Marching Optimal-Parameter Ridges: an Algorithm to Extract Shape Loci in 3D Images

Jacob D. Furst, Stephen M. Pizer

Medical Image Display and Analysis Group
University of North Carolina, Chapel Hill, NC 27599
furst@cs.unc.edu
fax: 1 (919) 962-1799

Abstract. This paper presents a method for identifying image loci that can be used as a basis for object segmentation and image registration. The focus is on 1D and 2D shape loci in 3D images. This method, called *marching ridges*, uses generalized height ridges, oriented medialness measures and a marching cubes like algorithm to extract optimal scale-orientation cores. This algorithm can can also be used for other image processing tasks such as finding intensity skeletons of objects and identifying object boundaries.

Keywords: marching ridges, scale-space, orientation, medialness

1 Introduction

There are a variety of problems in medical imaging for which height ridges are generally useful. For example, height ridges of intensity can be used to track the centers of blood vessels. Ridges of boundariness can also be used to identify edges using the Canny definition. Of particular interest to this paper are height ridges of medialness (called cores) that can be used as the basis for a figural shape model, the deformable shape locus, described by Pizer [13].

In this paper, we will concentrate on height ridges as they are defined in [7]. In particular, we will use the definition of optimal parameter ridges, in which certain parameters of the space in which we are finding ridges are distinguished. These ideas can be used for identifying boundaries, in which the boundary normal provides a distinguished orientation or in seaching for ridges (cores) in a scale-orientation space of medialness, in which we will distinguish scale and orientation.

This paper describes an algorithm for finding cores of 3D images, similar in structure to Lorensen's marching cubes [10] and Bloomenthals' tracked partitioning [1] (when finding 2D cores) and Thirion's [14] marching lines (when finding 1D cores). This paper presents algorithms and results for 2D cores. 1D cores in 3D images are the subject of current work.

2 Methods

The methods used in this paper combine much of the work described in Section 1 into a common algorithm called marching ridges. We measure an image using oriented medialness kernels and then find ridges within the marching framework using the optimal scale-orientation definition of height ridges.

2.1 Height ridges

Maximum convexity height ridges The maximum convexity height ridge is dealt with extensively in Eberly's fine book [3]. Here we will only repeat the definition of the maximum convexity height ridge:

Definition 21 *(Maximum Convexity Height Ridge) An m-dimensional (mD) maximum convexity height ridge of a function $M : \mathcal{X} \rightarrow \mathbb{R}$, or m-ridge of M, is an mD locus in the (nD, $n > m$) domain of M. In general, the definition of this locus involves*

1. *given the Hessian matrix of second derivatives $\mathcal{H}(M)$, order the Hessian's eigenvalues λ_i according to $\lambda_i < \lambda_{i+1}$. Let the vector $\boldsymbol{v}^i$ be the eigenvector associated with λ_i. The first $n - m$ such $\boldsymbol{v}^i$ are transverse directions.*
2. *require that M have a relative maximum in each of the transverse directions; that is,*
   (a) $\boldsymbol{v}^i \cdot \nabla(M)(\bar{x}) = 0$ *for* $i = 1, \ldots, n - m$, *where* $\nabla(M)$ *is the gradient of* M, *and*
   (b) $\lambda_{n-m} < 0$ *at* $\bar{x}$

Generalized height ridges We have expanded upon Eberly's definition to create a more generalized definition of height ridge, one that includes many definitions and algorithms already existing in the image processing literature (See [8, 9, 5]).

Definition 22 *(Height Ridge) An m-dimensional (mD) height ridge of a function $M : \mathcal{X} \rightarrow \mathbb{R}$, or m-ridge of M, is an mD locus in the (nD, $n > m$) domain of M. In general, the definition of this locus involves*

1. *a rule for choosing $n - m$ linearly independent directions, $\boldsymbol{v}^i$, transverse to the putative ridge at a location x, and*
2. *requiring that M have a relative maximum in each of the (1D) $\boldsymbol{v}^i$ directions; that is,*
   (a) $\boldsymbol{v}^i \cdot \nabla(M)(\bar{x}) = 0$ *for* $i = 1, \ldots, n - m$, *and*
   (b) $(\boldsymbol{v}^i)^t \, \mathcal{H}(M)(\bar{x}) \, \boldsymbol{v}^i < 0$ *for* $i = 1, \ldots, n - m$.

