ABSTRACT

A Beefy Frangi Filter for Noisy Vascular Segmentation and Network Connection in PCSVN

By

Lucas Wukmer

May 2018

Recent statistical analysis of placental features has suggested the usefulness of studying key features of the placental chorionic surface vascular network (PCSVN) as a measure of overall neonatal health. A recent study has suggested that reliable reporting of these features may be useful in identifying risks of certain neurodevelopmental disorders at birth. The necessary features can be extracted from an accurate tracing of the surface vascular network, but such tracings must still be done manually, with significant user intervention. Automating this procedure would not only allow more data acquisition to study the potential effects of placental health on later conditions, but may ideally serve as a real-time diagnostic for neonatal risk factors as well.

Much work has been to develop reliable vascular extraction methods for well-known image domains (such as retinal MRA images) using Hessian-based filters, namely the (multiscale) Frangi filter. It is desirable to extend these arguments to placental images, but this approach is greatly hindered by the inherent irregularity of the placental surface as a whole, which introduces significant noise into the image domain. A recent attempt was made to apply an additional local curvilinear filter to the Frangi result in an effort to remove some noise from the final extraction.

Here we propose an alternate extraction method. First, we use arguments from Frangis original paper to provide a proper selection of parameters for our particular image domain. Using the same arguments from differential geometry that gave rise to the Frangi filter, we calculate the leading principal direction (eigenvector of the Hessian) to indicate the directionality of curvilinear

features at a particular scale. We are then able to apply an appropriately-oriented morphological filter to our Frangi targets at select scales to remove noise. This approach differs significantly from previous efforts in that morphological filtering will take place at each scale space, rather than being performed one time following multiscale synthesis. Noise removal performed in this way is expected to aide in coherent interpretation of targets that should appear in a connected network.

Finally, we discuss an important advancement in implementation–scale space conversion for differentiation (i.e. gaussian blur) via Fast Fourier Transform (FFT) rather than a more traditional convolution with a gaussian kernel, which offers a significant speedup. This thesis will also contain a general, in depth summary of both multiscale Hessian filters and scale-space theory.

We demonstrate the effectiveness of our improved vascular extraction technique on several of the following image domains: a private database of barium-injected samples provided by University of Rochester, uninjected/raw placental samples from Placental Analytics LLC, a collection of simulated images, the DRIVE and STARE databases of retinal MRAs, and a new collection of computer-generated images with significant curvilinear content.

Time permitting, this research will be extended to include a method of network connection, so that a logically connected vascular network is realized (i.e. network completion).

A Beefy Frangi Filter for Noisy Vascular Segmentation and Network Connection in PCSVN

A THESIS

Presented to the Department of Department of Mathematics and Statistics

California State University, Long Beach

In Partial Fulfillment

of the Requirements for the Degree

Master of Science in Applied Mathematics

Committee Members:

Jen-Mei Chang, Ph.D. (Chair)
James von Brecht, Ph.D.
William Ziemer, Ph.D.

College Designee:

Tangan Gao, Ph.D.

By Lucas Wukmer

B.S., 2013, University of California, Los Angeles

May 2018

WE, THE UNDERSIGNED MEMBERS OF THE COMMITTEE,

HAVE APPROVED THIS THESIS

A Beefy Frangi Filter for Noisy Vascular Segmentation and Network Connection in PCSVN

By

Lucas Wukmer

COMMITTEE MEMBERS

Jen-Mei Chang, Ph.D. (Chair)	Mathematics and Statistics
James von Brecht, Ph.D.	Mathematics and Statistics
William Ziemer, Ph.D.	Mathematics and Statistics

ACCEPTED AND APPROVED ON BEHALF OF THE UNIVERSITY

Tangan Gao, Ph.D.
Department Chair, Mathematics and Statistics

California State University, Long Beach

May 2018

ACKNOWLEDGEMENTS

Acknowledgments go here.

TABLE OF CONTENTS

Page

APPENDIX Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

The Applied Problem

Reference Nen's paper and latest autism risk paper.

Context in Image Processing

- brief background of math image processing methods
- what's been tried in this applied problem
  - nen [1]
  - catalina's paper
  - kara's paper
  - other domains

Research Goals

Segue from previous paragraph, talk about strengths and weaknesses of other methods and what this research aims to accomplish. Include 'research questions' that could allow a reader to answer the question ”will this research work for my problem?”. ”Elevator pitch” maybe goes here.

Roadmap

Outline of the thesis (”firstly” bullshit)

CHAPTER 2

MATHEMATICAL METHODS

Overview of Differential Geometry in Image Processing

Basics, Definitions

Definition 2.1.1. *For theoretical purposes, we may view any 2D grayscale image as a continuous function* $L : \mathbb{R}^2 \to \mathbb{R}$ *with* $L \in \mathcal{C}^2\left(\mathbb{R}^2\right)$.

