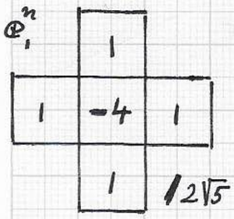
$$e_2^n = (-1,0,0,0,1)^T / \sqrt{2}$$

$$e_3^n = (-1,0,0,1,0)^T / \sqrt{2}$$

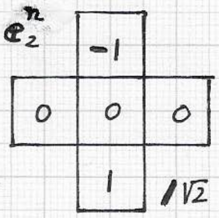
$$e_4^n = (-1,1,0,0,0)^T / \sqrt{2}$$

$$e_5^n = (1,1,1,1,1)^T / \sqrt{5}$$

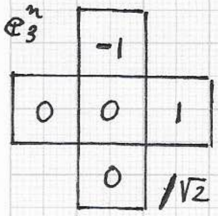
e_1^n



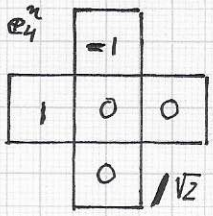
e_2^n



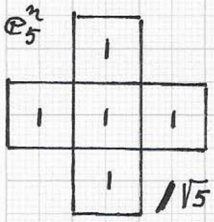
e_3^n



e_4^n



e_5^n



5-pixel mask: indexing, matrix of "connectivity", eigenvalues, normalized eigenvectors/-functions in tuple and convolution mask representations.

The 5×5 connectivity matrix shown in the figure (left, top) is based on the convention that an edge "going out" from a mask pixel i and connecting it to a neighbor pixel is "+"; an edge "coming into" mask pixel i from a neighbor pixel is "−". For example, mask pixel 3 has +4 outgoing edges. ... The set $\{e_i^n\}_{i=1}^5$ is a basis, and any piecewise-constant function f can be expressed as $f^* = \sum_{i=1}^5 c_i e_i^n$ over the 5-pixel mask region. The normal equations define the coefficient vector: $\mathbf{c} = \left[\langle e_i^n, e_j^n\rangle\right]^{-1}\left[\langle f, e_i^n\rangle\right]$.

$^*f = (m_1, m_2, m_3, m_4, m_5)^T$

...

■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions: … Computing the inner products leads to

$$\mathbb{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1/2 & 1/2 & 0 \\ 0 & 1/2 & 1 & 1/2 & 0 \\ 0 & 1/2 & 1/2 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} (m_1+m_2-4m_3+m_4+m_5)/2\sqrt{5} \\ (-m_1+m_5)/\sqrt{2} \\ (-m_1+m_4)/\sqrt{2} \\ (-m_1+m_2)/\sqrt{2} \\ (m_1+m_2+m_3+m_4+m_5)/\sqrt{5} \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 3/2 & -1/2 & -1/2 & 0 \\ 0 & -1/2 & 3/2 & -1/2 & 0 \\ 0 & -1/2 & -1/2 & 3/2 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} (m_1+m_2-4m_3+m_4+m_5)/2\sqrt{5} \\ (-m_1+m_5)/\sqrt{2} \\ (-m_1+m_4)/\sqrt{2} \\ (-m_1+m_2)/\sqrt{2} \\ (m_1+m_2+m_3+m_4+m_5)/\sqrt{5} \end{bmatrix}$$

$$= \begin{bmatrix} (m_1+m_2-4m_3+m_4+m_5)/2\sqrt{5} \\ (-m_1-m_2-m_4+3m_5)/2\sqrt{2} \\ (-m_1-m_2+3m_4-m_5)/2\sqrt{2} \\ (-m_1+3m_2-m_4-m_5)/2\sqrt{2} \\ (m_1+m_2+m_3+m_4+m_5)/\sqrt{5} \end{bmatrix}.$$

