The Local Structure of Space-Variant Images

Bruce Fischl=

Dept. Cognitive and Neural Systems

Boston University
677 Beacon Street
Boston, MA 02215
email: fischl@cns.bu.edu

Eric L. Schwartz=*

Dept. Cognitive and Neural Systems

Boston University
677 Beacon Street
Boston, MA 02215
email: eric@thing4.bu.edu
(617) 353-6179

Michael A. Cohen

Dept. Cognitive and Neural Systems

Boston University
677 Beacon Street
Boston, MA 02215
email: mike@cns.bu.edu

* Author to whom reprint requests should be made.

= Supported in part by the office of naval research (ONR N00014-95-1-0409).

The Local Structure of Space-Variant Images

Abstract - *Local image structure is widely used in theories of both machine and biological vision. The form of the differential operators describing this structure for space-invariant images has been well documented (e.g. Koenderink, 1984). Although space-variant coordinates are universally used in mammalian visual systems, the form of the operators in the space-variant coordinate system has received little attention. In this report we derive the form of the most common differential operators and surface characteristics in the space-variant domain and show examples of their use. The operators include the Laplacian, the gradient and the divergence, as well as the fundamental forms of the image treated as a surface. We illustrate the use of these results by deriving the space-variant form of corner detection and image enhancement algorithms. The latter is shown to have interesting properties in the complex log domain, implicitly encoding a variable grid-size integration of the underlying PDE, allowing rapid enhancement of large scale peripheral features while preserving high spatial frequencies in the fovea.*

Keywords: anisotropic diffusion, space-variant vision, log-polar, image enhancement

1. Introduction.

Images can be characterized in a number of ways. For example, one can consider an image as having been generated by a random process. This allows the tools of statistical mechanics and probability theory to be brought to bear on the problems of image analysis (e.g. Geman and Geman, 1984; Cooper et al., 1981). An alternative formulation is to view an image deterministically, and rely on a geometric analysis of its properties. Typically, computational constraints force the analysis to be a local one which leads to the use of differential geometry as an important tool in machine and biological vision. Within the deterministic framework one can further subdivide the types of approaches. Perhaps the most common is to consider the image intensity function to be a discrete sampling of a scalar field over a subset of the real plane. This leads to the use of multivariable calculus and integral transform techniques in image processing. A somewhat different viewpoint is to regard the image as a surface embedded in $\Re^3$, leading to the characterization of surface patches by quantities such as the coefficients of the first fundamental form (Nitzberg and Shiota, 1992), the Gaussian curvature (Deriche and Giraudon, 1993; Barth et al., 1993) and mean curvature (El-Fallah and Ford, 1994). Many other examples of the use of differential image structure by both machine and biological vision researchers can be found in the literature (e.g. Koenderink, 1984; Koenderink and van Doorn, 1990, Koenderink and van Doorn, 1994; Deriche et al., 1992; Deriche and Giraudon, 1993; Florack et al., 1992; Perona and Malik, 1987, Perona and Malik, 1990; Beaudet, 1978, Kitchen and Rosenfeld, 1982; Krueger and Phillips, 1989).

Almost without exception the use of differential image characteristics in machine and biological vision research has been performed in the space-invariant or Cartesian domain. However, it has been shown that the mapping from the mammalian retina to striate cortex is a space-variant one which can be well approximated by a complex log map (Schwartz, 1977; Schwartz, 1980;

Schwartz, 1994). Despite some notable advantages (dramatic pixel count reduction, quasi size and rotation invariance), the complex log map has not been widely used in the machine vision community. In large part this has been due to the lack of shape invariance under translation, which severely complicates object recognition. This drawback has recently been addressed (Bonmassar and Schwartz, 1994), allowing frequency domain techniques to be applied in the complex log domain.

In a similar manner we wish to transform a class of standard space domain computational vision algorithms, which make use of the differential structure of an image, so that they may be used in the complex log domain. The geometry of an image is modified by the log-mapping, and this transformation must be accounted for to properly compute geometric invariants in the range of the mapping. In section (3) of this paper we treat the image as a scalar field and the complex log mapping as a change of coordinates, deriving the metric tensor of the transformation, as well as the form of both the ∇ and the ∇^2 operators in the space-variant coordinate system. The modification of the operators takes the form of negatively weighted exponentials which account for the varying distance between pixels as the distance from the fovea increases. The conformal nature of the mapping insures that the weighting is uniform in both the radial and angular directions. In section (3.1.) we show that this is concisely expressed by the metric tensor of the log domain, which provides a means to calculate Cartesian distances in the log plane. Examples are presented which verify the form of the modified operators and illustrate their use.

In the latter part of this paper we treat the image as a surface embedded in $\Re^3$, and derive the first (section (4.1.)) and second (section (4.2.)) fundamental forms of the surface (Carmo, 1976) using a log-polar parameterization, which leads to expressions for the Gaussian and mean curvature of images in terms of complex log coordinates. Together, these results allow the computation

of differential image characteristics directly in the log domain, without referring to the original Cartesian image. In terms of artificial vision systems this implies that all calculations can be performed on the much smaller log-domain image, yielding a dramatic speed increase. This is illustrated by the transformation of both a corner detection algorithm (section (5.4.)), as well as an image enhancement technique based on anisotropic diffusion (section (5.5.)), for use directly in the log domain. The corner detection algorithm was chosen as it is a simple but useful application of differential geometry to image processing, and is hence a convenient example of the application of the types of results derived in this paper. The anisotropic diffusion is used as an example as the form the nonlinear diffusion equation takes in the log domain is advantageous when compared to the Cartesian form. Specifically, the non-uniform grid spacing inherent in the log domain allows the diffusion equation to be integrated with time steps which are exponential functions of retinal eccentricity, resulting in large scale structure enhancement in relatively few time steps. For biological vision, these results specify the form the computation of differential image structure must take if it is used in mammalian cortical systems.

2. Space-Variant Vision.

The mammalian retina is a space-variant sensor: the spacing of sensory neurons across the retinal surface is not uniform. The density of cells is greatest in the high acuity fovea, and falls off with retinal eccentricity. This allows the simultaneous achievement of high resolution and a wide field of view without requiring an enormous number of sensing elements. This anatomical feature has clear perceptual correlates. Visual acuity in the fovea is greater than in the periphery by at least a factor of 40 (Wertheim, 1894). This is the result of many factors including the optics of the eye (Campbell and Green, 1965), photoreceptor sampling density (Williams and Coletta, 1987),

spatial averaging due to the size of peripheral receptive fields (Merigan and Katz, 1990), as well as ganglion cell density (Wässle et al., 1990).

The mapping from the retina to striate cortex has been shown to be well approximated by a complex log map (Schwartz, 1977; Schwartz, 1980). This discovery has motivated the use of the complex log mapping in the construction of space variant sensors and algorithms for machine vision systems (Rojer and Schwartz, 1990; Weiman, 1988; Sandini and Dario, 1989, Sandini et al., 1989; Messner and Szu, 1986; Schenker et al., 1981; Bonmassar and Schwartz, 1994; Bonmassar and Schwartz, 1995; Bonmassar and Schwartz, 1996b; Bonmassar and Schwartz, 1996a). The log mapping expresses the variation in cortical area devoted to different regions of the retina.

2.1. Space-Variant Vision in Biology.

The investigation of the space-variant properties of the mammalian retino-cortical mapping dates back to the early 1960's. In order to characterize the transformation of visual data from retinal coordinates to primary visual cortex Daniel and Whitteridge (Daniel and Whitteridge, 1961) introduced the concept of the cortical magnification factor, M_c, measured in millimeters of cortex per degree of visual angle. The magnification factor is not constant across the retina, but rather varies as a function of eccentricity.[1] Experimentally, the cortical magnification factor has been found to be accurately approximated by

$$M_c(r) = \frac{C_1}{1 + C_2 r} \tag{2.1}$$

where r is the retinal eccentricity measured in degrees, and C_1 and C_2 are experimentally determined constants related to the foveal magnification and the rate at which magnification falls off

1. For example, approximately half of primary visual cortex is concerned with the central 8 degrees of the visual field (Wässle et al., 1990).

with eccentricity. Integrating equation (2.1) yields a relationship between retinal eccentricity and cortical distance ρ

$$\rho(r) = \int_0^r \frac{C_1}{1 + C_2 r'} dr' = \frac{C_1}{C_2} \log(1 + C_2 r) \tag{2.2}$$

Schwartz (Schwartz, 1977; Schwartz, 1980) has pointed out that the cortical magnification factor should be considered a vector quantity as opposed to a scalar one. The retino-cortical mapping can then be conveniently and concisely expressed as a conformal transformation. In this approach, a complex variable *z* is used to describe the retinal coordinates

$$z = re^{i\theta} = x + iy \tag{2.3}$$

Polar coordinates are then used to replace Cartesian ones in the retina

$$r = \sqrt{x^2 + y^2}\ ,\ \theta = \tan^{-1}\left(\frac{y}{x}\right) \tag{2.4}$$

The cortical point (ρ*(z)*,ϕ*(z)*) can then be specified by a single complex variable *w* as

$$w = \rho(z) + i\phi(z) = K \log(z + a), \quad Re(z) \geq 0 \tag{2.5}$$

where *K* is a scale factor determined by cortical area, which will be dropped in the following discussion, and *a* is a real positive constant, called the map parameter. The value of *a* determines the size of the quasi-linear region around *z*=0, and is generally believed to be in the range 0.3 to 0.7 degrees (see (Schwartz, 1994) for a discussion of the significance of *K* and *a*). The effect of modifying *a* on the mapping can be seen in the following way (see figure (2.1)). For small *z* (i.e. $z << a$), the mapping can be approximated using a series expansion around the point *z*=0:

$$w \approx \log(a) + \frac{z}{a} \tag{2.6}$$

Thus, in the fovea, the mapping is essentially linear. The magnitude of the derivative of the mapping gives an approximation to the cortical magnification factor:

$$\left|\frac{dw}{dz}\right| = \left|\frac{1}{z+a}\right| \tag{2.7}$$

Which is approximately constant for $z<<a$. The complex log transformation of equation (2.5) therefore smoothly varies from a linear map in the fovea, to a logarithmic map in the periphery, with the magnitude of a controlling the size of the region of approximate linearity. This is in contrast with other techniques which explicitly overlay a Cartesian fovea on a log image to obtain a similar effect (Sandini and Dario, 1989; Sandini et al., 1989).