Definition 2.1.2. *In the context of differential geometry, we wish to refer to its graph* $f : \mathbb{R}^2 \to \mathbb{R}^3$ *by* $(u,v) \mapsto (u,v,L(u,v))$.

Definition 2.1.3. *In situations where we wish to discuss a discrete image, we may refer to* $L_0 \in \mathbb{R}^{m\times n}$. *That is,* L_0 *is a matrix corresponding to the m-by-n digital grayscale image.*

Viewing the surface in R^3, we define the Hessian $\mathcal{H}$ of the surface L at a point (x,y) on the surface as the matrix of its second partial derivatives:

$$\mathcal{H}(x,y) = \begin{bmatrix} L_{xx}(x,y) & L_{xy}(x,y) \\ L_{yx}(x,y) & L_{yy}(x,y) \end{bmatrix} \tag{2.1}$$

At any point (x,y) we denote the two eigenpairs of $\mathcal{H}(x,y)$ as

$$\mathcal{H}u_i = \kappa_i u_i \ , \quad i = 1,2 \tag{2.2}$$

where κ_i and u_i are known as the *principal curvatures* and *principal directions* of $L(x,y)$, respectively, and we label such that $|\kappa_2| \geq |\kappa_1|$. Notably, $\mathcal{H}(x,y)$ is a real, symmetric matrix (since $L_{xy} = L_{yx}$ and L is a real function)and thus its eigenvalues are real and its eigenvectors are orthonormal to each other, as given by following lemma:

Lemma 2.1.1 (Principal Axis Theorem?). *Let A be a real, symmetric matrix. The eigenvalues of A are real and its eigenvectors are orthonormal to each other.*

Proof. Let $x \neq 0$ so that $Ax = \lambda x$. Then

$$\begin{aligned}
\|Ax\|_2^2 = \langle Ax, Ax \rangle &= (Ax)^* Ax \\
&= x^* A^* Ax = x^* A^T Ax = x * AAx \\
&= x^* A \lambda x = \lambda x^* Ax \\
&= \lambda x^* \lambda x = \lambda^2 x^* x = \lambda^2 \|x\|_2^2
\end{aligned}$$

Upon rearrangement, we have $\lambda^2 = \frac{\|Ax\|_2^2}{\|x\|_2^2} \geq 0 \implies \lambda$ is real.

To prove that a set of orthonormalizable eigenvectors exists, let A be real, symmetric as above and consider the eigenpairs $Av_1 = \lambda_1 v_1, Av_2 = \lambda_2 v_2$ with $v_1, v_2 \neq 0$. [1]

In the case that $\lambda_1 \neq \lambda_2$, we have

$$\begin{aligned}
(\lambda_1 - \lambda_2) v_1^T v_2 &= \lambda_1 v_1^T v_2 - \lambda_2 v_1^T v_2 \\
&= (\lambda_1 v_1)^T v_2 - v_1^T (\lambda_2 v_2) \\
&= (Av_1)^T v_2 - v_1^T (Av_2) \\
&= v_1^T A^T v_2 - v_1^T Av_2 \\
&= v_1^T Av_2 - v_1^T Av_2 = 0
\end{aligned}$$

Since $\lambda_1 \neq \lambda_2$, we conclude that $v_1^T v_2 = 0$.

In the case that $\lambda_1 = \lambda_2 =: \lambda$, we can define (as in Gram-Schmidt orthogonalization)

[1] To simplify notation, we simplify our argument to consider two explicit eigenvectors only, since we're only concerned with the 2×2 matrix $\mathcal{H}$ anyway.

$u = v_2 - \frac{v_1^T v_2}{v_1^T v_1} v_1$. This is an eigenvector for $\lambda = \lambda_2$, as

$$\begin{aligned} Au &= A\left(v_2 - \frac{v_1^T v_2}{v_1^T v_1} v_1\right) \\ &= Av_2 - \frac{v_1^T v_2}{v_1^T v_1} Av_1 \\ &= \lambda v_2 - \frac{v_1^T v_2}{v_1^T v_1} \lambda v_1 \\ &= \lambda\left(v_2 - \frac{v_1^T v_2}{v_1^T v_1} v_1\right) = \lambda u \end{aligned}$$

and is perpendicular to v_1, since

$$\begin{aligned} v_1^T u &= v_1^T \left(v_2 - \frac{v_1^T v_2}{v_1^T v_1} v_1\right) \\ &= v_1^T v_2 - \left(\frac{v_1^T v_2}{v_1^T v_1}\right) v_1^T v_1 \\ &= v_1^T v_2 - v_1^T v_2 (1) = 0. \end{aligned}$$

□

Thus we see that the two principal directions form an orthonormal frame at each point (x,y) within the continuous image $L(x,y)$.