Note that the maximum convexity height ridges are a specific example of the generalized height ridge in which the rule for choosing transverse directions involves an eigen analysis of the Hessian matrix. Also, the generalized definition provides a mechanism for computing Canny edges [2], for example, by choosing one direction transverse to the ridge as the gradient direction, and maximizing gradient magnitude in that direction.

Optimal parameter height ridges There are a number of tasks in image processing, to include finding skeletons [8] and edges [9], that require a choice of directions different than that of the maximal convexity definition. One way to do this is the optimal parameter height ridge. In this work, we choose to calculate optimal parameter ridges for three major reasons:

1. Distinguishing certain parameters seems a natural and intuitive choice when dealing with certain functions. Pizer [13] has described how position, radius, and orientation are natural parameters to separate for modeling shape. The maximum convexity height ridge does not guarantee such separation.
2. The incommensurability of position, radius and orientation can provide for non-Euclidean geometries in the domain of M. In the case of position and scale alone, Eberly [3] has shown a hyperbolic geometry in the domain of M. However, even then, there is a parameter involved relating derivatives in space to derivatives of radius. The optimal parameter hight ridge avoids this notion and guarantees Euclidean derivatives.
3. Algorithmically, the problems of discontinuity in eigensystems becomes unmanageable in higher dimensions. Morse [11] was able to show success in 2D, Eberly [4] in 3D, and Furst [6] some success in 4D, but both Eberly and Furst have failed to robustly generate maximum convexity height ridges for medical images using a 4D domain of position and scale.

Working from our definition of generalized height ridges, we can define an optimal parameter ridge as follows:

Definition 23 *(Optimal Parameter Height Ridge) An m-dimensional (mD) optimal parameter height ridge of a function $M : \mathcal{X} \times \mathcal{S} \rightarrow \mathbb{R}$, or m-ridge of M, is an mD locus in the (nD, $n > m$) domain of M. Define it as follows*

1. *given the Hessian matrix of M restricted to the subspace $\mathcal{S}$, $\mathcal{H}|_{\mathcal{S}}(M)$, let each of the eigenvectors of $\mathcal{H}|_{\mathcal{S}}(M)$ be a transverse direction, and have a rule for choosing the remaining transverse directions from $\mathcal{X}$ in a way possible dependent on the eigenvectors of $\mathcal{H}|_{\mathcal{S}}(M)$.*
2. *requiring that M have a relative maximum in each of the (1D) $\boldsymbol{v}^i$ directions; that is,*
   (a) *the gradient of M restricted to the subsace $\mathcal{S}$, $\nabla|_{\mathcal{S}}(M)$, vanishes, and $\boldsymbol{v}^i \cdot \nabla(M)(\bar{x}) = 0$ for each $\boldsymbol{v}^i \in \mathcal{X}$, and*
   (b) *$\mathcal{H}|_{\mathcal{S}}(M)$ is negative definite at $\bar{x}$ and $(\boldsymbol{v}^i)^t\ \mathcal{H}(M)(\bar{x})\ \boldsymbol{v}^i < 0$ for each $\boldsymbol{v}^i \in \mathcal{X}$.*

Let $M(\bar{x};\bar{s}) : \mathcal{X} \times \mathcal{S} \rightarrow \mathbb{R}$. Let $\mathcal{X}$ be called the underlying image space (typically $\mathbb{R}^2$ or $\mathbb{R}^3$; this paper uses $\mathbb{R}^3$) and let $\mathcal{S}$ be called the distinguished space whose spanning parameters will be optimized. We have identified distinguished spaces containing from one to five dimensions; this paper uses three. We can view these three as a radius and an angular orientation: $\mathbb{R}_+ \times S^2$ (spherical coordinates) or we can view them as a vector: $\mathbb{R}^3$ (Cartesian coordinates). The spherical coordinates provide an intuitive idea of the optimized parameters, while the Cartesian coordinates provide a computational advantage. When looking for a 2D ridge in a 6D space ($\mathbb{R}^3 \times \mathbb{R}^3$) we need four transverse directions. Requiring a maximum of M in the distinguished space $\mathcal{S}$ identifies three transverse directions. As required by Definition 23, these directions are eigenvectors of $\mathcal{H}|_{\mathcal{S}}(M)$. The last transverse direction is taken from the underlying image space $\mathcal{X}$. This direction can be chosen using the maximum convexity rule as given in Definition 21. However, given that one of our distinguished parameters is orientation, we can use the particular orientation that produces a maximum of M in $\mathcal{S}$ to determine the final transverse direction in $\mathcal{X}$. That is, if $\bar{s} \in \mathcal{S}$ is the point at which $M(\bar{x}; \bar{s})$ is maximal for a particular $x \in \mathcal{X}$, then $\boldsymbol{v}^4 = \frac{\bar{s}}{\|\bar{s}\|}$ in the Cartesian formulation of $\bar{s}$ or $\boldsymbol{v}^4 = [\cos s_2 \cos s_3, \sin s_2 \cos s_3, \sin s_3]$ in the spherical formulation of $\bar{s}$.

