Theory of Ridges

David Eberly, Geometric Tools, Redmond WA 98052
https://www.geometrictools.com/

This work is licensed under the Creative Commons Attribution 4.0 International License. To view a copy of this license, visit http://creativecommons.org/licenses/by/4.0/ or send a letter to Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.

Created: January 1, 1995
Last Modified: March 2, 2008

Contents

1 Ridges in Euclidean Geometry

1.1 Restricted Local Extrema

We now take a closer look at the definition for local extrema of a function $f \in C^2(\mathbb{R}^n, \mathbb{R})$. *Ridges* will be a generalization of local maxima whereby the test for maximality of $f(x)$ is made in a restricted neighborhood of x. A similar concept of *courses* generalizes local minima, but since local minima of f are local maxima of $-f$, it is sufficient to study only the concept of ridge.

Recall that a point x is a critical point for f if $v^{\mathsf{T}} Df(x) = 0$ for all directions v (equivalently, $Df(x) = 0$). At a critical point, $f(x)$ is a strict local maximum if $v^{\mathsf{T}} D^2 f(x) v < 0$ for all directions v (equivalently, $D^2 f(x)$ is negative definite) and in this case x is said to be a local maximum point. Let v_1 through v_n be linearly independent directions and define the $n \times n$ matrix $V = [v_1 \cdots v_n]$ whose columns are the given directions. The test for local maximum points becomes $V^{\mathsf{T}} Df(x) = 0$ and $V^{\mathsf{T}} D^2 f(x) V < 0$. In effect we are exploring the function values in a neighborhood of x which is locally represented by the affine space $x + \langle V \rangle = x + \mathbb{R}^n$. The function to be tested, $\phi : \mathbb{R}^n \to \mathbb{R}$, is defined by $\phi(s) = f(x + Vs)$. A local maximum point x occurs when $D\phi(0) = V^{\mathsf{T}} Df(x) = 0$ and $D^2\phi(0) = V^{\mathsf{T}} D^2 f(x) V < 0$.

Rather than testing for a local maximum in all n directions, it is possible to restrict attention to only $n - d$ directions for some d with $0 \le d < n$. Let the directions be denoted v_1 through v_{n-d} and define the $n \times (n-d)$ matrix $V = [v_1 \cdots v_{n-d}]$. Consider f restricted to the affine space $x + \langle V \rangle$. Define $\phi : \mathbb{R}^{n-d} \to \mathbb{R}$ by $\phi(s) = f(x + Vs)$. A point x is a *restricted local maximum point of type d relative to V* if $f(x)$ is a local maximum in the affine space $x + \langle V \rangle$. The test for such a point is symbolically just as in the case $d = 0$: $D\phi(0) = V^{\mathsf{T}} Df(x) = 0$ and $D^2\phi(0) = V^{\mathsf{T}} D^2 f(x) V < 0$. The definition is summarized below.

Definition 1 (Restricted Local Maximum). *Let $f \in C^2(\mathbb{R}^n, \mathbb{R})$. For a given d with $0 \le d < n$ and an $n \times (n-d)$ matrix V of rank $n-d$, the point x is a* restricted local maximum point of type d with respect to V *if $V^{\mathsf{T}} Df(x) = 0$ and $V^{\mathsf{T}} D^2 f(x) V < 0$.*

Note that $V^{\mathsf{T}} Df(x) = 0$ is a system of $n - d$ equations in n unknowns which, by the Implicit Function Theorem, typically has solutions which lie on d-dimensional manifolds. The terminology *type d* in the definition is used to reflect the expected dimensionality of the solution set. Also note that local maxima are just special cases of this definition when $d = 0$. Such points typically are isolated and are labeled as 0-dimensional structures.

In the case $d = 0$, any choice of V yields the same extrema. When $d > 0$, there are many choices for V. In this book we will concentrate on only a few choices for both d and V. Generally the choices will depend on the needs of a particular application.

Example 1. Let $n = 2$, $d = 1$, and $f(x, y) = x^2 y$. If $v_1 = (1, 0)$, then $V^{\mathsf{T}} Df = 2xy$ and $V^{\mathsf{T}} D^2 f V = 2y$. The restricted local maxima occur when $x = 0$ and $y < 0$. If $v_1 = (1, 1)$, then $V^{\mathsf{T}} Df = x(x + 2y)$ and $V^{\mathsf{T}} D^2 f V = 4x + 2y$. The restricted local maxima occur when $x = 0$ with $y < 0$ or $x + 2y = 0$ with $y > 0$.

In the previous two examples the direction vectors are constant. The search for restricted local maxima in these cases is equivalent to looking for local maxima of the function along straight lines in its domain. The direction vectors can depend on the point of application. For example, if $v_1 = (y, -x)$, then $V^{\mathsf{T}} Df = x(2y^2 - x^2)$ and $V^{\mathsf{T}} D^2 f V = 2y(y^2 - 2x^2)$. The restricted local maxima occur when $x = 0$ with $y < 0$ or

$x = \pm\sqrt{2}y$ with $y > 0$. The search for restricted local maxima is equivalent to looking for local maxima of the function along circles centered at the origin.

FIGURES GO HERE, ONE FIGURE FOR EACH OF THE THREE EXAMPLES.

1.2 Height Ridge Definition

The choice of V for the d-dimensional height ridge definition is based on convexity/concavity of the graph of f. The idea is that a ridge point on the graph is a place where f has a restricted local maximum in $n-d$ directions for which the graph of f is concave. The motivation comes from the case $n = 2$ where the graph is thought of as mountain terrain. Peaks of the terrain ($d = 0$) can be simply characterized by those points for which the function has a local maximum. As indicated in the introduction chapter, ridges of the terrain ($d = 1$) can be characterized in a variety of ways. The height ridge definition uses the heuristic that a ridge point should be a point for which the function has a local maximum in the direction for which the graph has the largest concavity. Figure (???) shows the graph of a function on which such points are marked.

FIGURE GOES HERE. (Gaussian graph with ridge shown, from dissertation)

Since eigenvalues of the $D^2 f$ measure convexity and concavity in the corresponding eigendirections, a natural definition for ridges is given below.

Definition 2 (Height Ridge Definition). *Let $f \in C^2(\mathbb{R}^n, \mathbb{R})$. Let λ_i and v_i, $1 \le i \le n$, be the eigenvalues and unit-length eigenvectors of $D^2 f$ with $\lambda_1 \le \cdots \le \lambda_n$. A point x is a* d-dimensional ridge point *if x is a restricted local maximum point of type d with respect to $V = [v_1 \cdots v_{n-d}]$. Since $V^{\mathsf{T}} D^2 f V =$* $\mathrm{Diag}\{\lambda_1, \ldots, \lambda_{n-d}\}$ *and since the eigenvalues are ordered, the test for a ridge point reduces to $V^{\mathsf{T}} \nabla f(x) = 0$ and $\lambda_{n-d}(x) < 0$.*

The definition can be modified to allow x to be a restricted local maximum point with respect to any subset of $n - d$ eigenvectors for which the eigenvalues are negative, but we do not consider that generalization in this book. An equivalent formulation for the more general definition in the case $n = 2$ and $d = 1$ is found in (cite LINDEBERG book).

Example 2. Consider $f(x, y) = x^2 y$. The eigenvalues of $D^2 f$ are $\lambda_1 = y - \sqrt{4x^2 + y^2}$ and $\lambda_2 = y + \sqrt{4x^2 + y^2}$. Observe that $\lambda_1(x, y) \le 0 \le \lambda_2(x, y)$ for all (x, y). Corresponding (non unit length) eigenvectors are

$$v_1 = \left\{ \begin{array}{ll} (-y + R, -2x), & y \le 0 \\ (2x, -y - R), & y \ge 0 \end{array} \right\} \quad \text{and} \quad v_2 = \left\{ \begin{array}{ll} (2x, -y + R), & y \le 0 \\ (-y - R, -2x), & y \ge 0 \end{array} \right\}$$

where $R = \sqrt{4x^2 + y^2}$. The two different sets of eigenvectors are used since at $x = 0$ one vector in each set degenerates to the zero vector. The first directional derivatives are

$$v_1^{\mathsf{T}} Df = \left\{ \begin{array}{ll} 2x(yR - x^2 - y^2), & y \le 0 \\ x^2(3y - R), & y \ge 0 \end{array} \right\} \quad \text{and} \quad v_2^{\mathsf{T}} Df = \left\{ \begin{array}{ll} x^2(3y + R), & y \le 0 \\ -2x(yR + x^2 + y^2), & y \ge 0 \end{array} \right\}.$$

Let V be the 2×2 matrix whose columns are v_1 and v_2. Since $D^2 f$ is a symmetric matrix, its eigenvectors v_1 and v_2 are orthogonal; thus we have $V\intercal D^2 f V = \text{Diag}\{\lambda_1|v_1|^2, \lambda_2|v_2|^2\}$.

There are no 0-dimensional ridges (*i.e.* no local maxima) since $\lambda_2 \geq 0$ for all (x, y). There are 1-dimensional ridges. Firstly, note that $\lambda_1(x, y) < 0$ as long as not both $x = 0$ and $y \geq 0$. Secondly, if $x = 0$ and $y < 0$, then $V^\intercal Df = 0$ and $V^\intercal D^2 f V < 0$. Also, if $y > 0$ and $x^2 = 2y^2$, then $V^\intercal Df = 0$ and $V^\intercal D^2 f V < 0$. Therefore, the 1-dimensional ridges lie on three rays with origin $(0, 0)$.

FIGURE GOES HERE.

Example 3. Consider $f(x, y, z) = -0.5(ax^2 + by^2 + cz^2)$ where $0 < a < b < c$. The first derivatives are $Df = -(ax, by, cz)$, and the second derivatives are $D^2 f = -\text{Diag}(a, b, c)$. The ordered eigenvalues are $\lambda_1 = -c$, $\lambda_2 = -b$, and $\lambda_3 = -a$, with corresponding eigenvectors $v_1 = (0, 0, 1)$, $v_2 = (0, 1, 0)$, and $v_3 = (1, 0, 0)$. The only 0-dimensional ridge point is the local maximum point $(0, 0, 0)$. The 1-dimensional ridge points consist of the x-axis since $v_1^\intercal Df = v_2^\intercal Df = 0$ imply $y = z = 0$. Intuitively this seems reasonable since the longest axes of the ellipsoidal level sets lie on the x-axis. The 2-dimensional ridge points consist of the xy-plane since $v_1^\intercal Df = 0$ implies $z = 0$. This set also makes intuitive sense since the ellipsoidal level sets are flattest in the z-direction.

FIGURES? How about (1) rendered ellipsoid, (2) slice perpendicular to the x-axis, (3) slices parallel to xy-plane.

1.3 1-Dimensional Ridges in $\mathbb{R}^2$

Constructing closed form representations for ridges is generally intractable. For dimensions 2 through 4 it is possible to solve symbolically for the eigenvalues and eigenvectors in closed form (REFER: CRC Handbook formulas). However, solving explicitly for the roots of the first derivative equations is not possible. For dimensions $n \geq 5$ there are no formulas for the roots of polynomials of degree n. (REFER: Galois result about roots of polynomials). Numerical algorithms for solving eigensystems must be used instead. Section (???) gives a discussion of such solvers. The remainder of this chapter provides ridge algorithms which lend themselves to numerical computation.

We resort now to index notation discussed in Section (???). The eigensystems for $D^2 f$ are given by $f_{,ij} u_j = \alpha u_i$ and $f_{,ij} v_j = \beta v_i$ where $\alpha \leq \beta$, $u_i u_i = 1$, $v_i v_i = 1$, $u_i v_i = 0$, and $e_{ij} u_i v_j = 1$ (the vectors u and v form a right-handed orthonormal system). Define $P = u_i f_{,i}$ and $Q = v_i f_{,i}$. According to the height ridge definition, a point $x \in \mathbb{R}^2$ is a 1-dimensional ridge point if $P(x) = 0$ and $\alpha(x) < 0$.

To facilitate detection of zeros of P, we require $P(x)$ to be at least a C^1 function. In a region where $\alpha < \beta$, the eigenvectors u and v are C^{k-2} functions as long as f is a C^k function. We will assume that $k \geq 3$ so that the eigenvectors are at least C^1 in such regions. Problems can occur at *umbilics*, those points for which $\alpha = \beta$. At umbilics the eigenvectors can become discontinuous. The two 1-dimensional eigenspaces when $\alpha \neq \beta$ merge into a 2-dimensional eigenspace when $\alpha = \beta$. For example, consider $f(x, y) = (x^4 + 6y^2)/12$. The second derivative matrix is $D^2 f = \text{Diag}\{x^2, 1\}$. The ordered eigenvalues are $\alpha = \min\{x^2, 1\}$ and

$\beta = \max\{x^2, 1\}$. For $x > 1$ the eigenvectors are $u = (0, 1)$ and $v = (1, 0)$. For $x < 1$ the eigenvectors are $u = (1, 0)$ and $v = (0, 1)$. The eigenvectors become discontinuous as x varies through 1; the eigenspaces are in a sense swapped.

Umbilics play the role of either endpoints or branch points for 1-dimensional ridges. The algorithm discussed in this section applies to ridges which lie in umbilic-free regions, that is, in regions for which $\alpha < \beta$ and $\alpha < 0$. However, the numerical algorithms can become ill-conditioned near an umbilic point, so the algorithms must detect umbilics and handle them appropriately. Section (???) deals with these issues.

The algorithm consists of finding an initial ridge point, then traversing the ridge by following the ridge direction. Both steps involve computing the gradient of P. To support the application of ridge finding where the function is given discretely as a table of values, it is helpful to have a closed form formula for DP which involves only explicit occurrences of the derivatives of function f and the eigenvalues and eigenvectors of $D^2 f$. In this case, splines may be used as a smooth representation of f. Section (???) gives a detailed discussion of the use of B-splines.

Since u and v form an orthonormal system we can write Df as $f_{,i} = Pu_i + Qv_i$. Moreover, $u_i u_i = 1$ implies $u_i u_{i,j} = 0$. Differentiating P yields $P_{,k} = u_i f_{,ik} + u_{i,k} f_{,i} = \alpha u_k + Q v_i u_{i,k}$. Differentiating $f_{,ij} u_j = \alpha u_i$ yields $f_{,ij} u_{j,k} + f_{,ijk} u_j = \alpha u_{i,k} + \alpha_{,k} u_i$ where we have used $f_{,ij} u_j = \alpha u_i$. Contracting with v_i and using $v_i f_{,ij} = \beta v_j$ and $u_i v_i = 0$ we obtain $\beta v_j u_{j,k} + f_{,ijk} v_i u_j = \alpha v_i u_{i,k}$. Therefore, $v_i u_{i,k} = f_{,ijk} v_i u_j / (\alpha - \beta)$. Substituting this in the previous equation for $P_{,k}$ yields

$$P_{,k} = \alpha u_k + \frac{Q}{\alpha - \beta} f_{,ijk} v_i u_j. \tag{1}$$

The eigenvectors and eigenvalues are computed using second derivatives of f, the quantity $Q = v_i f_{,i}$ requires first derivatives of f, and $P_{,k}$ additionally requires third derivatives of f. All calculations do not require explicit formulas for the derivatives of eigenvectors $u_{i,j}$ or $v_{i,j}$.

1.3.1 Ridge Flow

Given an initial approximation $\mathcal{A}$ to a ridge point, a flow path to the ridge is determined by gradient descent. Ridge points occur as absolute minimum points for the function $P^2(x)/2$ where $\alpha(x) < 0$. The gradient descent is modeled by

$$\frac{dx_i(t)}{dt} = -(P^2(x(t))/2)_{,i} = -P(x(t))P_{,i}(x(t)), \;\; x_i(0) = \mathcal{A}_i, \;\; i = 1, 2. \tag{2}$$

The solution curve terminates at time $T > 0$ if $P(x(T)) = 0$ or if a positive local minimum is reached, in which case a different starting point should be used. The point $\mathcal{R} = x(T)$ will be used as the starting ridge point for ridge traversal. Gradient descent and minimization are discussed in Section (???).

1.3.2 Ridge Traversal

Let $\mathcal{R}$ be the initial ridge point obtained by the construction in the previous subsubsection. If $T(x)$ is a tangent vector to the ridge, then the ridge can be traversed by solving a system of ordinary differential equations, $dx/dt = T(x)$. To determine $T(x)$, note that the ridge curve is a solution to $P(x) = 0$, so it is (part of) a level curve for P. The gradient of P is therefore normal to the ridge; a tangent to the ridge is orthogonal to the normal, so $T_i(x) = e_{ij}P_{,j}(x)$. The system of equations determining the traversal is therefore

$$\frac{dx_i(t)}{dt} = \pm\, e_{ij}P_{,j}(x(t)), \quad x_i(0) = \mathcal{R}_i, \quad i = 1,2 \tag{3}$$

where two traversals are required. Ordinary differential equation solvers are discussed in Section (???).

EXAMPLES.

- tube
- Y-junction
- T-junction
- head image (or something realistic)

1.4 1-Dimensional Ridges in $\mathbb{R}^3$

The ridge construction gets somewhat more complicated because of the effects of what we call *semi-umbilics*. These are points for which at least two eigenvalues are equal (but not necessarily all eigenvalues are equal). At semi-umbilics the eigenvectors can be discontinuous, and swapping of eigenspaces can occur.

Let $f_{,ij}u_j = \alpha u_i$, $f_{,ij}v_j = \beta v_i$, and $f_{i,j}w_j = \gamma w_i$ where $\alpha \le \beta \le \gamma$ and u, v, and w form a right-handed orthonormal system (the vectors are all unit length, mutually orthogonal, and $e_{ijk}u_iv_jw_k = 1$). Define $P = u_if_{,i}$, $Q = v_if_{,i}$, and $R = w_if_{,i}$. According to the height ridge definition, a point $x \in \mathbb{R}^3$ is a 1-dimensional ridge point if $P(x) = 0$, $Q(x) = 0$, and $\beta(x) < 0$.

We assume that f is at least a C^3 function so that the eigenvectors are C^3 functions within regions which contain no semi-umbilics. The semi-umbilics $\beta = \gamma$ or $\alpha = \gamma$ are analogous to the umbilic case when $n = 2$. Ridges tend to branch or end at such points. However, occurrence of the semi-umbilic $\alpha = \beta$ should not generally be a stopping condition for the ridge construction. This type of semi-umbilic will occur for data sets with cylindrical symmetry. For example, consider $f(x, y, z) = -(2x^2 + 2y^2 + z^2)$. The first derivatives are $Df = (-4x, -4y, -2z)$ and the second derivatives are $D^2f = \mathrm{Diag}(-4, -4, -2)$. Thus, $\alpha = \beta = -4$ and $\gamma = -2$. There are many choices for u and v, but for any such choice the equations $P = 0$ and $Q = 0$ provide a 1-dimensional ridge $(0, 0, z)$ for all $z \in \mathbb{R}$.

At semi-umbilics the eigenspace has dimension larger than 1. Although the eigenvectors may become discontinuous, it is possible to choose a smoothly varying basis for the eigenspace. Assume that $\gamma > \beta$ in the region for which we seek ridges. Let $\bar{u}$ and $\bar{v}$ be *smoothly varying* orthonormal vectors which span $\langle w\rangle^{\perp}$. The eigenvector basis and the smooth basis are related by

$$\begin{bmatrix} \bar{u}_k \\ \bar{v}_k \end{bmatrix} = \begin{bmatrix} c_{11} & c_{21} \\ c_{12} & c_{22} \end{bmatrix} \begin{bmatrix} u_k \\ v_k \end{bmatrix} = C^{\mathsf{T}} \begin{bmatrix} u_k \\ v_k \end{bmatrix}$$

where $C = [c_{ij}]$ is an orthogonal matrix. Define

$$\begin{bmatrix} \bar{P} \\ \bar{Q} \end{bmatrix} = C^{\mathsf{T}} \begin{bmatrix} P \\ Q \end{bmatrix},$$

so $P = 0$ and $Q = 0$ if and only if $\bar{P} = 0$ and $\bar{Q} = 0$. The ridge algorithms will require differentiating $\bar{P}$ and $\bar{Q}$. The following development provides closed form solutions for these derivatives.

Since $\bar{u}$, $\bar{v}$, and w form a smoothly varying orthonormal system, their derivatives satisfy

$$\left[\begin{array}{c}\bar{u}_{i,j}\\ \bar{v}_{i,j}\\ \hline w_{i,j}\end{array}\right] = \left[\begin{array}{cc|c}0 & 0 & a_j\\ 0 & 0 & b_j\\ \hline -a_j & -b_j & 0\end{array}\right]\left[\begin{array}{c}\bar{u}_i\\ \bar{v}_i\\ \hline w_i\end{array}\right]$$

for some choice of continuous vectors a_j and b_j. Without loss of generality, the vectors in the $(1,2)$ and $(2,1)$ positions of the matrix were set to zero since they represents rotations within the plane spanned by $\bar{u}$ and $\bar{v}$. Differentiating $\bar{P}$ and $\bar{Q}$ yields

$$\begin{bmatrix}\bar{P}_{,k}\\ \bar{Q}_{,k}\end{bmatrix} = \begin{bmatrix}\bar{u}_i f_{,ik} + \bar{u}_{i,k} f_{,i}\\ \bar{v}_i f_{,ik} + \bar{v}_{i,k} f_{,i}\end{bmatrix} = \begin{bmatrix}f_{,ki}\bar{u}_i + R a_k\\ f_{,ki}\bar{v}_i + R b_k\end{bmatrix}.$$

The a_k and b_k are determined by the eigensystem for w, namely $f_{,ij}w_j = \gamma w_i$. Differentiate this to obtain $f_{,ij}w_{j,k} + f_{,ijk}w_j = \gamma w_{i,k} + \gamma_{,k}w_i$. Substitute for $w_{i,j}$ and rearrange to obtain $(f_{,ij}\bar{u}_j - \gamma\bar{u}_i)a_k + (f_{,ij}\bar{v}_j - \gamma\bar{v}_i)b_k = f_{,ijk}w_j - \gamma_{,k}w_i$. Contracting with $\bar{u}$ and $\bar{v}$ yields $(\bar{u}_i f_{,ij}\bar{u}_j - \gamma)a_k + (\bar{u}_i f_{,ij}\bar{v}_j)b_k = f_{,ijk}\bar{v}_i w_j$ and $(\bar{v}_i f_{,ij}\bar{u}_j)a_k + (\bar{v}_i f_{,ij}\bar{v}_j - \gamma)b_k = f_{,ijk}\bar{v}_i w_j$. This is a system of two equations in the two unknown vectors a_k and b_k which can be solved explicitly as

$$\begin{bmatrix}a_k\\ b_k\end{bmatrix} = \begin{bmatrix}\bar{u}_i f_{,ij}\bar{u}_j - \gamma & \bar{u}_i f_{,ij}\bar{v}_j\\ \bar{v}_i f_{,ij}\bar{u}_j & \bar{v}_i f_{,ij}\bar{v}_j - \gamma\end{bmatrix}^{-1}\begin{bmatrix}f_{,ijk}\bar{u}_i w_j\\ f_{,ijk}\bar{v}_i w_j\end{bmatrix}.$$

Using the relationships between the eigenvectors and the smooth basis, we obtain

$$\begin{bmatrix}a_k\\ b_k\end{bmatrix} = C^{\mathsf{T}}\begin{bmatrix}\frac{1}{\alpha-\gamma} f_{,ijk}u_i w_j\\ \frac{1}{\beta-\gamma} f_{,ijk}v_i w_j\end{bmatrix}.$$

Substituting into the formulas for $\bar{P}_{,k}$ and $\bar{Q}_{,k}$ yields

$$\begin{bmatrix}\bar{P}_{,k}\\ \bar{Q}_{,k}\end{bmatrix} = C^{\mathsf{T}}\begin{bmatrix}\alpha u_k + \frac{R}{\alpha-\gamma} f_{,ijk}u_i w_j\\ \beta v_k + \frac{R}{\beta-\gamma} f_{,ijk}v_i w_j\end{bmatrix}$$

These formulas will be used both in finding an initial ridge point and in determining the ridge direction. Note that the matrix C is an unknown quantity, but we will see that the ridge flow is independent of C and the ridge traversal only requires knowing $\det(C) = \pm 1$. With this in mind, define

$$\begin{bmatrix}\tilde{P}_{,k}\\ \tilde{Q}_{,k}\end{bmatrix} = \begin{bmatrix}\alpha u_k + \frac{R}{\alpha-\gamma} f_{,ijk}u_i w_j\\ \beta v_k + \frac{R}{\beta-\gamma} f_{,ijk}v_i w_j\end{bmatrix} \tag{4}$$

These quantities will in effect play the role of the derivatives of $\bar{P}$ and $\bar{Q}$ despite the fact that they are not necessarily the derivatives of some functions $\tilde{P}$ and $\tilde{Q}$. The use of index/derivative notation is used suggestively to remind us how the quantities were obtained.

1.4.1 Ridge Flow

Given an initial approximation $\mathcal{A}$ to a ridge point, a flow path to the ridge is determined by gradient descent. Ridge points occur as absolute minimum points for the function $(\bar{P}^2(x) + \bar{Q}^2(x))/2$. Note that $[\ \bar{P} \quad \bar{Q}\ ] = [\ P \quad Q\ ]C$ and $[\ \bar{P}_{,k} \quad \bar{Q}_{,k}\ ] = [\ \tilde{P}_{,k} \quad \tilde{Q}_{,k}\ ]C$, so $\bar{P}\bar{P}_{,i} + \bar{Q}\bar{Q}_{,i} = P\tilde{P}_{,k} + Q\tilde{Q}_{,k}$. Therefore, the gradient descent is modeled by

$$\frac{dx_i(t)}{dt} = -P(x(t))\tilde{P}_{,i}(x(t)) - Q(x(t))\tilde{Q}_{,i}(x(t)), \quad x_i(0) = \mathcal{A}_i, \quad i = 1, 2, 3. \tag{5}$$

The solution curve terminates at time $T > 0$ if $P(x(T)) = 0$ and $Q(x(T)) = 0$, or if a positive local minimum is reached, in which case a different starting point should be used. The point $\mathcal{R} = x(T)$ will be used as the starting ridge point for ridge traversal. The nice consequence of this result is that one does not have to explicitly construct smoothly varying $\bar{u}$ and $\bar{v}$ in order to compute the flow direction to a ridge. The (possibly discontinuous) eigenvectors u and v can be used instead to produce the continuous flow direction.