✠The following is an **unverified claim** (which might be useful for later): The frame varies continuously along paths in $\mathbb{R}^2$ except at points where $\mathcal{H}(x,y)$ is singular. To make this explicit:

Theorem 2.1.2 (Continuity of the leading principal direction)**.** *Let $\theta : I := [0,1] \to \mathbb{R}^2$ be a parametrized regular curve in $\mathbb{R}^2$ and $H_\theta := \mathcal{H}_f \circ \theta(t)$ be the matrix-valued function (where $\mathcal{H}_f$ is the 2×2 Hessian of the smooth surface f) Let $U : I \to \mathbb{R}^2$ be the implicitly-defined vector valued function s.t. $U(t)$ is the leading eigenvector of H_θ (and therefore the leading principal direction of f). That is,*

$$H_\theta\, U(t) = \lambda U(t) \quad \textit{with} \quad \lambda = \rho(H_\theta) \tag{2.3}$$

In other words, $|\lambda| \geq |\tilde{\lambda}|$ for any $\tilde{\lambda} : H_\theta\, u = \tilde{\lambda} u$ for some $u \neq 0$.

Then, $U(t)$ is continuous in t whenever $H_f(t)$ is non-singular. Note: Maybe fix this so that the path avoids any nonsingular points? $U(t)$ isn't even well-defined at such points anyway.

Proof. First, we show that $U(t)$ is a well-defined function at all points t where $H_f(t)$ is non-singular. □

✠TODO

Eigenvectors of the Hessian are principal curvature directions

Formally, within the context of differential geometry, we should show a few things:

- definition of curvature of a surface (how it relates to curvature of a regular curve) and showing that it's invariant of which curve is taken–that is, it only depends on a tangent vector to the surface.
- normal curvature is given by the second fundamental form $\mathbf{II}(X,X) = \langle \mathsf{L}X, X\rangle$ where L is the Weingarten map.
- X is an extremal value of $\mathbf{II}(X,X)$ subject to $\mathbf{I}(X,X) = 1$ iff X is an eigenvector of L.
- In the case of a surface $f(u,v) = (u,v,h(u,v))$ (i.e. graph of a 2D function, which is the only situation we care about in image processing), the Weingarten map is exactly the Hessian matrix.

Defining normal curvature of a surface

In the context of a regular curve $c : I \to \mathbb{R}^3$ parametrized along some closed interval $I \in \mathbb{R}$ (that is, one where $c'(s) = 1 \ \ \forall s \in I$), curvature at a point $s \in I$ is defined simply as the magnitude of the curve's acceleration: $\kappa(s) := \|c''(s)\|$. To extend the notion of curvature of a surface f, we can consider at some point $u = (u_1, u_2)$ on the surface and consider a parametrized curve with parametrization $\theta : I \to U \subset \mathbb{R}^2$ so that we may speak of $c(u) = c \circ \theta(t)$ freely as the curve c at point $u \in U$ and simultaneously refer to it as $c(t)$ for $t \in I$. Of course, image$(c) \subset$ image(f) exactly–that is, the curve is on the surface.

We now present a main result that provides a notion of curvature of a surface.

Theorem 2.2.1 (Theorem of Meusnier)**.** *Given a point $u \in U$ and a tangent direction $X \in T_u\mathbb{R}^3$ any curve on the surface f such that $c'(u) = X$ will have the same curvature.*

Proof. Considering any such curve where $\frac{\partial c}{\partial t}(u = \theta(t)) = X$ where $X \in T_u\mathbb{R}^3$ is a normalized vector tangent to the surface at the point u, we wish to decompose its acceleration along the two orthogonal vectors X and the Gauss map $\nu := \nu(u_1, u_2) = \frac{\frac{\partial f}{\partial u_1} \times \frac{\partial f}{\partial u_2}}{\|\frac{\partial f}{\partial u_1} \times \frac{\partial f}{\partial u_2}\|}$. (Note that X and ν are indeed orthogonal, as both $\frac{\partial f}{\partial u_i} \in T_u\mathbb{R}^3,\ i = 1, 2$.) We then have

$$c'' = \langle c'', X\rangle X + \langle c'', \nu\rangle \nu \tag{2.4}$$

.