Equation (2.5) is analytic everywhere in the domain and is hence conformal, implying that local angles are preserved by the transformation (Churchill and Brown, 1984). The singularity at the origin for the more commonly used complex log mapping $w=log(z)$ is removed at the cost of mapping the two hemifields separately and managing a discontinuity along the vertical meridian. The full form of the mapping for both hemifields[1] is given by

$$w = \begin{cases} \log(z+a) & \text{Real}(z) \geq 0 \\ 2\log(a) - \log(-z+a) & \text{Real}(z)<0 \end{cases}, \tag{2.8}$$

Figure (2.1) shows an example of an image, and its complex log transformation for a variety of values of the map parameter a. As can be seen, decreasing the value of a (moving from left to right) increases the magnification of the map, corresponding to an increased foveal representation.

1. The complex log transformation requires a branch cut which divides the complex plane along the imaginary axis. This division was originally motivated by brain anatomy: the two half-planes in the range of the mapping correspond to the primary visual area in each hemisphere of the brain.

The original image on the left contains 512x512=256K pixels, as compared to the images after the complex log transform which contain 14,307 pixels, 5,000 pixels, and 496 pixels respectively (see (Rojer and Schwartz, 1990) for a complete discussion of the details involved in the design of a complex log transformation). Thus, algorithms can be expected run an order of magnitude or two faster in the log domain as compared to the Cartesian one in contemporary machine vision implementations[1].

Figure 2.1 here

3. Space-variant metric tensor and differential operators.

In this section we compute the metric tensor of the complex log mapping, as well as the form of the ∇ operator, which yields the space-variant form of the gradient, the divergence, as well as the ∇^2 operator in the new coordinate system. The metric tensor expresses how the standard Cartesian metric induces a metric in the complex log domain. Used in conjunction with the conformal nature of the mapping, the metric tensor yields a simple derivation of the form of the ∇ operator in the log plane.

1. Rojer and Schwartz (Rojer and Schwartz, 1990) estimate that for biological systems the increase in speed can be up to four orders of magnitude.

As noted in section (2), the complex log coordinate transform considered in this work is of the form:

$$w = \log(z + a),\ a \in \Re,\quad z, w \in C,\ Re(z) \geq 0 \tag{3.1}$$

More explicitly, the log coordinates (ρ,ϕ) are given in terms of their Cartesian counterparts (x,y) by:

$$\rho = \log\left(\sqrt{(x+a)^2 + y^2}\right),\ \phi = \tan^{-1}\left(\frac{y}{(x+a)}\right) \tag{3.2}$$

The inverse relations are:

$$x = e^{\rho}\cos\phi - a,\ y = e^{\rho}\sin\phi \tag{3.3}$$

The log mapping of equation (3.2) as well as the inverse mapping given by (3.3) are both analytic everywhere in their respective domains, and are hence conformal. This has a number of interesting and useful implications. First, the conformal nature of the mapping ensures that local angles are preserved (Churchill and Brown, 1984). This in turn implies that the log-polar coordinate basis is orthogonal when projected into Cartesian space. This fact will be used to simplify the derivation of the log domain gradient in section (3.2.). Second, given any conformal mapping $w(z)=u(x,y) + i\,v(x,y)$, the Cauchy-Riemann equations can be used to relate the directional derivatives of the coordinate functions as follows (Greenberg, 1988):

$$\frac{\partial u}{\partial x} = \frac{\partial v}{\partial y},\ \frac{\partial u}{\partial y} = -\frac{\partial v}{\partial x} \tag{3.4}$$

These relationships follow directly from the path-independence of the differentiability of w, and ensure that the metric tensor has a simple form, as will be shown in the next section.

3.1. Metric tensor of the complex log mapping.

A useful way to understand the effects of the complex log transformation on the standard Cartesian operators is in terms of the metric tensor of the complex log domain. The metric tensor expresses how distance in log space relates to distance in Cartesian space. Because the coordinate transform is space-variant, the metric tensor is not constant as it is in the Cartesian plane, but rather varies as a function of log coordinate. Formally, the metric tensor T of a transformation z from a (ρ,ϕ) coordinate system into an (x,y) coordinate system (such as equation (3.3)) is given by

$$T = \begin{bmatrix} \langle z_\rho, z_\rho \rangle & \langle z_\rho, z_\phi \rangle \\ \langle z_\phi, z_\rho \rangle & \langle z_\phi, z_\phi \rangle \end{bmatrix} \tag{3.5}$$

Where the z_i are vectors whose elements are the derivatives of the new coordinate functions with respect to the subscripted variable. The metric tensor is a multilinear map which describes what happens to an infinitesimal length element under the transformation. That is, given a differential vector $dv=(d\rho, d\phi)$ in the (ρ, ϕ) domain, the metric tensor can be used to calculate the Cartesian length of dv in the transformed coordinates

$$\langle dv, dv \rangle = dv^T T dv \tag{3.6}$$

where the components of T in the log plane can be computed from equation (3.3)

$$T = \begin{bmatrix} x_\rho x_\rho + y_\rho y_\rho & x_\rho x_\phi + y_\rho y_\phi \\ x_\rho x_\phi + y_\rho y_\phi & x_\phi x_\phi + y_\phi y_\phi \end{bmatrix} = \begin{bmatrix} e^{2\rho} & 0 \\ 0 & e^{2\rho} \end{bmatrix} \tag{3.7}$$

The diagonal form of T is not coincidental, but is rather a direct consequence of the conformal nature of the complex log map via the Cauchy-Riemann equations. In this case, the Cauchy-Rie-

mann equations insure that $x_\rho = y_\phi$ and $x_\phi = -y_\rho$. Given these relationships, it is apparent that the off diagonal terms in T must cancel, and that therefore the metric tensor of any conformal mapping has the form $T = A\,\delta_{ij}$ (i.e. diagonal, with equal elements along the diagonal).

Examining the metric tensor yields some insights into the geometry of the log plane under the induced metric. From equation (3.7) we can see that as distance from the fovea increases, the Cartesian length of a log domain vector v_l is scaled by e^ρ. Conversely, the length of a Cartesian vector v_c mapped into the log plane shrinks by a factor of $e^{-\rho}$ due to the compressive logarithmic nonlinearity. This fact has direct implications for differentiation in the log domain as will be shown in the next section.

3.2. Space-Variant form of ∇f.

A straightforward, although tedious way to compute the space-variant form of the gradient is to use the chain rule to express ∇f in the new (ρ,ϕ) coordinate system:

$$\nabla f = \left(\frac{\partial f}{\partial \rho}\frac{\partial \rho}{\partial x} + \frac{\partial f}{\partial \phi}\frac{\partial \phi}{\partial x}\right)\boldsymbol{i} + \left(\frac{\partial f}{\partial \rho}\frac{\partial \rho}{\partial y} + \frac{\partial f}{\partial \phi}\frac{\partial \phi}{\partial y}\right)\boldsymbol{j} \tag{3.8}$$

To complete the transformation, the partials ρ_x, ϕ_x, ρ_y, ϕ_y as well as the unit vectors $\boldsymbol{i}$ and $\boldsymbol{j}$ would have to be calculated in terms of the log coordinates ρ and ϕ.

Instead of taking this approach, we proceed as follows. As noted in the introduction to this section, the conformality of the log mapping implies that local angles are preserved by the transformation. This simplifies the derivation considerably. Specifically, it insures that the basis vectors of the (ρ,ϕ) space which are orthogonal in the log domain, are also orthogonal when projected into Cartesian space (see figure (3.1)). Since the gradient is the combination of the directional

derivative in *any* two orthogonal directions, we are assured that the gradient in the log space is of the form

$$\nabla f = A(\rho, \phi)\left(\frac{\partial f}{\partial \rho}\boldsymbol{e}_\rho + \frac{\partial f}{\partial \phi}\boldsymbol{e}_\phi\right) \tag{3.9}$$

where the $A(\rho,\phi)$ term accounts for the variation in length a vector experiences under the log mapping, and $\boldsymbol{e}_p$ and $\boldsymbol{e}_\phi$ are an orthonormal basis (in the induced metric) for the log domain. Note that equation (3.9) holds for *any* conformal mapping, with the specifics of the transformation expressed in the coefficient function *A*. Another way to see that the gradient must be of the form given in (3.9) is to observe that any inhomogenous scaling of the basis vectors would result in the angle between the gradient and the basis vectors being different in the two spaces, which cannot be the case since the mapping is conformal. Using the chain rule we compute the length of $\partial f/\partial x$ relative to $\partial f/\partial \rho$ (or, equivalently, the length of $\partial f/\partial x$ relative to $\partial f/\partial \phi$):

$$\left|\frac{\partial f}{\partial \rho}\right| = \left|\frac{\partial f}{\partial x}\frac{\partial x}{\partial \rho}\right| = \left|\frac{\partial f}{\partial x}e^\rho \cos\phi\right| = e^\rho\left|\frac{\partial f}{\partial x}\right| \tag{3.10}$$

Which, of course, is exactly what the metric tensor indicates as well. This exponential scaling is accounted for by the $A(\rho,\phi)$ term. Normalizing the vectors in terms of the induced metric gives us an expression for the gradient in the space-variant domain[1]:

$$\nabla f = e^{-\rho}\left(\frac{\partial f}{\partial \rho}\boldsymbol{e}_\rho + \frac{\partial f}{\partial \phi}\boldsymbol{e}_\phi\right) \tag{3.11}$$

1. Note that this derivation does not account for the varying support of each log pixel. As one moves into the periphery of the log plane, each log pixel is typically generated by averaging a larger region of Cartesian space, both in the mammalian retina and in machine vision systems. The averaging is done to avoid aliasing in the periphery, and attenuates high frequency information, partially offsetting the need for a negative exponential weighting to account for varying pixel separation.

From equation (3.11) it is apparent that the ∇ has the general form $e^{-\rho}(\partial/\partial\rho\ \boldsymbol{e}_\rho + \partial/\partial\phi\ \boldsymbol{e}_\phi)$ which allows the direct computation of quantities such as the divergence and the curl in the log plane.