1.4.2 Ridge Traversal

Let $\mathcal{R}$ be the initial ridge point obtained by ridge flow. If $T(x)$ is a unit length tangent vector to the ridge, then the ridge can be traversed by solving a system of ordinary differential equations, $dx/dt = T(x)$. To determine $T(x)$, note that the ridge curve is a solution to $\bar{P}(x) = 0$ and $\bar{Q}(x) = 0$. The curve is the intersection of the two implicitly defined surfaces whose normals are $D\bar{P}$ and $D\bar{Q}$. The curve direction must be perpendicular to both normals, so choose T to be the cross product of the normals, $T_i(x) = e_{ijk}\bar{P}_{,j}(x)\bar{Q}_{,k}(x)$.

As in ridge flow, we do not have to explicitly construct smoothly varying $\bar{u}$ and $\bar{v}$ to obtain a continuous ridge direction. Since $[\ D\bar{P} \quad D\bar{Q}\ ] = [\ D\tilde{P} \quad D\tilde{Q}\ ]C$ and C is orthogonal, the cross products are related by $D\bar{P}(x) \times D\bar{Q}(x) = \det(C(x))D\tilde{P}(x) \times D\tilde{Q}(x)$, where $|\det(C(x))| = 1$. The discontinuity of the cross product is captured entirely by $\det(C(x))$. If a semi-umbilic $\alpha = \beta$ causes eigenspaces to be swapped, or if the numerical eigensolver does not provide a smoothly varying set of eigenvectors as x varies, then such behavior will affect $\det(C(x))$ and can be detected in the implementation by comparing the angle between the previously computed direction and the currently computed direction. The system of equations determining the traversal is therefore

$$\frac{dx_i(t)}{dt} = \pm\, e_{ijk}\tilde{P}_{,j}(x(t))\tilde{Q}_{,k}(x(t)), \quad x_i(0) = \mathcal{R}_i, \quad i = 1, 2, 3, \tag{6}$$

where two traversals are required.

EXAMPLE. Maybe a simple blurred tube object displayed as a sequence of slices on which the ridge points have been marked?

1.5 1-Dimensional Ridges in $\mathbb{R}^n$

The problems with semi-umbilics occur for $n \geq 2$. The ideas for dimensions $n = 2$ generalize to higher dimensions. Let the eigenvalues and eigenvectors for $D^2 f$ be denoted λ_k and v_k for $1 \leq k \leq n$. The semi-umbilics $\lambda_k = \lambda_n$ correspond to branch points and end points of the ridge. The semi-umbilics $\lambda_i = \lambda_j$ ($i \neq n$, $j \neq n$) reflect symmetries of the data set, but should not cause termination of the ridge construction. We assume that the ridge construction is in regions where $\lambda_{n-1} < \lambda_n$ so that the eigenspaces corresponding to the first $n-1$ eigenvectors never swap or combine with the eigenspace corresponding to the last eigenvalue λ_n.

Let v_{ij} be an orthonormal matrix (with determinant 1) whose columns are unit length eigenvectors for $D^2 f$; the columns form a right-handed orthonormal system. All the eigensystems can be written as a single matrix equation $f_{,ij} v_{jk} = v_{ij} \lambda_{jk}$ where λ_{ii} is a diagonal matrix whose j^{th} diagonal entry is the eigenvalue corresponding to the eigenvector which is the j^{th} column of v_{ij}. Define $P_j = v_{ij} f_{,i}$. According to the height ridge definition, a point $x \in \mathbb{R}^n$ is a 1-dimensional ridge point if $P_j(x) = 0$ for $1 \le j \le n-1$ and $\lambda_{n-1}(x) < 0$.

The construction includes indices whose range is between 1 and $n-1$ rather than the full range between 1 and n. As a notational aid, indices in the range 1 through $n-1$ will be subscripted with a zero. Indices in the full range are unsubscripted. The index n is the dimension of the space and does not indicate a free index.

Define $w_i = v_{in}$, $\gamma = \lambda_n$, and $R = w_i f_{,i}$. Let $\bar{v}_{ij}$ denote a smoothly varying orthonormal matrix whose last column is w and whose first $n-1$ columns span $\langle w \rangle^\perp$. The first $n-1$ eigenvectors and the smooth basis for $\langle w \rangle^\perp$ are related by

$$\bar{v}_{ii_0} = v_{ij_0} c_{j_0 i_0}$$

where $c_{i_0 j_0}$ is an orthogonal matrix. Define

$$\bar{P}_{i_0} = \bar{v}_{ii_0} f_{,i} = P_{j_0} c_{j_0 i_0}$$

so $\bar{P}_{i_0} = 0$ if and only if $P_{i_0} = 0$. The ridge algorithms will require differentiating $\bar{P}_{i_0}$.

Orthonormality of $\bar{v}_{ij}$ implies $\bar{v}_{ki}\bar{v}_{kj} = \delta_{ij} = \bar{v}_{ik}\bar{v}_{jk}$. Differentiating yields $\bar{v}_{ki}\bar{v}_{kj,m} + \bar{v}_{ki,m}\bar{v}_{kj} = 0$. The derivatives $\bar{v}_{kj,m}$ can be written in terms of the orthonormal basis as $\bar{v}_{kj,m} = a_{j\ell m}\bar{v}_{k\ell}$ where $a_{j\ell m}$ is a continuous quantity that will be determined later. Replacing this in the previous equation yields $0 = \bar{v}_{ki} a_{j\ell m} \bar{v}_{k\ell} + a_{i\ell m}\bar{v}_{k\ell} u_{kj} = a_{jim} + a_{ijm}$. Thus, a_{ijm} is antisymmetric in its first two indices. Without loss of generality we can choose $a_{i_0 j_0 k} = 0$ ($i_0 \ne j_0$) since these components represent rotations of the vectors within the orthogonal complement of w. The vectors must be a solution to

$$\bar{v}_{ii_0,j} = a_{i_0 n j} w_i \;\text{ and }\; w_{i,j} = -a_{i_0 n j}\bar{v}_{ii_0}.$$

Differentiating $\bar{P}_{i_0}$ yields

$$\bar{P}_{i_0,k} = \bar{v}_{ii_0} f_{,ik} + \bar{v}_{ii_0,k} f_{,i} = f_{,ki}\bar{v}_{ii_0} + R a_{i_0 n k}.$$

The $a_{i_0 nk}$ are determined by the eigensystem for w, namely $f_{,ij} w_j = \gamma w_i$. Differentiate this to obtain $f_{,ij} w_{j,k} + f_{,ijk} w_j = \gamma w_{i,k} + \gamma_{,k} w_i$. Substitute for $w_{i,j}$ and rearrange to obtain $(f_{,ij}\bar{v}_{ji_0} - \gamma\bar{v}_{ii_0}) a_{i_0 nk} = f_{,ijk} w_j - \gamma_{,k} w_i$. Contracting with $\bar{v}_{ij_0}$ yields $(\bar{v}_{ij_0} f_{,ij}\bar{v}_{ji_0} - \gamma\delta_{i_0 j_0}) a_{i_0 nk} = f_{,ijk}\bar{v}_{ij_0} w_j$. This system of equations can be solved explicitly as follows.

Using the relationships between the eigenvectors and the smooth basis, we have

$$\begin{aligned}
\bar{v}_{ij_0} f_{,ij}\bar{v}_{ji_0} - \gamma\delta_{j_0 i_0} &= v_{ik_0} c_{k_0 j_0} f_{,ij} v_{jm_0} c_{\ell_0 i_0} - \gamma\delta_{j_0 i_0} \\
&= c_{k_0 j_0}(v_{ik_0} f_{,ij} v_{jm_0}) c_{m_0 i_0} - \gamma\delta_{j_0 i_0} \\
&= c_{k_0 j_0}(v_{ik_0} v_{ij} \lambda_{jm_0}) c_{m_0 i_0} - \gamma\delta_{j_0 i_0} \\
&= c_{k_0 j_0}(\delta_{k_0 j} \lambda_{jm_0}) c_{m_0 i_0} - \gamma\delta_{j_0 i_0} \\
&= c_{k_0 j_0} \lambda_{k_0 m_0} c_{m_0 i_0} - \gamma c_{k_0 j_0}\delta_{k_0 m_0} c_{m_0 i_0} \\
&= c_{k_0 j_0}(\lambda_{k_0 m_0} - \gamma\delta_{k_0 m_0}) c_{m_0 i_0}
\end{aligned}$$

and

$$f_{,ijk}\bar{v}_{ij_0}w_j = c_{i_0j_0}f_{,ijk}v_{ii_0}w_j.$$

The system can be reduced as shown where $\Delta_{i_0j_0}$ is the diagonal matrix whose k_0^{th} diagonal entry is $\lambda_{k_0}-\gamma$ and $\Delta_{i_0j_0}^{-1}$ is the diagonal inverse matrix:

$$\begin{aligned}
(\bar{v}_{ij_0}f_{,ij}\bar{v}_{ji_0}-\gamma\delta_{i_0j_0})a_{i_0nk} &= f_{,ijk}\bar{v}_{ij_0}w_j \\
c_{k_0j_0}(\lambda_{k_0m_0}-\gamma\delta_{k_0m_0})c_{m_0i_0}a_{i_0nk} &= c_{i_0j_0}f_{,ijk}v_{ii_0}w_j \\
(\lambda_{j_0m_0}-\gamma\delta_{j_0m_0})c_{m_0i_0}a_{i_0nk} &= f_{,ijk}v_{ij_0}w_j \\
c_{m_0i_0}a_{i_0nk} &= \Delta_{m_0j_0}^{-1}f_{,ijk}v_{ij_0}w_j \\
a_{i_0nk} &= c_{m_0i_0}\Delta_{m_0j_0}^{-1}f_{,ijk}v_{ij_0}w_j.
\end{aligned}$$

Substituting into the formula for $\bar{P}_{\alpha,k}$ yields

$$\begin{aligned}
\bar{P}_{i_0,k} &= \bar{v}_{ii_0}f_{,ik}+Ra_{i_0nk} \\
&= v_{ij_0}c_{j_0i_0}f_{,ik}+Rc_{m_0i_0}\Delta_{m_0j_0}^{-1}f_{,ijk}v_{ik_0}w_j \\
&= c_{j_0i_0}(v_{ij_0}f_{,ik}+R\Delta_{j_0k_0}^{-1}f_{,ijk}v_{ik_0}w_j) \\
&= c_{j_0i_0}(v_{kk_0}\lambda_{k_0j_0}+R\Delta_{k_0j_0}^{-1}f_{,ijk}v_{ik_0}w_j).
\end{aligned}$$

Define

$$\tilde{P}_{i_0,k} = v_{kj_0}\lambda_{j_0i_0}+R\Delta_{i_0j_0}^{-1}f_{,ijk}v_{ij_0}w_j. \tag{7}$$

These quantities play the role of the derivatives of $\bar{P}_{i_0}$ despite the fact that they are not necessarily the derivatives of some functions $\tilde{P}_{i_0}$.

1.5.1 Ridge Flow

Given an initial approximation $\mathcal{A}$ to a ridge point, a flow path to the ridge is determined by gradient descent. Ridge points occur as absolute minimum points for the function $P_{i_0}(x)P_{i_0}(x)/2$. Note that $\bar{P}_{i_0} = P_{j_0}c_{j_0i_0}$ and $\bar{P}_{i_0,k} = c_{j_0i_0}\tilde{P}_{j_0,k}$, so $\bar{P}_{i_0}\bar{P}_{i_0,k} = P_{i_0}\tilde{P}_{i_0,k}$. Therefore, the gradient descent is modeled by

$$\frac{dx_i(t)}{dt} = -P_{i_0}(x(t))\tilde{P}_{i_0,i}(x(t)), \;\; x_i(0)=\mathcal{A}_i, \;\; 1\le i\le n. \tag{8}$$

The solution curve terminates at time $T>0$ if $P_{i_0}(x(T))=0$ or if a positive local minimum is reached, in which case a different starting point should be used. The point $\mathcal{R}=x(T)$ will be used as the starting ridge point for ridge traversal. As in the case of 1-dimensional ridges in $\mathbb{R}^3$, the continuous flow direction is calculated directly from (possibly discontinuous) eigenvectors and eigenvalues.

1.5.2 Ridge Traversal

Let $\mathcal{R}$ be the initial ridge point obtained by ridge flow. If $T(x)$ is a unit length tangent vector to the ridge, then the ridge can be traversed by solving a system of ordinary differential equations, $dx/dt = T(x)$. To determine $T(x)$, note that the ridge curve is the intersection of $n-1$ surfaces implicitly defined by $\bar{P}_{i_0}(x)=0$.

The curve direction is therefore a vector which is orthogonal to all $n-1$ surface normals $P_{i_0,i}$. The way to construct a vector $T \in \mathbb{R}^n$ which is orthogonal to $n-1$ orthogonal vectors is to use the generalized cross product

$$T_i = e_{ii_1\cdots i_{n-1}} \bar{P}_{1,i_1} \cdots \bar{P}_{n-1,i_{n-1}}$$

where $e_{ii_1\cdots i_{n-1}}$ is the permutation tensor on n symbols. Let $\varepsilon_{j_1\cdots j_{n-1}}$ be the permutation tensor on $n-1$ symbols; then

$$\begin{aligned} T_i &= \tfrac{1}{(n-1)!} e_{ii_1\cdots i_{n-1}} \varepsilon_{j_1\cdots j_{n-1}} P_{j_1,i_1} \cdots P_{j_{n-1},i_{n-1}} \\ &= \tfrac{1}{(n-1)!} e_{ii_1\cdots i_{n-1}} \varepsilon_{j_1\cdots j_{n-1}} c_{k_1 j_1} \tilde{P}_{k_1 i_1} \cdots c_{k_{n-1} j_{n-1}} \tilde{P}_{j_{n-1} i_{n-1}} \\ &= (\varepsilon_{j_1\cdots j_{n-1}} c_{k_1 j_1} \cdots c_{k_{n-1} j_{n-1}}) \tfrac{1}{(n-1)!} e_{ii_1\cdots i_{n-1}} \tilde{P}_{k_1 i_1} \cdots \tilde{P}_{k_{n-1} i_{n-1}} \\ &= \det(C) \varepsilon_{k_1\cdots k_{n-1}} \tfrac{1}{(n-1)!} e_{ii_1\cdots i_{n-1}} \tilde{P}_{k_1 i_1} \cdots \tilde{P}_{k_{n-1} i_{n-1}} \\ &= \det(C) e_{ii_1\cdots i_{n-1}} \tilde{P}_{1 i_1} \cdots \tilde{P}_{n-1 i_{n-1}}. \end{aligned}$$

Any sign-changes or swapping of eigenvectors within $\langle w \rangle^\perp$ produced by semi-umbilics or the numerical eigensolver will be reflected in the determinant $\det(C)$. But since C is orthonormal, the determinant is either 1 or -1. The key result again is that you do not need to keep track of smoothly varying eigenfields. You simply compute the eigenvectors, compute the ridge direction, and choose the sign of the direction so that the current direction forms smallest angle with the previous direction. The system of equations determining the traversal is therefore

$$\frac{dx_i(t)}{dt} = \pm e_{i\, i_1\cdots i_{n-1}} \tilde{P}_{1\, i_1}(x(t)) \cdots \tilde{P}_{n-1\, i_{n-1}}(x(t)), \;\; x_i(0) = \mathcal{R}_i, \;\; 1 \le i \le n, \tag{9}$$

where two traversals are required.

1.6 2-Dimensional Ridges in $\mathbb{R}^3$

The ridge algorithms for dimensions $d \ge 2$ are similar to those for 1-dimensional ridges. The main difference is ridge traversal. Rather than traversing a curve in space, we now need to traverse manifolds. The construction of 2-dimensional ridges in $\mathbb{R}^3$ is similar to the construction of 1-dimensional ridges in $\mathbb{R}^2$.

Let $f_{,ij} u_j = \alpha u_i$, $f_{,ij} v_j = \beta v_i$, and $f_{,ij} w_j = \gamma w_i$ where $\alpha \le \beta \le \gamma$ and u, v, and w form an a right-handed orthonormal system. Define $P = u_i f_{,i}$, $Q = v_i f_{,i}$, and $R = w_i f_{,i}$. According to the height ridge definition, a point $x \in \mathbb{R}^3$ is a 2-dimensional ridge point if $P(x) = 0$ and $\alpha(x) < 0$. Typically the solution should be a 2-dimensional surface in $\mathbb{R}^3$.

We assume that ridges are sought in regions where $\alpha < \beta$. This guarantees that the α-eigenspace is not swapped with those for β and γ. Consequently, u can be chosen smoothly and P is a C^1 function. A normal to the ridge surface is $P_{,k}$ given by $P_{,k} = u_i f_{,ik} + u_{i,k} f_{,i} = \alpha u_k + Q v_i u_{i,k} + R w_i u_{i,k}$ where we have used $f_{,i} = P u_i + Q v_i + R w_i$. Differentiate $f_{,ij} u_j = \alpha u_i$ to obtain $f_{,ij} u_{j,k} + f_{,ijk} u_j = \alpha u_{i,k} + \alpha_k u_i$. Contract with v_i and w_i to obtain $v_i u_{i,k} = f_{,ijk} v_i u_j / (\alpha - \beta)$ and $w_i u_{i,k} = f_{,ijk} w_i u_j / (\alpha - \gamma)$. Substitute in the formula for gradient of P to obtain

$$P_{,k} = \alpha u_k + \frac{Q}{\alpha - \beta} f_{,ijk} v_i u_j + \frac{R}{\alpha - \gamma} f_{,ijk} w_i u_j. \tag{10}$$

1.6.1 Ridge Flow

Given an initial approximation $\mathcal{A}$ to a ridge point, a flow path to the ridge is determined by gradient descent. Ridge points occur as absolute minimum points for the function $P^2(x)/2$ where $\alpha(x) < 0$. The gradient descent is modeled by

$$\frac{dx_i(t)}{dt} = -P(x(t))P_{,i}(x(t)), \;\; x_i(0) = \mathcal{A}_i, \;\; i = 1, 2, 3. \tag{11}$$

The solution curve terminates at time $T > 0$ if $P(x(T)) = 0$ or if a positive local minimum is reached, in which case a different starting point should be used. The point $\mathcal{R} = x(T)$ will be used as the starting ridge point for ridge traversal.

1.6.2 Ridge Traversal

From surface theory discussed in Section (???), a convenient coordinate system to impose on a surface is one using lines-of-curvature. The tangents to these curves are principal direction vectors. The resulting coordinates are orthogonal and the implied metric tensor is diagonal. Of course, the coordinate grid can degenerate at umbilics. If ϕ and ψ are unit-length principal direction vectors, $N = \phi \times \psi$ is the unit-length normal, and the metric tensor is $G = \mathrm{Diag}(\alpha^2, \beta^2)$, then the surface parameterization $x(s, t)$ satisfies $x_s = \alpha\phi$ and $x_t = \beta\psi$ with

$$\begin{bmatrix} \phi_s \\ \psi_s \\ N_s \end{bmatrix} \begin{bmatrix} 0 & -\frac{1}{\beta}\alpha_t & \alpha\kappa_\phi \\ \frac{1}{\beta}\alpha_t & 0 & 0 \\ -\alpha\kappa_\phi & 0 & 0 \end{bmatrix} \begin{bmatrix} \phi \\ \psi \\ N \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \phi_t \\ \psi_t \\ N_t \end{bmatrix} \begin{bmatrix} 0 & \frac{1}{\alpha}\beta_s & 0 \\ -\frac{1}{\alpha}\beta_s & 0 & \beta\kappa_\phi \\ 0 & -\beta\kappa_\psi & 0 \end{bmatrix} \begin{bmatrix} \phi \\ \psi \\ N \end{bmatrix}$$

where κ_T is the curvature of the normal section determined by N and tangent vector T. It is also necessary that

$$\begin{aligned} &\tfrac{\partial}{\partial t}(\alpha\kappa_\phi) - \kappa_\psi \tfrac{\partial\alpha}{\partial t} = 0 \\ &\tfrac{\partial}{\partial s}(\beta\kappa_\psi) - \kappa_\phi \tfrac{\partial\beta}{\partial s} = 0 \\ &\tfrac{\partial}{\partial s}\left(\tfrac{1}{\alpha}\tfrac{\partial\beta}{\partial s}\right) + \tfrac{\partial}{\partial t}\left(\tfrac{1}{\beta}\tfrac{\partial\alpha}{\partial\alpha}\partial t\right) + \alpha\beta\kappa_\phi\kappa_\psi = 0. \end{aligned}$$

The difficulty in implementing this scheme as stated is the problem with updating the metric components α and β during traversal. Equations providing explicit access to α_s and β_t are not immediate derivable from these equations.

The following construction is easier to implement, but generally only allows *local* traversal of the surface. The key idea is that a manifold is locally Euclidean. The construction provides *geodesic coordinates* at the point of application.

Define $N = DP/|DP|$ to be the unit normal to the surface $P = 0$. Let ϕ and ψ be chosen so that ϕ, ψ, and N form a smooth right-handed orthonormal system. The derivatives of the normal and tangents are related by

$$\left[\begin{array}{c} \phi_{i,j} \\ \psi_{i,j} \\ \hline N_{i,j} \end{array}\right] = \left[\begin{array}{cc|c} 0 & 0 & a_j \\ 0 & 0 & b_j \\ \hline -a_j & -b_j & 0 \end{array}\right] \left[\begin{array}{c} \phi_i \\ \psi_i \\ \hline N_i \end{array}\right]$$

for some continuous vectors a_j and b_j. Without loss of generality, the vectors in the $(1,2)$ and $(2,1)$ positions were chosen to be zero since they only represents rotations within $\langle \phi, \psi \rangle$. Differentiating N_i yields

$$N_{i,j} = (\delta_{ik} - N_i N_k) \frac{P_{,kj}}{|DP|}.$$

Contracting with ϕ and ψ produces

$$a_j = -\phi_i N_{i,j} = -\phi_i \frac{P_{,ij}}{|DP|} \text{ and } b_j = -\psi_i N_{i,j} = -\psi_i \frac{P_{,ij}}{|DP|}.$$

Let $x(s,t)$ be a parameterization of the surface implicitly defined by $P(x(s,t)) \equiv 0$. Ridge traversal will be according to the system of partial differential equations $\partial x/\partial s = \phi(x)$ and $\partial x/\partial t = \psi(x)$. The equality of mixed partial derivatives, $\partial^2 x/\partial s \partial t = \partial^2 x/\partial t \partial s$ imposes the constraint on the tangent vectors, $\phi_{i,j}\psi_j = \psi_{i,j}\phi_j$. But this condition already holds for our choice of tangents and normal since

$$\begin{aligned} \phi_{i,j}\psi_j - \psi_{i,j}\phi_j &= (a_j\psi_j - b_j\phi_j)N_i \\ &= \left(-\phi_i \tfrac{P_{,ij}}{|DP|}\psi_j + \psi_i \tfrac{P_{,ij}}{|DP|}\phi_j\right) N_i \\ &= 0 \end{aligned}$$

where we have used the symmetry $P_{,ij} = P_{,ji}$. The system of partial differential equations which governs ridge traversal is

$$\begin{array}{ll} \frac{\partial x}{\partial s} = \phi, & \frac{\partial x}{\partial t} = \psi, \\ \frac{\partial \phi}{\partial s} = -\left(\phi^\mathsf{T} \frac{D^2P}{|DP|}\phi\right) N, & \frac{\partial \phi}{\partial t} = -\left(\phi^\mathsf{T} \frac{D^2P}{|DP|}\psi\right) N, \\ \frac{\partial \psi}{\partial s} = -\left(\psi^\mathsf{T} \frac{D^2P}{|DP|}\phi\right) N, & \frac{\partial \psi}{\partial t} = -\left(\psi^\mathsf{T} \frac{D^2P}{|DP|}\psi\right) N. \end{array} \tag{12}$$

Note that $\partial \phi/\partial s = \kappa_\phi N$ and $\partial \psi/\partial t = \kappa_\psi N$ where κ_T is the curvature of the normal section determined by N and tangent vector T. Although the system of equations includes second derivatives of P, the numerical implementation avoids explicitly calculating them. Section (???) is a discussion of the surface traversal implementation.

1.7 2-Dimensional Ridges in $\mathbb{R}^4$

Let $f_{,ij}u_j = \alpha u_j$, $f_{,ij}v_j = \beta v_j$, $f_{,ij}w_j = \gamma w_j$, and $f_{,ij}\xi_j = \delta \xi_j$ where $\alpha \le \beta \le \gamma \le \delta$ and u, v, w, and ξ form a right-handed orthonormal system. Define $P = u_i f_{,i}$, $Q = v_i f_{,i}$, $R = w_i f_{,i}$, and $S = \xi_i f_{,i}$. According to the height ridge definition, a point $x \in \mathbb{R}^4$ is a 2-dimensional ridge point if $P(x) = 0$, $Q(x) = 0$, and $\beta(x) < 0$. Typically the solution should be a 2-dimensional surface in $\mathbb{R}^4$.

We need to worry about the semi-umbilics $\alpha = \beta$ and $\gamma = \delta$ where eigenspaces may merge and cause discontinuities in the corresponding eigenvectors. We assume that $\beta < \gamma$ to avoid vector swaps between the uv-plane and the $w\xi$-plane. Let $\bar{u}$, $\bar{v}$, $\bar{w}$, and $\bar{\xi}$ be smoothly varying orthonormal vectors such that $\langle \bar{u}, \bar{v} \rangle = \langle u, v \rangle$ and $\langle \bar{w}, \bar{\xi} \rangle = \langle w, \xi \rangle$. We seek restricted local maxima for f in $\langle \bar{u}, \bar{v} \rangle$. The eigenvector basis and the smooth basis are related by

$$\begin{bmatrix} \bar{u}_k \\ \bar{v}_k \\ \hline \bar{w}_k \\ \bar{\xi}_k \end{bmatrix} = \left[\begin{array}{cc|cc} \phi_{11} & \phi_{21} & 0 & 0 \\ \phi_{12} & \phi_{22} & 0 & 0 \\ \hline 0 & 0 & \psi_{11} & \psi_{21} \\ 0 & 0 & \psi_{12} & \psi_{22} \end{array}\right] \begin{bmatrix} u_k \\ v_k \\ \hline w_k \\ \xi_k \end{bmatrix}$$

where $\Phi = [\phi_{ij}]$ and $\Psi = [\Psi_{ij}]$ are orthogonal matrices. Define

$$\begin{bmatrix} \bar{P} \\ \bar{Q} \\ \hline \bar{R} \\ \bar{S} \end{bmatrix} = \left[\begin{array}{c|c} \Phi^{\mathsf{T}} & 0 \\ \hline 0 & \Psi^{\mathsf{T}} \end{array}\right] \begin{bmatrix} P \\ Q \\ \hline R \\ S \end{bmatrix}$$

so $\bar{P} = 0$ and $\bar{Q} = 0$ if and only if $P = 0$ and $Q = 0$. The ridge algorithms will require differentiating $\bar{P}$ and $\bar{Q}$. The following development provides closed form solutions for these derivatives.