Once having determined this direction, we must satisfy the derivative conditions to label it a ridge point. We do this by assuming that the locus of points $\{(\bar{x}; \bar{s}) | (\bar{x}; \bar{s}) \in \mathcal{X} \times \mathcal{S}\}$ for which $M(\bar{x}; \bar{s})$ is maximal in $\mathcal{S}$ creates a well-defined manifold $\mathcal{M}$. Except for non-generic situations, this assumption holds. (See [7] for failure of genericity).

We then define a coordinate chart $\bar{i} : \mathcal{X} \to \mathcal{M}$. This coordinate chart is well defined except at folds of $\mathcal{M}$ with respect to its projection onto $\mathcal{X}$. These folds, however, occur only where one of the eigenvalues of $\mathcal{H}|_{\mathcal{S}}(M)$ vanishes. Since a ridge point is possible only where $\mathcal{H}|_{\mathcal{S}}(M)$ is negative definite, the coordinate chart will be well defined at all ridge points.

We then define $\hat{M} : \mathcal{X} \to \mathcal{S} = f \circ i$. We must then satisfy the following two conditions to establish a ridge point at x

1. the first derivative of $\hat{M}$ in the $\boldsymbol{v}^4$ direction ($\hat{M}_{\boldsymbol{v}^4}$) vanishes, and
2. the second derivative of $\hat{M}$ in the $\boldsymbol{v}^4$ direction ($\hat{M}_{\boldsymbol{v}^4\boldsymbol{v}^4}$) is less than 0.

See Section 2.3 for a the mathematical implementation of these conditions.

Any point $(\bar{x}; \bar{s})$ that satisfies these two conditions and for which $M|_{\mathcal{S}}(\bar{x}; \bar{s})$ is maximal is an optimal scale-orientation ridge point of M. The collection of all such points forms the optimal scale-orientation ridge.

Subdimensional maxima Given the definition of a generalized height ridge, there is no guarantee that a point satisfying the derivative conditions in some subset of transverse directions will be a local maximum of the function restricted to the space spanned by those transverse directions. This concern was detailed in [7]. However, we have proved that for optimal parameter ridges, each ridge point is a maximum of the functions M restricted to the space spanned by the directions transverse to the ridge.

2.2 Marching ridges

Both marching cubes and marching lines share the common characteristic of finding implicitly defined manifolds by finding intersections of the manifold with line segments of the space. Marching cubes does this with a characteristic functions and line segments whose endpoints are sample points of the original image. Marching lines does this with zero-trapping along line segments whose endpoints are image sample points and then zero-trapping along line segments whose endpoints are the previously defined zeroes. Marching ridges incorporates both these strategies for finding both curve and surface ridges in 3D images. However, the ridge definition requires more than just that we identify zero-crossings of first derivatives. It also requires that second derivatives are less than zero. This condition produces boundaries of the ridge, a situation not encountered in marching cubes or marching lines, in which the surfaces and curves, respectively, are closed.

Marching ridges is a general purpose algorithm for finding height ridges; we have used it to calculate skeletons, edges, and cores using a variety of ridge definitions. This paper uses a specific implementation of the algorithm for finding optimal parameter ridges of medialness, which I will hereafter refer to as marching cores.

The marching cores algorithm consists of the following steps:

1. **Initialization** Using a mouse to specify a spatial location in a target image and sliders to specify distinguished parameters, the user identifies a starting point in $\mathcal{X} \times \mathcal{S}$ for the marching.
2. **Search** Given the starting spatial point, the algorithm constructs a cube containing the initial point and the seven points whose coordinate values are 1 greater than the initial point for some subset of coordinates. This cube serves as the structure for the rest of the algorithm. E.g., starting at image coordinate (34,123,12) produces the cube with vertices at (34,123,12), (35,123,12), (35,124,12), (34,124,12), (34,124,13), (35,124,13), (35,123,13) and (34,123,13).
3. **Maximization** Each vertex of the cube maximizes the medialness M with respect to the distinguished parameters, producing values for each spatial point.
4. **First derivative calculation** As described in Section 2.1, each vertex of the cube calculates first derivatives of $\hat{M}$. This is done using weighting functions as described in Section 2.3.
5. **Trap zeroes** If the value of $\hat{M}_{\boldsymbol{v}^4}$ at any two vertices joined by an edge of the cube differ in sign, we use a linear approximation of $\hat{M}_{\boldsymbol{v}^4}$ to interpolate the location along the edge where $\hat{M}_{\boldsymbol{v}^4} = 0$.
6. **Second derivative calculation** Each such zero-crossing of $\hat{M}_{\boldsymbol{v}^4}$ then performs an optimization of distinguished parameters and a subsequent calculation of second derivatives of $\hat{M}$. If $\hat{M}_{\boldsymbol{v}^4\boldsymbol{v}^4} < 0$, then the point is a ridge point and is labeled appropriately in the image.
7. **Expansion** Each face of the original cube that contains any ridge points among its edges identifies an adjacent cube to be searched for ridge points. Each such cube is entered into a list.
8. **Marching** The algorithm explores each cube in the list in a breadth-first pattern in a manner similar to the initial cube with the exception that initial values for the optimization of distinguished parameters is the average of the optimal values already determined among its vertices.

The marching cores algorithm continues this loop until there are no more cubes to process. This occurs as the ridges ends, closes or exits the image.

2.3 Oriented medialness

Pizer [12, 13] describes a variety of options for producing medialness in 3D images. The medialness used to produce the examples in this paper is the semi-linear, central medialness described in [12]. It is designed to reduce the effects of interfigural interference on the calculation of cores, as well as to give a more accurate width estimation of objects. More importantly, the orientation component of the medialness kernel can be used in an optimal parameter ridge to determine the final transverse direction as described in Section 2.1.

The underlying medialness is thus $M(\bar{x};\boldsymbol{n}) = L_{\boldsymbol{nn}}(\bar{x};\|\boldsymbol{n}\|)$, where $L(\bar{x};r) = I(\bar{x}) * G(\bar{x};r)$, the original image convolved with a Gaussian of standard deviation r. The algorithm does not, however, calculate medialness, since only derivatives of medialness are required for ridge identification. Since the medialness is a function of image derivatives, the derivatives of medialness are also derivatives of the original image intensity. First derivatives of medialness require third derivatives of image intensity, while second derivatives of medialness require fourth derivatives of image intensity.

Given that $\hat{M}(\bar{x}) = M(\bar{x}, \bar{i}(\bar{x}))$ and that M can be defined in terms of derivatives of the original image, the derivatives necessary for ridge calculation are as follows:

$$\begin{aligned}
\hat{M}_{\boldsymbol{n}} &= M_{\bar{i}(\boldsymbol{n})} \\
&= L_{\boldsymbol{nnn}} \\
\hat{M}_{\boldsymbol{nn}} &= M_{\bar{i}(\boldsymbol{n})\bar{i}(\boldsymbol{n})} - (\nabla|_{\mathcal{S}}(M_{\bar{i}(\boldsymbol{n})}))^t \ (\mathcal{H}^{-1}|_{\mathcal{S}}(M)) \ (\nabla|_{\mathcal{S}}(M_{\bar{i}(\boldsymbol{n})})) \\
&= L_{\boldsymbol{nnnn}} - (\nabla|_{\mathcal{S}}(L_{\boldsymbol{nnn}}))^t \ (\mathcal{H}^{-1}|_{\mathcal{S}}(L_{\boldsymbol{nn}})) \ (\nabla|_{\mathcal{S}}(L_{\boldsymbol{nnn}}))
\end{aligned}$$

Where $\bar{i}$ is the coordinate chart described in Section 2.1. Each of these derivatives is implemented as a spherical weighting function applied to the original image intensities.

3 Results

Figure 1 shows the result of using the marching ridges algorithm on a human brain ventricle in an MR image. Only four slices of the head are shown; the core extends further in the ventricle on higher and lower slices.

4 Discussion

4.1 Skeletons and object edges by marching ridges

The marching ridges algorithm has been used to produce intensity ridges (skeletons) of objects in 2D images. It has also been used in 3D images to find curvilinear skeletons of tube-like objects and surface skeletons for more general objects.