Figure 3.1 here

3.3. Space-Variant form of $\nabla\cdot\boldsymbol{f}$.

The form of the divergence of a vector field in the log plane can be calculated in a straightforward manner using the form of the ∇ operator derived in the prior section. To do so we will require the derivatives of the log plane orthonormal basis vectors $\boldsymbol{e}_\rho$ and $\boldsymbol{e}_\phi$ with respect to the log coordinates. Like their polar counterparts, $\boldsymbol{e}_\rho$ and $\boldsymbol{e}_\phi$ do not change in the radial direction and hence both derivatives with respect to ρ are zero. To calculate the change in the basis vector with respect to the angular log coordinate we use the chain rule as follows:

$$\boldsymbol{e}_\rho = \cos\phi\frac{\partial}{\partial x} + \sin\phi\frac{\partial}{\partial y},\ \boldsymbol{e}_\phi = \cos\phi\frac{\partial}{\partial y} - \sin\phi\frac{\partial}{\partial x} \tag{3.12a}$$

$$\frac{\partial \boldsymbol{e}_\rho}{\partial\phi} = \cos\phi\frac{\partial}{\partial y} - \sin\phi\frac{\partial}{\partial x} = \boldsymbol{e}_\phi,\ \frac{\partial \boldsymbol{e}_\phi}{\partial\phi} = -\sin\phi\frac{\partial}{\partial x} - \cos\phi\frac{\partial}{\partial y} = -\boldsymbol{e}_\rho \tag{3.12b}$$

Given these relations, the divergence of an arbitrary vector field whose components expressed in the orthonormal log basis ($\boldsymbol{e}_\rho$, $\boldsymbol{e}_\phi$) are (f^ρ, f^ϕ) can be calculated as:

$$\nabla \cdot f = e^{-\rho}\left(\frac{\partial}{\partial\rho}\boldsymbol{e}_\rho + \frac{\partial}{\partial\phi}\boldsymbol{e}_\phi\right)\cdot\left(f^\rho\boldsymbol{e}_\rho + f^\phi\boldsymbol{e}_\phi\right) \tag{3.13}$$

Using equations (3.12b) and the orthonormality of the basis vectors, the divergence simplifies to

$$\nabla \cdot f = e^{-\rho}\left(f^\rho_\rho + f^\phi_\phi + f^\rho\right) \tag{3.14}$$

where as before the subscripts indicate partial differentiation with respect to the subscripted variable.

3.4. Space-variant form of the Laplacian.

Given the divergence and the gradient, the calculation of the space-variant form of the Laplacian is also straightforward. Using the divergence in the log domain as given by equation (3.14) with the components of the vector field being the components of the log domain gradient ($f_\rho e^{-\rho}\boldsymbol{e}_\rho + f_\phi e^{-\rho}\boldsymbol{e}_\phi$) as specified by equation (3.11) yields a simple derivation of the Laplacian:

$$\nabla^2 f = \nabla\cdot\nabla f = e^{-\rho}\left(\left(f_\rho e^{-\rho}\right)_\rho + \left(f_\phi e^{-\rho}\right)_\phi + f_\rho e^{-\rho}\right) \tag{3.15a}$$

$$\Rightarrow \nabla^2 f = e^{-\rho}\left(f_{\rho\rho}e^{-\rho} - f_\rho e^{-\rho} + f_{\phi\phi}e^{-\rho} + f_\rho e^{-\rho}\right) = e^{-2\rho}\left(f_{\rho\rho} + f_{\phi\phi}\right) \tag{3.15b}$$

Note, that as was the case for the gradient, the Laplacian is the sum of the second directional derivatives in any two orthogonal directions. The conformal nature of the mapping again implies the preservation of angles which in turn insures that the Laplacian is of the form $B(\rho,\phi)$ $(f_{\rho\rho}+f_{\phi\phi})$

The Laplacian, similar to the gradient, can be obtained by treating the log plane as a Cartesian one, then weighting the result of calculating the standard Laplacian with a negative exponential in twice the radial coordinate.

3.5. Space-Variant form of $\nabla x \boldsymbol{f}$.

Using ∇ as a vector operator we now compute the curl of a vector field whose components in the orthonormal log basis ($\boldsymbol{e}_\rho$, $\boldsymbol{e}_\phi$) are (f^ρ, f^ϕ).

$$\nabla \times f = e^{-\rho}\left(\frac{\partial}{\partial \rho}\boldsymbol{e}_\rho + \frac{\partial}{\partial \phi}\boldsymbol{e}_\phi\right) \times \left(f^\rho \boldsymbol{e}_\rho + f^\phi \boldsymbol{e}_\phi\right) \tag{3.16a}$$

$$= \boldsymbol{e}_\rho \times e^{-\rho}\frac{\partial}{\partial \rho}\left(f^\rho \boldsymbol{e}_\rho + f^\phi \boldsymbol{e}_\phi\right) + \boldsymbol{e}_\phi \times e^{-\rho}\frac{\partial}{\partial \phi}\left(f^\rho \boldsymbol{e}_\rho + f^\phi \boldsymbol{e}_\phi\right) \tag{3.16b}$$

$$= \boldsymbol{e}_\rho \times \left(e^{-\rho} f^\rho_\rho \boldsymbol{e}_\rho + e^{-\rho} f^\phi_\rho \boldsymbol{e}_\phi\right) + \boldsymbol{e}_\phi \times \left(e^{-\rho} f^\rho_\phi \boldsymbol{e}_\rho + e^{-\rho} f^\phi_\phi \boldsymbol{e}_\phi + e^{-\rho} f^\rho \boldsymbol{e}_\phi - e^{-\rho} f^\phi \boldsymbol{e}_\rho\right) \tag{3.16c}$$

$$\Rightarrow \nabla \times f = e^{-\rho}\left(f^\phi_\rho - f^\rho_\phi + f^\phi\right) \tag{3.16d}$$

4. Differential Geometry.

In the prior section we treated an image as a scalar field over a subset of $\Re^2$. However, an alternate and equally valid approach is to consider it as a surface embedded in $\Re^3$. In this formulation the image is given by a set of points S

$$S = \{ (x, y, z) \mid z - I(x, y) = 0 \} \tag{4.1}$$

where $I(x,y)$ is the intensity function of the image. The set S is said to be a *regular surface* if it is locally diffeomorphic to $\Re^2$ (Carmo, 1976). Given this property, the surface can be parameterized by a set of maps $\boldsymbol{x_i}: U_i \rightarrow S$, $U_i \subset R^2$(note that bold faced symbols denote maps in this section). Each

$\boldsymbol{x_i}$ takes points in a neighborhood U_i of $\Re^2$ and maps them into points on the surface, thus describing how to construct S from pieces of the real plane. Together, the maps $\boldsymbol{x_i}$ cover the surface and define a coordinate system on it. The significance of the $\boldsymbol{x_i}$ is that they allow calculus to be applied to the surface through the mappings (assuming they themselves are sufficiently smooth). Symbolically the maps are written as

$$\mathbf{x}_i(u, v) = (x(u, v), y(u, v), z(u, v)), \qquad u, v \in U_i \subset \Re^2, \quad x, y, z \in \Re^3 \tag{4.2}$$

A simple and natural parameterization of an image surface S is given by:

$$\mathbf{x}(u, v) = (u, v, I(u, v)), \qquad u, v \in \Re^2 \tag{4.3}$$

A parameterization of a surface is not necessarily unique. A different parameterization can be obtained by defining a mapping from an open set U' in the plane of some new parameters ρ-ϕ to the domain U of $\boldsymbol{x}$ in the u-v plane. If the mapping is 1-1 and the determinant of the Jacobian of the transformation is non-zero in the domain, then the mapping is called an *allowable parameter transformation* (Lipschutz, 1969, page 157), and can be used to pass back and forth between the two domains[1]. The log mapping of equation (2.8) is an allowable parameterization as

$$\left|\frac{\partial(\rho, \phi)}{\partial(u, v)}\right| = \left|\begin{bmatrix} \rho_u & \rho_v \\ \phi_u & \phi_v \end{bmatrix}\right| = \left|\begin{bmatrix} e^{\rho}\cos\phi & e^{\rho}\sin\phi \\ -e^{\rho}\sin\phi & e^{\rho}\cos\phi \end{bmatrix}\right| = e^{2\rho} \neq 0 \tag{4.4}$$

Thus, the parameter transformation $u(\rho, \phi)$, $v(\rho, \phi)$ is allowable, and we can use it to parameterize the surface in the following way[2] (see figure (4.1)):

1. The inverse of an allowable parameter transformation is also allowable.

$$\boldsymbol{x}\left(u(\rho,\phi), v(\rho,\phi)\right) = \left(e^{\rho}\cos\phi - a, e^{\rho}\sin\phi, I\left(e^{\rho}\cos\phi - a, e^{\rho}\sin\phi\right)\right), \quad (4.5)$$
$$\log(a) \le \rho \le \log(a + R_{max}), \quad -\frac{\pi}{2} \le \phi \le \frac{\pi}{2}, \quad a > 0$$

Where R_{max} is the maximum radius in the visual field, a is a real positive constant, and ρ and ϕ are the coordinates of the log-polar domain as defined in section (2). In a slight abuse of notation, in this section we will use $I(\rho,\phi)$ to denote the image in log coordinates, that is $I(\rho,\phi)=I(e^{\rho}cos\phi-a,e^{\rho}sin\phi)$.

Figure 4.1 here

Just as a curve in $\Re^3$ can be uniquely specified by two local invariant properties (the curvature and the torsion), so too can a surface be uniquely defined by two local invariant properties known as the first (***I***) and the second (***II***) fundamental forms. These properties are invariant in the sense that they are functions of the surface, and not the parameterization (i.e. they are invariant under a parameter transformation). The components of the first and second fundamental forms are widely

2. To cover the surface we need a second parameter transformation $u(\rho',\phi')$, $v(\rho',\phi')$ corresponding to Real(z)<0 in equation (2.8), whose domain is the left hemifield $-\pi/2 \le \phi \le \pi/2$. This extension is straightforward, and for brevity's sake we leave it as an exercise for the reader.

used in computational vision, and in this section we will derive the form they take in the log domain using the parameterization of equation (4.5).