Since $\bar{u}$, $\bar{v}$, $\bar{w}$, and $\bar{\xi}$ form a smoothly varying orthonormal system, their derivatives satisfy

$$\begin{bmatrix} \bar{u}_{i,j} \\ \bar{v}_{i,j} \\ \hline \bar{w}_{i,j} \\ \bar{\xi}_{i,j} \end{bmatrix} = \left[\begin{array}{cc|cc} 0 & 0 & a_j & b_j \\ 0 & 0 & c_j & d_j \\ \hline -a_j & -c_j & 0 & 0 \\ -b_j & -d_j & 0 & 0 \end{array}\right] \begin{bmatrix} \bar{u}_i \\ \bar{v}_i \\ \hline \bar{w}_i \\ \bar{\xi}_i \end{bmatrix}$$

for some choice of continuous vectors a_j, b_j, c_j, and d_j. The vectors related to interactions between $\bar{u}$ and $\bar{v}$ and between $\bar{w}$ and $\bar{\xi}$ are set to 0 since they only represent rotations within the respective 2-dimensional subspaces. Differentiating $\bar{P}$ and $\bar{Q}$ yields

$$\begin{bmatrix} \bar{P}_{,k} \\ \bar{Q}_{,k} \end{bmatrix} = \begin{bmatrix} \bar{u}_i f_{,ik} + \bar{u}_{i,k} f_{,i} \\ \bar{v}_i f_{,ik} + \bar{v}_{i,k} f_{,i} \end{bmatrix} = \begin{bmatrix} f_{,ki}\bar{u}_i + \bar{R}a_k + \bar{S}b_k \\ f_{,ki}\bar{v}_i + \bar{R}c_k + \bar{S}d_k \end{bmatrix}.$$

The vectors a_k, b_k, c_k, and d_k are determined by the eigensystems for w and ξ. Some calculations show that

$$\begin{bmatrix} f_{,ij}\bar{w}_j \\ f_{,ij}\bar{\xi}_j \end{bmatrix} = \Psi^{\mathsf{T}}\,\mathrm{Diag}(\gamma,\delta)\Psi \begin{bmatrix} \bar{w}_i \\ \bar{\xi}_j \end{bmatrix}.$$

The matrix $M = [m_{ij}] = \Psi^{\mathsf{T}}\,\mathrm{Diag}(\gamma,\delta)\Psi$ must be differentiable since $D^2 f$, $\bar{w}$, and $\bar{\xi}$ are all differentiable. Differentiating the two equations and contracting both with $\bar{u}_i$ and $\bar{v}_i$ yields four equations in the four unknown vectors

$$\begin{bmatrix} \bar{u}_i f_{,ij}\bar{u}_j - m_{11} & \bar{u}_i f_{,ij}\bar{v}_j & -m_{12} & 0 \\ \bar{v}_i f_{,ij}\bar{u}_j & \bar{v}_i f_{,ij}\bar{v}_j - m_{11} & 0 & -m_{12} \\ -m_{21} & 0 & \bar{u}_i f_{,ij}\bar{u}_j - m_{11} & \bar{u}_i f_{,ij}\bar{v}_j \\ 0 & -m_{21} & \bar{v}_i f_{,ij}\bar{u}_j & \bar{v}_i f_{,ij}\bar{v}_j - m_{11} \end{bmatrix} \begin{bmatrix} a_k \\ c_k \\ b_k \\ d_k \end{bmatrix} = \begin{bmatrix} f_{,ijk}\bar{u}_i\bar{w}_j \\ f_{,ijk}\bar{v}_i\bar{w}_j \\ f_{,ijk}\bar{u}_i\bar{\xi}_j \\ f_{,ijk}\bar{v}_i\bar{\xi}_j \end{bmatrix}.$$

Using the relationships between the eigenvectors and the smooth basis, we obtain

$$\begin{bmatrix} a_k \\ c_k \end{bmatrix} = \Phi\mathsf{T} \begin{bmatrix} \frac{1}{\alpha-\gamma} f_{,ijk} u_i w_j \\ \frac{1}{\beta-\gamma} f_{,ijk} v_i w_j \end{bmatrix} \text{ and } \begin{bmatrix} b_k \\ d_k \end{bmatrix} = \Phi\mathsf{T} \begin{bmatrix} \frac{1}{\alpha-\delta} f_{,ijk} u_i \xi_j \\ \frac{1}{\beta-\delta} f_{,ijk} v_i \xi_j \end{bmatrix}.$$

Substituting into the formulas for $\bar{P}_{,k}$ and $\bar{Q}_{,k}$ yields

$$\begin{bmatrix} \bar{P}_{,k} \\ \bar{Q}_{,k} \end{bmatrix} = \Phi\mathsf{T} \begin{bmatrix} \alpha u_k + \frac{R}{\alpha-\gamma} f_{,ijk} u_i w_j + \frac{S}{\alpha-\delta} f_{,ijk} u_i \xi_j \\ \beta v_k + \frac{R}{\beta-\gamma} f_{,ijk} v_i w_j + \frac{S}{\beta-\delta} f_{,ijk} v_i \xi_j \end{bmatrix}.$$

These formulas will be used both in finding an initial ridge point and in determining the tangent spaces at each point on the 2-dimensional ridge. The matrix Φ is an unknown quantity, but the ridge flow is independent of Φ and the ridge traversal only requires knowing $\det(\Phi) = \pm 1$. Define

$$\begin{bmatrix} \tilde{P}_{,k} \\ \tilde{Q}_{,k} \end{bmatrix} = \begin{bmatrix} \alpha u_k + \frac{R}{\alpha-\gamma} f_{,ijk} u_i w_j + \frac{S}{\alpha-\delta} f_{,ijk} u_i \xi_j \\ \beta v_k + \frac{R}{\beta-\gamma} f_{,ijk} v_i w_j + \frac{S}{\beta-\delta} f_{,ijk} v_i \xi_j \end{bmatrix}. \tag{13}$$

These quantities will in effect play the role of the derivatives of $\bar{P}$ and $\bar{Q}$ despite the fact that they are not necessarily the derivatives of some functions $\tilde{P}$ and $\tilde{Q}$. The use of index/derivative notation is used suggestively to remind us how the quantities were obtained.

1.7.1 Ridge Flow

Given an initial approximation $\mathcal{A}$ to a ridge point, a flow path to the ridge is determined by gradient descent. Ridge points occur as absolute minimum points for the function $(\bar{P}^2(x) + \bar{Q}^2(x))/2$. Note that $[\ \bar{P} \quad \bar{Q}\ ] = [\ P \quad Q\ ]\Phi$ and $[\ \bar{P}_{,k} \quad \bar{Q}_{,k}\ ] = [\ \tilde{P}_{,k} \quad \tilde{Q}_{,k}\ ]\Phi$, so $\bar{P}\bar{P}_{,k} + \bar{Q}\bar{Q}_{,k} = P\tilde{P}_{,k} + Q\tilde{Q}_{,k}$. Therefore, the gradient descent is modeled by

$$\frac{dx_i(t)}{dt} = -P(x(t))\tilde{P}_{,i}(x(t)) - Q(x(t))\tilde{Q}_{,i}(x(t)), \quad x_i(0) = \mathcal{A}_i, \quad i = 1,2,3,4. \tag{14}$$

The solution curve terminates at time $T > 0$ if $P(x(T)) = 0$ and $Q(x(T)) = 0$ or if a positive local minimum is reached, in which case a different starting point should be used. The point $\mathcal{R} = x(T)$ will be used as the starting ridge point for ridge traversal.

1.7.2 Ridge Traversal

As in Section (???), the following construction provides geodesic coordinates for the surface, so the traversal is only local. The gradients $D\bar{P}$ and $D\bar{Q}$ are normals to the 2-dimensional manifold which is the intersection of the two hypersurfaces defined implicitly by $\bar{P} = 0$ and $\bar{Q} = 0$. By Gram-Schmidt orthonormalization we can find two smoothly varying orthonormal vectors N and M such that

$$N_i = u_{11}\bar{P}_{,i} \text{ and } M_i = u_{21}\bar{P}_{,i} + u_{22}\bar{Q}_{,i}$$

where the u_{ij} are differentiable. Let ϕ and ψ be chosen so that ϕ, ψ, N, and M form a smooth right-handed orthonormal system. The derivatives of the normals and tangents are related by

$$\left[\begin{array}{c} \phi_{i,j} \\ \psi_{i,j} \\ \hline N_{i,j} \\ M_{i,j} \end{array}\right] = \left[\begin{array}{rr|rr} 0 & 0 & a_j & b_j \\ 0 & 0 & c_j & d_j \\ \hline -a_j & -c_j & 0 & e_j \\ -b_j & -d_j & -e_j & 0 \end{array}\right] \left[\begin{array}{c} \phi_i \\ \psi_i \\ \hline N_i \\ M_i \end{array}\right]$$

for some continuous vectors a_j, b_j, c_j, d_j, and e_j. Without loss of generality, the vectors in the $(1,2)$ and $(2,1)$ positions are chosen to be zero since they only represent rotations within $\langle\phi,\psi\rangle$. Differentiating N_i and M_i yields

$$N_{i,j} = u_{11}\bar{P}_{,ij} + u_{11,j}\bar{P}_{,i} \text{ and } M_{i,j} = u_{21}\bar{P}_{,ij} + u_{21,j}\bar{P}_{,i} + u_{22}\bar{Q}_{,ij} + u_{22,j}\bar{Q}_{,i}.$$

Contracting with ϕ and ψ produces

$$a_j = -\phi_i N_{i,j}\ ,\ b_j = -\phi_i M_{i,j}\ ,\ c_j = -\psi_i N_{i,j}\ , \quad \text{and } d_j = -\psi_i M_{i,j}.$$

Let $x(s,t)$ be a parameterization of the surface defined implicitly by $\bar{P}(x(s,t)) \equiv 0$ and $\bar{Q}(x(s,t)) \equiv 0$. Ridge traversal will be according to the system of partial differential equations $\partial x/\partial s = \phi(x)$ and $\partial x/\partial t = \psi(x)$. The equality of mixed partial derivatives, $\partial^2 x/\partial s\partial t = \partial^2 x/\partial t\partial s$, imposes the constraint on the tangent vectors, $\phi_{i,j}\psi_j = \psi_{i,j}\phi_j$. But this condition already holds for our choice of tangents and normals since

$$\begin{aligned} \phi_{i,j}\psi_j - \psi_{i,j}\phi_j &= (a_j\psi_j - c_j\phi_j)N_i + (b_j\psi_j - d_j\phi_j)M_i \\ &= (-\phi_i N_{,ij}\psi_j + \psi_i N_{,ij}\phi_j)\,N_i + (-\phi_i M_{,ij}\psi_j + \psi_i M_{,ij}\phi_j)\,M_i \\ &= \left(-u_{11}\phi_i\bar{P}_{,ij}\psi_j + u_{11}\psi_i\bar{P}_{,ij}\phi_j\right)N_i \\ &\quad + \left(-u_{21}\phi_i\bar{P}_{,ij}\psi_j - u_{22}\phi_i\bar{Q}_{,ij}\psi_j + u_{21}\psi_i\bar{P}_{,ij}\phi_j + u_{22}\psi_i\bar{Q}_{,ij}\phi_j\right)M_i \\ &= 0 \end{aligned}$$

where we have used the symmetries $\bar{P}_{,ij} = \bar{P}_{,ji}$ and $\bar{Q}_{,ij} = \bar{Q}_{,ji}$.

The system of partial differential equations which governs ridge traversal is

$$\begin{array}{rclclcl} \frac{\partial x_i}{\partial s} &=& \phi_i \\ \frac{\partial x_i}{\partial t} &=& \psi_i \\ \frac{\partial \phi_i}{\partial s} &=& \phi_{i,j}\frac{\partial x_j}{\partial s} &=& \phi_{i,j}\phi_j &=& -(\phi_k N_{k,j}\phi_j)N_i - (\phi_k M_{k,j}\phi_j)M_i \\ \frac{\partial \phi_i}{\partial t} &=& \phi_{i,j}\frac{\partial x_j}{\partial t} &=& \phi_{i,j}\psi_j &=& -(\phi_k N_{k,j}\psi_j)N_i - (\phi_k M_{k,j}\psi_j)M_i \\ \frac{\partial \psi_i}{\partial s} &=& \psi_{i,j}\frac{\partial x_j}{\partial s} &=& \psi_{i,j}\phi_j &=& \frac{\partial \phi_i}{\partial t} \\ \frac{\partial \psi_i}{\partial t} &=& \psi_{i,j}\frac{\partial x_j}{\partial t} &=& \psi_{i,j}\psi_j &=& -(\psi_k N_{k,j}\psi_j)N_i - (\psi_k M_{k,j}\psi_j)M_i. \end{array} \tag{15}$$

We need to compute the quadratic forms involving the matrices $N_{k,j}$, $M_{k,j}$, and input vectors ϕ and ψ.

Consider

$$\begin{aligned}
(\phi_k N_{k,j}\phi_j)N_i + (\phi_k M_{k,j}\phi_j)M_i &= [\phi_k(u_{11}\bar{P}_{,kj} + u_{11,j}\bar{P}_{,k})\phi_j][u_{11}\bar{P}_{,i}] \\
&\quad + [\phi_k(u_{21}\bar{P}_{,kj} + u_{21,j}\bar{P}_{,k} + u_{22}\bar{Q}_{,kj} + u_{22,j}\bar{Q}_{,k})\phi_j][u_{21}\bar{P}_{,i} + u_{22}\bar{Q}_{,i}] \\
&= \left[\begin{array}{cc} \phi_k\bar{P}_{,kj}\phi_j & \phi_k\bar{Q}_{,kj}\phi_j \end{array}\right] \left[\begin{array}{cc} u_{11}^2 + u_{21}^2 & u_{21}u_{22} \\ u_{21}u_{22} & u_{22}^2 \end{array}\right] \left[\begin{array}{c} \bar{P}_{,i} \\ \bar{Q}_{,i} \end{array}\right] \\
&= \left[\begin{array}{cc} \phi_k\bar{P}_{,kj}\phi_j & \phi_k\bar{Q}_{,kj}\phi_j \end{array}\right] \left[\begin{array}{cc} \bar{P}_{,m}\bar{P}_{,m} & \bar{P}_{,m}\bar{Q}_{,m} \\ \bar{Q}_{,m}\bar{P}_{,m} & \bar{Q}_{,m}\bar{Q}_{,m} \end{array}\right]^{-1} \left[\begin{array}{c} \bar{P}_{,i} \\ \bar{Q}_{,i} \end{array}\right].
\end{aligned}$$

We have used the following information for the last displayed equality. Let A be the matrix whose columns are N and M. Let B be the matrix whose columns are $D\bar{P}$ and $D\bar{Q}$. Let $U = [u_{ij}]$ be the lower triangular matrix relating N and M to $D\bar{P}$ and $D\bar{Q}$. Gram-Schmidt orthonormalization yielded $A = BU^{\mathsf{T}}$ where A is orthogonal. Thus, $I = A^{\mathsf{T}}A = UB^{\mathsf{T}}BU^{\mathsf{T}}$, where I is the identity matrix, which implies $U^{\mathsf{T}}U = (B^{\mathsf{T}}B)^{-1}$. Similar formulas can be derived:

$$(\phi_k N_{k,j}\psi_j)N_i + (\phi_k M_{k,j}\psi_j)M_i = \left[\begin{array}{cc} \phi_k\bar{P}_{,kj}\psi_j & \phi_k\bar{Q}_{,kj}\psi_j \end{array}\right] \left[\begin{array}{cc} \bar{P}_{,m}\bar{P}_{,m} & \bar{P}_{,m}\bar{Q}_{,m} \\ \bar{Q}_{,m}\bar{P}_{,m} & \bar{Q}_{,m}\bar{Q}_{,m} \end{array}\right]^{-1} \left[\begin{array}{c} \bar{P}_{,i} \\ \bar{Q}_{,i} \end{array}\right]$$

and

$$(\psi_k N_{k,j}\psi_j)N_i + (\psi_k M_{k,j}\psi_j)M_i = \left[\begin{array}{cc} \psi_k\bar{P}_{,kj}\psi_j & \psi_k\bar{Q}_{,kj}\psi_j \end{array}\right] \left[\begin{array}{cc} \bar{P}_{,m}\bar{P}_{,m} & \bar{P}_{,m}\bar{Q}_{,m} \\ \bar{Q}_{,m}\bar{P}_{,m} & \bar{Q}_{,m}\bar{Q}_{,m} \end{array}\right]^{-1} \left[\begin{array}{c} \bar{P}_{,i} \\ \bar{Q}_{,i} \end{array}\right]$$

Note that surface traversal requires computing second derivatives of $\bar{P}$ and $\bar{Q}$. We have not derived these formulas because the numerical implementation of the traversal uses only first derivative information. In this implementation the quantities $\tilde{P}_{,i}$ and $\tilde{Q}_{,i}$ will again play the role of $\bar{P}_{,i}$ and $\bar{Q}_{,i}$, so an explicit construction of smoothly varying eigenvectors is not required. Section (???) is a discussion of this implementation.

1.8 d-Dimensional Ridges in $\mathbb{R}^n$

We present the general method for constructing d-dimensional ridges in $\mathbb{R}^n$ by computing the d-dimensional tangent spaces. The ideas are similar to what we have seen in earlier sections. Smoothly varying vectors are used for the directional derivatives whose zeros define the ridge. After algebraic reductions, it is shown that the tangent spaces can be calculated using the eigenvectors rather than explicitly constructing the smoothly varying vectors. The construction includes indices whose range is limited to either 1 through $n-d$ or $n-d+1$ through n. As a notational aid, indices in the range 1 through $n-d$ will be subscripted with a 0. Indices in the range $n-d+1$ through n will be subscripted with a 1. Indices without subscripts have the full range 1 through n. The summation convention is still used on the subscripted indices, but the sums are over the appropriate ranges.

The eigensystems are given by $f_{,ij}v_{jk} = v_{ij}\lambda_{jk}$ where λ_{jk} is a diagonal matrix whose diagonal entries are the eigenvalues of $f_{,ij}$ and where v_{jk} is an orthonormal matrix whose columns are the eigenvectors. We assume $\lambda_{n-d} < \lambda_{n-d+1}$ so that the eigenspaces corresponding to the first $n-d$ eigenvalues never swap or combine with the eigenspaces corresponding to the last d eigenvalues. Define $P_i = v_{ji}f_{,j}$. The d-dimensional ridge points are solutions to $P_i(x) = 0$ for $1 \leq i \leq n-d$ and $\lambda_{n-d}(x) < 0$.

Let $\mathbb{R}^n = S \oplus S^\perp$ be a direct sum of $\mathbb{R}^n$ into orthogonal subspaces S and $S^\perp$ where S is spanned by the first $n-d$ eigenvectors and $S^\perp$ is spanned by the remaining d eigenvectors. Let $\bar{v}_{ij}$ denote a smoothly varying orthonormal matrix whose first $n-d$ columns span S and whose remaining d columns span $S^\perp$; then

$$\bar{v}_{ii_0} = v_{ij_0} c_{j_0 i_0} \text{ and } \bar{v}_{ii_1} = v_{ij_1} c_{j_1 i_1},$$

where $c_{j_0 i_0}$ is an $(n-d)\times(n-d)$ orthogonal matrix and $c_{j_1 i_1}$ is a $d \times d$ orthogonal matrix. Define

$$\bar{P}_{i_0} = \bar{v}_{ii_0} f_{,i} = P_{j_0} c_{j_0 i_0} \text{ and } \bar{P}_{i_1} = \bar{v}_{ii_1} f_{,i} = P_{j_1} c_{j_1 i_1},$$

so $\bar{P}_{i_0} = 0$ if and only if $P_{i_0} = 0$. The ridge algorithm requires derivatives of $\bar{P}_{i_0}$, provided by the following construction.

Since $\bar{v}_{ij}$ is smooth and orthonormal, the derivatives satisfy

$$\bar{v}_{ii_0,j} = \bar{v}_{ii_1} a_{i_0 i_1 j} = v_{ij_1} c_{j_1 i_1} a_{i_0 i_1 j} \text{ and } \bar{v}_{ii_1,j} = -\bar{v}_{ii_0} a_{i_0 i_1 j} = -v_{ij_0} c_{j_0 i_0} a_{i_0 i_1 j}$$

for some choice of continuous tensors $a_{i_0 i_1 j}$. As in previous sections, rotations within S and $S^\perp$ are irrelevant in the construction, so $a_{i_0 j_0 k} = 0$ and $a_{i_1 j_1 k} = 0$. Differentiating $\bar{P}_{i_0}$ yields

$$\begin{aligned}
\bar{P}_{i_0,j} &= \bar{v}_{ii_0} f_{,ij} + \bar{v}_{ii_0,j} f_{,i} \\
&= f_{,ji} v_{ij_0} c_{j_0 i_0} + f_{,i} v_{ij_1} c_{j_1 i_1} a_{i_0 i_1 j} \\
&= v_{jk_0} \lambda_{k_0 j_0} c_{j_0 i_0} + P_{j_1} c_{j_1 i_1} a_{i_0 i_1 j} \\
&= c_{j_0 i_0} (v_{jk_0} \lambda_{k_0 j_0} + P_{j_1} c_{j_0 k_0} c_{j_1 i_1} a_{k_0 i_1 j}) \\
&= c_{j_0 i_0} (v_{jk_0} \lambda_{k_0 j_0} + P_{j_1} \psi_{j_0 j_1 j})
\end{aligned}$$

where $\psi_{j_0 j_1 j} = c_{j_0 i_0} c_{j_1 i_1} a_{i_0 i_1 j}$.

Now consider

$$\begin{aligned}
f_{,ij} \bar{v}_{ji_0} &= f_{,ij} v_{jj_0} c_{j_0 i_0} \\
&= v_{ik_0} \lambda_{k_0 j_0} c_{j_0 i_0} \\
&= \bar{v}_{i\ell_0} (c_{k_0 \ell_0} \lambda_{k_0 j_0} c_{j_0 i_0}) \\
&= \bar{v}_{i\ell_0} b_{\ell_0 i_0}
\end{aligned}$$

where $b_{\ell_0 i_0} = c_{k_0 \ell_0} \lambda_{k_0 j_0} c_{j_0 i_0}$. Contracting with $\bar{v}_{im_0}$ and using $\bar{v}_{im_0} \bar{v}_{i\ell_0} = \delta_{m_0 \ell_0}$ yields $b_{m_0 i_0} = \bar{v}_{im_1} f_{,ij} \bar{v}_{ji_1}$, which implies $b_{m_0 i_0}$ is differentiable. Similarly,

$$f_{,ij} \bar{v}_{ji_1} = \bar{v}_{ij_1} b_{j_1 i_1}$$

where $b_{j_1 i_1} = c_{k_1 j_1} \lambda_{k_1 \ell_1} c_{\ell_1 i_1} = \bar{v}_{ij_1} f_{,ij} \bar{v}_{ji_1}$ is differentiable. Differentiating the last displayed equation, substituting in the third derivatives of the $\bar{v}$ tensor, and using the definitions of the b tensors, yields

$$\begin{aligned}
f_{,ij} \bar{v}_{ji_1,k} + f_{,ijk} \bar{v}_{ji_1} &= \bar{v}_{ij_1} b_{j_1 i_1,k} + \bar{v}_{ij_1,k} b_{j_1 i_1}, \\
f_{,ij} (-\bar{v}_{ji_0} a_{i_0 i_1 k}) + f_{,ijk} \bar{v}_{ji_1} &= \bar{v}_{ij_1} b_{j_i i_1,k} + b_{j_1 i_1} (-\bar{v}_{ii_0} a_{i_0 j_1 k}), \\
-\bar{v}_{ij_0} f_{,ij} \bar{v}_{ji_0} a_{i_0 i_1 k} + f_{,ijk} \bar{v}_{ij_0} \bar{v}_{ji_1} &= \bar{v}_{ij_0} \bar{v}_{ij_1} b_{j_1 i_1,k} - \bar{v}_{ij_0} \bar{v}_{ii_0} b_{j_1 i_1} a_{i_0 j_1 k}, \\
b_{j_0 i_0} a_{i_0 i_1 k} - b_{j_1 i_1} a_{j_0 j_1 k} &= f_{,ijk} \bar{v}_{ij_0} \bar{v}_{ji_1},
\end{aligned}$$

where we have used $\bar{v}_{ij_0}\bar{v}_{ij_1} = 0$ and $\bar{v}_{ij_0}\bar{v}_{ii_0} = \delta_{j_0 i_0}$.