The first term is zero, since for a regular curve,

$$\langle c'', X\rangle = \langle c'', c'\rangle = 0$$

Using chain rule, we can rewrite the second coefficient of (2.4) as

$$\begin{aligned} \langle c'', \nu\rangle &= \frac{\partial}{\partial t}\left[\langle c', \nu\rangle\right] - \langle c', \frac{\partial \nu}{\partial t}\rangle &(2.5)\\ &= \frac{\partial}{\partial t}\left[\langle X, \nu\rangle\right] - \langle c', \frac{\partial \nu}{\partial t}\rangle &(2.6)\\ &= 0 - \langle X, \frac{\partial \nu}{\partial t}\rangle &(2.7) \end{aligned}$$

Thus, we can express the curvature at this point on our selected curve as

$$\|c''\| = -\langle X, \frac{\partial \nu}{\partial t}\rangle \|\nu\| = -\langle X, \frac{\partial \nu}{\partial t}\rangle$$

which only depends on the point u and the selected direction X. □

In fact, we refer to this invariant quantity as the normal curvature of the surface.

Definition 2.2.1. *The normal curvature of a surface at point u in the direction X is given by* $\kappa_\nu := -\langle X, \frac{\partial \nu}{\partial t}\rangle$.

In other contexts, this quantity is referred to as the second fundamental form at the point $u \in U$; that is, $\mathbf{II}(X,X) := -\langle X, \frac{\partial \nu}{\partial t}\rangle$.

Our final goal to is to characterize such normal curvatures and establish, at a given point $u \in U$, a method of determining in which directions an extremal normal curvature occurs. To do so, we shall consider the relationship between the direction X and the normal curvature κ_ν in that direction at some specified u. That is, we trace $X \in T_u f$ (which may be expanded in the orthogonal basis $\left\{\frac{\partial f}{\partial u_1}, \frac{\partial f}{\partial u_2}\right\}$) back to its associated surface element $f(u)$ and then expand it along the basis $\left\{\frac{\partial \nu}{\partial u_1}, \frac{\partial \nu}{\partial u_2}\right\}$.

Formally, we can write this map as $\mathcal{L} = -d\nu \circ (df)^{-1}$.

That is, $\mathsf{L}\frac{\partial f}{\partial u_i} = -\frac{\partial \nu}{\partial u_i}$. where we refer to L we can write a matrix representation of the map $\mathcal{L}$ as the change of basis matrix

$$\mathsf{L} := \begin{pmatrix} \langle \frac{\partial f}{\partial u_1}, -\frac{\partial \nu}{\partial u_1}\rangle & \langle \frac{\partial f}{\partial u_1}, -\frac{\partial \nu}{\partial u_2}\rangle \\ \langle \frac{\partial f}{\partial u_2}, -\frac{\partial \nu}{\partial u_1}\rangle & \langle \frac{\partial f}{\partial u_2}, -\frac{\partial \nu}{\partial u_2}\rangle \end{pmatrix} \tag{2.8}$$

First, we need the following lemma:

Lemma 2.2.2. *If $A \in R^{n\times n}$ is a symmetric real matrix, $v \in R^n$ and given the dot product $\langle \cdot, \cdot \rangle$, we have $\nabla_v \langle v, Av\rangle = 2Av$. In particular, when $A = I$ the identity matrix, we have $\nabla_v \langle v, v\rangle = 2v$.*

Proof. The result is uninterestingly obtained by tracking each (the 'ith') component of $\nabla_v \langle v, Av\rangle$:

$$
\begin{aligned}
\Big(\nabla_v \langle v, Av\rangle\Big)_i = \frac{\partial}{\partial v_i}\Big[\langle v, Av\rangle\Big] &= \frac{\partial}{\partial v_i}\left[\sum_{j=1}^{n} v_j (Av)_j\right] && (2.9)\\
&= \frac{\partial}{\partial v_i}\left[\sum_{j=1}^{n} v_j \sum_{k=1}^{n} a_{jk} v_k\right] && (2.10)\\
&= \frac{\partial}{\partial v_i}\left[a_{ii}v_i^2 + v_i \sum_{k\neq i} a_{ik}v_k + v_i \sum_{j\neq i} a_{ji}v_j + \sum_{j\neq i}\sum_{k\neq i} v_j a_{jk} v_k\right] && (2.11)\\
&= 2a_{ii}v_i + \sum_{k\neq i} a_{ik}v_k + \sum_{j\neq i} a_{ji}v_j + 0 && (2.12)\\
&= 2a_{ii}v_i + 2\sum_{k\neq i} a_{ik}v_k = 2\sum_{k=1}^{n} a_{ik}v_k = 2\big(Av\big)_i && (2.13)\\
&\implies \nabla_v \langle v, Av\rangle = 2Av. && (2.14)
\end{aligned}
$$

□

Theorem 2.2.3 (Theorem of Olinde Rodrigues)**.** *Fixing a point $u \in U$, a direction $X \in T_u\mathbb{R}^3$ minimizes the normal curvature $\kappa_\nu = \boldsymbol{II}(X,X)$ subject to $\boldsymbol{I}(X,X) = 1$ iff X is a (normalized) eigenvector of the Weingarten map L.*

Proof. Recall first the definition of the first two fundamental forms, $\mathbf{II}(X,X) = \langle \mathsf{L}X, X\rangle$ and $\mathbf{I}(X,X) = \langle X, X\rangle$.