4.1. The First Fundamental Form.

The first fundamental form of differential geometry, denoted ***I***, is used to compute metric properties of a surface *S*, such as length, area, and angle, and is closely related to the metric tensor computed in section (3.1.). ***I*** is a mapping which takes tangent vectors in the plane of the parameterization to tangent vectors on the surface. The surface tangent vectors can then be used in the same way as tangent vectors in the plane, i.e. to compute the angle of intersection of two curves, integrated to yield the length of a curve on the surface, etc.... Formally, the first fundamental form ***I*** of a surface *S* at a point $p=(u_0,v_0)$, given a parameterization ***x***, is a quadratic positive definite form defined on a vector $w=(du, dv)$ in the *u*-*v* plane given by:

$$\boldsymbol{I}_p(w) = \langle w, w \rangle_p = \begin{bmatrix} du & dv \end{bmatrix} \begin{bmatrix} E & F \\ F & G \end{bmatrix} \begin{bmatrix} du \\ dv \end{bmatrix} \tag{4.6}$$

where *E*, *F*, and *G*, the coefficients of the first fundamental form, are functions of *p*, and are the components of the metric tensor of the surface:

$$E(u_0, v_0) = \langle \mathbf{x}_u, \mathbf{x}_u \rangle_p, \quad F(u_0, v_0) = \langle \mathbf{x}_u, \mathbf{x}_v \rangle_p, \quad G(u_0, v_0) = \langle \mathbf{x}_v, \mathbf{x}_v \rangle_p \tag{4.7}$$

where the subscripts *u* and *v* indicate differentiation with respect to the subscripted variable, and the inner products are evaluated at the point *p*. The first fundamental form uses the metric tensor of the parameterization to express how the standard Cartesian metric in $\Re^3$ induces a metric on the tangent plane of the surface at each point.

We can obtain an expression for the first fundamental form using the Cartesian basis and the parameterization given by equation (4.3):

$$\mathbf{x}_u = (1, 0, I_u)\ ,\ \mathbf{x}_v = (0, 1, I_v) \tag{4.8a}$$

$$E(u_0, v_0) = 1 + I_u^2, \quad F(u_0, v_0) = I_u I_v\ , \quad G(u_0, v_0) = 1 + I_v^2 \tag{4.9a}$$

$$\boldsymbol{I}_p(du, dv) = \begin{bmatrix} du & dv \end{bmatrix} \begin{bmatrix} 1 + I_u^2 & I_u I_v \\ I_u I_v & 1 + I_v^2 \end{bmatrix} \begin{bmatrix} du \\ dv \end{bmatrix} \tag{4.9b}$$

Calculating the first fundamental form using the log domain parameterization is now straight-forward. We apply equation (4.7) to the log domain parameterization given by (4.5) at the point $q=(\rho_0,\phi_0)$. To do so, we must calculate the first derivatives of the mapping

$$\mathbf{x}_\rho = \left(e^\rho \cos\phi, e^\rho \sin\phi, I_\rho \right), \mathbf{x}_\phi = \left(-e^\rho \sin\phi, e^\rho \cos\phi, I_\phi \right) \tag{4.10}$$

The coefficients of $\boldsymbol{I}$ in terms of the complex-log variables are then given by

$$E(\rho_0, \phi_0) = e^{2\rho} + I_\rho^2,\ F(\rho_0, \phi_0) = I_\rho I_\phi,\ G(\rho_0, \phi_0) = e^{2\rho} + I_\phi^2 \tag{4.11}$$

The native log domain form of the first fundamental form is thus given by:

$$\boldsymbol{I} = \begin{bmatrix} d\rho & d\phi \end{bmatrix} \begin{bmatrix} E' & F' \\ F' & G' \end{bmatrix} \begin{bmatrix} d\rho \\ d\phi \end{bmatrix} = \begin{bmatrix} d\rho & d\phi \end{bmatrix} \begin{bmatrix} e^{2\rho} + I_\rho^2 & I_\rho I_\phi \\ I_\rho I_\phi & e^{2\rho} + I_\phi^2 \end{bmatrix} \begin{bmatrix} d\rho \\ d\phi \end{bmatrix} \tag{4.12}$$

From equation (4.12) it is clear that for a constant surface the first fundamental form reduces to the metric tensor of the coordinate transformation given in equation (3.7), as one would expect.

Example: Consider the upper hemisphere of the unit sphere covered by the parameterization

$$\mathbf{x}(u, v) = \left(u, v, \sqrt{1 - u^2 - v^2} \right), \left(u^2 + v^2 \right) < 1 \tag{4.13}$$

In Cartesian coordinates we have:

$$\mathbf{x}_u = \left(1, 0, -u\left(1 - u^2 - v^2 \right)^{-0.5} \right), \mathbf{x}_v = \left(0, 1, -v\left(1 - u^2 - v^2 \right)^{-0.5} \right) \tag{4.14}$$

$$E = \frac{1 - v^2}{1 - u^2 - v^2}, F = \frac{uv}{1 - u^2 - v^2}, G = \frac{1 - u^2}{1 - u^2 - v^2} \tag{4.15}$$

The first fundamental form can be used to calculate the length of a curve $\alpha(t)$ along the surface by integrating the square root of $\boldsymbol{I}$ of the tangent to the curve along $\alpha(t)$. For example, the curve $\alpha(t)=(t,0)$ in the u-v plane gives rise to a curve on the sphere whose length is given by

$$s(\alpha(t)) = \int_0^T \sqrt{\mathbf{I}_{\alpha(t)}(\alpha'(t))}\, dt = \int_0^T \sqrt{\frac{1}{1 - t^2}} dt = \tan^{-1}\left(T\left(1 - T^2 \right)^{-0.5} \right) \tag{4.16}$$

In terms of log coordinates, the same region of the sphere is parameterized by:

$$\mathbf{x}(\rho, \phi) = \left(e^{\rho}\cos\phi - a, e^{\rho}\sin\phi, \sqrt{d} \right), \quad d > 0 \tag{4.17}$$

where we have defined $d = 1-a^2-e^{2\rho}+2ae^{\rho}\cos\phi$ for conciseness. We can now directly apply equations (4.11) to calculate the coefficients of the first fundamental form in the log plane:

$$E = e^{2\rho} + \frac{e^{2\rho}\left(a\cos\phi - e^{\rho} \right)^2}{d}, F = \frac{ae^{2\rho}\left(a\cos\phi - e^{\rho} \right)\sin\phi}{d}, G = e^{2\rho} + \frac{a^2 e^{2\rho}\sin\phi^2}{d} \tag{4.18}$$

Using equation (3.2) we calculate the form of $\alpha(t)$ in log coordinates to be $\alpha(t)=(\log(t+a),0)$ which implies $\alpha'(t)=(d\rho,d\phi)=((t+a)^{-1},0)$. The length of $\alpha(t)$ on the sphere calculated using the log parameterization is then

$$s(\alpha(t)) = \int_0^T \sqrt{\left((t+a)^2 + \frac{(t+a)^2 t^2}{1 - a^2 - (t+a)^2 + 2a(t+a)} \right) \frac{1}{(t+a)^2}} dt = \int_0^T \sqrt{\frac{1}{1-t^2}} dt \quad (4.19)$$

which serves both as a check that (4.12) is correct as well as an example of the invariance of $\boldsymbol{I}$ under a change of parameters.

4.2. The Second Fundamental Form.

The second fundamental form of a surface is a second order quadratic form which yields information about the local characteristics of the surface relative to the embedding space $\Re^3$. For example, the coefficients of the second fundamental form determine which side of its tangent plane the surface lies on in a neighborhood of a point, and therefore can be used to classify the point as hyperbolic, parabolic, elliptic, or planar. Like the first fundamental form, it is invariant under a parameter transformation[1]. Together, the first and second fundamental forms uniquely determine a surface up to a rigid transformation.

The second fundamental form is defined in terms of the differential of the normal vector field of the surface. Given a regular surface S together with a parameterization $\boldsymbol{x}$ defined on $U \subset \Re^2$, we can construct a unit normal vector field[2] $\boldsymbol{N}$ to the surface by taking the vector product of two linearly independent tangent vectors to the surface at each point:

$$\mathbf{N}(q) = \frac{\mathbf{x}_u \wedge \mathbf{x}_v}{|\mathbf{x}_u \wedge \mathbf{x}_v|} \quad (4.20)$$

The mapping $\boldsymbol{N}: S \to S^2$ from the surface to the unit sphere is called the *Gauss map* of S. The differential of the Gauss map, $d\boldsymbol{N}_q$ measures how rapidly the normal vector N pulls away from

1. If a parameter transformation reverses the orientation of the normal vector field then the second fundamental form changes its sign.
2. The choice of sign for the unit normal is arbitrary.

N(q) in a neighborhood of the point *q*. When represented as a matrix in the basis $\{\boldsymbol{x}_u,\boldsymbol{x}_v\}$ the determinant of $d\boldsymbol{N}_q$ is the *Gaussian curvature K* of *S* at *q*, and the negative of half the trace is known as the *mean curvature H* of *S* at *q*. The second fundamental form of *S* which takes a vector in T_qS and returns a measure of the curvature in that direction is also defined in terms of the differential of the Gauss map by (Carmo, 1976, page 141)

$$\boldsymbol{II}_q(w) = -\langle d\mathbf{N}_q(w), w\rangle \tag{4.21}$$

If *w* is a unit vector then the second fundamental form gives the normal curvature of any curve parameterized by arc length passing through *q* tangent to *w.* The maximum and minimum curvatures at a point are denoted k_1 and k_2 respectively and are known as the *principal curvatures* of *S* at *q*. The corresponding directions are called *principal directions*. The Gaussian and mean curvature are given by the product and average of the principal curvatures respectively. In the basis $\{\boldsymbol{x}_u,\boldsymbol{x}_v\}$ the second fundamental form is given by

$$\boldsymbol{II}_q(du, dv) = \begin{bmatrix} du & dv \end{bmatrix} \begin{bmatrix} e & f \\ f & g \end{bmatrix} \begin{bmatrix} du \\ dv \end{bmatrix} \tag{4.21c}$$

where the coefficients *e*, *f*, and *g* are[1]

$$e = \langle \mathbf{N}, \mathbf{x}_{uu}\rangle_q, \quad f = \langle \mathbf{N}, \mathbf{x}_{uv}\rangle_q, \quad g = \langle \mathbf{N}, \mathbf{x}_{vv}\rangle_q \tag{4.22}$$