Contracting with $c_{k_0 j_0}$ and $c_{k_1 i_1}$ yields

$$\begin{aligned}
(c_{k_0 j_0} b_{j_0 i_0})(c_{k_1 i_1} a_{i_0 i_1 k} - (b_{j_1 i_1} c_{k_1 i_1})(c_{k_0 j_0} a_{j_0 j_1 k}) &= f_{,ijk}(\bar{v}_{ij_0} c_{k_0 j_0})(\bar{v}_{ji_1} c_{k_1 i_1}), \\
(\lambda_{k_0 \ell_0} c_{\ell_0 i_0})(c_{k_1 i_1} a_{i_0 i_1 k}) - (c_{\ell_1 j_1} \lambda_{\ell_1 k_1})(c_{k_0 j_0} a_{j_0 j_1 k}) &= f_{,ijk} v_{ik_0} v_{jk_1}, \\
\lambda_{k_0 \ell_0} \psi_{\ell_0 k_1 k} - \lambda_{\ell_1 k_1} \psi_{k_0 \ell_1 k} &= f_{,ijk} v_{ik_0} v_{jk_1}, \\
\Delta_{k_0 k_1 i_0 i_1} \psi_{i_0 i_1 k} &= f_{,ijk} v_{ik_0} v_{jk_1},
\end{aligned}$$

where $\Delta_{k_0 k_1 i_0 i_1} = 0$ if $i_0 \neq k_0$ or $i_1 \neq k_1$, and $\Delta_{k_0 k_1 i_0 i_1} = \lambda_{k_0 k_0} - \lambda_{k_1 k_1}$ if $i_0 = k_0$ and $i_1 = k_1$ (no sum over k_0 or k_1). Consequently,

$$\psi_{i_0 i_1 k} = \Delta^{-1}_{i_0 i_1 k_0 k_1} f_{,ijk} v_{ik_0} v_{jk_1},$$

where $\Delta^{-1}_{i_0 i_1 k_0 k_1} = 0$ if $k_0 \neq i_0$ or $k_1 \neq i_1$, and $\Delta^{-1}_{i_0 i_1 k_0 k_1} = (\lambda_{i_0 i_0} - \lambda_{i_1 i_1})^{-1}$ if $k_0 = i_0$ and $k_1 = i_1$ (no sum over i_0 or i_1).

Substituting into the formula for $\bar{P}_{i_0,k}$ yields

$$\bar{P}_{i_0,k} = c_{j_0 i_0}(v_{kk_0} \lambda_{k_0 j_0} + P_{j_1} \Delta^{-1}_{j_0 j_1 \ell_0 \ell_1} f_{,ijk} v_{i\ell_0} v_{j\ell_1}).$$

These formulas will be used both in finding an initial ridge point and in determining the tangent spaces at each point on the d-dimensional ridge. The matrix $C_0 = [c_{j_0 i_0}]$ is an unknown quantity, but the ridge flow is independent of C_0 and the ridge traversal only requires knowing $\det(C_0) = \pm 1$. Define

$$\tilde{P}_{i_0,k} = v_{kk_0} \lambda_{k_0 i_0} + P_{j_1} \Delta^{-1}_{i_0 j_1 \ell_0 \ell_1} f_{,ijk} v_{i\ell_0} v_{j\ell_1}. \tag{16}$$

These quantities will in effect play the role of the derivatives of $\bar{P}_{i_0}$ despite the fact that they are not necessarily the derivatives of some functions $\tilde{P}_{i_0}$. The use of index/derivative notation is used suggestively to remind us how the quantities were obtained.

1.8.1 Ridge Flow

Given an initial approximation $\mathcal{A}$ to a ridge point, a flow path to the ridge is determined by gradient descent. Ridge points occur as absolute minimum points for the function $P_{i_0}(x) P_{i_0}(x)/2$. Note that $\bar{P}_{i_0} \bar{P}_{i_0,j} = P_{k_0} c_{k_0 i_0} c_{j_0 i_0} \tilde{P}_{j_0,j} = P_{j_0} \tilde{P}_{j_0,j}$. Therefore, the gradient descent is modeled by

$$\frac{dx_i(t)}{dt} = -P_{i_0}(x(t)) \tilde{P}_{i_0,i}(x(t)), \;\; x_i(0) = \mathcal{A}_i, \;\; 1 \leq i \leq n. \tag{17}$$

The solution curve terminates at time $T > 0$ if $P_{i_0}(x(T)) = 0$ or if a positive local minimum is reached, in which case a different starting point should be used. The point $\mathcal{R} = x(T)$ will be used as the starting ridge point for ridge traversal.

1.8.2 Ridge Traversal

As in previous sections, the construction provides geodesic coordinates for a local traversal of the surface. The gradients $\bar{P}_{i_0,i}$ are normals to the d-dimensional manifold which is the intersection of the $n - d$ hypersurfaces defined implicitly by $\bar{P}_{i_0} = 0$. By Gram-Schmidt orthonormalization we can find $n - d$ smoothly varying orthonormal vectors N_{ii_0} (written as an $n \times (n-d)$ matrix) such that

$$N_{ii_0} = u_{i_0 j_0} \bar{P}_{j_0,i}$$

where the $u_{i_0 j_0}$ are differentiable and $u_{i_0 j_0} = 0$ for $i_0 < j_0$. Let ϕ_{ii_1} represent smoothly varying, unit length, tangent vectors to the ridge surface. The derivatives of the normals and tangents are related by

$$\begin{aligned} \phi_{ii_1,j} &= N_{ii_0} a_{i_0 i_1 j}, \\ N_{ii_0,j} &= -\phi_{ii_1} a_{i_0 i_1 j} + N_{ij_0} a_{j_0 i_0 j}, \end{aligned}$$

for some skew-symmetric continuous tensor a_{ijk}. Without loss of generality, the tangent derivatives are chosen not to depend on tangent vectors themselves; that is, rotations within the tangent space are irrelevant to the construction. However, note that the normal derivatives do have dependence on normal vectors. Differentiating the normals yields

$$N_{ii_0,j} = u_{i_0 j_0} \bar{P}_{j_0,ij} + u_{i_0 j_0,j} \bar{P}_{j_0,i}.$$

Contracting with the tangent vectors produces

$$a_{i_0 j_1 j} = -\phi_{ij_1} N_{ii_0,j}.$$

Let $x(s)$ be a parameterization of the surface defined implicitly by $\bar{P}_{i_0}(x(s)) = 0$ where $s \in \mathbb{R}^d$. Ridge traversal will be according to the system of partial differential equations $\partial x_i / \partial s_{j_1} = \phi_{ij_1}(x)$. The equality of mixed partial derivatives, $\partial^2 x_i / \partial s_{j_1 k_1} = \partial^2 x_i / \partial s_{k_1 j_1}$, imposes the constraint on the tangent vectors: $\phi_{ij_1,k}\phi_{kk_1} \equiv \phi_{ik_1,k}\phi_{kj_1}$. But this condition already holds for our choice of tangents and normals since

$$\begin{aligned} \phi_{ij_1,k}\phi_{kk_1} - \phi_{ik_1,k}\phi_{kj_1} &= (N_{ii_0} a_{i_0 j_1 k})\phi_{kk_1} - (N_{ii_0} a_{i_0 k_1 k})\phi_{kj_1} \\ &= N_{ii_0}(a_{i_0 j_1 k}\phi_{kk_1} - a_{i_0 k_1 k}\phi_{kj_1}) \\ &= N_{ii_0}(-\phi_{ij_1} N_{ii_0,k}\phi_{kk_1} + \phi_{ik_1} N_{ii_0,k}\phi_{kj_1}) \\ &= N_{ii_0}[-\phi_{ij_1}(u_{i_0 j_0}\bar{P}_{j_0,ik})\phi_{kk_1} + \phi_{ik_1}(u_{i_0 j_0}\bar{P}_{j_0,ik})\phi_{kj_1}] \\ &= N_{ii_0} u_{i_0 j_0}(-\phi_{ij_1}\bar{P}_{j_0,ik}\phi_{kk_1} + \phi_{ik_1}\bar{P}_{j_0,ik}\phi_{kj_1}) \\ &= 0 \end{aligned}$$

where we have used the symmetry $\bar{P}_{j_0,ik} = \bar{P}_{j_0,ki}$.

The system of partial differential equations which governs ridge traversal is

$$\begin{aligned} \frac{\partial x_i}{\partial s_{i_1}} &= \phi_{ii_1}, \\ \frac{\partial \phi_{ii_1}}{\partial s_{j_1}} &= \phi_{ii_1,j}\frac{\partial x_j}{\partial s_{j_1}} \\ &= \phi_{ii_1,j}\phi_{jj_1} \\ &= N_{ii_0}(-\phi_{ki_1} N_{ki_0,j}\phi_{jj_1}) \\ &= -[\phi_{ki_1}(u_{i_0 j_0}\bar{P}_{j_0,kj})\phi_{jj_1}][u_{i_0 k_0}\bar{P}_{k_0,i}] \\ &= -(\phi_{ki_1}\bar{P}_{j_0,kj}\phi_{jj_1})(u_{i_0 j_0} u_{i_0,k_0})\bar{P}_{k_0,i} \\ &= -(\phi_{ki_1}\bar{P}_{j_0,kj}\phi_{jj_1})(\bar{P}_{j_0,m}\bar{P}_{k_0,m})^{-1}\bar{P}_{k_0,i}. \end{aligned} \tag{18}$$

We have used the following information for the last displayed equality:

$$\delta_{i_0 j_0} = N_{ii_0} N_{ij_0} = (u_{i_0 k_0} \bar{P}_{k_0,i})(u_{j_0 m_0} \bar{P}_{m_0,i}) = u_{i_0 k_0} (\bar{P}_{k_0,i} \bar{P}_{m_0,i}) u_{j_0 m_0}.$$

The $(n-d) \times (n-d)$ matrices U and U^{T} may be inverted to obtain $u_{i_0 j_0} u_{i_0 k_0} = (\bar{P}_{j_0,m} \bar{P}_{k_0,m})^{-1}$ where the inverse operation is applied to the matrix implied by the contraction of $\bar{P}$ with itself. The traversal requires computing second derivatives of $\bar{P}$, which we have not derived here. The numerical implementation, discussed in Section (???), only requires first derivatives of $\bar{P}$.

1.9 Ridges Implementation

The Euclidean height ridge implementations are found in the applications directory, `magic3/appls/ridge?d`, where `?` is 2, 3, or 4. The code is in `ridge?d.zip`.

2 Ridges in Riemannian Geometry

Chapter 3 discussed the fundamental concepts of restricted local extrema and height ridges. The concepts were applied to functions $f : \mathbb{R}^n \to \mathbb{R}$ where $\mathbb{R}^n$ is the set of n-tuples of real numbers. An implicit assumption was made that $\mathbb{R}^n$, as a geometric entity, is standard Euclidean space whose metric tensor is the identity. The same concepts are definable even if $\mathbb{R}^n$ is assigned an arbitrary positive definite metric tensor. The extension to Riemannian geometry requires tensor calculus which is discussed in Section (???). Most notably the constructions involve the ideas of covariant and contravariant tensors and of covariant differentiation.

We present in this chapter the reiteration of the definitions for restricted local maxima and height ridges, but in the context of a metric assigned to $\mathbb{R}^n$, call it g_{ij}. The corresponding Christoffel symbols are denoted Γ^k_{ij} and the covariant derivatives through order three are given by

$$f_{,i} = \frac{\partial f}{\partial x_i}, \quad f_{,ij} = \frac{\partial f_{,i}}{\partial x_j} - \Gamma^\ell_{ij} f_{,\ell}, \quad \text{and} \quad f_{,ijk} = \frac{\partial f_{,ij}}{\partial x_k} - \Gamma^\ell_{ik} f_{,\ell j} - \Gamma^\ell_{jk} f_{,i\ell}.$$

Recall that the covariant and contravariant representations of a vector v are related by $v_i = g_{ij} v^j$ and $v^i = g^{ij} v_j$.

2.1 Restricted Local Extrema

Let $f \in C^2(\mathbb{R}^n, \mathbb{R})$ where $\mathbb{R}^n$ is assigned a Riemannian geometry. A point x is a critical point for f if $v^i f_{,i}(x) = 0$ for all directions v. At a critical point, $f(x)$ is a strict local maximum if $v^i f_{,ij}(x) v^j < 0$ for all directions v. Let v_1 through v_n be linearly independent vectors and define the tensor $v^r_{\cdot c}$ where, as a matrix, the c^{th} column is the vector v_c. The test for local maximum points becomes $v^j_{\cdot i} f_{,j} = 0$ and $v^k_{\cdot i} v^\ell_{\cdot j} f_{,k\ell}$ is negative definite. It is possible to restrict attention to only $n-d$ directions for some d with $0 \le d < n$. Let the directions be denoted v_1 through v_{n-d} and define the $n \times (n-d)$ matrix $V = [v^r_{\cdot c}]$ where $1 \le r \le n$ and $1 \le c \le n-d$.

Definition 3 (Restricted Local Maximum). *A point x is a restricted local maximum point of type d relative to V if $f(x)$ is a local maximum in the affine space $x + \langle V \rangle$. The test for such a point is therefore*

$v^j_{\cdot i} f_{,j} = 0$ *for* $1 \le i \le n-d$ *and* $v^k_{\cdot i} v^\ell_{\cdot j} f_{,k\ell}$ *is an* $(n-d) \times (n-d)$ *negative definite tensor.*

2.2 Height Ridge Definition

The ideas for height ridges in Euclidean space carry over immediately to Riemannian spaces. We present the general height ridge definition and provide summaries of its application to various dimensions n and various ridge dimensions d.

Definition 4 (Height Ridge Definition). *Let* $f \in C^2(\mathbb{R}^n, \mathbb{R})$ *where* $\mathbb{R}^n$ *is assigned a Riemannian geometry. Let* λ_i *and* v_i, $1 \le i \le n$ *be the generalized eigenvalues and eigenvectors for* $f_{,ij}$ *in the following sense. Define the diagonal tensor* $\lambda^r_{\cdot c}$ *whose* i^{th} *entry is* λ_i *and where* $\lambda_1 \le \cdots \le \lambda_n$. *Define the tensor* $v^r_{\cdot c}$ *whose* c^{th} *column as a matrix is the vector* v_c *and for which* $v^k_{\cdot i} v_{kj} = \delta_{ij}$. *Finally, let the tensors satisfy* $f_{,ij} v^j_{\cdot k} = v_{ij} \lambda^j_{\cdot k}$.

A point x *is a* d*-dimensional ridge point if it is a restricted local maximum point of type* d *with respect to* $v^r_{\cdot c}$. *Since* $v^k_{\cdot i} f_{,k\ell} v^\ell_{\cdot j} = \delta_{i\ell} \lambda^\ell_{\cdot j}$ *is diagonal and since the eigenvalues are ordered, the test for a ridge point reduces to* $v^j_{\cdot i} f_{,j}(x) = 0$ *for* $1 \le i \le n-d$ *and* $\lambda_{n-d}(x) < 0$.

2.2.1 1-Dimensional Ridges in $\mathbb{R}^2$

We summarize the height ridge definition which was discussed in detail in Section (???), but extend it to the case of Riemannian geometry. Let $f_{,ij} u^j = \alpha u_i$ and $f_{,ij} v^j = \beta v_i$ where $\alpha \le \beta$, $u_i u^i = 1$, $v_i v^i = 1$, and $u_i v^i = 0$. Define $P = u^i f_{,i}$ and $Q = v^i f_{,i}$. According to the height ridge definition, a point $x \in \mathbb{R}^2$ is a 1-dimensional ridge point if $P(x) = 0$ and $\alpha(x) < 0$.

Equation (1) generalizes to

$$P_{,k} = \alpha u_k + \frac{Q}{\alpha - \beta} f_{,ijk} v^i u^j, \tag{19}$$

which specifies the covariant derivative of P. The model for ridge flow given by equation (2) generalizes as follows. As a parameterized curve $x^i(t)$, the tangent to the ridge flow path is the contravariant vector dx^i/dt. The gradient $P_{,i}(x)$ is covariant, so we need its contravariant counterpart. Thus, the ridge flow is modeled by

$$\frac{dx^i}{dt} = -P g^{ij} P_{,j} = -P P^i_{,\cdot} \ \ , \ \ x^i(0) = \mathcal{A}^i, \ \ i = 1, 2, \tag{20}$$

where $\mathcal{A}$ is an initial approximation to the ridge. If the flow terminates at time $T > 0$ where $P(x(T)) = 0$, then $\mathcal{R} = x(T)$ is used as the starting ridge point for ridge traversal. Ridge traversal given by equation (3) generalizes to

$$\frac{dx^i}{dt} = \pm g^{ij} e_{jk} g^{k\ell} P_{,\ell} = \pm e^{ij} P_{,j} \ \ , \ \ x^i(0) = \mathcal{R}^i(0), \ \ i = 1, 2. \tag{21}$$

Note that raising of indices on both covariant tensors e_{ij} and $P_{,i}$ is required to produce a contravariant vector dx^i/dt.

2.2.2 1-Dimensional Ridges in $\mathbb{R}^3$

The height ridge definition which was discussed in detail in Section (???) is extended to the case of Riemannian geometry. Let $f_{,ij}u^j = \alpha u_i$, $f_{,ij}v^j = \beta v_i$, and $f_{,ij}w^j = \gamma w_i$ where $\alpha \le \beta \le \gamma$, $u_i u^i = v_i v^i = w_i w^i = 1$, and $u_i v^i = u_i w^i = v_i w^i = 0$. Define $P = u^i f_{,i}$, $Q = v^i f_{,i}$, and $R = w^i f_{,i}$. According to the height ridge definition, a point $x \in \mathbb{R}^2$ is a 1-dimensional ridge point if $P(x) = 0$, $Q(x) = 0$, and $\beta(x) < 0$.

Equation (4) generalizes to

$$\begin{bmatrix} \tilde{P}_{,k} \\ \tilde{Q}_{,k} \end{bmatrix} = \begin{bmatrix} \alpha u_k + \frac{R}{\alpha-\gamma} f_{,ijk} u^i w^j \\ \beta v_k + \frac{R}{\beta-\gamma} f_{,ijk} v^i w^j \end{bmatrix} \tag{22}$$

which specifies the quantities that play the role of the covariant derivatives of P and Q despite the fact that neither $\tilde{P}_{,k}$ nor $\tilde{Q}_{,k}$ are the covariant derivatives of some tensors $\tilde{P}$ or $\tilde{Q}$. The model for ridge flow given by equation (5) generalizes to

$$\frac{dx^i}{dt} = -g^{ij}\left(P\tilde{P}_{,j} + Q\tilde{Q}_{,j}\right) = -P\tilde{P}^i_{,\cdot} - Q\tilde{Q}^i_{,\cdot} \;\;,\;\; x^i(0) = \mathcal{A}^i, \;\; i = 1, 2, 3, \tag{23}$$

where $\mathcal{A}$ is an initial approximation to the ridge. If the flow terminates at time $T > 0$ where $P(x(T)) = 0$, then $\mathcal{R} = x(T)$ is used as the starting ridge point for ridge traversal. Ridge traversal given by equation (6) generalizes to

$$\frac{dx^i}{dt} = \pm g^{ij} e_{jk\ell} g^{kr} g^{\ell s} \tilde{P}_{,r} \tilde{Q}_{,s} = \pm e^{ijk} \tilde{P}_{,j} \tilde{Q}_{,k} \;\;,\;\; x^i(0) = \mathcal{R}^i(0), \;\; i = 1, 2, 3. \tag{24}$$

2.2.3 1-Dimensional Ridges in $\mathbb{R}^n$

The general height ridge definition for 1-dimensional ridges which was discussed in detail in Section (???) is extended to the case of Riemannian geometry. Let $f_{,ij} v^j_{\cdot k} = v^i_{\cdot j} \lambda^j_{\cdot k}$ where λ is a diagonal tensor whose j^{th} diagonal entry is the generalized eigenvalue corresponding to the generalized eigenvector which is the j^{th} column of $v^i_{\cdot j}$ treated as an $n \times n$ matrix. Define $P_j = v^i_{\cdot j} f_{,i}$. According to the height ridge definition, a point $x \in \mathbb{R}^n$ is a 1-dimensional ridge point if $P_j(x) = 0$ for $1 \le j \le n-1$ and $\lambda_{n-1}(x) < 0$.

Using the same convention for index notation as in Section (???), we can extend the equations in that section. Equation (7) generalizes to

$$\tilde{P}_{i_0,k} = v_{kj_0} \lambda^{j_0}_{i_0} + R\Delta^{-1}_{i_0 j_0} f_{,ijk} v^i_{j_0} w^j. \tag{25}$$

where $\Delta^{-1}_{i_0 j_0}$ is the diagonal inverse tensor for the diagonal tensor $\Delta_{i_0 j_0}$ whose k_0^{th} diagonal entry is $\lambda_{k_0} - \gamma$. The vector w is the generalized eigenvector corresponding to eigenvalue $\gamma = \lambda_n$ and the value $R = w^i f_{,i}$. The summation over j_0 is not intended as a tensor summation, but is a simple arithmetic sum, so the rule for pairing a contravariant and a covariant index does not apply here. The model for ridge flow given by equation (8) generalizes to

$$\frac{dx^i}{dt} = -g^{ij} P_{i_0} \tilde{P}_{i_0,j} = P_{i_0} \tilde{P}^i_{i_0,\cdot} \;\;,\;\; x^i(0) = \mathcal{A}^i, \;\; 1 \le i \le n \tag{26}$$

where $\mathcal{A}$ is an initial approximation to the ridge. If the flow terminates at time $T > 0$ where $P(x(T)) = 0$, then $\mathcal{R} = x(T)$ is used as the starting ridge point for ridge traversal. The summation over i_0 is not intended as a tensor summation, but is just a simple arithmetic one, so the index convention of pairing a covariant with a contravariant index does not apply here. Finally, ridge traversal in equation (9) generalizes to

$$\frac{dx^i}{dt} = \pm g^{ij} e_{j\,j_1 \cdots j_{n-1}} g^{j_1 i_1} \tilde{P}_{1\,i_1} \cdots g^{j_{n-1} i_{n-1}} \tilde{P}_{n-1\,i_{n-1}} = \pm e^{i\,i_1 \cdots i_{n-1}} \tilde{P}^{i_1}_{1\cdot} \cdots \tilde{P}^{i_{n-1}}_{n-1\cdot} \;\;,\;\; x^i(0) = \mathcal{R}_i,\;\; 1 \le i \le n. \tag{27}$$

2.2.4 2-Dimensional Ridges in $\mathbb{R}^3$

The height ridge definition which was discussed in detail in Section (???) is extended to the case of Riemannian geometry. Let $f_{,ij}u^j = \alpha u_i$, $f_{,ij}v^j = \beta v_i$, and $f_{,ij}w^j = \gamma w_j$ where $\alpha \le \beta \le \gamma$, $u_i u^i = v_i v^i = w_i w^i = 1$, and $u_i v^i = u_i w^i = v_i w^i = 0$. Define $P = u^i f_{,i}$, $Q = v^i f_{,i}$, and $R = w^i f_{,i}$. According to the height ridge definition, a point $x \in \mathbb{R}^2$ is a 2-dimensional ridge point if $P(x) = 0$, and $\alpha(x) < 0$.

Equation (10) generalizes to

$$P_{,k} = \alpha u_k + \frac{Q}{\alpha - \beta} f_{,ijk} v^i u^j + \frac{R}{\alpha - \gamma} f_{,ijk} w^i u^j. \tag{28}$$

which specifies the covariant derivative of P. Ridge flow given by equation (11) generalizes to

$$\frac{dx^i}{dt} = -g^{ij} P P_{,j} = -P P^{i}_{,\cdot} \;\;,\;\; x_i(0) = \mathcal{A}_i,\;\; i = 1, 2, 3, \tag{29}$$

where $\mathcal{A}$ is an initial approximation to the ridge. If the flow terminates at time $T > 0$ where $P(x(T)) = 0$, then $\mathcal{R} = x(T)$ is used as the starting ridge point for ridge traversal. Ridge traversal given by equation (12) generalizes to

$$\begin{array}{ll} \frac{\partial x^i}{\partial s} = \phi^i, & \frac{\partial x^i}{\partial t} = \psi^i, \\ \frac{\partial \phi^i}{\partial s} = -\left(\phi^j \frac{P_{,jk}}{|DP|} \phi^k\right) N^i, & \frac{\partial \phi^i}{\partial t} = -\left(\phi^j \frac{P_{,jk}}{|DP|} \psi^k\right) N^i, \\ \frac{\partial \psi^i}{\partial s} = -\left(\psi^j \frac{P_{,jk}}{|DP|} \phi^k\right) N^i, & \frac{\partial \psi^i}{\partial t} = -\left(\psi^j \frac{P_{,jk}}{|DP|} \psi^k\right) N^i \end{array} \tag{30}$$

where $|DP| = sqrt P_{,i} g^{ij} P_{,j}$, N is a unit normal to the surface, and ϕ and ψ are unit tangent vectors to the surface. The covariant form for the normal is $N_i = P_{,i}/|DP|$ and the contravariant form is $N^i = g^{ij} N_j$. As in Section (???), this system of equations generates a local traversal for the surface, but does not necessarily allow a global traversal.

2.2.5 2-Dimensional Ridges in $\mathbb{R}^4$

The height ridge definition which was discussed in detail in Section (???) is extended to the case of Riemannian geometry. Let $f_{,ij}u^j = \alpha u_i$, $f_{,ij}v^j = \beta v_i$, $f_{,ij}w^j = \gamma w_j$, and $f_{,ij}\xi_j = \delta \xi_j$ where $\alpha \le \beta \le \gamma \le \delta$ and u, v, w, and ξ form a right-handed orthonormal system with respect to the metric. Define $P = u^i f_{,i}$, $Q = v^i f_{,i}$, $R = w^i f_{,i}$, and $S = \xi^i f_{,i}$. According to the height ridge definition, a point $x \in \mathbb{R}^4$ is a 2-dimensional ridge point if $P(x) = 0$, $Q(x) = 0$, and $\beta(x) < 0$.