c_1: $\begin{array}{|c|c|c|} \hline & 1 & \\ \hline 1 & -4 & 1 \\ \hline & 1 & \\ \hline \end{array} \;/2\sqrt{5}$ c_2: $\begin{array}{|c|c|c|} \hline & -1 & \\ \hline -1 & 0 & -1 \\ \hline & 3 & \\ \hline \end{array} \;/2\sqrt{2}$

c_3: $\begin{array}{|c|c|c|} \hline & -1 & \\ \hline -1 & 0 & 3 \\ \hline & -1 & \\ \hline \end{array} \;/2\sqrt{2}$ c_4: $\begin{array}{|c|c|c|} \hline & -1 & \\ \hline 3 & 0 & -1 \\ \hline & -1 & \\ \hline \end{array} \;/2\sqrt{2}$

c_5: $\begin{array}{|c|c|c|} \hline & 1 & \\ \hline 1 & 1 & 1 \\ \hline & 1 & \\ \hline \end{array} \;/\sqrt{5}$

5-pixel mask and "weights" that must be used to compute a coefficient c_i, $i=1\ldots5$.

The computation of the coefficients c_i is summarized in the figure (left). When the eigenfunction-based expansion $\mathbb{f} = \sum_{i=1}^{5} c_i \mathbb{e}_i^n$ is computed for a 5-pixel region, the mask's domain, then the pixels' mass values m_i will have to be combined linearly using the pixel weights shown in the figure.

Note. The eigenvalue tuple is $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5) = (5, 1, 1, 1, 0)$. Eigenvalue $\lambda_5 = 0$ corresponds to eigenfrequency $\sqrt{\lambda_5} = 0$, which implies that the term $c_5 \mathbb{e}_5^n$ represents the average mass value for the 5-pixel mask region. …

• **The 3 different eigenvalues λ_1, $\lambda_2=\lambda_3=\lambda_4$ and λ_5 define eigenfunctions capturing 3 levels of scale/detail: The term $c_5 \mathbb{e}_5^n$ represents the average; the sum $c_2\mathbb{e}_2^n + c_3\mathbb{e}_3^n + c_4\mathbb{e}_4^n$ is the "first level of detail"; the term $c_1\mathbb{e}_1^n$ represents the "second level of detail."**

!

■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions: ... Eigenvalues λ_2, λ_3 and λ_4 have the same value ($\lambda_2 = \lambda_3 = \lambda_4 = 1$), corresponding to the eigenfrequency $\sqrt{1} = 1$, which implies that the sum $\sum_{i=2}^{4} c_i \mathbb{e}_i^n$ represents another level of detail. (The value 1 is an eigenvalue of multiplity 3, and the 3 eigenfunctions $\mathbb{e}_2^n$, $\mathbb{e}_3^n$ and $\mathbb{e}_4^n$ associated with the eigenvalues λ_2, λ_3 and λ_4 are NOT unique and NOT mutually orthogonal to each other; these 3 eigenfunctions can be defined in several ways, as long as they are satisfying linear independence.) Eigenvalue $\lambda_1 = 5$ corresponds to eigenfrequency $\sqrt{\lambda_1} = \sqrt{5}$, which implies that the term $c_1 \mathbb{e}_1^n$ represents the "finest level of detail" of $\mathbb{f}$'s expansion.

Since a segment is given as a finite set of pixels/voxels, one must adjust this multi-scale local analysis approach to process the segment's boundary. One must employ a boundary-specific convolution mask that does not "stick out," does not include pixels/voxels outside the segment's interior. ...