Using the method of Lagrange multipliers, we define the Lagrangian:

$$
\begin{aligned}
\mathscr{L}(X;\lambda) &= \mathbf{II}(X,X) - \lambda\Big(\mathbf{I}(X,X) - 1\Big)\\
&= \langle \mathsf{L}X, X\rangle - \lambda\Big(\langle X, X\rangle - 1\Big)
\end{aligned}
\tag{2.15}
$$

Extremal values occur when $\nabla_{X,\lambda}\mathscr{L}(X;\lambda) = 0$, which becomes the two equations

$$
\begin{cases}
\nabla_X \langle \mathsf{L}X, X\rangle - \lambda \nabla_X \left(\langle X, X\rangle - 1\right) = 0\\
\langle X, X\rangle - 1 = 0
\end{cases}
\tag{2.16}
$$

The second requirement is simply the constraint that X is normalized. Using the previous lemma, we can simplify the first result as follows:

$$\nabla_X \langle \mathsf{L}X, X\rangle - \lambda \nabla_X (\langle X, X\rangle - 1) = 0 \tag{2.17}$$

$$2\mathsf{L}X - \lambda(2X) = 0 \tag{2.18}$$

$$\implies LX - \lambda X = 0 \implies LX = \lambda X. \tag{2.19}$$

Thus the two hypotheses are exactly equivalent when X is normalized. It is also worth remarking that the corresponding eigenvalue λ is the Lagrangian multiplier itself. $\square$

Our final goal is to determine that in the case of $f(u,v) = (u,v,L(u,v))$, the Weingarten map itself is defined as

Definition 2.2.2 (The Weingarten map)**.** *We define the Weingarten map $\mathcal{L}$ at point $u \in U$ as mapping tangent vectors (in the basis of $\left\{\frac{\partial f}{\partial u_i}\right\}$) to the (negative) derivatives of normal vectors (in the basis $\left\{-\frac{\partial \nu}{\partial u_i}\right\}$).*

We are now ready to show that in the case of a graph surface, this matrix is exactly the Hessian matrix.

Theorem 2.2.4. *When $f : U \to \mathbb{R}^3$ is given by $(u,v) \mapsto (u,v,L(u,v))$, the matrix representation of the Weingarten map is exactly the Hessian matrix given in (2.1).*

Proof. First, we can (using chain rule) rewrite each component as

$$\langle \frac{\partial f}{\partial u_i}, -\frac{\partial \nu}{\partial u_j}\rangle = \langle \frac{\partial^2 f}{\partial u_i \partial u_j}, \nu\rangle$$

Now, given our particular surface f, we can calculate each of these components directly. We have:

$$\begin{gathered} f_u = (1,0,L_u), \quad f_v = (0,1,L_v) \\ f_{uu} = (0,0,L_{uu}), \quad f_{uv} = (0,0,L_{uv}) = f_{vu}, \quad f_{vv} = (0,0,L_{vv}) \end{gathered} \tag{2.20}$$

and we have the unit normal vector (Gauss map)

$$\nu(u_1,u_2) = \frac{\frac{\partial f}{\partial u_1} \times \frac{\partial f}{\partial u_2}}{\|\frac{\partial f}{\partial u_1} \times \frac{\partial f}{\partial u_2}\|} \tag{2.21}$$

$$= \frac{(1,0,L_u) \times (0,1,L_v)}{\|\cdots\|} \tag{2.22}$$

□

Calculating Derivatives of Discrete Images

Discrete derivatives. ✠**TODO:** Develop the following:

- Take derivative with gradient / divided difference.
- Derivatives should be taken on Gaussian blur (equivalent to scale space development) [Lindeberg]. That is, you can take the derivative of either the convolved image, or you can take derivatives of the Gaussian itself, *then* convolve.
- Pseudocode for `np.gradient` which is used in calculating Hessian (code below)

```
gaussian_filtered = fftgauss(image, sigma=sigma)
Lx, Ly = np.gradient(gaussian_filtered)
Lxx, Lxy = np.gradient(Lx)
Lxy, Lyy = np.gradient(Ly)
```

The Frangi Filter

Intro to Hessian-based filters

Hessian-based filters are a family of curvilinear filters that employ the Hessian and its eigenspaces to determine regions of significant curvature within an image. Several such filters exist –see Sato [13] and Lorenz[10]. These filters use information about the principal curvatures (eigenvalues of the Hessian) at each point to

Overview of Frangi vesselness measure

The Frangi filter is a widely used [citation needed] Hessian-based filter that relies on the principal curvatures–that is, the eigenvalues of $\mathcal{H}_\sigma(x,y)$ at some particular scale σ at each point (x,y) in the image.