Equation (13) generalizes to

$$\begin{bmatrix} \tilde{P}_{,k} \\ \tilde{Q}_{,k} \end{bmatrix} = \begin{bmatrix} \alpha u_k + \frac{R}{\alpha - \gamma} f_{,ijk} u^i w^j + \frac{S}{\alpha - \delta} f_{,ijk} u^i \xi^j \\ \beta v_k + \frac{R}{\beta - \gamma} f_{,ijk} v^i w^j + \frac{S}{\beta - \delta} f_{,ijk} v^i \xi^j \end{bmatrix} \tag{31}$$

which specifies the quantities that play the role of the covariant derivatives of P and Q despite the fact that neither $\tilde{P}_{,k}$ nor $\tilde{Q}_{,k}$ are the covariant derivatives of some tensors $\tilde{P}$ or $\tilde{Q}$. The model for ridge flow given by equation (14) generalizes to

$$\frac{dx^i}{dt} = -g^{ij}\left(P\tilde{P}_{,j} + Q\tilde{Q}_{,j}\right) = -P\tilde{P}^{i}_{,\cdot} - Q\tilde{Q}^{i}_{,\cdot} \ \ , \ \ x^i(0) = \mathcal{A}^i, \ \ i = 1,2,3,4, \tag{32}$$

where $\mathcal{A}$ is an initial approximation to the ridge. If the flow terminates at time $T > 0$ where $P(x(T)) = 0$, then $\mathcal{R} = x(T)$ is used as the starting ridge point for ridge traversal. Ridge traversal given by equation (15) generalizes to

$$\begin{array}{rcl}
\frac{\partial x^i}{\partial s} & = & \phi^i \\
\frac{\partial x^i}{\partial t} & = & \psi^i \\
\frac{\partial \phi^i}{\partial s} & = & -(\phi^k N_{k,j}\phi^j)N^i - (\phi^k M_{k,j}\phi^j)M^i \\
\frac{\partial \phi^i}{\partial t} & = & -(\phi^k N_{k,j}\psi^j)N^i - (\phi^k M_{k,j}\psi^j)M_i \\
\frac{\partial \psi_i}{\partial s} & = & \frac{\partial \phi^i}{\partial t} \\
\frac{\partial \psi^i}{\partial t} & = & -(\psi^k N_{k,j}\psi^j)N^i - (\psi^k M_{k,j}\psi^j)M^i,
\end{array} \tag{33}$$

where the covariant forms of N and M are as in Section (???) and where

$$(\phi^k N_{k,j}\psi^j)N^i + (\phi^k M_{k,j}\psi^j)M^i = g^{ir}\left[\begin{array}{cc} \phi^k \bar{P}_{,kj}\psi^j & \phi^k \bar{Q}_{,kj}\psi^j \end{array}\right]\left[\begin{array}{cc} \bar{P}^{m}_{,\cdot}\bar{P}_{,m} & \bar{P}^{m}_{,\cdot}\bar{Q}_{,m} \\ \bar{Q}^{m}_{,\cdot}\bar{P}_{,m} & \bar{Q}^{m}_{,\cdot}\bar{Q}_{,m} \end{array}\right]^{-1}\left[\begin{array}{c} \bar{P}_{,r} \\ \bar{Q}_{,r} \end{array}\right]$$

and

$$(\psi^k N_{k,j}\psi^j)N^i + (\psi^k M_{k,j}\psi^j)M^i = g^{ir}\left[\begin{array}{cc} \psi^k \bar{P}_{,kj}\psi^j & \psi^k \bar{Q}_{,kj}\psi^j \end{array}\right]\left[\begin{array}{cc} \bar{P}^{m}_{,\cdot}\bar{P}_{,m} & \bar{P}^{m}_{,\cdot}\bar{Q}_{,m} \\ \bar{Q}^{m}_{,\cdot}\bar{P}_{,m} & \bar{Q}^{m}_{,\cdot}\bar{Q}_{,m} \end{array}\right]^{-1}\left[\begin{array}{c} \bar{P}_{,r} \\ \bar{Q}_{,r} \end{array}\right].$$

2.2.6 d-Dimensional Ridges in $\mathbb{R}^n$

The general height ridge definition for d-dimensional ridges which was discussed in detail in Section (???) is extended to the case of Riemannian geometry. Let $f_{,ij}v^{j}_{\cdot k} = v^{i}_{\cdot j}\lambda^{j}_{\cdot k}$ where λ is a diagonal tensor whose j^{th} diagonal entry is the generalized eigenvalue corresponding to the generalized eigenvector which is the j^{th} column of $v^{i}_{\cdot j}$ treated as an $n \times n$ matrix. Define $P_j = v^{i}_{\cdot j}f_{,i}$. According to the height ridge definition, a point $x \in \mathbb{R}^n$ is a 1-dimensional ridge point if $P_j(x) = 0$ for $1 \leq j \leq n - d$ and $\lambda_{n-d}(x) < 0$.

Using the same convention for index notation as in Section (???), we can extend the equations in that section. Equation (16) generalizes to

$$\tilde{P}_{i_0,k} = v_{kk_0}\lambda^{k_0}_{i_0} + P_{j_1}\Delta^{-1}_{i_0 j_1 \ell_0 \ell_1} f_{,ijk} v_{i\ell_0} v_{j\ell_1}. \tag{34}$$

where $\Delta^{-1}_{k_0 k_1 i_0 i_1} = 0$ if $i_0 \neq k_0$ or $i_1 \neq k_1$ and $\Delta^{-1}_{k_0 k_1 i_0 i_1} = (\lambda_{k_0} - \lambda_{k_1})^{-1}$ if $k_0 = i_0$ and $k_1 = i_1$. The summations over j_1, ℓ_0, and ℓ_1 are not intended as tensor summations, but are simple arithmetic sums, so the rule for pairing contravariant and covariant indices does not apply here. The model for ridge flow given by equation (17) generalizes to

$$\frac{dx^i(t)}{dt} = -g^{ij}P_{i_0}\tilde{P}_{i_0,j} = -P_{i_0}\tilde{P}^{i}_{i_0,\cdot} \ \ , \ \ x^i(0) = \mathcal{A}^i, \ \ 1 \leq i \leq n \tag{35}$$

where $\mathcal{A}$ is an initial approximation to the ridge. If the flow terminates at time $T > 0$ where $P(x(T)) = 0$, then $\mathcal{R} = x(T)$ is used as the starting ridge point for ridge traversal. The summation over i_0 is not intended as a tensor summation, but is just a simple arithmetic one, so the index convention of pairing a covariant with a contravariant index does not apply here. Finally, ridge traversal in equation (18) generalizes to

$$\begin{aligned} \frac{\partial x^i}{\partial s_{i_1}} &= \phi^i_{i_1}, \\ \frac{\partial \phi^i_{i_1}}{\partial s_{j_1}} &= -(\phi^k_{i_1} \bar{P}_{j_0,kj} \phi^j_{j_1})(\bar{P}^m_{j_0,\cdot} \bar{P}_{k_0,m})^{-1} g^{ir} \bar{P}_{k_0,r} \end{aligned} \tag{36}$$

where the summations over j_0 and k_0 are not intended as tensor summations, but are just simple arithmetic ones. The inverse operation indicates computing the inverse of the matrix given by the tensor product $\bar{P}^m_{j_0,\cdot} \bar{P}_{k_0,m}$.

3 Ridges of Functions Defined on Manifolds

In Section (???) we apply the definitions of Chapter 4 to finding height ridges of functions defined on manifolds. The first case is simplest where we construct 1-dimensional ridges of a function defined on a 2-dimensional surface embedded in $\mathbb{R}^3$. The second case is the most general where we construct d-dimensional ridges of a function defined on an n-dimensional manifold embedded in $\mathbb{R}^p$. Section (???) provides an alternative definition for ridges based on principal curvatures and principal directions. This definition has proved useful in registration of 3-dimensional images via embedded 2-dimensional manifolds (REFERENCE French people). Section (???) discusses a ridge definition which has also been used successfully. This definition is essentially an extension of the principal direction definition.

3.1 Height Ridge Definition

3.1.1 1-Dimensional Ridges in $M^2 \subset \mathbb{R}^3$

Let $\bar{x} : \mathbb{R}^2 \to \mathbb{R}^3$ define a surface where the parameterization is denoted $\bar{x}(x)$. Assume that the tangent vectors $\partial\bar{x}/\partial x^1$ and $\partial\bar{x}/\partial x^2$ are always linearly independent. A unit normal to the surface is therefore

$$N = \frac{\frac{\partial \bar{x}}{\partial x^1} \times \frac{\partial \bar{x}}{\partial x^2}}{\left| \frac{\partial \bar{x}}{\partial x^1} \times \frac{\partial \bar{x}}{\partial x^2} \right|}.$$

As discussed in Section (???), the metric tensor for the surface is

$$g_{ij} = \frac{\partial \bar{x}}{\partial x^i} \cdot \frac{\partial \bar{x}}{\partial x^j}$$

and the Christoffel symbols are

$$\Gamma^k_{ij} = \frac{\partial^2 \bar{x}}{\partial x^i \partial x^j} \cdot g^{k\ell} \frac{\partial \bar{x}}{\partial x^\ell}.$$

If $f : \mathbb{R}^2 \to \mathbb{R}$ is a function defined on the same parameter space as the surface, say $f = f(x)$, then the height ridge definition in Chapter 4 may be directly applied to f.

However, a more typical application occurs where the function values at points on the surface are inherited from a function $\phi : \mathbb{R}^3 \to \mathbb{R}$ by the embedded surface. In this case the function is $f(x) = \phi(\bar{x}(x))$. The height ridge definition requires computing up through third order covariant derivatives of f. Some calculations will show that

$$\begin{aligned} f_{,i} &= \phi_{,\ell}\bar{x}_{\ell,i} \\ f_{,ij} &= \phi_{,\ell}\bar{x}_{\ell,ij} + \phi_{,\ell m}\bar{x}_{\ell,i}\bar{x}_{m,j} \\ f_{,ijk} &= \phi_{,\ell}\bar{x}_{\ell,ijk} + \phi_{,\ell m}\left(\bar{x}_{\ell,i}\bar{x}_{m,jk} + \bar{x}_{\ell,j}\bar{x}_{m,ik} + \bar{x}_{\ell,k}\bar{x}_{m,ij}\right) + \phi_{,\ell mn}\bar{x}_{\ell,i}\bar{x}_{m,j}\bar{x}_{n,k} \end{aligned}$$

where $\phi_{,\ell}$, $\phi_{,\ell m}$, and $\phi_{,\ell mn}$ are the Cartesian partial derivatives up through third order of ϕ. The covariant derivatives of the surface are

$$\begin{aligned} \bar{x}_{m,i} &= \tfrac{\partial \bar{x}_m}{\partial x^i} \\ \bar{x}_{m,ij} &= \tfrac{\partial \bar{x}_{m,i}}{\partial x^j} - \Gamma^{\ell}_{ij}\bar{x}_{m,\ell} = -b_{ij}N_m \\ \bar{x}_{m,ijk} &= \tfrac{\partial \bar{x}_{m,ij}}{\partial x^k} - \Gamma^{\ell}_{ik}\bar{x}_{m,\ell j} - \Gamma^{\ell}_{jk}\bar{x}_{m,i\ell} = -b_{ij,k}N_m - b_{ij}b^{\ \ell}_{k\cdot}\bar{x}_{m,\ell} \end{aligned}$$

where b_{ij} is the matrix representing the second fundamental form for the surface. Note that the index m is in Cartesian space, so it is irrelevant whether or not it is listed as covariant or contravariant. Once again, the height ridge definition from Chapter 4 applies to the function f. Notice that the explicit parameterization of the surface must be known in order to compute the ridges.

A more general situation occurs when the surface is defined implicitly as the zero level set for a function defined on $\mathbb{R}^3$. In this case a parameterization is not known, so some more calculations are needed to compute ridges. Let the surface be implicitly defined by $H(\bar{x}) \equiv 0$ for some function $H : \mathbb{R}^3 \to \mathbb{R}$. Assuming that the surface is parameterized as $\bar{x}(x)$, we have

$$0 \equiv H(\bar{x}(x)).$$

Differentiating this equation yields

$$0 \equiv H_{,\ell}\bar{x}_{\ell,i}.$$

A unit normal to the surface is $N_\ell = H_{,\ell}/L$ where $L = \sqrt{H_{,m}H_{,m}}$. The tangents to the surface are $\bar{x}_{\ell,i}$ for $i = 1, 2$. Differentiating again yields

$$0 \equiv H_{,\ell}\bar{x}_{\ell,ij} + H_{,\ell m}\bar{x}_{\ell,i}\bar{x}_{m,j} = -Lb_{ij} + H_{,\ell m}\bar{x}_{\ell,i}\bar{x}_{m,j}$$

Solving this equation yields

$$b_{ij} = \frac{H_{,\ell m}}{L}\bar{x}_{\ell,i}\bar{x}_{m,j}. \tag{37}$$

Differentiating one more time yields

$$\begin{aligned} 0 &= H_{,\ell}\bar{x}_{\ell,ijk} + H_{,\ell m}\left(\bar{x}_{\ell,i}\bar{x}_{m,jk} + \bar{x}_{\ell,j}\bar{x}_{m,ik} + \bar{x}_{\ell,k}\bar{x}_{m,ij}\right) + H_{,\ell mn}\bar{x}_{\ell,i}\bar{x}_{m,j}\bar{x}_{n,k} \\ &= -Lb_{ij,k} - N_\ell H_{,\ell m}\left(\bar{x}_{m,i}b_{jk} + \bar{x}_{m,j}b_{ik} + \bar{x}_{m,k}b_{ij}\right) + H_{,\ell mn}\bar{x}_{\ell,i}\bar{x}_{m,j}\bar{x}_{n,k} \end{aligned}$$

Solving this equations yields

$$b_{ij,k} = \left[-\frac{N_r}{L^2}\left(H_{,r\ell}H_{,mn} + H_{,rm}H_{,\ell n} + H_{,rn}H_{,\ell m}\right) + \frac{H_{,\ell mn}}{L}\right]\bar{x}_{\ell,i}\bar{x}_{m,j}\bar{x}_{n,k}. \tag{38}$$

From the Riemannian ridge definition we had $f_{,ij}u^j = \alpha u_i$ and $f_{,ij}v^j = \beta v_i$ with $\alpha \le \beta$, $u_iu^i = 1$, $v_iv^i = 1$, and $u_iv^i = 0$. Also, we had $P = u^if_{,i}$ and $Q = v^if_{,i}$. A point $x \in \mathbb{R}^2$ is a ridge point if $P(x) = 0$ and

$\alpha(x) < 0$. Now define $\bar{u}_\ell = \bar{x}_{\ell,i}u^i$ and $\bar{v}_\ell = \bar{x}_{\ell,i}v^i$. The vectors $\bar{u}$ and $\bar{v}$ are tangents to the surface $\bar{x}(x)$. Note that $P = u^i f_{,i} = u^i \phi_{,\ell}\bar{x}_{\ell,i} = \bar{u}_\ell \phi_{,\ell}$. Similarly, $Q = \bar{v}_\ell \phi_{,\ell}$. To complete the set of directional derivatives, define $R = N_\ell \phi_{,\ell}$.

From Section (??? quad. forms), the eigenvector u is the vector which minimizes the quadratic form $u^i f_{,ij} u^j$. We wish to construct a similar problem involving the vector $\bar{u}$. Observe that

$$\begin{aligned} u^i f_{,ij} u^j &= \phi_{,\ell}\bar{x}_{\ell,ij}u^i u^j + \phi_{,\ell m}\bar{x}_{\ell,i}\bar{x}_{m,j}u^i u^j \\ &= -R\bar{u}_\ell \tfrac{H_{,\ell m}}{L}\bar{u}_m + \bar{u}_\ell \phi_{,\ell m}\bar{u}_m \\ &= \bar{u}_\ell \left(\phi_{,\ell m} - R\tfrac{H_{,\ell m}}{L}\right)\bar{u}_m \\ &=: \bar{u}_\ell A_{\ell m}\bar{u}_m. \end{aligned}$$

Minimizing the quadratic form $u^i f_{,ij} u^j$ is therefore equivalent to minimizing the quadratic form $\bar{u}_\ell A_{\ell m}\bar{u}_m$ with the restriction $\bar{u} \in \langle N \rangle^\perp$. As shown in Section (???), the vector $\bar{u}$ which minimizes the restricted quadratic form must satisfy

$$(\delta_{ik} - N_i N_k)A_{kj}\bar{u}_j = \alpha \bar{u}_i.$$

Since all the tensor quantities live in $\mathbb{R}^3$ (which is Euclidean), there is no distinction between covariant and contravariant indices. The matrix for this eigensystem, $(I - NN^\mathsf{T})A$, is not symmetric. To reduce the system to one involving symmetric matrices (hence, eigensolvers for symmetric systems can be used), consider the following. Let $\bar{T}_1$ and $\bar{T}_2$ be two vectors such that $\bar{T}_1$, $\bar{T}_2$, and N form an orthonormal frame field. Since $\bar{u}$ is a tangent vector, $\bar{u}_\ell = \mu_m \bar{T}_{m\ell}$. The eigensystem reduces to

$$\bar{T}_{ik}A_{k\ell}\bar{T}_{\ell j}\mu_j = \alpha \mu_i.$$

The matrix for this eigensystem is symmetric, so the eigensolvers for symmetric systems can be used. The minimum eigenvalue for the system is α and the corresponding eigenvector μ yields $\bar{u}$ as indicated earlier. The maximum eigenvalue for the system is β and the corresponding eigenvector ν yields $\bar{v}_\ell = \nu_m \bar{T}_{m\ell}$. Thus, without knowing a parameterization for the implicitly defined surface, we can still construct ridges. A point $\bar{x} \in \mathbb{R}^3$ on the surface $H(\bar{x}) \equiv 0$ is a 1-dimensional ridge point for ϕ restricted to the surface if $P = \bar{u}_\ell \phi_{,\ell} = 0$ and $\alpha < 0$ where $\bar{u}$ and α are constructed as previously shown.

For ridge flow and ridge traversal, we need to compute the first-order covariant derivatives of P. Recall that

$$\begin{aligned} P_{,r} &= \alpha u_r + \tfrac{Q}{\alpha-\beta} f_{,ijr}u^i v^j \\ &= \left(\alpha + \tfrac{Q}{\alpha-\beta} f_{,ijk}u^i v^j u^k\right)u_r + \left(\tfrac{Q}{\alpha-\beta} f_{,ijk}u^i v^j v^k\right)v_r. \end{aligned}$$

Thus, we need only compute formulas for the tensor products involving $f_{,ijk}$ which depend on P, Q, R, $\bar{u}$, $\bar{v}$, and N. Observe that

$$\begin{aligned} f_{,ijk}u^i v^j u^k &= \phi_{,\ell}\left(-b_{ij,k}u^i v^j u^k N_\ell - b_{ij}u^i v^j b_{k\cdot}^{m}\bar{x}_{\ell,m}u^k\right) \\ &\quad -\phi_{,\ell m}N_m\left(\bar{u}_\ell b_{jk}v^j u^k + \bar{v}_\ell b_{ik}u^i u^k + \bar{u}_\ell b_{ij}u^i v^j\right) \\ &\quad +\phi_{,\ell mn}\left(\bar{u}_\ell \bar{v}_m \bar{u}_n\right) \\ &= P\left(\Theta_{\bar{u}\bar{u}}\Theta_{\bar{u}\bar{v}}\right) + Q\left(\Theta_{\bar{u}\bar{v}}\Theta_{\bar{u}\bar{v}}\right) + R\left(2\Theta_{\bar{u}N}\Theta_{\bar{u}\bar{v}} + \Theta_{\bar{v}N}\Theta_{\bar{u}\bar{u}} - \tfrac{H_{,\ell mn}}{L}\bar{u}_\ell \bar{v}_m \bar{u}_n\right) \\ &\quad -2\Theta_{\bar{u}\bar{v}}\Psi_{\bar{u}N} - \Theta_{\bar{u}\bar{u}}\Psi_{\bar{v}N} + \phi_{,\ell mn}\bar{u}_\ell \bar{v}_m \bar{u}_n \end{aligned}$$

where $\Theta_{XY} = X^{\mathsf{T}}(D^2 H/L)Y$ and $\Psi_{XY} = X^{\mathsf{T}}(D^2\phi)Y$ are quadratic forms where X and Y are any of $\bar{u}$, $\bar{v}$, or N. Similarly,

$$\begin{aligned} f_{,ijk}u^i v^j u^k &= P\left(\Theta_{\bar{u}\bar{v}}\Theta_{\bar{u}\bar{v}}\right) + Q\left(\Theta_{\bar{u}\bar{v}}\Theta_{\bar{v}\bar{v}}\right) + R\left(2\Theta_{\bar{v}N}\Theta_{\bar{u}\bar{v}} + \Theta_{\bar{u}N}\Theta_{\bar{v}\bar{v}} - \tfrac{H_{,\ell mn}}{L}\bar{u}_\ell \bar{v}_m \bar{v}_n\right) \\ &\quad -2\Theta_{\bar{u}\bar{v}}\Psi_{\bar{v}N} - \Theta_{\bar{v}\bar{v}}\Psi_{\bar{u}N} + \phi_{,\ell mn}\bar{u}_\ell \bar{v}_m \bar{v}_n. \end{aligned}$$

3.1.2 d-Dimensional Ridges in $M^n \subset \mathbb{R}^p$

Let $\bar{x} : \mathbb{R}^n \to \mathbb{R}^p$ define an n-dimensional manifold M^n where the parameterization is denoted $\bar{x}(x)$. Assume that the manifold has codimension $q = p - n$ everywhere, which implies that each tangent space has dimension n. To help keep track of all the myriad indices, we use notation similar to that introduced in Section (???). An index with range 1 through p will be subscripted with a p. An index with range 1 through q will be subscripted with a q. An index with range 1 through n will be unsubscripted. However, if such an index has subrange 1 through $n - d$, it will be subscripted with a 0, and if it has subrange $n - d + 1$ through n, it will be subscripted with a 1.

Tangent vectors to the manifold are given by the partial derivatives of the parameterization, $T_{ii_p} = \partial\bar{x}_{i_p}/\partial x^i$. Let $N_{i_q i_p}$ denote a smoothly varying orthonormal set of normal vectors for the manifold. The metric tensor and the Christoffel symbols are as before,

$$g_{ij} = \frac{\partial\bar{x}}{\partial x^i}\cdot\frac{\partial\bar{x}}{\partial x^j} \;\text{ and }\; \Gamma^k_{ij} = \frac{\partial^2\bar{x}}{\partial x^i \partial x^j}\cdot g^{k\ell}\frac{\partial\bar{x}}{\partial x^\ell}.$$

If $f : \mathbb{R}^n \to \mathbb{R}$ is a function defined on the same parameter space as the manifold, say $f = f(x)$, then the height ridge definition in Chapter 4 may be directly applied to f.

Consider now the application where the function values at points on the manifold are inherited from a function $\phi : \mathbb{R}^p \to \mathbb{R}$ by the embedded manifold. In this case the function is $f(x) = \phi(\bar{x}(x))$. The height ridge definition requires computing up through third order covariant derivatives of f. As in the last subsubsection, calculations will show that

$$\begin{aligned} f_{,i} &= \phi_{,i_p}\bar{x}_{i_p,i} \\ f_{,ij} &= \phi_{,i_p}\bar{x}_{i_p,ij} + \phi_{,i_p j_p}\bar{x}_{i_p,i}\bar{x}_{j_p,j} \\ f_{,ijk} &= \phi_{,i_p}\bar{x}_{i_p,ijk} + \phi_{,i_p j_p}\left(\bar{x}_{i_p,i}\bar{x}_{j_p,jk} + \bar{x}_{i_p l,j}\bar{x}_{j_p,ik} + \bar{x}_{i_p,k}\bar{x}_{j_p,ij}\right) + \phi_{,i_p j_p k_p}\bar{x}_{i_p,i}\bar{x}_{j_p,j}\bar{x}_{k_p,k} \end{aligned}$$

where $\phi_{,i_p}$, $\phi_{,i_p j_p}$, and $\phi_{,i_p j_p k_p}$ are the Cartesian partial derivatives up through third order of ϕ. The covariant derivatives of the surface are (NEED TO PROVE THIS?)

$$\begin{aligned} \bar{x}_{i_p,i} &= \tfrac{\partial\bar{x}_{i_p}}{\partial x^i} \\ \bar{x}_{i_p,ij} &= -b_{ijk_q}N_{k_q i_p} \\ \bar{x}_{i_p,ijk} &= -b_{ijk_q,k}N_{k_q i_p} - b_{ijk_q}b^{\ell}_{k\cdot k_q}\bar{x}_{i_p,\ell} \end{aligned}$$

where b_{ijk_q} is the generalization of the second fundamental form for a 2-dimensional surface embedded in $\mathbb{R}^3$. Note that the index i_p is in Cartesian space, so it is irrelevant whether or not it is listed as covariant or contravariant. Once again, the height ridge definition from Chapter 4 applies to the function f. Notice that the explicit parameterization of the manifold must be known in order to compute the ridges.

Finally, consider the more general situation where the surface is defined implicitly as the zero level set for a vector-valued function defined on $\mathbb{R}^p$. In this case a parameterization is not known, so some more calculations are needed to compute ridges. Let the manifold be implicitly defined by $H(\bar{x}) \equiv 0$ for some vector-valued function $H : \mathbb{R}^p \to \mathbb{R}^q$. Assuming that the manifold is parameterized as $\bar{x}(x)$, we have

$$0 \equiv H_{i_q}(\bar{x}(x)).$$

Differentiating this equation yields

$$0 \equiv H_{i_q,i_p}\bar{x}_{i_p,i}$$

which implies that $H_{i_q,\cdot}$ are normal vectors to the manifold. We will come back to this point later. Differentiating again yields

$$0 \equiv H_{i_q,i_p}\bar{x}_{i_p,ij} + H_{i_q,i_pj_p}\bar{x}_{i_p,i}\bar{x}_{j_p,j}.$$

Differentiating one more time yields

$$0 \equiv H_{i_q,i_p}\bar{x}_{i_p,ijk} + H_{i_q,i_pj_p}\left(\bar{x}_{i_p,i}\bar{x}_{j_p,jk} + \bar{x}_{i_p,j}\bar{x}_{j_p,ik} + \bar{x}_{i_p,k}\bar{x}_{j_p,ij}\right) + H_{i_q,i_pj_pk_p}\bar{x}_{i_p,i}\bar{x}_{j_p,j}\bar{x}_{k_p,k}.$$

Since $H_{i_q,\cdot}$ are q normal vectors to the manifold, we can apply the QR algorithm to obtain q orthonormal vectors $N_{i_q\cdot}$. This construction produces a matrix L such that $L_{i_qk_q}L_{j_qk_q} = H_{i_q,i_p}H_{j_q,i_p}$ and $N_{i_qi_p} = L^{-1}_{i_qj_q}H_{j_q,i_p}$ where the inverse matrix L^{-1} satisfies $L^{-1}_{i_qj_q}L_{j_qk_q} = \delta_{i_qk_q}$.