Using the above definitions, we can compute ***N(q)*** for the standard Cartesian parameterization given by equation (4.3) as

$$\boldsymbol{N}(q) = \frac{(-I_u, -I_v, 1)}{\sqrt{1 + I_u^2 + I_v^2}} \tag{4.23}$$

1. Some authors use the notation *L, M, N* for the coefficients of the second fundamental form.

The coefficients are then given by

$$e = \frac{I_{uu}}{\sqrt{1 + I_u^2 + I_v^2}}, \quad f = \frac{I_{uv}}{\sqrt{1 + I_u^2 + I_v^2}}, \quad g = \frac{I_{vv}}{\sqrt{1 + I_u^2 + I_v^2}} \tag{4.24}$$

The Gaussian curvature K can then be computed from the coefficients of $\boldsymbol{I}$ and $\boldsymbol{II}$ as follows:

$$K = \frac{eg - f^2}{EG - F^2} = \frac{I_{uu}I_{vv} - I_{uv}^2}{\left(1 + I_u^2 + I_v^2\right)^2} \tag{4.25}$$

Similarly, the mean curvature H is

$$H = \frac{1}{2}\frac{eG - 2fF + gE}{EG - F^2} = \frac{\left(1 + I_u^2\right)I_{vv} - 2I_uI_vI_{uv} + \left(1 + I_v^2\right)I_{uu}}{2\left(1 + I_u^2 + I_v^2\right)^{\frac{3}{2}}} \tag{4.26}$$

The same procedure can now be used to calculate these quantities in the complex-log domain. Using the parameterization of equation (4.5) we obtain an expression for the surface normal in log-polar coordinates:

$$\boldsymbol{N}(q) = \frac{\mathbf{x}_\rho \wedge \mathbf{x}_\phi}{\left|\mathbf{x}_\rho \wedge \mathbf{x}_\phi\right|} = \frac{\left(I_\phi \sin\phi - I_\rho \cos\phi, \quad -I_\phi \cos\phi - I_\rho \sin\phi, \quad e^{2\rho}\right)}{\sqrt{e^{2\rho} + I_\rho^2 + I_\phi^2}} \tag{4.27}$$

The second partials of the log domain parameterization are

$$\mathbf{x}_{\rho\rho} = \left(e^\rho \cos\phi, \quad e^\rho \sin\phi, \quad I_{\rho\rho}\right) \tag{4.28a}$$

$$\mathbf{x}_{\rho\phi} = \left(-e^\rho \sin\phi, \quad e^\rho \cos\phi, \quad I_{\rho\phi}\right) \tag{4.28b}$$

$$\mathbf{x}_{\phi\phi} = \left(-e^\rho \cos\phi, \quad -e^\rho \sin\phi, \quad I_{\phi\phi}\right) \tag{4.28c}$$

where we have assumed things are sufficiently smooth such that all mixed partials are equal. The coefficients of the second fundamental form in the log domain are then given by

$$e = \langle \mathbf{N}, \mathbf{x}_{\rho\rho} \rangle_q = \frac{e^{\rho}\,(I_{\rho\rho} - I_{\rho})}{\sqrt{e^{2\rho} + I_{\rho}^2 + I_{\phi}^2}} \tag{4.29a}$$

$$f = \langle \mathbf{N}, \mathbf{x}_{\rho\phi} \rangle_q = \frac{e^{\rho}\,(I_{\rho\phi} - I_{\phi})}{\sqrt{e^{2\rho} + I_{\rho}^2 + I_{\phi}^2}} \tag{4.29b}$$

$$g = \langle \mathbf{N}, \mathbf{x}_{\phi\phi} \rangle_q = \frac{e^{\rho}\,(I_{\phi\phi} + I_{\rho})}{\sqrt{e^{2\rho} + I_{\rho}^2 + I_{\phi}^2}} \tag{4.29c}$$

As before, we calculate the Gaussian and mean curvature using the coefficients of the first and second fundamental forms

$$K = \frac{-I_{\phi}^2 - I_{\phi\phi} I_{\rho} - I_{\rho}^2 + 2 I_{\phi} I_{\rho\phi} - I_{\rho\phi}^2 + I_{\phi\phi} I_{\rho\rho} + I_{\rho} I_{\rho\rho}}{\left(e^{2\rho} + I_{\phi}^2 + I_{\rho}^2 \right)^2} \tag{4.30}$$

$$H = \frac{e^{2p} I_{\phi\phi} + I_{\phi}^2 I_{\rho} + I_{\phi\phi} I_{\rho}^2 + I_{\rho}^3 - 2 I_{\phi} I_{\rho} I_{\rho\phi} + e^{2p} I_{\rho\rho} + I_{\phi}^2 I_{\rho\rho}}{2 e^{p} \left(e^{2\rho} + I_{\phi}^2 + I_{\rho}^2 \right)^{\frac{3}{2}}} \tag{4.31}$$

Example: Continuing our prior example, we again consider the upper hemisphere of the unit sphere covered by the parameterization

$$\mathbf{x}(u, v) = \left(u, v, \sqrt{1 - u^2 - v^2} \right), \left(u^2 + v^2 \right) < 1 \tag{4.32}$$

Since the curvature in any direction at every point on the sphere is given by $1/r$, it is clear that the principal curvatures are both equal to 1. This implies that the Gaussian and mean curvature are also unity. Verifying this is a straightforward computation. In Cartesian coordinates we have:

$$I_{uu} = \frac{\left(v^2-1\right)}{\left(1-u^2-v^2\right)^{\frac{3}{2}}},\ I_{uv} = \frac{-uv}{\left(1-u^2-v^2\right)^{\frac{3}{2}}},\ I_{vv} = \frac{\left(u^2-1\right)}{\left(1-u^2-v^2\right)^{\frac{3}{2}}} \tag{4.33}$$

The coefficients of the second fundamental form (equation (4.24)) are then given by

$$e = \frac{v^2-1}{1-u^2-v^2},\ f = \frac{-uv}{1-u^2-v^2},\ g = \frac{u^2-1}{1-u^2-v^2} \tag{4.34}$$

Computing the Gaussian and mean curvatures from the coefficients we obtain

$$K = \frac{eg-f^2}{EG-F^2} = \frac{\left(1-u^2-v^2\right)^{-1}}{\left(1-u^2-v^2\right)^{-1}} = 1,\ H = \frac{1}{2}\frac{eG-2fF+gE}{EG-F^2} = 1 \tag{4.35}$$

As noted previously, the same region of the sphere is parameterized in terms of log coordinates by

$$\mathbf{x}\left(u\left(\rho,\phi\right),v\left(\rho,\phi\right)\right) = \left(e^{\rho}\cos\phi-a,\ e^{\rho}\sin\phi,\ \sqrt{d}\right),\quad d>0 \tag{4.36}$$

where as before, $d = 1-a^2-e^{2\rho}+2ae^{\rho}\cos\phi$ for conciseness. The first and second order partials of the intensity function with respect to the log coordinates are given by

$$I_{\rho} = e^{\rho}\left(a\cos\phi-e^{\rho}\right)d^{-0.5},\ I_{\phi} = \left(-ae^{\rho}\sin\phi\right)d^{-0.5} \tag{4.37a}$$

$$I_{\rho\rho} = e^{\rho}\left(a\cos\phi-e^{\rho}\right)d^{-0.5}-e^{2\rho}\left(a\cos\phi-e^{\rho}\right)^2 d^{\frac{-3}{2}}-e^{2\rho}d^{-0.5} \tag{4.37b}$$

$$I_{\phi\phi} = \left(-ae^{\rho}\cos\phi\right)d^{-0.5}-\left(ae^{\rho}\sin\phi\right)^2 d^{\frac{-3}{2}} \tag{4.37c}$$

$$I_{\rho\phi} = \left(ae^{\rho}\sin\phi\right)\left(a^2-1-ae^{\rho}\cos\phi\right)d^{\frac{-3}{2}} \tag{4.37d}$$

We now apply equations (4.30) to calculate the Gaussian curvature directly in the log domain

$$K = \frac{-I_\phi^2 - I_{\phi\phi}I_\rho - I_\rho^2 + 2I_\phi I_{\rho\phi} - I_{\rho\phi}^2 + I_{\phi\phi}I_{\rho\rho} + I_\rho I_{\rho\rho}}{\left(e^{2\rho} + I_\phi^2 + I_\rho^2\right)^2} = 1 \qquad (4.38)$$

which illustrates the invariance of the Gaussian curvature under a change of parameters.

5. Results.

In this section we present a few examples to illustrate how results derived in the prior sections can be used to modify a variety of image processing algorithms to run directly in the log domain. In the first two parts of this section we illustrate the use of the gradient and Laplacian in the log plane on simple synthetic images, and verify the results analytically. In section (5.3.) we use the metric tensor of the mapping to convert between degrees of the visual field and millimeters of primary visual cortex. In section (5.4.) we modify a more sophisticated algorithm, the corner detection technique of Deriche and Giraudon (Deriche and Giraudon, 1993), for use directly in the space-variant coordinate system. Finally, in section (5.5.) we derive a numerical implementation of anisotropic diffusion (Perona and Malik, 1987, Perona and Malik, 1990) for multi-scale image enhancement in log coordinates. The form of the simple numerical implementation in the log plane is shown to be equivalent to a sophisticated variable grid size integration of the underlying PDE, and may have implications for the possible neurophysiological correlates of perceptual filling-in in primary visual cortex.

5.1. Gradient magnitude.

As an illustration of the validity of equation (3.11) for gradient calculation in the log domain we construct an image which is a ramp function in the *x* coordinate *I(x,y)=x*. In Cartesian space,

the gradient of this image has constant magnitude ($|\nabla I|=1$). Figure (5.1) depicts the image, its log transform, and the two methods of gradient magnitude calculation. The third image from the left is the magnitude of the gradient treating the image as a Cartesian one. As can be seen, the gradient magnitude varies across the image, growing with increasing radial coordinate. In contrast, the image at the far right is generated by using equation (3.11) to calculate the gradient magnitude. It is approximately constant as is appropriate. Analytically this result is straightforward. Using

Figure 5.1 here

equation (3.11) to calculate the gradient in terms of complex log coordinates yields:

$$\nabla I\left(e^{\rho}\cos\phi - a, e^{\rho}\sin\phi \right) = \nabla\left(e^{\rho}\cos\phi - a \right) = \left(\cos\phi \mathbf{e}_{\rho} - \sin\phi \mathbf{e}_{\phi} \right) \qquad (5.1)$$

Computing the magnitude results in $|\nabla I|=\cos^2\phi+\sin^2\phi=1$ as expected.