Implementation Details: Convolution Speedup via FFT

As described above, the actual computation of derivatives is achieved via convolution with a gaussian. In practice, this is very slow for large scales. In **TODO** .

- find image processing papers that find hessian from FFT / who uses this?
- with above: downsides?
- SIDE BY SIDE comparison?

2D Discrete Fourier Transform Convolution Theorem . [2]

Theorem 2.4.1 (2D DFT Convolution Theorem)**.** *Given two discrete functions are sequences with the same length*[3]*, that is:* $f(x,y)$ *and* $h(x,y)$ *for integers* $0 < x < M$ *and* $0 < y < N$*, we can take the discrete fourier transform (DFT) of each:*

$$F(u,v) := \mathcal{D}\{f(x,y)\} = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)e^{-2\pi i\left(\frac{ux}{M}+\frac{vy}{N}\right)} \tag{2.23}$$

$$H(u,v) := \mathcal{D}\{h(x,y)\} = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} h(x,y)e^{-2\pi i\left(\frac{ux}{M}+\frac{vy}{N}\right)} \tag{2.24}$$

and given the convolution of the two functions

$$(f \star h)(x,y) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f(m,n)h(x-m,y-n) \tag{2.25}$$

then $(f \star h)(x,y)$ *and* $MN \cdot F(u,v)H(u,v)$ *are transform pairs, i.e.*

$$(f \star h)(x,y) = \mathcal{D}^{-1}\{MN \cdot F(u,v)H(u,v)\} \tag{2.26}$$

[2] the following was adapted in a large part from DFT: an owner's manual. cite?

[3] If they're not actually the same length, DIP-GW suggests to make the final length at least $P = A + C - 1$ and $Q = B + D - 1$ in the case that the sizes are $A \times B$ and $C \times D$ for $f(x,y)$ and $h(x,y)$ respectively. Not sure if that matters.

The proof follows from the definition of convolution, substituting in the inverse-DFT of f and h, and then rearrangement of finite sums.

Proof.

$$(f \star h)(x,y) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f(m,n)h(x-m,y-n) \tag{2.27}$$

$$= \sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\left(\sum_{p=0}^{M-1}\sum_{q=0}^{N-1} F(p,q)e^{2\pi i\left(\frac{mp}{M}+\frac{nq}{N}\right)}\right)\left(\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} H(u,v)e^{2\pi i\left(\frac{u(x-m)}{M}+\frac{v(y-n)}{N}\right)}\right) \tag{2.28}$$

$$= \left(\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} H(u,v)e^{2\pi i\left(\frac{ux}{M}+\frac{vy}{N}\right)}\right)\left(\sum_{p=0}^{M-1}\sum_{q=0}^{N-1} F(p,q)\left(\sum_{m=0}^{M-1} e^{2\pi i\left(\frac{m(p-u)}{M}\right)}\right)\left(\sum_{n=0}^{N-1} e^{2\pi i\left(\frac{n(q-v)}{N}\right)}\right)\right) \tag{2.29}$$

$$= \left(\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} H(u,v)e^{2\pi i\left(\frac{ux}{M}+\frac{vy}{N}\right)}\right)\left(\sum_{p=0}^{M-1}\sum_{q=0}^{N-1} F(p,q)\left(M\cdot\hat{\delta}_M(p-u)\right)\left(N\cdot\hat{\delta}_M(q-v)\right)\right) \tag{2.30}$$

$$= \left(\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} H(u,v)e^{2\pi i\left(\frac{ux}{M}+\frac{vy}{N}\right)}\right)\cdot MNF(u,v) \tag{2.31}$$

$$= MN\cdot\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F(u,v)H(u,v)e^{2\pi i\left(\frac{ux}{M}+\frac{vy}{N}\right)} \tag{2.32}$$

$$= MN\cdot\mathcal{D}^{-1}\{FH\} \tag{2.33}$$

where

$$\hat{\delta}_N(k) = \begin{cases} 1 & \text{when } k=0 \mod N \\ 0 & \text{else} \end{cases} \tag{2.34}$$

$\square$

Above, we make use of the following lemma:

Lemma 2.4.2. *Let j and k be integers and let N be a positive integer. Then*

$$\sum_{n=0}^{N-1} e^{2\pi i\left(\frac{n(j-k)}{N}\right)} = N\cdot\hat{\delta}_N(j-k) \tag{2.35}$$