We can now solve the previous displayed equations for the tensor b_{ijk_q} and its derivatives $b_{ijk_q,k}$. Observe that

$$\begin{aligned}
0 &\equiv H_{i_q,i_p}\left(-b_{ijk_q}N_{k_qi_p}\right) + H_{i_q,i_pj_p}\bar{x}_{i_p,i}\bar{x}_{j_p,j} \\
&= -H_{i_q,i_p}L^{-1}_{k_qj_q}H_{j_q,i_p}b_{ijk_q} + H_{i_q,i_pj_p}\bar{x}_{i_p,i}\bar{x}_{j_p,j} \\
&= -\left(H_{i_q,i_p}H_{j_q,i_p}\right)L^{-1}_{k_qj_q}b_{ijk_q} + H_{i_q,i_pj_p}\bar{x}_{i_p,i}\bar{x}_{j_p,j} \\
&= -L_{i_q\ell_q}L_{j_q\ell_q}L^{-1}_{k_qj_q}b_{ijk_q} + H_{i_q,i_pj_p}\bar{x}_{i_p,i}\bar{x}_{j_p,j} \\
&= -L_{i_qk_q}b_{ijk_q} + H_{i_q,i_pj_p}\bar{x}_{i_p,i}\bar{x}_{j_p,j},
\end{aligned}$$

which leads to

$$b_{ijk_q} = L^{-1}_{k_qi_q}H_{i_q,i_pj_p}\bar{x}_{i_p,i}\bar{x}_{j_p,j}. \tag{39}$$

The following also uses properties of the tensor L:

$$\begin{aligned}
0 \equiv\ & H_{i_q,i_p}\left(-b_{ijk_q,k}N_{k_qi_p} - (???)\bar{x}_{i_p,(???)}\right) \\
& -H_{i_q,i_pj_p}N_{k_qj_p}\left(\bar{x}_{i_p,i}b_{jkk_q} + \bar{x}_{i_p,j}b_{ikk_q} + \bar{x}_{i_p,k}b_{ijk_q}\right) \\
& +H_{i_q,i_pj_pk_p}\bar{x}_{i_p,i}\bar{x}_{j_p,j}\bar{x}_{k_p,k} \\
=\ & -L_{i_qk_q}b_{ijk_q,k} \\
& -H_{i_q,i_pj_p}N_{k_qj_p}L^{-1}_{k_q\ell_q}H_{\ell_q,r_ps_p}\left(\bar{x}_{i_p,i}\bar{x}_{r_p,j}\bar{x}_{s_p,k} + \bar{x}_{i_p,j}\bar{x}_{r_p,i}\bar{x}_{s_p,k} + \bar{x}_{i_p,k}\bar{x}_{r_p,i}\bar{x}_{s_p,j}\right) \\
& +H_{i_q,i_pj_pk_p}\bar{x}_{i_p,i}\bar{x}_{j_p,j}\bar{x}_{k_p,k}
\end{aligned}$$

which leads to

$$\begin{aligned}
b_{ijk_q,k} = \Big[& -N_{m_qj_p}L^{-1}_{k_qi_q}L^{-1}_{m_qr_q}\left(H_{i_q,r_pj_p}H_{r_q,s_pt_p} + H_{i_q,s_pj_p}H_{r_q,r_pt_p} + H_{i_q,t_pj_p}H_{r_q,r_ps_p}\right) \\
& +L^{-1}_{k_qi_q}H_{i_q,r_ps_pt_p}\Big]\bar{x}_{r_p,i}\bar{x}_{x_p,j}\bar{x}_{t_p,k}.
\end{aligned} \tag{40}$$

From the Riemannian ridge definition we had to solve eigensystems of the form $f_{,ij}v^j_{\cdot k} = v_{ij}\lambda^j_{\cdot k}$. The directional derivatives were given by $P_j = v^i_{\cdot j} f_{,i}$. A point $x \in \mathbb{R}^n$ is a d-dimensional ridge point if $P_j(x) = 0$ for $1 \le j \le n-d$ and $\lambda_{n-d}(x) < 0$. Now define $\bar{v}_{i_p j} = \bar{x}_{i_p,i} v^i_{\cdot j}$, which are tangents to the manifold $\bar{x}(x)$. Note that $P_j = f_{,i} v^i_{\cdot j} = \phi_{,i_p} \bar{x}_{i_p,i} v^i_{\cdot j} = \phi_{,i_p} \bar{v}_{i_p j}$.

We again wish to convert and optimize the quadratic form $f_{,ij}u^i u^j$ in terms of tangent vectors to the manifold. Observe that

$$\begin{aligned} f_{,ij}u^i u^j &= \phi_{,i_p}\bar{x}_{i_p,ij}u^i u^j + \phi_{.i_p j_p}\bar{x}_{i_p,i}\bar{x}_{j_p,j}u^i u^j \\ &= \left(\phi_{,i_p}N_{k_q i_p}\right)\left(-b_{ijk_q}u^i u^j\right) + \phi_{,i_p j_p}\bar{u}_{i_p}\bar{u}_{j_p} \\ &= -\left(\phi_{,i_p}N_{k_q i_p}\right)\left(L^{-1}_{k_q i_q}H_{i_q,i_p j_p}\bar{u}_{i_p}\bar{u}_{j_p}\right) + \phi_{.i_p j_p}\bar{u}_{i_p}\bar{u}_{j_p} \\ &= \bar{u}_{i_p}\left[\phi_{,i_p j_p} - \left(N_{k_q i_p}\phi_{,i_p}\right)L^{-1}_{k_q i_q}H_{i_q,i_p j_p}\right]\bar{u}_{j_p} \\ &=: \bar{u}_{i_p}A_{i_p j_p}\bar{u}_{j_p} \end{aligned}$$

Minimizing the quadratic form $u^i f_{,ij}u^j$ is therefore equivalent to minimizing the quadratic form $\bar{u}_\ell A_{\ell m}\bar{u}_m$ with the restriction $\bar{u}$ is in the tangent space to the manifold. Let $\bar{T}_{i_p i}$ be an orthonormal set of tangent vectors. Since $\bar{u}$ is a tangent vector, $\bar{u}_{i_p} = \mu_i \bar{T}_{i_p i}$. The restricted quadratic form displayed previously reduces to

$$\begin{aligned} \bar{u}_{i_p}A_{i_p j_p}\bar{u}_{j_p} &= \left(\mu_i \bar{T}_{i_p i}\right)A_{i_p j_p}\left(\mu_j \bar{T}_{j_p j}\right) \\ &= \mu_i\left(\bar{T}_{i_p i}A_{i_p j_p}\bar{T}_{j_p j}\right)\mu_j \\ &= \mu_i B_{ij}\mu_j. \end{aligned}$$

The critical vectors for this quadratic form are eigenvectors for the system $B_{ij}\mu_j = \lambda\mu_i$. The matrix for this eigensystem is symmetric, so the eigensolvers for symmetric systems can be used. Thus, without knowing a parameterization for the implicitly defined manifold, we can still construct ridges. A point $\bar{x} \in \mathbb{R}^p$ on the manifold defined by $H_{i_q}(\bar{x}) = 0$ is a d-dimensional ridge point for ϕ restricted to the manifold if $P_j = \phi_{,i_p}\bar{v}_{i_p,j} = 0$ for $1 \le j \le n-d$ and $\lambda_{n-d} < 0$.

NEED TO DERIVE FORMULAS FOR $P_{i,j}$ FOR GENERAL RIDGE FLOW AND TRAVERSAL.

For ridge flow and ridge traversal, we need to compute the quantities that play the role of the first-order covariant derivatives of P. Recall that

$$\tilde{P}_{i_0,k} = v_{kk_0}\lambda^{k_0}_{\cdot i_0} + P_{j_1}\Delta^{-1}_{i_0 j_1 \ell_0 \ell_1} f_{,ijk} v^i_{\cdot \ell_0} v^j_{\cdot \ell_1}$$

where $\Delta^{-1}_{k_0 k_1 i_0 i_1} = (\lambda_{k_0} - \lambda_{k_1})^{-1}$ if $i_0 = k_0$ and $i_1 = k_1$, but is zero otherwise. We can write $\tilde{P}_{i_0,k} = C_{i_0 r} v_{kr}$ where

$$\begin{aligned} C_{i_0 r} &= \tilde{P}_{i_0,k} v^k_{\cdot r} \\ &= \lambda^r_{\cdot i_0} + P_{j_1}\Delta^{-1}_{i_0 j_1 \ell_0 \ell_1} f_{,ijk} v^i_{\cdot \ell_0} v^j_{\cdot \ell_1} v^k_{\cdot r}. \end{aligned}$$

Thus, we need only compute formulas for the tensor products $f_{,ijk} v^i_{\cdot \ell_0} v^j_{\cdot \ell_1} v^k_{\cdot r}$ in terms of the directional derivatives P_i, the tangent vectors $\bar{v}_{\cdot i}$, and the normal vectors $N_{\cdot i}$. Observe that

$$\begin{aligned} f_{,ijk} v^i_{\cdot \ell_0} v^j_{\cdot \ell_1} v^k_{\cdot r} &= -\phi_{,i_p}\left[\left(b_{ijk_q,k} v^i_{\cdot \ell_0} v^j_{\cdot \ell_1} v^k_{\cdot r}\right)N_{k_q i_p} + \left(b_{ijk_q} v^i_{\cdot \ell_0} v^j_{\cdot \ell_1}\right) v^k_{\cdot r} b^\ell_{k \cdot k_q} \bar{x}_{i_p,\ell}\right] \\ &\quad -\phi_{,i_p j_p} N_{k_q j_p}\left[\bar{v}_{i_p \ell_0}\left(b_{jkk_q} v^j_{\cdot \ell_1} v^k_{\cdot r}\right) + \bar{v}_{i_p \ell_1}\left(b_{ikk_q} v^i_{\cdot \ell_0} v^k_{\cdot r}\right) + \bar{v}_{i_p r}\left(b_{ijk_q} v^i_{\cdot \ell_0} v^j_{\cdot \ell_1}\right)\right] \\ &\quad +\phi_{,i_p j_p k_p}\left[\bar{v}_{i_p \ell_0}\bar{v}_{j_p \ell_1}\bar{v}_{k_p r}\right] \\ &= \end{aligned}$$

*********** where $\Theta_{XY} = X^{\mathsf{T}}(D^2H/L)Y$ and $\Psi_{XY} = X^{\mathsf{T}}(D^2\phi)Y$ are quadratic forms where X and Y are any of $\bar{u}$, $\bar{v}$, or N. Similarly,

$$\begin{aligned} f_{,ijk}u^iv^ju^k &= P\left(\Theta_{\bar{u}\bar{v}}\Theta_{\bar{u}\bar{v}}\right) + Q\left(\Theta_{\bar{u}\bar{v}}\Theta_{\bar{v}\bar{v}}\right) + R\left(2\Theta_{\bar{v}N}\Theta_{\bar{u}\bar{v}} + \Theta_{\bar{u}N}\Theta_{\bar{v}\bar{v}} - \tfrac{H_{,\ell mn}}{L}\bar{u}_\ell\bar{v}_m\bar{v}_n\right) \\ &\quad -2\Theta_{\bar{u}\bar{v}}\Psi_{\bar{v}N} - \Theta_{\bar{v}\bar{v}}\Psi_{\bar{u}N} + \phi_{,\ell mn}\bar{u}_\ell\bar{v}_m\bar{v}_n. \end{aligned}$$

3.2 Principal Direction Ridge Definition

The principal direction definition for ridges and valleys is motivated by the differential geometry of n-dimensional hypersurfaces in $\mathbb{R}^{n+1}$. We define creases as loci of extrema of *principal curvatures* along associated *lines of curvature*. The curvature measurements are made with respect to the metric on the tangent hyperplanes.

In standard differential geometry textbooks, hypersurfaces are described by a parameterization which is used in obtaining principal curvatures and principal directions. In computational vision applications, typically one obtains a surface as a collection of points with no underlying parameterization. Such surfaces are assumed to be implicitly defined, so we also want to construct principal curvatures and principal directions for surfaces defined as level sets of functions $F : \mathbb{R}^{n+1} \to \mathbb{R}$. Assume that F is a C^4 function for which $\nabla F \neq 0$. The normal vectors to the surface are $N = \nabla F/|\nabla F|$.

Construction With Parameterization. Let the surface be parameterized by position $x : \mathbb{R}^n \to \mathbb{R}^{n+1}$, say $x = x(u)$. Define $J = \partial x/\partial u$, an $(n+1)\times n$ matrix which has rank n and satisfies the property $N\mathsf{T}J = 0$. That is, the columns of J are a basis of the tangent space and are orthogonal to N at position $x(u)$. The first and second fundamental forms are given by the $n \times n$ matrices $\mathbf{I} = J\mathsf{T}J$ and $\mathbf{II} = -J\mathsf{T}\partial N/\partial u$, respectively. The matrix representing the shape operator on the tangent space is $S = \mathbf{I}^{-1}\mathbf{II}$. Consider the eigenvector problem $Sp = \kappa p$. Each eigenvector p is a principal direction. The corresponding eigenvalue κ is a principal curvature. The vector p is an n-vector given in terms of tangent space coordinates, but its representation in $\mathbb{R}^{n+1}$ is $\xi = Jp$.

Construction without Parameterization. Define $W = -\partial N/\partial x$, an $(n+1) \times (n+1)$ matrix. We claim that if $Sp = \kappa p$, then $\xi = Jp$ satisfies $W\xi = \kappa\xi$. Firstly, we have $\mathbf{I} = J\mathsf{T}J$. Secondly, by the chain rule we have $\partial N/\partial u = (\partial N/\partial x)J$, so $\mathbf{II} = J\mathsf{T}WJ$. The eigenvector problem $(\mathbf{II} - \kappa\mathbf{I})p = 0$ is therefore transformed to $J\mathsf{T}(W - \kappa E)\xi = 0$, where E is the $(n+1) \times (n+1)$ identity matrix and where $\xi = Jp$.

Since $J\mathsf{T}$ has full rank n, its generalized inverse is given by $(J\mathsf{T})^+ = J(J\mathsf{T}J)^{-1}$. If p is a principal direction, that is $Sp = \kappa p$, then $WJp = JSp = \kappa JP$, so $\xi = Jp$ is an eigenvector of W with corresponding eigenvalue κ. Conversely, if ξ is an eigenvector of W, that is $W\xi = \kappa\xi$, and ξ is a tangent vector, say $\xi = Jp$, then $J^+\xi = p$, $JJ^+\xi = JP = \xi$, and

$$SJ^+\xi = (J^+WJ)J^+\xi = J^+W\xi = \kappa J^+\xi,$$

so $J^+\xi$ is a principal directions with corresponding principal curvature κ. Additionally, W has an identically zero eigenvalue, but the corresponding eigenvector is not a tangent vector. This follows from the identity $W = (E - NN\mathsf{T})\,\mathrm{Hess}\,(F)/|\nabla F|$ which can be derived by explicitly computing $\partial N_i/\partial x_j$ for $N = \nabla F/|\nabla F|$. The eigenvector is $\mathrm{adj}(\,\mathrm{Hess}\,(F))\nabla F$ where adj indicates the adjoint of a matrix. A short computation shows that $W\,\mathrm{adj}(\,\mathrm{Hess}\,(f))\nabla F = 0$.

3.2.1 Creases on Graphs

Let $f : \mathbb{R}^n \to \mathbb{R}$ be a C^4 function with graph $g : \mathbb{R}^n \to \mathbb{R}^{n+1}$ given by $g(x) = (x, f(x))$. The principal curvatures κ_i and directions p_i, $1 \le i \le n$, are determined by $Sp_i = \kappa_i p_i$ where S is the shape operator described earlier. Assume that the curvatures are ordered as $\kappa_i \ge \ldots \ge \kappa_n$. In the height definition, we defined creases as generalized extrema of a single real-valued function. The principal direction definition is different in that creases will occur as extreme points of each principal curvature with respect to its principal direction. Moreover, the classification of an extreme point will depend on following the integral curves of the principal direction vector field, so we need to use the more general second directional derivative test. Like the height definition, we will characterize the creases according to the dimension of the manifold we expect when finding roots to equations. We also can refine the definitions to include the concepts of strong and weak creases. In the definition, assume that $1 \le d \le n$.

- The point x is a *ridge point of type* $n-d$ if $\kappa_d(x) > 0$, and $D_{p_i}\kappa_i(x) = 0$ and $D_{p_i}D_{p_i}\kappa_i(x) < 0$ for $1 \le i \le d$. Additionally x is a *strong ridge point* if $\kappa_d(x) > |\kappa_n(x)|$; otherwise it is a *weak ridge point.*
- The point x is a *valley point of type* $n-d$ if $\kappa_{n-d+1}(x) < 0$, and $D_{p_i}\kappa_i(x) = 0$ and $D_{p_i}D_{p_i}\kappa_i(x) > 0$ for $n-d+1 \le i \le n$. Additionally x is a *strong valley point* if $|\kappa_{n-d+1}(x)| > \kappa_1(x)$; otherwise it is a *weak valley point.*

We briefly contrast the height and principal direction definitions. In the height definition, we searched for the local extrema of a *single* function f whose domain was restricted to a subspace of $\mathbb{R}^n$ (so we searched in *multiple* directions). That is, if V is the $n \times d$ matrix whose columns span the desired subspace, and if s is a $d \times 1$ vector-valued parameter, then we searched for extrema of $\phi(s) = f(x+Vs)$ using the standard definition for extrema. The second derivative test involved determining the definiteness of the second derivative matrix for $\phi(s)$ when $s = 0$. In the principal direction definition, we are searching for local extrema of *multiple* functions κ_i. Each such function has a *single* direction p_i associated with it, so the construction of extrema is the usual one for functions of a single real variable. That is, if s is a real variable, then for each i we search for extrema along a path $\xi(s)$ of a function $\phi_i(s) = \kappa_i(\xi(s))$, where the path is determined by $\xi'(s) = p_i(\xi(s))$, $\xi(0) = x$. The second derivative test involves testing the sign of $\phi_i''(0)$.

3.2.2 Creases on Level Surfaces

Let $F : \mathbb{R}^{n+1} \to \mathbb{R}$ be a C^4 function, and consider the hypersurface defined implicitly by $F(x) = 0$. As shown before, we do not need to find a parameterization of the hypersurface to construct its principal curvatures $\kappa_i(x)$ and principal directions $\xi_i(x) \in \mathbb{R}^{n+1}$. They are the eigenvalues and non-tangential eigenvectors of the matrix $W = -\partial N/\partial x$, where $N = \nabla F/|\nabla F|$ and $\xi(x)\mathsf{T}N(x) = 0$. In the definition, assume that $1 \le d \le n$. Also, the points $x \in \mathbb{R}^n$ of interest must be solutions to $F(x) = 0$.

- The point x is a *ridge point of type* $n-d$ if $\kappa_d(x) > 0$, and $D_{\xi_i}\kappa_i(x) = 0$ and $D_{\xi_i}D_{\xi_i}\kappa_i(x) < 0$ for $1 \le i \le d$. Additionally x is a *strong ridge point* if $\kappa_d(x) > |\kappa_n(x)|$; otherwise it is a *weak ridge point.*
- The point x is a *valley point of type* $n-d$ if $\kappa_{n-d+1}(x) < 0$, and $D_{\xi_i}\kappa_i(x) = 0$ and $D_{\xi_i}D_{\xi_i}\kappa_i(x) > 0$ for $n-d+1 \le i \le n$. Additionally x is a *strong valley point* if $|\kappa_{n-d+1}(x)| > \kappa_1(x)$; otherwise it is a *weak valley point.*

3.2.3 Graph Examples

Example 1: In dimension $n = 1$, the matrix S is 1×1 and its single entry is $\kappa = -f_{xx}/(1 + f_x^2)^{3/2}$. The graph of $f(x)$ is a planar curve whose curvature at $(x, f(x))$ is $\kappa(x)$. Ridges (valleys) are local maxima (minima) of $\kappa(x)$. For example, let $f(x) = x^p$ where p is a positive even integer. The curvature is $\kappa = -p(p-1)x^{p-2}/(1 + p^2 x^{2p-2})^{3/2}$. The solutions to $\kappa_x = 0$ are

$$x = 0,\ \pm\left(\frac{p-2}{p^2(2p-1)}\right)^{1/(2p-2)}.$$

For $p = 2$ the only solution is $x = 0$. The curvature has a negative local minimum of -2, so $x = 0$ is a valley point. For $p > 2$, κ has a local maximum of 0 at $x = 0$, so the graph of f has a flat spot which is neither a ridge nor a valley. At the other two critical points, κ has negative local minima, so the points are valley points. Note that as $p \to \infty$, the graph of x^p approaches 0 pointwise on $(-1, 1)$ and the valley points approach ± 1. Figure 3.1 shows the graphs and valley points for two different values of p.

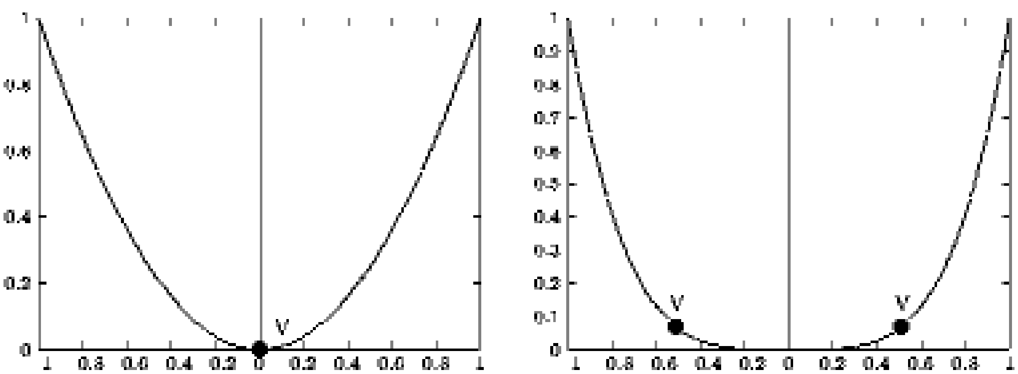

Figure 1: graph of $f(x) = x^2$, graph of $f(x) = x^4$

The valley points are labeled on the graphs as V. When $p = 4$, the valleys are $\pm(1/56)^{1/6} \doteq 0.51$.

The example $f(x) = x^4$ shows that creases according to the principal direction definition are not necessarily local extrema in the function. However, the creases obtained may be better suited for functions which correspond to measurements other than intensity or for which the independent variable is not a spatial one. For example, f might be a function of time for which we are interested in knowing a first time when f has a transition between slowly decreasing and greatly decreasing. The crease points can be viewed as such transitions.

Example 2: Let $f(x, y) = x^2 y$. The matrix for the shape operator is

$$S = \frac{1}{L^3}\begin{bmatrix} -2y(x^4 - 1) & 2x(1 + x^4) \\ 2x(1 + 2x^2y^2) & -4x^4 y \end{bmatrix}$$

where $L = \sqrt{1 + |\nabla f|^2} = \sqrt{1 + 4x^2y^2 + x^4}$. The principal curvatures are $\kappa = (y(3x^4 - 1) \pm R)/L^3$ where $R = \sqrt{(1 + x^4)[y^2(9x^4 + 1) + 4x^2]}$, and corresponding principal directions are

$$p_1 = \left\{ \begin{array}{ll} (-(1 + x^4)y + R, -2x(1 + 2x^2y^2)), & y < 0 \\ (2x(1 + x^4), -(1 + x^4)y - R), & y > 0 \end{array} \right\} \text{ and } p_2 = \left\{ \begin{array}{ll} (2x(1 + x^4), -(1 + x^4)y + R), & y < 0 \\ (-(1 + x^4)y - R, -2x(1 + 2x^2y^2)), & y > 0 \end{array} \right\}.$$

A closed form solution for the ridges is not tractable. The ridges consist of a ray $(0, y)$ for $y < \alpha < 0$ and some constant α, an s-shaped curve in the fourth and first quadrants starting at $(0, \alpha)$ and passing through the x-axis at some point $(\beta, 0)$ for constant $\beta > 0$. This curve's reflection through the y-axis is also a ridge.

3.2.4 Level Surface Example

EXAMPLE 3: Consider an ellipsoid defined as a level surface of the function $F(x, y, z) = (ax^2 + by^2 + cz^2)/2$, say $F(x, y, z) = p > 0$, where $0 < a < b < c$. The unit normal vectors are $N = (ax, by, cz)/L$ where $L = \sqrt{a^2x^2 + b^2y^2 + c^2z^2}$. The matrix W is

$$W = \frac{1}{L^3}\begin{bmatrix} a(b^2y^2 + c^2z^2) & -ab^2xy & -ac^2xz \\ -ba^2xy & b(a^2x^2 + c^2z^2) & -bc^2yz \\ -ca^2xz & -cb^2yz & c(a^2x^2 + b^2y^2) \end{bmatrix}.$$

The principal curvatures of the surface are $\kappa_1 = (\alpha + \sqrt{\beta})/L^3$ and $\kappa_2 = (\alpha - \sqrt{\beta})/L^3$ where $\alpha = a^2(b + c)x^2 + b^2(a + c)y^2 + c^2(a + b)z^2$ and $\beta = a^4(b - c)^2x^4 + b^4(a - c)^2y^4 + c^4(a - b)^2z^4 + 2(a - c)(b - c)a^2b^2x^2y^2 + 2(a - b)(c - b)a^2c^2x^2z^2 + 2(b - a)(c - a)b^2c^2y^2z^2$. Corresponding principal directions are

$$\begin{aligned} p_1 &= ab(cxz, cyz, -ax^2 - by^2) + L\kappa_1(acxz, bcyz, -a^2x^2 - b^2y^2) \text{ and} \\ p_2 &= ab(-y(abx^2 + b^2y^2 + c^2z^2), x(a^2x^2 + aby^2 + c^2z^2), c(a - b)xyz) + L^3\kappa_1(-by, ax, 0). \end{aligned}$$

Clearly $\kappa_1 > 0$ for all (x, y, z), so let us attempt to locate ridges of type 1. Taking derivatives, we obtain the formula

$$L^3\nabla\kappa_1 + 3L^2\kappa\nabla L = \nabla\alpha + \frac{1}{2\sqrt{\beta}}\nabla\beta.$$

At $z = 0$, some calculations will show that $p_1 = -\alpha(0, 0, 1)$. It is easily shown that $p_1 \cdot \nabla\alpha = p_1 \cdot \nabla\beta = p_1 \cdot \nabla L = 0$ when $z = 0$. Thus, $D_{p_1}\kappa_1(x, y, 0) = 0$ for all x and y (which lie on the curve $ax^2 + by^2 = 2p$).