• Approximation hierarchy of function $\mathbb{f}$:

REFINEMENT ↓

- Approximation 0: $\mathbb{a}_0 = c_5 \mathbb{e}_5^n$
- Approximation 1: $\mathbb{a}_1 = \mathbb{a}_0 + \sum_{i=2}^{4} c_i \mathbb{e}_i^n$
- Approximation 2: $\mathbb{a}_2 = \mathbb{a}_1 + c_1 \mathbb{e}_1^n$

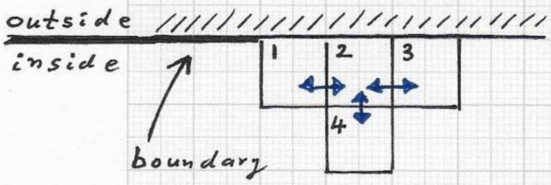
Approximation 2, $\mathbb{a}_2$, is a lossless representation of $\mathbb{f}$ in the eigenfunction basis of the 5-pixel convolution mask.

⇒ "Function $\mathbb{f}$ is expanded in a multi-scale fashion via the '3-levels-of-detail basis' $\{\mathbb{e}_i^n\}$."

!

■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• **Laplacian eigenfunctions:** ... We present possible convolution masks for analyzing the boundary of a 2D segment. Depending on the geometry and given interior pixels of a segment, one can use 4-pixel, 3-pixel, 2-pixel and, ultimately, 1-pixel masks for the local (multi-scale) analysis of those parts of a segment where the 5-pixel convolution mask "does not fit."



$$\begin{array}{c|c} & \begin{array}{cccc}1&2&3&4\end{array}\\ \hline \begin{array}{c}1\\2\\3\\4\end{array} & \begin{bmatrix} 1 & -1 & 0 & 0\\ -1 & 3 & -1 & -1\\ 0 & -1 & 1 & 0\\ 0 & -1 & 0 & 1\end{bmatrix}\end{array} \qquad \lambda = (4,1,1,0) \qquad \lambda_2 = \lambda_3$$

$$e_1^n = (1,-3,1,1)^T / 2\sqrt{3}$$
$$e_2^n = (-1,0,0,1)^T / \sqrt{2}$$
$$e_3^n = (-1,0,1,0)^T / \sqrt{2}$$
$$e_4^n = (1,1,1,1)^T / 2$$
$$\Rightarrow f = \sum_{i=1}^{4} c_i e_i^n$$
$$\Rightarrow c = \ldots = \begin{bmatrix} (m_1 - 3m_2 + m_3 + m_4)/2\sqrt{2}\\ (-2m_1 - 2m_3 + 4m_4)/3\sqrt{2}\\ (-2m_1 + 4m_3 - 2m_4)/3\sqrt{2}\\ (m_1 + m_2 + m_3 + m_4)/2 \end{bmatrix}$$

c_1

1	-3	1
	1	

$/2\sqrt{2}$

c_2

-2	0	-2
	4	

$/3\sqrt{2}$

c_3

-2	0	4
	-2	

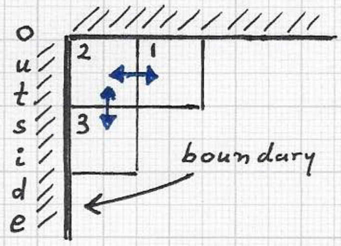
$/3\sqrt{2}$

c_4

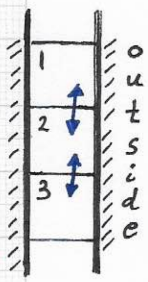
1	1	1
	1	

$/2$

4-pixel mask on boundary: indexing, connectivity matrix, eigenvalues, normalized eigenvectors/-functions, coefficients c_i and final convolution mask.



$$\begin{array}{c|c} & \begin{array}{ccc}1&2&3\end{array}\\ \hline \begin{array}{c}1\\2\\3\end{array} & \begin{bmatrix} 1 & -1 & 0\\ -1 & 2 & -1\\ 0 & -1 & 1\end{bmatrix}\end{array} \qquad \lambda = (3,1,0)$$