5.2. Laplacian.

To illustrate the use of the modified Laplacian operator, we generate a Cartesian image which yields a constant value when operated on by ∇^2. In polar coordinates the intensity function is

given by $I(r,\theta)=r^2$ where r is the distance from the center of the image (i.e. a paraboloid). The Laplacian of this image is given by $\nabla\cdot\nabla(x^2+y^2)=4$. Figure (5.2) shows the results of applying an unmodified Laplacian (second image from the right) which yields varying values across the image, and the modified Laplacian (far right) of equation (3.15b) which is appropriately uniform.

Figure 5.2 here

As before, we can easily verify the result analytically using equation (3.15b):

$$\nabla^2 I\left(e^{\rho}\cos\phi - a, e^{\rho}\sin\phi \right) = \nabla^2\left(e^{2\rho} + a^2 - 2ae^{\rho}\cos\phi \right) = e^{-2\rho}\left(4e^{2\rho} \right) = 4 \qquad (5.2)$$

5.3. Metric Tensor: conversion between degrees of visual field and millimeters of cortex.

One use for the metric tensor of the coordinate transformation is the conversion between degrees of visual field and millimeters of mammalian primary visual cortex. This is useful in the context of experiments which posit a neural substrate of a visual percept. The metric tensor allows conversion of distance and velocity in the visual field to the corresponding quantities in primary visual cortex, permitting comparisons of visual phenomena with physiological estimates of neural

conduction velocities. In that vein we will use Chervin and colleague's estimate of the lateral signal velocity in primary visual cortex of 0.06-0.09 m/sec (Chervin et al., 1988).

In order to compute the number of millimeters of primary visual cortex that a neural signal must transit in order to follow a given path in visual space we must calculate the length of the path in Cartesian space using the intrinsic metric of the log domain (or, more formally, the metric induced on Cartesian space by the inverse log mapping). This involves inverting the metric tensor of equation (3.7) and expressing it in Cartesian coordinates:

$$T^{-1} = \begin{bmatrix} \frac{1}{(x+a)^2 + y^2} & 0 \\ 0 & \frac{1}{(x+a)^2 + y^2} \end{bmatrix} \tag{5.3}$$

Given a parameterized curve $\alpha(t)=(x(t),y(t))$ in Cartesian coordinates, the cortical length of the path can be written in terms of the metric tensor as follows:

$$s(\alpha(t))_{cortex} = \int_{T_1}^{T_2} \sqrt{\left(\begin{bmatrix} x'(t) & y'(t) \end{bmatrix} T^{-1} \begin{bmatrix} x'(t) \\ y'(t) \end{bmatrix} \right)} dt = \int_{T_1}^{T_2} \sqrt{\left(\frac{x'^2 + y'^2}{(x+a)^2 + y^2} \right)} dt \tag{5.4}$$

where T_1 and T_2 are the starting and ending eccentricities respectively. To compute the length of a path in millimeters of primate cortex, we require a reasonable value for a, as well as reinstating the scale factor K dropped in section (2.1.). Estimates of the value of a vary widely in the literature from as low as 0.1^o to as high as 4^o (Schwartz, 1994; Van Essen et al., 1984; Dow et al., 1985; Tootell et al., 1988; Levi et al., 1985). Setting a=0.6 degrees, K=5 mm/degree (Rojer and

Schwartz, 1992) for macaque and K=7 mm/deg for human[1], and choosing a path along the x axis given by $\alpha(t)=(t,0)$ which results in a straight line in the range of the mapping, we obtain

$$s(\alpha(t))_{cortex} = K\int_{T_1}^{T_2}\sqrt{\left(\frac{1}{(t+a)^2}\right)}dt = K(\log(a+T_2) - \log(a+T_1)) \qquad (5.5)$$

Thus, a horizontal visual path from 0^o to 2.5^o corresponds to a cortical length of approximately 8.2 mm. Since primary visual cortex in the macaque is about 40 mm in length, we find that a path crossing the central 5 degrees of the visual field maps to a cortical path of about 16.4 mm or approximately 41% of the total length of V1 in the macaque, in rough agreement with the figure cited by (Wilson et al., 1990). Setting T_1=0 and T_2=4 yields 20 mm of cortex devoted to the central 8^o, as noted in section (2.1.). One point to note in this context is that the variance in the estimates of a and K introduce errors in the estimates of cortical velocities which are nonlinearly dependent on eccentricity, and can be as much as a factor of 5.

The first example of the use of these equations will be taken from the physiological experiments of Tai Sing Lee (Lee, 1995). In this research, monkeys were shown images of texture squares and strips while recording were taken from cells in primary visual cortex. Cells in V1 near the center or medial axis of the square experienced a delayed enhancement in their firing rate, approximately 50-120 milliseconds slower than V1 response latency. The stimulus size was 4^ox4^o with the monkey initially foveating the center of the stimulus, indicating that a corresponding neural signal laterally traversing primary visual cortex would have an initial eccentricity of 2^o, and would propagate inwards to the center of the fovea. Thus

1. Human primary visual cortex has about twice the area of a rhesus monkey, and therefore units of area are scaled by 2, and units of length by √2 (Schwartz, 1980).

$$s(\alpha(t))_{cortex} = 5(\log(0.6+2) - \log(0.6)) = 7.33mm \quad (5.6)$$

Dividing by the measured delay range yields a required signal velocity in the range 0.06-0.15 mm/msec.

Our second example comes from psychophysical experiments performed by Paradiso and Nakayama (Paradiso and Nakayama, 1991). In an effort to probe the temporal characteristics of perceptual filling in they presented bright targets followed by a delayed mask, and measured the longest delay at which the mask could inhibit the perception of brightness at the center of the target. Using a least-squares fit they estimate the rate of perceptual filling in as roughly 110-150 degrees/second. Paradiso and Nakayama then translate this into cortical velocities using Levi's (Levi et al., 1985) estimate of magnification in human primary visual cortex, arriving at cortical velocity estimates in the range 0.15-0.40 mm/msec. Paradiso and Nakayama note that this computation ignores the possibility that speed may change with eccentricity (Paradiso and Nakayama, 1991, page 1233). Using equation (5.5) with the human estimate of K=7 mm/deg, together with the data they present in Fig. 5 (distance between target and mask versus time to half maximum), we compute a required cortical conduction velocity of between 0.015-0.064 mm/msec, well under the maximum velocity cited above.

5.4. Corner Detection.

Deriche and Giraudon (Deriche and Giraudon, 1993) outline a sophisticated technique for detecting and accurately localizing corners using the determinant of the Hessian of the intensity function ($I_{xx} I_{yy} - I_{xy}^2$) at a variety of scales. The method is based on the observation that a corner gives rise to an elliptic (positive curvature) maximum of the determinant of the Hessian in all directions[1] which moves through scale space. They note that the corner bisector line passes

through the maxima at different scales as well as the exact location of the corner. They construct the bisector line by finding elliptical maxima at 2 or more scales, then search for a zero crossing of the Laplacian along the bisector line, using it to specify the corner location.

We illustrate the transformation of this algorithm into the log domain using a synthetic smoothed corner at the Cartesian origin generated by

$$\frac{1}{4}\left(1 + erf\left(\frac{x}{\sqrt{C}}\right)\right)\left(1 + erf\left(\frac{y}{\sqrt{C}}\right)\right) \tag{5.7}$$

where C is a real positive constant which modulates the degree of smoothing, and *erf* is the standard error function defined by

$$erf(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}\,dt \tag{5.8}$$

Using either the chain rule or the invariance of the Gaussian curvature to a reparameterization, we can calculate the determinant of the Hessian in the log plane as

$$DET = \frac{-I_\phi^2 - I_{\phi\phi}I_\rho - I_\rho^2 + 2I_\phi I_{\rho\phi} - I_{\rho\phi}^2 + I_{\phi\phi}I_{\rho\rho} + I_\rho I_{\rho\rho}}{e^{4\rho}} \tag{5.9}$$

Choosing the map parameter a to be $1/(2\pi)$ we transform the corner into the log plane, then compute the determinant of the Hessian directly in the space-variant domain using equation (5.9), as depicted in figure (5.3). The local maximum in the log domain Hessian for C=2 occurs at (ρ_1,ϕ_1)=(0.5725,0.7219) which, using the inverse transformation of equation (3.3), maps to (1.171,1.171) in Cartesian coordinates. Repeating the procedure using C=8 yields

1. The determinant of the Hessian and the Gaussian curvature always have the same sign.

(ρ_2,ϕ_2)=(1.234,0.753) which corresponds to *(x,y)*=(2.343, 2.343). Together these two points

Figure 5.3 here

define the corner bisector line on which the actual corner point is constrained to lie. Transforming *y=mx+b* into log coordinates yields the equation for a log line:

$$\rho(\phi) = \log\left(\frac{ma-b}{m\cos\phi-\sin\phi}\right) \tag{5.10}$$

Next, the coordinates of the maxima of the Hessian (ρ_1,ϕ_1), (ρ_2,ϕ_2) are used to solve for the slope and intercept, obtaining *b*=0, *m*=1, which specifies the proper Cartesian bisector line. Finally, we search the log line $\rho(\phi)=log(a/(cos(\phi)-sin(\phi)))$ for zero-crossing of the log domain

Laplacian given by equation (3.15b), and find that the zero of the Laplacian occurs at $(\rho_c,\phi_c)=(log(a),0)$ corresponding to the Cartesian origin, which is the correct corner location.