The second directional derivative when $z = 0$ can be shown to be $D_{p_1p_1}\kappa_1 = \alpha[\kappa_{zz} - \alpha\nabla\kappa^{\mathsf{T}}(\partial p_1/\partial z)]$, where all quantities involved are evaluated at $z = 0$. Some tedious algebraic calculations lead to

$$D_{p_1p_1}\kappa_1(x, y, 0) = \frac{2c^2}{L^3\sqrt{\beta}}\left(\omega_1a^4x^4 + \omega_2a^2b^2x^2y^2 + \omega_3b^4y^4\right)$$

where $\omega_1 = (c-b)[a(4b+5c) - (b+c)(b+6c)]$, $\omega_2 = \{(c-a)[ac+3(b+c)(b-6c)] + (c-b)[bc+3(a+c)(a-6c)]\}$, and $\omega_3 = (c - a)[b(4a + 5c) - (a + c)(a + 6c)]$. Using $0 < a < b < c$, it can be shown that all $\omega_i < 0$, so $D_{p_1p_1}\kappa_1(x, y, 0) < 0$. Therefore the points on the ellipsoid for which $z = 0$ are ridges of type 1.

Additional calculations will show that when $y = 0$ and $z = 0$, $D_{p_2}\kappa_2 = 0$ and $D_{p_2p_2}\kappa_2 < 0$. The vertices $(\pm\sqrt{2p/a}, 0, 0)$ are therefore ridges of type 0.

3.2.5 Invariance Properties

For parameterized hypersurfaces, the ridge construction is invariant under diffeomorphisms on the parameter space. This result follows from standard differential geometry where the principal curvatures and principal directions do not change under these transformations. With regard to transformations applied to the entire space $\mathbb{R}^{n+1}$ in which the level surface $F(x) = 0$ lives, the ridges constructed are invariant under spatial translations and spatial rotations (Euclidean motions), and under uniform spatial magnifications.

In the special case of a graph defined by $F(x, z) = z - f(x) = 0$, where $x \in \mathbb{R}^n$ and $z \in \mathbb{R}$, the ridge construction is not invariant to uniform magnification in x. Note that the magnification is not a reparameterization of the original surface; the transformation does change the surface. For example, if $n = 1$ and $f(x) = 1-x^4$ for $x > 0$, a ridge is $x_0 = 56^{-1/6}$. Let $x = c\bar{x}$ for some $c > 0$ and define $\bar{f}(\bar{x}) = f(x) = 1-c^4\bar{x}^4$. The ridge for this new function is $\bar{x_0} = (56c^8)^{-1/6} \neq x_0/c$, so the ridge is not invariant.

3.3 Level Set Ridge Definition

3.3.1 Creases on Level Surfaces

Let $f : \mathbb{R}^n \to \mathbb{R}$, $n \geq 2$, be a C^4 function such that $\nabla f \neq 0$ (except at isolated points). The domain of f can be partitioned into its level sets defined by $f(x) = c$ for constants c. Note that a single level set can be viewed as a hypersurface in $\mathbb{R}^n$ implicitly defined by $F(x) = f(x) - c = 0$. Therefore, the principal direction definition may be applied to find creases on the hypersurface for each c in the range of f. The normals for the hypersurface are $N = \nabla f/|\nabla f|$ and the eigenvalues and (tangential) eigenvectors of the matrix $W = -\partial N/\partial x$ are the principal curvatures and principal directions. We will construct creases on the graph of f by applying the principal direction definition to each of its level surfaces. The set of all creases of all the level surfaces make up the creases of the graph.

The eigenvalues of W are $\kappa_1 \geq \ldots \geq \kappa_{n-1}$ and 0, with corresponding eigenvectors $\xi_1, \ldots, \xi_{n-1}$ and $\mathrm{adj}(\mathrm{Hess}\,(f))\nabla f$. The $\kappa_i(a)$ and $\xi_i(a)$ are the principal curvatures and principal directions for the level surface $f(x) = f(a)$. We can attempt to construct crease sets of dimension d where $1 \leq d \leq n-1$. The definition is similar to the principal direction definition, but with one subtle difference. In the principal direction definition, creases were solutions to equations of the type $D_\xi \kappa_i(x) = 0$ where $F(x) = 0$ for a single function F. In the level definition, creases are solutions to the same equations, but we now have an entire family of functions $F(x; c) = f(x) - c = 0$.

- The point x is a *ridge point of type* $n-1-d$ if $\kappa_d(x) > 0$, and $D_{\xi_i}\kappa_i(x) = 0$ and $D_{\xi_i}D_{\xi_i}\kappa_i(x) < 0$ for $1 \leq i \leq d$. Additionally x is a *strong ridge point* if $\kappa_d(x) > |\kappa_n(x)|$; otherwise it is a *weak ridge point.*
- The point x is a *valley point of type* $n-1-d$ if $\kappa_{n-d}(x) < 0$, and $D_{\xi_i}\kappa_i(x) = 0$ and $D_{\xi_i}D_{\xi_i}\kappa_i(x) > 0$ for $n-d \leq i \leq n-1$. Additionally x is a *strong valley point* if $|\kappa_{n-d}(x)| > \kappa_1(x)$; otherwise it is a *weak valley point.*

Example 1: Consider the case $n = 2$. Normal and tangent vectors to the level curves are given by $N(x, y) = (f_x, f_y)/(f_x^2 + f_y^2)^{1/2}$ and $T(x, y) = (f_y, -f_x)/(f_x^2 + f_y^2)^{1/2}$, and the curvature of the level curves is $\kappa(x, y) = -(f_x^2 f_{yy} - 2f_x f_y f_{xy} + f_y^2 f_{xx})/(f_x^2 + f_y^2)^{3/2}$.

Consider the function $f(x, y) = x^2 y$ for $x > 0$ and $y > 0$. The tangents to level curves are $T = (x, -2y)/(x^2 + 4y^2)^{1/2}$. The curvature and its derivative in the T direction are $\kappa = 6xy/(x^2+4y^2)^{3/2}$ and $D_T\kappa = -24xy(x^2 - 5y^2)/(x^2 + 4y^2)^3$. Setting $D_T\kappa = 0$ in the first quadrant yields $x = \sqrt{5}y$, $y > 0$. Some short calculations will show that $D_{TT}\kappa < 0$, so the points are ridge points.

3.3.2 1-Dimensional Creases from Mean Curvature

In the numerical implementation of the level definition, one must compute the eigenvalues and eigenvectors for the matrix W at each point in an image. This process is typically time-consuming. A variation on the level definition for constructing 1-dimensional creases computes the local extrema of the *mean curvature* $\mu = \text{trace}(W)/(n-1)$ rather than computing local extrema of principal curvatures. The trace of W is more easily computed than its eigenvalues.

Let $f : \mathbb{R}^n \to \mathbb{R}$ and let $x : \mathbb{R}^{n-1} \to \mathbb{R}^n$ denote a level surface; thus, $f(x(s)) \equiv c$ for some constant c and for all s. Let $\phi(s) = \mu(x(s))$ be the mean curvature of the level surface at position $x(s)$. In the construction we use the following abbreviations for the tensor quantities for the gradients and Hessians of the functions of interest.

$$\nabla\phi = \frac{\partial\phi}{\partial s_i}, \nabla\mu = \frac{\partial\mu}{\partial x_k}, \nabla f = \frac{\partial f}{\partial x_k}$$

and

$$\text{Hess}\,\phi = \frac{\partial^2\phi}{\partial s_i \partial s_j}, \text{Hess}\,\mu = \frac{\partial^2\mu}{\partial x_k \partial x_m}, \text{Hess}\,f = \frac{\partial^2 f}{\partial x_k \partial x_m}.$$

We also use the following abbreviations for the tensor quantitites for the first and second derivatives of position $x(s)$:

$$x'(s) = \frac{\partial x_k}{\partial s_i} \text{ and } x''(s) = \frac{\partial^2 x_k}{\partial s_i \partial s_j}.$$

The local extrema of ϕ occur when $\nabla\phi = 0$ and $\text{Hess}\,\phi$ is positive definite (local maximum) or negative definite (local minimum). We would like to determine the local extrema without having to choose a particular parameterization $x(s)$. Thus we need to select a smoothly varying basis of tangent vectors $v_i(x)$, $1 \le i \le n-1$, which can be used in the derivative tests instead of the tangent vectors $\partial x/\partial s_i$. For now let us assume that we have such a basis. Let V be the $n \times (n-1)$ matrix whose columns are $v_i(x)$. The two matrices V and $x'(s)$ are related by an invertible $(n-1) \times (n-1)$ matrix C (a change of basis), $x'(s) = VC$.

Using the chain rule, the derivatives of ϕ are

$$\nabla\phi = x'(s)^{\mathsf{T}}\nabla\mu \text{ and } \text{Hess}\,\phi = x'(s)^{\mathsf{T}}\,\text{Hess}\,\mu x'(s) + x''(s)\nabla\mu.$$

The first derivative test is $0 = \nabla\phi = C^{\mathsf{T}}V^{\mathsf{T}}\nabla\mu$. Since C is invertible, the critical points are solutions to $V^{\mathsf{T}}\nabla\mu = 0$. The second derivative test involves second derivatives of position, which we would like to avoid computing. Note that $V^{\mathsf{T}}\nabla\mu = 0$ implies that $\nabla\mu$ is orthogonal to the tangent space (at a critical point); that is, $\nabla\mu = \rho\nabla f$ where $\rho = (\nabla\mu \cdot \nabla f)/(\nabla f \cdot \nabla f)$. Moreover, since $f(x(s)) \equiv c$, taking derivatives yields

$$0 = x'(s)^{\mathsf{T}}\nabla f \text{ and } 0 = x'(s)^{\mathsf{T}}\,\text{Hess}\,f x'(s) + x''(s)\nabla f.$$

At a critical point we consequently have

$$x''(s)\nabla\mu = \rho x''(s)\nabla f = -\rho x'(s)^{\mathsf{T}}\,\text{Hess}\,f x'(s),$$

so the second derivative of ϕ at such points is

$$\text{Hess}\,\phi = x'(s)^{\mathsf{T}}\left(\text{Hess}\,\mu - \rho\,\text{Hess}\,f\right)x'(s).$$

In terms of the matrix V we have

$$C^{-t}\,\text{Hess}\,\phi C^{-1} = V^{\mathsf{T}}\left(\text{Hess}\,\mu - \rho\,\text{Hess}\,f\right)V.$$

By Sylvester's Theorem, $C^{-t}\,\text{Hess}\,\phi C^{-1}$ and $\text{Hess}\,\phi$ have the same definiteness, so we need only check the definiteness of $V^{\mathsf{T}}(\text{Hess}\,\mu - \rho\,\text{Hess}\,f)V$ for the second derivative test.

The remaining problem is to find a smoothly varying basis v_i for the tangent space to level surfaces which is easier to compute than the principal directions. Such a basis is given by the columns of a rotation matrix which maps the vector $e_n = (0, \ldots, 0, 1)$ to the normal $N = \nabla f / |\nabla f|$ of the surface. A rotation matrix is given in block form by

$$R = \left[\begin{array}{c|c} E + (N_n - 1)PP^{\mathsf{T}} & Q \\ \hline -Q^{\mathsf{T}} & N_n \end{array} \right]$$

where E is the $(n-1) \times (n-1)$ identity matrix, $Q = (N_1, \ldots, N_{n-1})^{\mathsf{T}}$ are the first $n-1$ components of the normal vector, N_n is the last component of N, and $P = Q/|Q|$ when $Q \neq 0$. If $Q = 0$, the rotation is just the identity matrix.

The crease definitions for this variation are given below. Let V be the matrix whose columns are the v_i vectors. A point $x \in \mathbb{R}^n$ is

- a ridge point if $\mu(x) > 0$, $V^{\mathsf{T}}\nabla\mu(x) = 0$, and $V^{\mathsf{T}}(\text{Hess}\,\mu - \rho\,\text{Hess}\,f)V$ is negative definite;
- a valley point if $\mu(x) < 0$, $V^{\mathsf{T}}\nabla\mu(x) = 0$, and $V^{\mathsf{T}}(\text{Hess}\,\mu - \rho\,\text{Hess}\,f)V$ is positive definite,

where $\rho(x) = (\nabla\mu \cdot \nabla f)/(\nabla f \cdot \nabla f)$. The directional derivatives and eigenvalues are all evaluated at the point in question. Note that this definition is identical to the original one when $n = 2$. The qualitative differences between ridges obtained by the level definition and those obtained by the variation involving mean curvature should be minimal in convex regions (all $\kappa_i > 0$). Some noticeable differences may occur in hyperbolic regions.

3.3.3 Invariance Properties

The ridges constructed by the level definition are invariant with respect to spatial translations, spatial rotations, and uniform spatial magnifications, just as in the principal direction definition since the ridges are located on level surfaces using the principal definition.

The ridges are also invariant under monotonic transformations of the function $f : \mathbb{R}^n \to \mathbb{R}$. Intuitively, you can think of $\mathbb{R}^n$ as a 1-parameter family of level sets of f, each having its function value as an "attribute". Monotonic transformations on f will not change the geometric structure of the level sets; rather it will only change the attributes of the level sets. Since the level sets have the same structure, the ridges will not change.

4 Core Theory

4.1 Linear Scale Space

An axiomatic approach to the processing of an image by a front-end vision system is given here. The term front-end denotes the primary stage of the vision system. It is a syntactical system which represents the input data in a format that can be interpreted by semantical systems in later stages of processing. Essentially a front-end vision system is a geometry engine whose input are intensity data and whose output is concise geometric information describing the original data.

4.1.1 Requirement of Scale

A typical goal in processing an image is to recognize basic objects called *figures* in the image. For a vision system to recognize a figure it must make measurements at various locations with an aperture size, or *scale*, which is proportional to the size, or *width*, of the figure. Let position be denoted by $\mathbf{x}$ and let scale be denoted by σ. As motivation, consider a figure which is a disk of radius r. The width is $w = 2r$, the diameter of the disk. The relationship of scale to width is quantified by $\sigma = \rho r$ where $\rho > 0$ is a constant of proportionality. To recognize the disk, the system must use this aperture size σ. Generally figures are not disks, but can vary in size and shape. The scale-to-width relationship is more appropriate quantified in terms of how changes in width, $dw = 2\,dr$, are related to changes in scale, $d\sigma$. The relationship is concisely

$$d\sigma = \rho dr. \tag{41}$$

For a vision system to recognize all objects in an image, it must use a range of scales. The smallest scale of the system, $\sigma_0 > 0$, is called *inner scale* and is determined by the resolution of the sampling devices of the system. The largest scale of the system, $\sigma_1 > \sigma_0$, is called *outer scale* and is determined by the field of view. Thus, the vision system performs a multiscale analysis by building a *scale space* consisting of the input image at inner scale and measurements derived from the image at all locations and all scales between inner and outer scale.

The input intensities are denoted by $L_0(\mathbf{x})$. For simplicity we assume that the input is defined for all $\mathbf{x} \in \mathbb{R}^n$. The vision system produces multiscale data $L(\mathbf{x}, \sigma)$ for $(\mathbf{x}, \sigma) \in \mathbb{R}^n \times [\sigma_0, \sigma_1]$. The function L can be described generally by the operator equation

$$\begin{aligned} L(\mathbf{x}, \sigma) &= \mathcal{O} L_0(\mathbf{x}),\ (\mathbf{x}, \sigma) \in \mathbb{R}^n \times [\sigma_0, \sigma_1], \\ L(\mathbf{x}, \sigma_0) &= L_0(\mathbf{x}), \end{aligned} \tag{42}$$

where the operator $\mathcal{O}$ is determined by additional requirements imposed on the front-end vision system.

4.1.2 Requirements of Cauality, Linearity, and Invariance

The fundamental constraint on a scale space is that it be *causal*; that is, no spurious detail should be generated with increasing scale. We also assume that the front-end vision system processes its input using the principle of superposition. If $I_0(\mathbf{x})$ and $J_0(\mathbf{x})$ are two input images, then

$$\mathcal{O}(I_0 + J_0)(\mathbf{x}) = \mathcal{O} I_0(\mathbf{x}) + \mathcal{O} J_0(\mathbf{x}). \tag{43}$$

This assumption is valid only for a short period of time after input is received by the vision system. This assumption can be relaxed so that the scale space metric has nonconstant density. In fact, the parameters in the scale space metric can be dependent on the input image.

The front-end vision system should sample and preprocess its input in a symmetric way. Recognizing a figure should be independent of the location orientation, and size of that figure. The measurements we make about the figure must be *invariant* with respect to location (invariance with respect to spatial location), orientation (invariance with respect to spatial rotations), and size (invariance with respect to units of measurement or, equivalently, to uniform magnifications in both the spatial and scale units, *zoom*). If M represents a measurement by the system and if T represents one of the transformations of translation, rotation, or zoom, then invariance means

$$M = T^{-1}MT \tag{44}$$

If a figure is translated (rotated, zoomed), its width measured at some location, and translated back (rotated back, zoomed back), the measured width is the same as that obtained by measuring the original figure with no translation (rotation, zooming).

The requirements of causality, linearity (43), and invariance with respect to translations, rotations, and zoom (44) imply that scale space (42) is generated by linear diffusion,

$$\begin{aligned} &\tfrac{dL}{d\sigma} = \tfrac{1}{\rho}\nabla \cdot (\sigma L),\ (\mathbf{x}, \sigma) \in \mathbb{R}^n \times [\sigma_0, \sigma_1], \\ &L(\mathbf{x}, \sigma_0) = L_0(\mathbf{x}). \end{aligned} \tag{45}$$

In terms of the physical model for the heat equation, ρ is the density function and σ is the conductance function.

Multiscale data is generated by running the linear diffusion equation on the initial image. The system is

$$\begin{aligned} &\rho\sigma u_\sigma = \sigma^2 \nabla^2 u,\ \ x \in \mathbb{R}^n, \sigma > \sigma_0 \\ &u(x, \sigma_0) = I(x) \end{aligned}$$

where $I(x)$ is the initial image and σ_0 is the inner scale. Letting $\tau = \sqrt{\sigma^2 - \sigma_0^2}/\rho$ and $v(x, \tau) = u(x, \sigma)$ tranforms the system to

$$\begin{aligned} &\tau v_\tau = \tau^2 \nabla^2 v,\ \ x \in \mathbb{R}^n, \tau > 0 \\ &v(x, 0) = I(x) \end{aligned}$$

For $n = 1$, assume that the initial data is represented as

$$I(x) = \sum_{k=0}^{n-1} I_k \delta(x - x_k)$$

where $\{I_k\}_{k=0}^{n-1}$ is the discrete signal data and where the x_k are uniformly spaced by $\Delta x > 0$. The solution to the linear diffusion equation is

$$u(x, \sigma) = \frac{\rho}{\sqrt{2\pi}\sqrt{\sigma^2 - \sigma_0^2}} \sum_{j=0}^{n-1} I_j F\left(\frac{\rho(x - x_j)}{\sqrt{\sigma^2 - \sigma_0^2}}\right) = \frac{1}{\sqrt{2\pi}\sqrt{r^2 - r_0^2}} \sum_{j=0}^{n-1} I_j F\left(\frac{\rho(x - x_j)}{\sqrt{r^2 - r_0^2}}\right)$$

where $F(z) = \exp(-z^2/2)$, $r = \sigma/\rho$, and $r_0 = \sigma_0/\rho$. At the grid points x_i, the solution to the diffusion is

$$u(x_i, \sigma) = \frac{1}{\sqrt{2\pi}\sqrt{r^2 - r_0^2}} \sum_{j=0}^{n-1} I_j F\left(\frac{\Delta x(i - j)}{\sqrt{r^2 - r_0^2}}\right).$$

For $n = 2$, assume that the initial data is represented as

$$I(x, y) = \sum_{k=0}^{n-1} \sum_{\ell=0}^{m-1} I_{k\ell} \delta(x - x_k)\delta(y - y_\ell)$$

where $\{I_{k\ell}\}_{k=0,\ell=0}^{n-1,m-1}$ is the discrete image data, the x_k are uniformly spaced by $\Delta x > 0$, and y_ℓ are uniformly spaced by $\Delta y > 0$. The solution to the linear diffusion equation is

$$\begin{aligned} u(x,y,\sigma) &= \frac{\rho^2}{2\pi(\sigma^2-\sigma_0^2)} \sum_{k=0}^{n-1} \sum_{\ell=0}^{m-1} I_{k\ell} F\left(\frac{\rho(x-x_k)}{\sqrt{\sigma^2-\sigma_0^2}}\right) F\left(\frac{\rho(y-y_\ell)}{\sqrt{\sigma^2-\sigma_0^2}}\right) \\ &= \frac{1}{2\pi(r^2-r_0^2)} \sum_{k=0}^{n-1} \sum_{\ell=0}^{m-1} I_{k\ell} F\left(\frac{(x-x_k)}{\sqrt{r^2-r_0^2}}\right) F\left(\frac{(y-y_\ell)}{\sqrt{r^2-r_0^2}}\right). \end{aligned}$$

At the grid points x_i, y_j, and σ_k, the solution to the diffusion is

$$u(x_i,y_j,\sigma) = \frac{1}{2\pi(r^2-r_0^2)} \sum_{k=0}^{n-1} \sum_{\ell=0}^{m-1} I_{k\ell} F\left(\frac{\Delta x(i-k)}{\sqrt{r^2-r_0^2}}\right) F\left(\frac{\Delta y(j-\ell)}{\sqrt{r^2-r_0^2}}\right).$$

Similar formulas hold for dimensions $n \geq 3$. If derivatives of u are required, you'll need to compute derivatives of $F(z)$. Here is a recursive formula for generating those derivatives. The $P_k(z)$ are polynomials of degree k:

$$\begin{aligned} F^{(k)}(z) &= P_k(z)F(z) \\ P_0(z) &\equiv 1 \\ P_{k+1}(z) &= P_k'(z) - zP_k(z) \end{aligned}$$

4.2 Diffusion and Metric

The diffusion process *generates* the multiscale data. The front-end vision system also must *interpret* this data in a geometric way in order to construct the representations that later stages of processing will use. Geometric interpretation requires imposing a metric on scale space. To obtain the desired invariances, note that a measured *spatial difference* is meaningful only in the context of the scale at which it is measured. Similarly, when making multiscale measurements, a measured *scale difference* is meaningful only in the context of the scale at which it is measured. These assumptions suggest specifying differential forms as the measurement tools. As a Euclidean space the 1-forms used for $\mathbb{R}^n \times [\sigma_0, \sigma_1]$ are dx_i, $1 \leq i \leq n$, and $d\sigma$. However, to retain the desired invariances, the dimensionless 1-forms to be used for scale space measurements are dx_i/σ, $1 \leq i \leq n$, and $d\sigma/\sigma$. The geometry of scale space is determined by the metric involving these forms:

$$ds^2 = \frac{d\mathbf{x} \cdot d\mathbf{x}}{\sigma^2} + \frac{d\sigma^2}{\rho^2\sigma^2}. \tag{46}$$

In order to compare spatial differences $d\mathbf{x}$ and scale differences $d\sigma$ we need to use the proportionality constant between width and scale (41). The "units" of $d\mathbf{x}$ and $d\sigma/\rho$ are the same. The inclusion of ρ in the metric allows us to combine space and scale so that geometric information can be properly interpreted. We will see in a later section that scale space with metric (46) has a non-Euclidean geometry.

The metric and the diffusion process are intimately linked together. If $M(\mathbf{x}, \sigma)$ represents a real-valued measurement in scale space, then changes in M are measured as

$$dM = \sum_{i=1}^{n} \frac{\partial M}{\partial x_i} dx_i + \frac{\partial M}{\partial \sigma} d\sigma = \sum_{i=1}^{n} \sigma \frac{\partial M}{\partial x_i} \frac{dx_i}{\sigma} + \rho\sigma \frac{\partial M}{\partial \sigma} \frac{d\sigma}{\rho\sigma}.$$

The natural derivatives to take in scale space are therefore $\sigma \partial M/\partial x_i$, $1 \leq i \leq n$, and $\rho\sigma\partial M/\partial\sigma$. Note that these quantities are dimensionless. Now the diffusion can be viewed as

$$\rho\sigma \frac{\partial}{\partial \sigma} L = (\sigma\nabla) \cdot (\sigma\nabla) L,$$

where the left-hand side is a single application of the scale space derivative with respect to scale and the right-hand side is a repeated application of the scale space spatial gradient. We will also see in a later section how measurements of figures, such as figure width, are based on the scale space differentiation.

4.3 Boundary Measurements

Our own vision system appears to recognize a figure in an image by locating and pairing opposing boundaries of the figure. The boundaries are usually noticeable because of high contrast in luminance at those locations. High contrast is related to locally large directional gradients in luminance. However, the pairing of opposing boundaries is a global task which requires the full power of multiscale analysis and scale space. We propose that the vision system makes a measure of *boundariness* at each position $\mathbf{x}$, scale σ, and orientation $\mathbf{u}$. Denote this function as $B(\mathbf{x}, \sigma, \mathbf{u})$. Specific choices for B might depend on the system and the task, but reasonable ones appear to be

$$B(\mathbf{x}, \sigma, \mathbf{u}) = \mathbf{u} \cdot \sigma \nabla L(\mathbf{x}\sigma) \tag{47}$$

for bright (dark) figures on a dark (bright) background, or

$$B(\mathbf{x}, \sigma, \mathbf{u}) = |\mathbf{u} \cdot \sigma \nabla L(\mathbf{x}\sigma)| \tag{48}$$

for figures where the foreground/background intensity ratios vary through 1 as the figure boundary is traversed.

IMPLEMENTATION DETAILS FOR BOUNDARINESS. Maybe suggest alternatives such as Gabor functions?

4.4 Medialness Functions

This document describes medialness functions which are obtained by linear convolution with a scale-dependent kernel (scale measured as radial units r). The continuous model is described first, followed by the discretization and implementation details.