Proof. For any particular $n \in 0..N-1$,consider the complex number $e^{2\pi in(j-k)/N}$. Note first, that this is an N-th root of unity, since

$$\left(e^{2\pi in(j-k)/N}\right)^N = e^{2\pi in(j-k)} = \left(e^{2\pi i}\right)^{n(j-k)} = 1^{n(j-k)} = 1$$

Thus, we understand that the complex number $e^{2\pi in(j-k)/N}$ is a root of $z^N - 1 = 0$, which we rewrite as

$$z^N - 1 \ = \ (z-1)(z^{n-1} + \cdots + z + 1) \ = \ (z-1)\sum_{n=0}^{N-1} z^n.$$

We consider two cases: in the case that $j-k$ is a multiple of N, we of course have $e^{2\pi in(j-k)/N} = 1$ for any n, and thus

$$\sum_{n=0}^{N-1} e^{2\pi i\left(\frac{n(j-k)}{N}\right)} = \sum_{n=0}^{N-1} (1) = N$$

.

In the case that $j-k$ is *not* a multiple of N, then clearly $\left(e^{2\pi in(j-k)/N}\right) - 1 \neq 0$, and thus it must be that

$$\left(e^{2\pi in(j-k)/N}\right)^N - 1 = 0 \implies \sum_{n=0}^{N-1} \left(e^{2\pi in(j-k)/N}\right)^n = 0$$

Combining these two cases gives the result of the lemma. □

FFT .

As noted, the above result applies to the Discrete Fourier Transform. As noted, we actually achieve a convolution speedup using a Fast Fourier Transform (FFT) instead.

- Basic theory.
- Show speedup and equivalency.

Odds & Ends of Fourier Analysis .

- Sampling theory?
- Wraparound error?

- Any other kinks introduced in going from continuous to discrete.

Linear Scale Space Theory

Koenderink showed/asserted that ”any image can be embedded in a one-parameter family of derived images (with resolution as the parameter) in essentially only one unique way” given a few of the so-called *scale space axioms*. They showed in particular that any such family must satisfy the heat equation

$$\Delta K(x,y,\sigma) = K_\sigma(x,y,\sigma) \text{ for } \sigma \geq 0 \text{ such that } K(x,y,0) = u_0(x,y). \tag{2.36}$$

where $K : \mathbb{R}^3 \to \mathbb{R}$ and $u_0 : \mathbb{R}^2 \to \mathbb{R}$: is the original image (viewed as a continuous surface) and σ is a resolution parameter. Much work blah blah blah has been done to formalize this approach.

Significance of the convolution operation

This section shows that if one constructs the family using an operation and a source image, the necessary operation is necessarily equivalent to convolution. I am guessing at this, but I think the axiom that causes this is wanting no space in the image to be preferred.

Uniqueness of the Gaussian Kernel

This is the result of most of these axioms.

$G_\sigma \star u_0$ **solves the heat equation**

given u_0 as a continuous image (unscaled), we construct PDE with this as a boundary condition.

$$u : \mathbb{R}^2 \supset \Omega \to \mathbb{R} \text{ with } u(\boldsymbol{x},t) : \begin{cases} \frac{\partial u}{\partial t}(\boldsymbol{x},t) = \Delta u(\boldsymbol{x},t) & , t \geq 0 \\ u(\boldsymbol{x},0) = u_0(\boldsymbol{x}) \end{cases} \tag{2.37}$$

We show that

$$u(\boldsymbol{x},t) = \left(G_{\sqrt{2t}} \star u_0\right)(\boldsymbol{x}) \tag{2.38}$$

solves (the above tagged equation), where

$$G_\sigma := \frac{1}{2\pi\sigma^2} e^{\left(-|x|^2/(2\sigma^2)\right)}$$

First, we need a quick lemma regarding differentiation a continuous convolution.

Lemma 2.5.1. *Derivative of a convolution is the way that it is (obviously rewrite this).*

Proof. For a single variable,

$$\begin{aligned}
\frac{\partial}{\partial\alpha}\left[f(\alpha)\star g(\alpha)\right] &= \frac{\partial}{\partial\alpha}\left[\int f(t)g(\alpha-t)dt\right] &(2.39)\\
&= \int f(t)\frac{\partial}{\partial\alpha}\left[g(\alpha-t)\right]dt &(2.40)\\
&= \int f(t)\left(\frac{\partial g}{\partial\alpha}\right)g(\alpha-t)dt &(2.41)\\
&= f(\alpha)\star g'(\alpha) &(2.42)
\end{aligned}$$

By symmetry of convolution we can also conclude

$$\frac{\partial}{\partial\alpha}\left[f(\alpha)\star g(\alpha)\right] = f'(\alpha)\star g(\alpha)$$