$$e_1^n = (1,-2,1)^T / \sqrt{6}$$
$$e_2^n = (-1,0,1)^T / \sqrt{2}$$
$$e_3^n = (1,1,1)^T / \sqrt{3}$$
$$\Rightarrow f = \sum_{i=1}^{3} c_i e_i^n$$
$$\Rightarrow c = \begin{bmatrix} (m_1 - 2m_2 + m_3)/\sqrt{6}\\ (-m_1 + m_3)/\sqrt{2}\\ (m_1 + m_2 + m_3)/\sqrt{3} \end{bmatrix}$$

c_1

-2	1
1	

$/\sqrt{6}$

c_2

0	-1
1	

$/\sqrt{2}$

c_3

1	1
1	

$/\sqrt{3}$

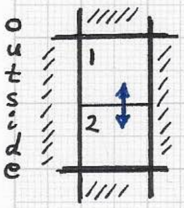
3-pixel mask on boundary: used in the "corner case" (left) and "1-pixel thickness case" (right). ...

Bernd Hamann 12/4/2021

Stratovan

■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions: ... When using the 5-pixel convolution mask one must also consider and use the mask's possible "sub-masks," i.e., the described 4-pixel, 3-pixel, 2-pixel and 1-pixel masks that represent the combinatorially possible "sub-mask symmetry classes." It is necessary to apply these smaller "sub-masks" to a segment's fragment regions where the complete 5-pixel mask is "to large." As a consequence, the smaller "sub-masks" can only capture and represent a smaller number of scales. The need to apply these masks/sub-masks to a segment's multiple fragments is evident by considering this example:



$$\begin{array}{c c} & \begin{array}{cc} 1 & 2 \end{array} \\ \begin{array}{c} 1 \\ 2 \end{array} & \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \end{array}$$

$$\boldsymbol{\lambda} = (2, 0)$$

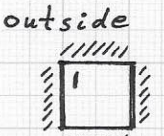
$$\mathbf{e}_1^n = (-1, 1)^T / \sqrt{2}$$

$$\mathbf{e}_2^n = (1, 1)^T / \sqrt{2}$$

$$\Rightarrow \mathbf{f} = \sum_{i=1}^{2} c_i \mathbf{e}_i^n$$

$$\Rightarrow \mathbf{C} = \begin{bmatrix} (-m_1 + m_2)/\sqrt{2} \\ (m_1 + m_2)/\sqrt{2} \end{bmatrix}$$

c_1: $\begin{array}{|c|}\hline -1 \\ \hline 1 \\ \hline \end{array} / \sqrt{2}$ c_2: $\begin{array}{|c|}\hline 1 \\ \hline 1 \\ \hline \end{array} / \sqrt{2}$



$$\begin{array}{cc} & 1 \\ 1 & [0] \end{array} \qquad \boldsymbol{\lambda} = (0)$$

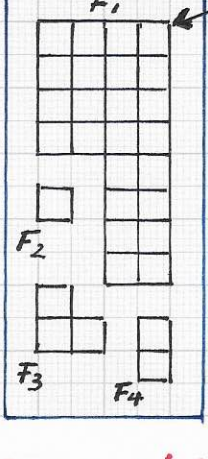
$$\mathbf{e}_1^n = (1)^T$$

$$\Rightarrow \mathbf{f} = c_1 \mathbf{e}_1^n$$

$$\Rightarrow \mathbf{C} = [\, m_1 \,]$$

c_1: $\begin{array}{|c|}\hline 1 \\ \hline \end{array}$

2-pixel and 1-pixel masks: used for segments – or fragments of segments – consisting of only two or one pixels.



boundary of 2D image/scan

1 segment consisting of 4 fragments F_1, F_2, F_3 and F_4

Fragment F_1 consists of pixels that can be analyzed with the 5-pixel, 4-pixel and 3-pixel masks.

Fragments F_2, F_3 and F_4 must be analyzed with the 1-pixel, 3-pixel and 2-pixel masks, respectively.

For overall segment analysis, one must meaningfully 'combine' eigenvalue-based information related to $\boldsymbol{\lambda}$-tuples.

~ BH