Let $I : \mathbb{R}^n \to \mathbb{R}$ be an image, say $I = I(x)$ where $x \in \mathbb{R}^n$. A medialness function, $M : \mathbb{R}^n \times (0, \infty) \to \mathbb{R}$, is given by the linear convolution

$$M(x, r) = K(x, r) \oplus I(x) = \int_{\mathbb{R}^n} K(\bar{x}, r) I(x - \bar{x})\, d\bar{x}_1 \cdots d\bar{x}_n$$

where kernel $K(x, r)$ is subject to the constraints

$$\int_{\mathbb{R}^n} K(x, r)\, dx_1 \cdots dx_n = 0 \quad \text{and} \quad \int_{\mathbb{R}^n} |K(x, r)|\, dx_1 \cdots dx_n = 1.$$

For now I will concentrate on kernels that are radially symmetric in space and self-similar and dissipative in scale; that is, kernels are of the form

$$K(x, r) = \frac{C}{r^n} F\left(\sqrt{\frac{|x|}{r}}\right)$$

where constant C is chosen so that the kernel satisfies the given integral constraints. Notice that F captures the effects of radial symmetry and self-similarity. The factor r^{-n} represents the dissipative behavior (the "amplitude" of F decreases as r increases). The prototypical example is the normalized Laplacian of a Gaussian kernel where $F(R) = R^{n-1}(n - R^2)\exp(-R^2/2)$.

4.4.1 2D Medialness

For two spatial dimensions, the kernels are of the form

$$K(x,y,r) = \frac{1}{2\pi r^2} F(\sqrt{x^2+y^2}/r).$$

Using the change of variables $x = rR\cos\theta$ and $y = rR\sin\theta$, the integral constraints for K reduce to ones for F:

$$\int_0^\infty RF(R)\,dR = 0 \text{ and } \int_0^\infty R|F(R)|\,dR = 1.$$

Using the change of variables $\bar{x} = rR\cos\theta$ and $\bar{y} = rR\sin\theta$, medialness for 2 spatial dimensions is

$$\begin{aligned} M(x,y,r) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \frac{1}{2\pi r^2} F(\sqrt{\bar{x}^2+\bar{y}^2}/r) I(x-\bar{x}, y-\bar{y})\,d\bar{x}d\bar{y} \\ &= \int_0^\infty RF(R) \left(\frac{1}{2\pi}\int_0^{2\pi} I(x-rR\cos\theta, y-rR\sin\theta)\,d\theta\right) dR. \end{aligned}$$

Define the function

$$A(x,y,\lambda) = \frac{1}{2\pi}\int_0^{2\pi} I(x-\lambda\cos\theta, y-\lambda\sin\theta)\,d\theta. \tag{49}$$

This function is the average of the image values on a circle centered at (x,y) with radius λ. The medialness is therefore

$$M(x,y,r) = \int_0^\infty RF(R)A(x,y,rR)\,dR. \tag{50}$$

If the circular averages are computed as a preprocessing step, then any medialness function can be computed as a one-dimensional integral rather than as a multidimensional convolution.

Equation (49) is discretized as follows. For a specified radius λ, N_λ uniformly spaced points are selected on the circle centered at (x,y) with radius λ. The i^{th} point is $(x_i,y_i) = (\lambda\cos\theta_i, \lambda\sin\theta_i)$. The angular difference between consecutive points is $\Delta\theta = 2\pi/N_\lambda$. The integral is approximated with a Riemann sum as

$$A(x,y,\lambda) \doteq \frac{1}{N_\lambda}\sum_{i=0}^{N_\lambda-1} I(x-x_i, y-y_i). \tag{51}$$

To calculate the image values at the (nonintegral) circle points requires interpolation. However, I avoid this by using Bresenham's circle algorithm to generate a set of points which are densely packed and approximately on the circle of the desired radius, λ, which now takes on only integer values. The number of samples N_λ becomes the number of points generated by the circle algorithm.

Equation (50) is discretized as follows. A cutoff for the integration is selected, call it C, so

$$M(x,y,r) \doteq \int_0^C RF(R)A(x,y,rR)\,dR.$$

The number of tabulated values for $F(R)$ is constant throughout the medialness calculations. For small r, the restriction that $\lambda = rR$ take on only integer value would lead to an inaccurate approximation of this integral (using Riemann sums). For example, if $C = 2$ and $r = 1$, then R can be only 0, 1, or 2 in order that rR be an integer. The Riemann sum approximation is $M(x,y,1) \doteq 0 + F(1)A(x,y,1) + 2F(2)A(x,y,2)$. For typical kernels, $F(C) = 0$, so the approximation to $M(x,y,1)$ depends on a single kernel value $F(1)$.

Instead, I treat $A(x, y, \lambda)$ as a piecewise constant function in the continuous variable λ where the constant values of A are known for integer values of λ. Let the number of tabulated values for $F(R)$ be denoted S. The samples are uniformly spaced, say $R_k = Ck/S$ and $F_k = F(R_k)$ for $0 \leq k < S$. The integral (over $[0, C]$) of medialness becomes

$$\int_0^C RF(R)A(x, y, rR)\, dR = \frac{C}{S} \sum_{k=0}^{S-1} R_k F_k A(x, y, \lceil rR_k \rceil)$$

where $\lceil v \rceil$ is the ceiling function which returns the smallest integer greater than v. Rather than store only the values $R_k F_k$ and continually sum them, it is possible to precompute the partial sums and do a table lookup for the integration. More precisely, the circular averages $A(x, y, i)$ are computed for $0 \leq i < rC$ for a given integer radius $r \geq 1$. For a given r and a selected i, the range of k indices for which the corresponding $R_k F_k$ multiply $A(x, y, i)$ is

$$\frac{Si}{Cr} \leq k < \frac{S(i+1)}{Cr}.$$

The summation over k can be replaced by summation over i, and the medialness is approximated as

$$M(x, y, r) \doteq \sum_{i=1}^{\lfloor rC \rfloor} \left(\sum_{k=\frac{Si}{Cr}}^{\frac{S(i+1)}{Cr}-1} R_k F_k \right) A(x, y, i). \quad (52)$$

4.4.2 2D Code

The objects which handle medialness computations are implemented in files `medial2d.{h,c}`. The class is named `Medial2D`. Its description is given below.

```
#include <Magic.h>

typedef struct {
    int samples;   // number of samples S for F(R)
    float* data;   // data = new float[samples] stores the F_k values
    float cutoff;  // 0 <= R <= cutoff C
} Kernel2D;

class Medial2D : public Spline3
{
private:
    Image_FLOAT& im;    // initial image
    Image_FLOAT& med;   // medialness image

    // callback to Spline3 object
    static Medial2D* object;
    static void compute_function_values ();

    int kradii;       // class copy of S/C
    float kcutoff;    // class copy of C
    float** kernel;   // kernel[r][i] are partial sums of R_k F_k
```

```
    void build_kernel (Kernel2D& kern);
        // kern[i] is F(R_i), this routine produces R_i F(R_i) and
        // adjusts the values so that the integral(R F(R)) = 0 and
        // integral(R |F(R)|) = 1

    Shell2D* circle;
        // circle[i] is Bresenham mask for circle of radius i+1

public:
    Medial2D (Image_FLOAT& _im, Image_FLOAT& _med, Kernel2D& kern);
    ~Medial2D ();

    float medialness (int x, int y, int r);

    static Kernel2D morse_kernel (int samples, float rho);
    static Kernel2D fritsch_kernel (int samples, float cutoff);
};
```

The `Kernel2D` structure keeps information about the function $F(R)$. The field `samples` stores the number of samples S. The field `cutoff` stores the maximum value C for the variable R. The value `data[k]`, $0 \le k < S$, stores $F_k = F(R_k)$ where $R_k = kC/S$. The caller of `Medial2D` is responsible for allocating and deallocating the `data` field in the `Kernel2D` passed to the constructor. Two standard kernels are provided (Morse, Fritsch). A typical usage would be

```
Image_FLOAT im("head.im");
int rmax = (im.bound(0) < im.bound(1) ? im.bound(0)/4 : im.bound(1)/4);
Image_FLOAT med(im.bound(0),im.bound(1),rmax);
Kernel2D kern = Medial2D::morse(256,0.25);
Medial2D medial(im,med,kern);
delete[] kern.data;
```

I'll hack a `Kernel2D` class later to avoid having to explicitly delete the kernel data.

Data member `im` is a reference to the 2D image whose medialness is to be computed. The medialness values at integer locations (x, y, r) will be stored in a 3D image. The caller of `Medial2D` is responsible for allocating this image. Data member `med` is a reference to this 3D image.

The inherited spline object `Spline3` is a B-spline interpolator. The `Medial2D` constructor sets the interpolator to be bicubic. The degree may be changed, but I needed at least degree 3 for applications which require continuous second-order derivatives of medialness. Typically the buffer `med` supplied by the user has no medialness values stored *a priori*, so the buffer should be initialized to `MAXFLOAT`. The routine `void compute_function_values()` is a callback to the `Spline3` class. If a `MAXFLOAT` is encountered by the spline object, then the callback function is executed to fill in the missing data. The callback itself makes calls to the member function `float medialness(int,int,int)`.

Data members `kradii` and `kcutoff` are initialized by the member function `void build_kernel(Kernel2D&)` to S/C and S, respectively. The double array `kernel[r][i]` is created by the same member function to store the sums

$$\sum_{k=\frac{Si}{Cr}}^{\frac{S(i+1)}{Cr}-1} R_k F_k$$

which appear in equation (52). These values are used in the convolution which is performed by function member `float medialness(int, int,int)`.

The single array `circle[r]`, $0 \le r < C * r_{\max}$ stores the points for a circle (centered at the origin) of radius $r+1$. The value of $r_{\max}$ is the r-bound for the medialness buffer supplied by the caller.

Medialness at the integer grid points are computed implicitly as needed by the spline object. If there is a need to compute them explicitly, use the function member `float medialness(int,int,int)`.

```
Medial2D m(image,med,kern);
int xi = 10, yi = 20, ri = 4
float xf = 10.5, yf = 20.5, rf = 3.8;

// compute medialness at an integer point
float result1 = m.medialness(xi,yi,ri);

// compute medialness at a noninteger points (uses spline)
float result2 = m(xf,yf,rf);

// compare results of two calls
if ( m.medialness(10,20,4) != m(10,20,4) )
    cout << "results not the same" << endl;
```

The results produced by the two calls at an integer grid point are generally not equal. The B-spline interpolation is based on the values produced by method `medialness` and it is not an exact interpolation algorithm.

4.4.3 Fritsch Medialness

The Fritsch medialness kernel is based on computing medialness as

$$M(x,y,\sigma) = -\sigma \frac{\partial L}{\partial \sigma}(x,y,\sigma)$$

where L is the solution to the diffusion equation $\sigma L_\sigma = \sigma^2 \nabla^2 L$ for $\sigma \ge \sigma_0 > 0$ with initial conditions $L(x,y,\sigma_0) = I(x,y)$. The kernel is $K(x,y,\sigma) = -\sigma^2 \nabla^2 G(x,y,\sigma)$ where G is a circular Gaussian with standard deviation σ. More explicitly,

$$K(x,y,\sigma) = \frac{1}{2\pi\sigma^2}\left(2 - \left(\frac{x}{\sigma}\right)^2 - \left(\frac{y}{\sigma}\right)^2\right)\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right).$$

In this case the scale σ and radius r are treated as the same parameter, so without loss of generality we use r. Setting $R = \sqrt{(x^2+y^2)/r}$, the nondissipative part of the kernel is

$$F(R) = (2 - R^2)\exp(-R^2/2).$$

The `Medial2D` method to compute this function is straightforward.

4.4.4 Morse Medialness

The Morse medialness kernel is based on computing medialness as

$$M(x, y, \sigma) = -\int_0^{2\pi} u(\theta) \cdot \sigma \nabla L((x, y) + r u(\theta), \sigma)\, d\theta$$

where L is also the solution to the diffusion equation $\sigma L_\sigma = \sigma^2 \nabla^2 L$ for $\sigma \geq \sigma_0 > 0$ with initial conditions $L(x, y, \sigma_0) = I(x, y)$. The vector $u(\theta) = (\cos\theta, \sin\theta)$ is a direction vector. The idea is to compute the medialness of point (x, y) relative to (possible) boundaries at a distance r units away. The function M is in effect the average of directional Gaussian derivatives of intensity around a circle centered at (x, y) with radius r, the orientation of the Gaussians being towards the circle center.

This medialness can be written as a linear convolution $M(x, y, \sigma) = K(x, y, \sigma) \oplus I(x, y)$ where

$$K(x, y, \sigma) = -\int_0^{2\pi} u(\theta) \cdot \sigma \nabla G((x, y) + r u(\theta), \sigma)\, d\theta$$

where G is a circular Gaussian with standard deviation σ. Some calculations will show that

$$K(x, y, \sigma) = \int_0^{2\pi} \left(\frac{x}{\sigma}\cos\theta + \frac{y}{\sigma}\sin\theta + \frac{r}{\sigma}\right) \frac{1}{2\pi\sigma^2} \exp\left(-0.5\left[\left(\frac{x}{\sigma} + \frac{r}{\sigma}\cos\theta\right)^2 + \left(\frac{y}{\sigma} + \frac{r}{\sigma}\sin\theta\right)^2\right]\right) d\theta.$$

The relationship between scale σ and radius r is allowed to be slightly more complicated. We assume that $\sigma = \rho r$ for some positive constant ρ. Experiments have shown that $0.125 \leq \rho \leq 0.5$ is a suitable range. For $\rho = 1$, the kernel is very similar to the Fritsch kernel for medialness. Using the scale-radius relationship, and setting $R = \sqrt{x^2 + y^2}/r$ and $\phi = \tan^{-1}(y/x)$, it can be shown that

$$K(x, y, r) = \frac{1}{2\pi\rho^3 r^2} \int_0^{2\pi} (1 + R\cos(\theta - \phi)) \exp\left(-\frac{1}{2\rho^2}\left(R^2 + 2R\cos(\theta - \phi) + 1\right)\right) d\theta.$$

Note that ϕ depends on x and y, so at first glance the integral is not simply a function of R. However, cosine is a periodic function and the interval of integration is over a full period. The integral value is therefore independent of the phase, so

$$K(x, y, r) = \frac{1}{2\pi\rho^3 r^2} \int_0^{2\pi} (1 + R\cos\theta) \exp\left(-\frac{1}{2\rho^2}(R^2 + 2R\cos\theta + 1)\right) d\theta.$$

The kernel is now in the form $Cr^{-2}F(R)$ for some constant C where $R = \sqrt{x^2 + y^2}/r$. The nondissipative part of the kernel is

$$F(R) = \int_0^{2\pi} (1 + R\cos\theta) \exp\left(-\frac{1}{2\rho^2}(R^2 + 2R\cos\theta + 1)\right) d\theta.$$

Function $F(R)$ is related to *modified Bessel functions.* The modified Bessel function of order zero is an even function given by

$$I_0(z) = \frac{1}{\pi} \int_0^{\pi} \exp(z\cos\theta)\, d\theta = \sum_{m=0}^{\infty} \frac{(z^2/4)^m}{(m!)^2}.$$

This function is also a solution to the second-order linear differential equation: $z^2 w'' + zw' - z^2 w = 0$. The modified Bessel function of order one is an odd function given by

$$I_1(z) = I_0'(z) = \frac{1}{\pi} \int_0^\pi \cos\theta \exp(z \cos\theta)\, d\theta$$

and it is a solution to the differential equation: $z^2 w'' + zw' - (z^2 + 1)w = 0$. We can rewrite

$$F(R) = 2\pi \exp\left(-\frac{R^2 + 1}{2\rho^2}\right) \left[I_0(R/\rho^2) - RI_1(R/\rho^2)\right].$$

We can take advantage of this relationship to compute F by calculating the modified Bessel functions. Polynomial approximations to these functions can be found in the section on special functions (and in `specfunc.c`).

4.4.5 3D Medialness

For three spatial dimensions, the kernels are of the form

$$K(x, y, z, r) = \frac{1}{4\pi r^3} F(\sqrt{x^2 + y^2 + z^2}/r).$$

Using the change of variables $x = rR\cos\theta \sin\phi$, $y = rR\sin\theta\sin\phi$, and $z = rR\cos\phi$, the integral constraints for K reduce to ones for F:

$$\int_0^\infty R^2 F(R)\, dR = 0 \text{ and } \int_0^\infty R^2 |F(R)|\, dR = 1.$$

Using the change of variables $\bar{x} = rR\cos\theta\sin\phi$, $\bar{y} = rR\sin\theta\sin\phi$, and $\bar{z} = rR\cos\phi$, medialness for 3 spatial dimensions is

$$\begin{aligned} &M(x, y, z, r) \\ &= \int_{-\infty}^\infty \int_{-\infty}^\infty \int_{-\infty}^\infty \tfrac{1}{4\pi r^3} F(\sqrt{\bar{x}^2 + \bar{y}^2 + \bar{z}^2}/r) I(x - \bar{x}, y - \bar{y}, z - \bar{z})\, d\bar{x} d\bar{y} d\bar{z} \\ &= \int_0^\infty R^2 F(R) \left(\tfrac{1}{4\pi} \int_0^{2\pi} \int_0^\pi I(x - rR\cos\theta\sin\phi, y - rR\sin\theta\sin\phi, z - rR\cos\phi)\, \sin\phi d\phi d\theta\right) dR. \end{aligned}$$

Define the function

$$A(x, y, z, \lambda) = \frac{1}{4\pi} \int_0^{2\pi} \int_0^\pi I(x - \lambda\cos\theta\sin\phi, y - \lambda\sin\theta\sin\phi, z - \lambda\cos\phi)\, \sin\phi d\phi d\theta. \quad (53)$$

This function is the average of the image values on a sphere centered at (x, y, z) with radius λ. The medialness is therefore

$$M(x, y, z, r) = \int_0^\infty R^2 F(R) A(x, y, z, rR)\, dR. \quad (54)$$

If the spherical averages are computed as a preprocessing step, then any medialness function can be computed as a one-dimensional integral rather than as a multidimensional convolution.

Equation (53) is discretized as follows. For a specified radius λ, N_λ densely packed points are selected on the sphere using a Bresenham-like algorithm as in the 2D case. The integral is approximated with a Riemann sum as

$$A(x,y,z,\lambda) \doteq \frac{1}{N_\lambda} \sum_{i=0}^{N_\lambda - 1} I(x - x_i, y - y_i, z - z_i). \tag{55}$$

Equation (54) is discretized as follows. A cutoff for the integration is selected, call it C, so

$$M(x,y,z,r) \doteq \int_0^C R^2 F(R) A(x,y,z,rR)\, dR.$$

The number of tabulated values for $F(R)$ is constant throughout the medialness calculations. As in the 2D case, I treat $A(x,y,z,\lambda)$ for $\lambda = 1, 2, \ldots$ as a piecewise constant function in the continuous variable λ. Let the number of tabulated values for $F(R)$ be denoted S. The samples are uniformly spaced, say $R_k = Ck/S$ and $F_k = F(R_k)$ for $0 \le k < S$. The integral (over $[0, C]$) of medialness becomes

$$\int_0^C R^2 F(R) A(x,y,z,rR)\, dR = \frac{C}{S} \sum_{k=0}^{S-1} R_k^2 F_k A(x,y,z,\lceil rR_k \rceil).$$

Partial sums of $R_k^2 F_k$ are precomputed. The spherical averages $A(x,y,z,i)$ are computed for $0 \le i < rC$ for a given integer radius $r \ge 1$. The summation over k can be replaced by summation over i, and the medialness is approximated as

$$M(x,y,z,r) \doteq \sum_{i=1}^{\lfloor rC \rfloor} \left(\sum_{k=\frac{Si}{Cr}}^{\frac{S(i+1)}{Cr}-1} R_k^2 F_k \right) A(x,y,z,i). \tag{56}$$

4.4.6 3D Code

The objects which handle medialness computations are implemented in files `medial3d.{h,c}`. The class is named `Medial3D`. Its description is given below.

```
#include <Magic.h>

typedef struct {
    int samples;    // number of samples S for F(R)
    float* data;    // data = new float[samples] stores the F_k values
    float cutoff;   // 0 <= R <= cutoff C
} Kernel3D;

class Medial3D : public Spline4
{
private:
    Image_FLOAT& im;    // initial image
    Image_FLOAT& med;   // medialness image

    // for callback to Spline4 object
```

```
    static Medial3D* object;
    static void compute_function_values ();

    // kernel computation and convolution
    int kradii;       // class copy of S/C
    float kcutoff;    // class copy of C
    float** kernel;   // kernel[r][i] are partial sums of R_k^2 F_k
    void build_kernel (Kernel3D& kern);
        // kern[i] is F(R_i), this routine produces R_i F(R_i) and
        // adjusts the values so that the integral(R F(R)) = 0 and
        // integral(R |F(R)|) = 1
    Shell3D* sphere;
        // sphere[i] is Bresenham mask for sphere of radius i+1

public:
    Medial3D (Image_FLOAT& _im, Image_FLOAT& _med, Kernel3D& kern);
    ~Medial3D ();

    float medialness (int* xi);

    static Kernel3D morse_kernel (int samples, float rho);
    static Kernel3D fritsch_kernel (int samples, float cutoff);
};
```

The descriptions of all the members is nearly identical to that of the 2D case. Rather than circle masks, this class has spherical masks. The value `sphere[r]` is the mask for a sphere centered at the origin of radius $r + 1$.

4.4.7 Fritsch Medialness

The Fritsch medialness kernel is based on computing medialness as

$$M(x, y, z, \sigma) = -\sigma \frac{\partial L}{\partial \sigma}(x, y, \sigma)$$

where L is the solution to the diffusion equation $\sigma L_\sigma = \sigma^2 \nabla^2 L$ for $\sigma \geq \sigma_0 > 0$ with initial conditions $L(x, y, z, \sigma_0) = I(x, y, z)$. The kernel is $K(x, y, z, \sigma) = -\sigma^2 \nabla^2 G(x, y, z, \sigma)$ where G is a circular Gaussian with standard deviation σ. More explicitly,

$$K(x, y, z, \sigma) = \frac{1}{(2\pi)^{3/2}\sigma^3} \left(3 - \left(\frac{x}{\sigma}\right)^2 - \left(\frac{y}{\sigma}\right)^2 - \left(\frac{z}{\sigma}\right)^2\right) \exp\left(-\frac{x^2 + y^2 + z^2}{2\sigma^2}\right).$$

Although the constant in the kernel is not $1/(4\pi)$, we will treat σ and r as the same. The code will make sure the kernel has absolute integral equal to one. Setting $R = \sqrt{(x^2 + y^2 + z^2)/r}$, the nondissipative part of the kernel is

$$F(R) = (3 - R^2)\exp(-R^2/2).$$

The `Medial2D` method to compute this function is straightforward.

4.4.8 Morse Medialness

The Morse medialness kernel is based on computing medialness as

$$M(x, y, z, \sigma) = -\int_0^{2\pi}\int_0^{\pi} u(\theta, \phi) \cdot \sigma\nabla L((x, y.z) + ru(\theta, \phi), \sigma)\ \sin\phi d\phi d\theta$$

where L is also the solution to the diffusion equation $\sigma L_\sigma = \sigma^2\nabla^2 L$ for $\sigma \geq \sigma_0 > 0$ with initial conditions $L(x, y, z, \sigma_0) = I(x, y, z)$. The vector $u(\theta, \phi) = (\cos\theta\sin\phi, \sin\theta\sin\phi, \cos\phi)$ is a direction vector. The idea is to compute the medialness of point (x, y, z) relative to (possible) boundaries at a distance r units away. The function M is in effect the average of directional Gaussian derivatives of intensity around a sphere centered at (x, y, z) with radius r, the orientation of the Gaussians being towards the sphere center.

This medialness can be written as a linear convolution $M(x, y, z, \sigma) = K(x, y, z, \sigma) \oplus I(x, y, z)$ where

$$K(x, y, z, \sigma) = -\int_0^{2\pi}\int_0^{\pi} u(\theta, \phi) \cdot \sigma\nabla G((x, y.z) + ru(\theta, \phi), \sigma)\ \sin\phi d\phi d\theta.$$

As in the 2D case, let $\sigma = \rho r$ for some positive constant ρ. Some calculations will show that

$$K(x, y, z, r) = \frac{1}{(2\pi)^{3/2}\rho^3 r^3}\int_0^{2\pi}\int_0^{\pi}(1 + T)\exp\left(-\frac{1}{2\rho^2}(R^2 + 2T + 1)\right)\ \sin\phi d\phi d\theta$$

where

$$T(x, y, z, r, \theta, \phi) = \frac{x}{r}\cos\theta\sin\phi + \frac{y}{r}\sin\theta\sin\phi + \frac{z}{r}\cos\phi.$$

Although not proved here, the claim is that the integral is a function only of $R = \sqrt{x^2 + y^2 + z^2}/r$. Consequently the integral may be simplified by considering $x = 0$, $y = 0$, and $z = rR$:

$$\begin{aligned} K(R, r) &= \tfrac{1}{(2\pi)^{3/2}\rho^3 r^3}\int_0^{2\pi}\int_0^{\pi}(1 + R\cos\phi)\exp\left(-\tfrac{1}{2\rho^2}(R^2 + 2R\cos\phi + 1)\right)\ \sin\phi d\phi d\theta \\ &= \tfrac{\exp\left(-\frac{R^2+1}{2\rho^2}\right)}{(2\pi)^{1/2}\rho^3 r^3}\int_0^{\pi}(1 + R\cos\phi)\exp\left(-\tfrac{1}{\rho^2}R\cos\phi\right)\ \sin\phi d\phi. \end{aligned}$$

The presence of $\sin\phi d\phi$ allows the integral to be evaluated in closed form:

$$K(R, r) = \frac{2\sqrt{2\pi}\exp\left(-\frac{R^2+1}{2\rho^2}\right)}{\rho^2 r^3 R}\left[(1 + \rho^2)\sinh(R/\rho^2) - R\cosh(R/\rho^2)\right].$$

4.5 Cores Implementation

The core implementations are found in the applications directory, `magic3/appls/core?d`, where `?` is 2 or 3. The code only finds 1-dimensional ridges in 3D scale space or 4D scale space. The code is in `core?d.zip`.