If f and g are twice differentiable, we can compound this result to show a similar statement holds for second derivatives, and then, given the additivity of convolution, we may conclude

$$\Delta(f\star g) = \Delta(f)\star g = f\star\Delta(g) \tag{2.43}$$

□

Theorem 2.5.2. $u(\boldsymbol{x},t) = \left(G_{\sqrt{2t}}\star u_0\right)(\boldsymbol{x})$ *solves the heat equation.*

Proof. We focus on the particular kernel

$$G_{\sqrt{2t}} = \frac{1}{4\pi t} e^{\left(-|x|^2/(4t)\right)}$$

□

Then

$$\frac{\partial u}{\partial t}(\boldsymbol{x},t) = \frac{\partial}{\partial t}\left(G_{\sqrt{2t}}(\boldsymbol{x},t) \star u_0(\boldsymbol{x})\right) \tag{2.44}$$

$$= \frac{\partial}{\partial t}\left(G_{\sqrt{2t}}(\boldsymbol{x},t)\right) \star u_0(\boldsymbol{x}) \tag{2.45}$$

$$= \frac{\partial}{\partial t}\left(\frac{1}{4\pi t}e^{\left(-|x|^2/(4t)\right)}\right) \star u_0(\boldsymbol{x}) \tag{2.46}$$

$$= \left[-\frac{1}{4\pi t^2}e^{\left(-|x|^2/(4t)\right)} + \frac{1}{4\pi t}\left(\frac{-|x|^2}{4t^2}\right)e^{-|x|^2/(4t)}\right] \star u_0(\boldsymbol{x}) \tag{2.47}$$

$$= -\frac{1}{4t^2}\left(e^{\left(-|x|^2/(4t)\right)} + |\boldsymbol{x}|^2 G_{\sqrt{2t}}(\boldsymbol{x},t)\right) \star u_0(\boldsymbol{x}) \tag{2.48}$$

and from the previous lemma,

$$\Delta u(\boldsymbol{x},t) = \Delta\left(G_{\sqrt{2t}} \star u_0(\boldsymbol{x})\right) = \Delta\left(G_{\sqrt{2t}}\right) \star u_0(\boldsymbol{x})$$

We explicitly calculate the Laplacian of $G_\sigma(x,y) = A\exp(-\frac{x^2+y^2}{2\sigma^2})$ as follows:

$$\begin{aligned}
\frac{\partial}{\partial x}G_\sigma(x,y) &= A\left(\frac{-2x}{2\sigma^2}\right)\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \\
\implies \frac{\partial^2}{\partial^2 x}G_\sigma(x,y) &= A\cdot\frac{\partial}{\partial x}\left[-\frac{x}{\sigma^2}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)\right] \\
&= A\left[-\frac{1}{\sigma^2}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) + \frac{x}{\sigma^2}\cdot\frac{2x}{2\sigma^2}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)\right] \\
&= A\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)\left[-\frac{1}{\sigma^2} + \frac{x^2}{\sigma^4}\right] \\
&= \frac{1}{\sigma^2}G_\sigma(x,y)\left[\frac{x^2}{\sigma^2} - 1\right]
\end{aligned}$$

By symmetry of argument we also may conclude

$$\frac{\partial^2}{\partial y^2}G_\sigma(x,y) = \frac{1}{\sigma^2}G_\sigma(x,y)\left[\frac{y^2}{\sigma^2} - 1\right]$$

and so

$$\Delta G_\sigma(x,y) = \frac{\partial^2}{\partial x^2}(G_\sigma) + \frac{\partial^2}{\partial y^2}(G_\sigma) = \frac{1}{\sigma^2}G_\sigma(x,y)\left[\frac{x^2+y^2}{\sigma^2} - 2\right] \tag{2.49}$$

Morphology

Methods of merging multiscale methods.

Other Odds and Ends

Finding the placental plate. Preprocessing. Could go in next section as well.

CHAPTER 3

RESEARCH PROTOCOL

List. All. Decisions. You Make. Be very explicit.

Pseudocode?

CHAPTER 4

RESULTS AND ANALYSIS

Data Set

Specifics of your data set. Barium placentas, other datasets.

NOTE: FIND YOUR TRACING PROTOCOL.

NOTE: Be specific about each so it's easy to gauge for others whether or not these methods would apply to their own research

Preprocessing

Results

Show stuff who the fuck knows

Answer Research Questions

CHAPTER 5

CONCLUSION

What it did well. What it didn't. People who want to expand–where should they start?

APPENDICES

APPENDIX A

APPENDIX TITLE

Put code here (and on github)

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Nen Huynh. *A filter bank approach to automate vessel extraction with applications*. PhD thesis, California State University, Long Beach, 2013.