5.5. Anisotropic Diffusion.

Another example of a sophisticated image processing technique which can be modified to run in the log plane is the anisotropic diffusion for image enhancement proposed by Perona and Malik (Perona and Malik, 1987, Perona and Malik, 1990). They suggested that a nonlinear diffusion equation in which the value of the conduction coefficient is inversely related to the local gradient magnitude could be used to enhance edges while smoothing regions. Thus, they treat image intensity as a conserved substance and allow it to diffuse over time by integrating the following nonlinear partial differential equation:

$$I_t = \nabla \cdot (c\,(|\nabla I|)\,\nabla I) \tag{5.11}$$

where I_t is the derivative of the intensity function with respect to time, the ∇ operator is with respect to the spatial coordinates, and the conductance coefficient *c()* is a function of the image intensity gradient magnitude.

The Perona-Malik equation (5.11) is a nonlinear partial differential equation of a type which is difficult to analyze. It has been suggested (Nitzberg and Shiota, 1992) that (5.11) is unstable for some parameter regimes, although this is still a point of investigation (Perona et al., 1994). Furthermore, it can amplify small scale noise which gives rise to high gradient magnitudes. Many variants of the Perona and Malik scheme have been proposed to improve its sensitivity to noise, its instability, and its equilibrium behavior (Alvarez et al., 1992; Catte et al., 1992; Dang et al., 1994; El-Fallah and Ford, 1994; Engquist et al., 1989; Illner and Neunzert, 1993; Li and Chen, 1994; Nitzberg and Shiota, 1992; Nördstrom, 1990; Osher and Rudin, 1990; Pauwels et al., 1993;

Price et al., 1990; Whitaker and Pizer, 1991; Whitaker, 1993; Cromartie and Pizer, 1993; Kacur and Mikula, 1995; Malladi and Sethian, 1995; Shah, 1996).

Using the space variant forms of the gradient (equation (3.11)) and the divergence (equation (3.14)) we can write equation (5.11) in log coordinates as

$$I_t = \nabla \cdot \left(ce^{-\rho} I_\rho \boldsymbol{e}_\rho + ce^{-\rho} I_\phi \boldsymbol{e}_\phi \right) = e^{-\rho} \left(\left(ce^{-\rho} I_\rho \right)_\rho + \left(ce^{-\rho} I_\phi \right)_\phi + ce^{-\rho} I_\rho \right) \tag{5.12a}$$

$$= e^{-\rho} \left(e^{-\rho} (cI_\rho)_\rho - ce^{-\rho} I_\rho + e^{-\rho} (cI_\phi)_\phi + ce^{-\rho} I_\rho \right) = e^{-2\rho} \left((cI_\rho)_\rho + (cI_\phi)_\phi \right) \tag{5.12b}$$

where the ρ, $\phi$and t subscripts denote partial differentiation with respect to the subscripted variable, and we have suppressed the arguments to *c()* and *I()* in the interests of conciseness. Using equation (5.12b) and substituting it into a Taylor series expansion of I around $t=t_0$ yields the first order approximation

$$I(t_0 + \Delta t) \approx I(t_0) + \Delta t \left(e^{-2\rho} (cI_\rho(t_0))_\rho + (cI_\phi(t_0))_\phi \right) \tag{5.13}$$

Using a discrete lattice with $\Delta\rho=\Delta\phi=1$, and considering the central pixel (ρ_0, ϕ_0), and its four connected neighbors (ρ_0, ϕ_{-1}), (ρ_0, ϕ_1), (ρ_{-1}, ϕ_0), and (ρ_1, ϕ_0) we use a centered difference approximation of the derivatives in (5.13). Labelling these pixels with superscripts 0, N, W, E, S respectively, we have

$$\left(c^0(t_0) I^0{}_\rho(t_0) \right)_\rho \approx \frac{c^E(t_0) I^E{}_\rho(t_0) - c^W(t_0) I^W{}_\rho(t_0)}{2} \tag{5.14}$$

$$\left(c^0(t_0) I^0{}_\phi(t_0) \right)_\phi \approx \frac{\left(c^S(t_0) I^S{}_\phi(t_0) - c^N(t_0) I^N{}_\phi(t_0) \right)}{2} \tag{5.15}$$

We use both backwards and forward differences to approximate the partial derivatives with respect to the spatial variables so as to limit the domain of our numerical implementation to the four nearest neighbors of the central pixel

$$\begin{aligned} I^{W}{}_{\rho}(t_0) &= I^{0}(t_0) - I^{W}(t_0), I^{E}{}_{\rho}(t_0) = I^{E}(t_0) - I^{0}(t_0), \\ I^{N}{}_{\phi}(t_0) &= I^{0}(t_0) - I^{N}(t_0), I^{S}{}_{\phi}(t_0) = I^{S}(t_0) - I^{0}(t_0) \end{aligned} \tag{5.16}$$

Substituting (5.14), (5.15) and (5.16) into (5.13) we arrive at

$$I^{0}(t_0 + \Delta t) \approx I^{0}(t_0)\left(1 - 0.5e^{-2\rho}\Delta t\left(\sum_{i \neq 0} c^{i}(t_0)\right)\right) + 0.5e^{-2\rho}\Delta t\left(\sum_{i \neq 0} c^{i}(t_0)\, I^{i}(t_0)\right) \tag{5.17}$$

Equation (5.17) can equivalently be written as the correlation of the image with a set of space and time varying masks

$$I(\rho, \phi, t_0 + \Delta t) \approx \sum_{\rho'}\sum_{\phi'} K^{t_0}{}_{\rho,\phi}(\rho', \phi')\, I(\rho + \rho', \phi + \phi', t_0) \tag{5.18}$$

where the mask weights are given by

$$K^{t_0}{}_{x,y} = \frac{e^{-2\rho}\Delta t}{2}\begin{bmatrix} 0 & c^{N}(t_0) & 0 \\ c^{W}(t_0) & \dfrac{2e^{2\rho}}{\Delta t} - \left(\sum_{i \neq 0} c^{i}(t_0)\right) & c^{E}(t_0) \\ 0 & c^{S}(t_0) & 0 \end{bmatrix} \tag{5.19}$$

In 2 dimensions the two components of the spatial gradient used in the computation of the conductance function are calculated using a Sobel operator with a negative exponential weight as specified by equation (3.11). At first sight equation (5.19) is distressing. It implies that diffusion falls off exponentially in the periphery. This makes perfect sense as diffusion is a random walk

process, and hence its propagation rate is proportional to $t^{1/2}$. This implies that it requires exponentially more time to diffuse across a peripheral pixel as compared to a foveal one. However, with the increased pixel spacing in the periphery comes increased numerical stability. An upper bound on the allowable stable time step Δt can be computed using Fourier-von Neumann stability analysis. In Cartesian space the numerical implementation will be stable if (Haberman, 1987):

$$\Delta t \leq \frac{(\Delta x)^2}{4c} \tag{5.20}$$

If we choose c to be in the range (0,1] and let Δx=1, we then have Δt<=0.25 (the lower bound on c is necessary given the ill-posed nature of the backwards heat equation (Haberman, 1987, page 74)). In the complex log plane the spatial grid of equation (5.20) is nonuniform. The inter-pixel distance is an exponential function of the radial coordinate, which implies that the stability constraint for an allowable time step in the log domain becomes

$$\Delta t \leq \frac{e^{2\rho}}{4c} \tag{5.21}$$

Equation (5.21) has important implications. It suggests that the nonlinear PDE (5.11) can be integrated using exponentially large time steps in the periphery, resulting in large scale structure enhancement in relatively few iterations. That is, we assume that t is approximately constant for a pixel and its four nearest neighbors, and allow the integration to proceed at different rates across the log domain image. Of course this is at the cost of fine scale image structure, but since such details aren't preserved in the periphery by the log mapping this is not a concern. Effectively, the space-variant time step allows different regions of the log plane to move through scale space at

different rates - faster in the periphery and slower in the foveal region. Figure (5.4) is an example using the above numerical implementation with a conductance function of the form

$$c(\nabla I) = e^{-\left(\frac{|\nabla I|}{k}\right)^2} \tag{5.22}$$

where k is a real constant which controls the relationship between edge strength and amount of diffusion. Note that the log plane diffusion is accomplished after only 5 iterations of the numerical technique, yet large scale structures such as the edge of the license plate and the boy's cheek are significantly enhanced in that time. To achieve comparable enhancement in the Cartesian domain requires between 50 and 100 iterations. Thus, the complex log mapping itself provides one order of magnitude compression, while the exponential diffusion rates allowable in the periphery of the log map yield another factor of 10 speed increase by reducing the required number of iterations. For example, the pepper image in the bottom row contains 512x152=262,144 pixels. Allowing the Cartesian image to diffuse[1] for 50 iterations requires almost 1,400 seconds to complete on a Sparc-20. The log mapped image in the third column is made up of 64x106=6,784 pixels, a compression factor of about 40. The large scale enhancement seen in the image at the lower right is achieved after only 5 iteration of the diffusion equation directly in the log domain, requiring a mere 4 seconds, a speed increase of a factor of 350. Given current DSP architectures, the complex log mapping provides a means for performing anisotropic diffusion for image enhancement in real-time.

1. The Cartesian diffusion is actually performed using the conductance function suggested by (El-Fallah and Ford, 1994) as the Perona-Malik scheme cannot handle the types of noise present in these images.

Figure 5.4 here

6. Conclusion.

In a deterministic framework, images are frequently treated in one of two ways: either as a scalar field over a subset of R^2, or as a surface embedded in R^3. The former approach leads to the use of the standard tools of multivariable calculus and integral transforms such as Fourier analysis. The latter formulation brings the tools of differential geometry to bear on image analysis, yielding invariant quantities such as the first and second fundamental forms. In this paper we have shown how computational vision algorithms can be implemented directly in the complex log

domain using both approaches. We have derived the form of the most common differential operators, ∇ and ∇^2, which can then be used in the calculation of the gradient, curl, divergence or the Laplacian in the log plane. In addition, we have viewed the space-variant image as a surface using a different parameterization from the standard Cartesian one. This led to the development of the first and second fundamental forms in the log-domain, which yields both the Gaussian and mean curvature.

These results specify the proper way to calculate differential image structure in space-variant images, and allows the modification of a variety of powerful image processing algorithms for use in the log domain (e.g. Perona and Malik, 1987; Nitzberg and Shiota, 1992; Deriche and Giraudon, 1993). From a computational standpoint, the form of the ∇ and ∇^2 operator is particularly appealing, as they can be calculated using existing algorithms for the Cartesian Laplacian and ∇ operators, the results of which need only be multiplied by a precalculated set of weights. In addition, the form of the these operators has implications for biological systems which employ a space-variant mapping, and in fact the space-variant form of ∇ has appeared in a recent model of the superior colliculus (Optican, 1995).