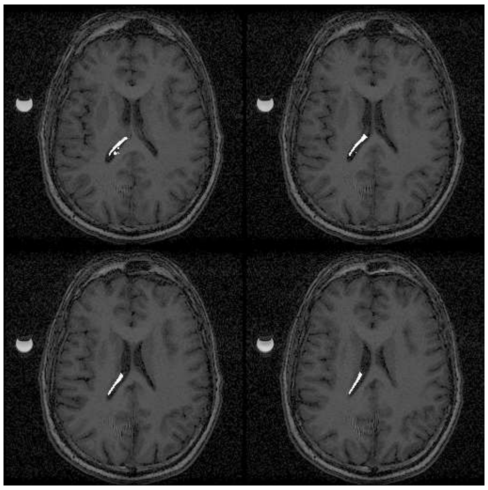

Fig. 1. Optimal scale-orientation core of brain ventricle in 4 adjacent MR image slices.

The marching ridges algorithm has also been used to produce edges of objects in both 2D and 3D images using the Canny definition.

4.2 1D cores in 3D images

This paper presents a method for calculating cores of general objects in 3D images, objects whose cores will generally be surface manifolds. There are, however, structures in the human body which are better described by curvilinear cores, e.g., blood vessels and bronchial tubes. We have modified the marching cores algorithm to calculate cores for these structures; space limitations prevent the inclusion of our results here.

4.3 Medialness with two orientations

The medialness we are currently using is a weighting function of spatial position, radius, and a single orientation. We have begun using Blum-like medialness measures that are functions of spatial position, radius, and two orientations.

5 Acknowledgements

The work presented here was accomplished partially under the support of NIH grants P01 CA47982, R01 CA67812 and R01 LM05508. The UNC Department of Computer Science also provided important support. We gratefully acknowledge the ideas of James Damon, Jason Miller and Rob Keller which have contributed to the mathematical aspects of this work.

References

1. J. Bloomenthal. Polygonization of implicit surfaces. *Computer Aided Geometric Design*, pages 341–355, 1988.
2. J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, November 1986.
3. D. Eberly. *Ridges in Image and Data Analysis.* Computational Imaging and Vision Series. Kluwer Academic Publishers, Dordrecht, NL, 1996.
4. D. Eberly and S. Pizer. Ridge flow models for image segmentation. In *SPIE Proceedings on Mathematical Methods in Medical Imaging II*, pages 54–64, 1994.
5. M. Fidrich. Iso-sruface extraction in 4d with applications related to scale space. Technical Report 2833, Institut National de Recherche en Informatique et en Automatique, March 1996.
6. J. Furst, S. Pizer, and D. Eberly. Marching cores: a mehtod for extracting cores from 3d medical images. In *Proceedings of the workshop on Mathematical Methods in Biomedical Image Analysis*, pages 124–130. IEEE Computer Society Technical Committee on Pattern Analysis and Machine Intelligence, June 1996.
7. Jacob D. Furst, Robert S. Keller, Jason E. Miller, and Stephen M. Pizer. Image loci are ridges in geometric spaces. In Bart ter Haar Romeny, Luc Florack, Jan Koenderink, and Max Viergever, editors, *Scale Space Theory in Computer Vision*, Lecture Notes in Computer Science, pages 176–187. Springer-Verlag, 1997.
8. R. M. Haralick. Ridges and valleys on digital images. *Computer Vision, Graphics, and Image Processing*, 22:28–38, 1983.
9. T. Lindeberg. Edge detection and ridge detection with automatic scale selection. Technical Report ISRN KTH/NA/P-96/06-SE, KTH (Royal Institute of Technology), 1996.
10. W. E. Lorensen and H. E. Cline. Marching cubes: a high resolution 3d surface construction algorithm. *Computer Graphics*, 21(4):163–169, July 1987.
11. B. S. Morse, S. M. Pizer, and A. Liu. Multiscale medial analysis of medical images. *Proceedings on Information Processing in Medical Images*, 687:112–131, 1993.
12. S. M. Pizer, D. Eberly, B. S. Morse, and D. S. Fritsch. Zoom-invariant vision of figural shape: The mathematics of cores. *Computer Vision and Image Understanding*, 1998. to appear.
13. Stephen M. Pizer, Daniel S. Fritsch, Kah-Chan Low, and Jacob D. Furst. 2d & 3d figural models of anatomic objects from medical images. In *Proceedings of ISMM98*, 1988. to appear.
14. J. Thirion and A. Gourdon. The marching lines algorithm: new results and proofs. Technical Report 1881, Institut National de Recherche en Informatique et en Automatique, April 1993.