The transformation of anisotropic diffusion for image enhancement also has desirable properties in the log plane, resulting in a simple method for using a space-variant time step to achieve rapid large scale enhancement in the periphery while preserving fine details in the fovea. Finally, from a biological standpoint, any model of the mammalian visual system which includes the complex log mapping must also account for the geometric warping the image undergoes in passing to the new coordinate system, using techniques such as the ones we have outlined.

7. Bibliography.

Alvarez, L., Lions, P.-L., and Morel, J.-M. (1992). Image selective smoothing and edge detection by nonlinear diffusion. II. *SIAM Journal of Numerical Analysis*, 29(3):845–866.

Barth, E., Caelli, T., and Zetzsche, C. (1993). Image encoding, labeling, and reconstruction from differential geometry. *CVGIP: Graphical models and image processing*, 55(6):428–446.

Beaudet, P. (1978). Rotational invariant image operators. *4th International Conference on Pattern Recognition*, pages 579–583.

Bonmassar, G. and Schwartz, E. (1995). Space-variant fourier analysis: the exponential chirp transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, submitted.

Bonmassar, G. and Schwartz, E. (1996a). Fourier analysis and cortical architectures: the exponential chirp transform. *Real Time Imaging*, In press.

Bonmassar, G. and Schwartz, E. L. (1994). Geometric invariance in space-variant vision systems: the exponential chirp transform. In *ICPR Proceedings*, ICPR-12, pages 204–207. International Conference on Pattern Recognition.

Bonmassar, G. and Schwartz, E. L. (1996b). Lie groups, space-variant fourier analysis and the exponential chirp transform. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, June 18-20, 1996, San Francisco CA, USA. 492-498*.

Campbell, F. and Green, D. (1965). Optical quality of the human eye. *Journal of Physiology, London*, 186:558–578.

Carmo, M. P. D. (1976). *Differential Geometry of curves and surfaces*. Prentice-Hall, Englewood Cliffs, NJ.

Catte, F., Lions, P.-L., Morel, J.-M., and Coll, T. (1992). Image selective smoothing and edge detection by nonlinear diffusion. *SIAM Journal of Numerical Analysis*, 29(1):182–193.

Chervin, R., Pierce, P. A., and Connors, B. (1988). Periodicity and directionality in the propagation of epileptiform discharges across neocortex. *Journal of Neurophysiology*, 60(5):1695–1713.

Churchill, R. V. and Brown, J. W. (1984). *Complex variables and applications*. McGraw-Hill Book Company, New York.

Cooper, D., Elliot, H., Cohen, F., Reiss, L., and Symosek, P. (1981). Stochastic boundary estimation and object recognition. In Rosenfeld, A., editor, *Image Modeling*. Academic Press.

Cromartie, R. and Pizer, S. (1993). Structure-sensitive adaptive contrast enhancement methods and their evaluation. *Image and Vision Computing*, 11(8):460–467.

Dang, T., Olivier, J., and Maitre, H. (1994). An image segmentation technique based on edge preserving smoothing filter and anisotropic diffusion. In *Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation*, pages 65–69.

Daniel, P. M. and Whitteridge, D. (1961). The representation of the visual field on the cerebral cortex in monkeys. *Journal of Physiology*, 159:203–221.

Deriche, R., Faugeras, O., Giraudon, G., Papadopoulo, T., Vailland, R., and Viéville, T. (1992). Four applications of differential geometry to computer vision. In Orban, G. and Nagel, H.-H., editors, *Artificial and Biological Vision Systems*, pages 93–141. Springer-Verlag.

Deriche, R. and Giraudon, G. (1993). A computational approach for corner and vertex detection. *International journal of computer vision*, 10(2):101–124.

Dow, B., Vautin, R., and Bauer, R. (1985). The mapping of visual space onto foveal striate cortex in the macaque monkey. *The Journal of Neuroscience*, 5(4):890–902.

El-Fallah, A. I. and Ford, G. E. (1994). Nonlinear adaptive image filtering based on inhomogeneous diffusion and differential geometry. *SPIE Image and Video Processing II*, 2182:49–63.

Engquist, B., Lotstedt, P., and Sjogreen, B. (1989). Nonlinear filters for efficient shock computation. *Mathematics of Computation*, 52(186):509–537.

Florack, L. M., ter Haar Romeny, B. M., Koenderink, J. J., and Viergever, M. A. (1992). Scale and the differential structure of images. *Image and Vision Computing*, 10(6):376–388.

Geman, S. and Geman, D. (1984). Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6:721–741.

Greenberg, M. D. (1988). *Advanced Engineering Mathematics*. Prentice-Hall, Englewood Cliffs, NJ.

Haberman, R. (1987). *Elementary applied partial differential equations*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, second edition.

Illner, R. and Neunzert, H. (1993). Relative entropy maximization and directed diffusion equations. *Mathematical Methods in the Applied Sciences*, 17:545–554.

Kacur, J. and Mikula, K. (1995). Solution of nonlinear diffusion appearing in image smoothing and edge detection. *Applied Numerical Mathematics*, 50:47–59.

Kitchen, L. and Rosenfeld, A. (1982). Gray-level corner detection. *Pattern Recognition Letters*, 1:95–102.

Koenderink, J. (1984). The structure of images. *Biological Cybernetics*, 50:363–370.

Koenderink, J. and van Doorn, A. (1990). Receptive field families. *Biological Cybernetics*, 63:291–298.

Koenderink, J. J. and van Doorn, A. J. (1994). Two-plus-one dimensional differential geometry. *Pattern Recognition Letters*, 15:439–443.

Krueger, W. M. and Phillips, K. (1989). The geometry of differential operators with application to image processing. *IEEE Transactions on pattern analysis and machine intelligence*, 11(12):1252–1264.

Lee, T. S. (1995). A bayesian framework for understanding texture segmentation in the primary visual cortex. *Vision Research*, 35(18):2643.

Levi, D. M., Klein, S. A., and Aitsebaomo, A. (1985). Vernier acuity, crowding, and cortical magnification. *Vision Research*, 25(7):963–977.

Li, X. and Chen, T. (1994). Nonlinear diffusion with multiple edginess thresholds. *Pattern Recognition*, 27(8):1029–1037.

Lipschutz, M. M. (1969). *Differential Geometry*. Schaum's outline series. McGraw-Hill Book Company, New York.

Malladi, R. and Sethian, J. A. (1995). Image processing via level set curvature flow. *Proceedings of the National Academy of Sciences*, 92:7046–7050.

Merigan, W. H. and Katz, L. M. (1990). Spatial resolution across the macaque retina. *Vision Research*, 30(7):985–991.

Messner, R. and Szu, H. (1986). An image processing architecture for real time generation of scale and rotation invariant patterns. *Computer Vision, Graphics and Image Processing*, 31:50–66.

Nitzberg, M. and Shiota, T. (1992). Nonlinear image filtering with edge and corner enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(8):826–833.

Nördstrom, N. K. (1990). Biased anisotropic diffusion: a unified regularization and diffusion approach to edge detection. *Image and Vision Computing*, 8(4):318–327.

Optican, L. M. (1995). A field theory of saccade generation: temporal-to-spatial transform in the superior colliculus. *Vision Research*, 35(23/24):3313.

Osher, S. and Rudin, L. I. (1990). Feature-oriented image enhancement using shock filters. *SIAM Journal of Numerical Analysis*, 27(4):919–940.

Paradiso, M. A. and Nakayama, K. (1991). Brightness perception and filling-in. *Vision Research*, 31(7/8):1221–1236.

Pauwels, E. J., Proesmans, M., Gool, L. J. V., Moons, T., and Oosterlinck, A. (1993). Segmentation and image enhancement using coupled anisotropic diffusion. *SPIE*, 2094:836–847.

Perona, P. and Malik, J. (1987). Scale space and edge detection using anisotropic diffusion. In *Proceedings of the IEEE Computer Society Workshop on Computer Vision*, pages 16–27, Miami, FL.

Perona, P. and Malik, J. (1990). Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629–639.

Perona, P., Shiota, T., and Malik, J. (1994). Anisotropic diffusion. In Romeny, B. M. T. H., editor, *Geometry Driven Diffusion in Computer Vision*, chapter 3, pages 73–92. Kluwer.

Price, C. B., Wambacq, P., and Oosterlinck, A. (1990). Image enhancement and analysis with reaction-diffusion paradigm. *IEE Proceedings*, 137 Pt. I(3):136–145.

Rojer, A. S. and Schwartz, E. L. (1990). Design considerations for a space-variant visual sensor with complex-logarithmic geometry. In *10th International Conference on Pattern Recognition*, volume 2, pages 278–285.

Rojer, A. S. and Schwartz, E. L. (1992). Biological basis for space-variant sensor design II: Implications for VLSI sensor design. In *SPIE Program on Computer Vision and Intelligent Robots IX: Advances in Intelligent Machines*.

Sandini, G., Bosero, F., Bontino, F., and Ceccherini, A. (1989). The use of an anthropomorphic visual sensor for motion estimation and object tracking. *Proceedings of OSA optical meeting on image understanding and machine vision*.

Sandini, G. and Dario, P. (1989). Active vision based on space-variant sensing. *International symposium on robotics research*.

Schenker, P., Cande, E., Wong, K., and Patternson, W. (1981). New sensor geometries for image processing: computer vision in the polar exponential grid. *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing.*, pages 1144–1148.

Schwartz, E. (1994). Computational studies of the spatial architecture of primate visual cortex: Columns, maps, and protomaps. In Peters, A. and Rocklund, K., editors, *Primary visual cortex in primates*, volume 10 of *Cerebral Cortex*, chapter 9. Plenum Press, New York.

Schwartz, E. L. (1977). Spatial mapping in the primate sensory projection: analytic structure and relevance to perception. *Biological Cybernetics*, 25:181–194.

Schwartz, E. L. (1980). Computational anatomy and functional architecture of striate cortex: a spatial mapping approach to perceptual coding. *Vision Research*, 20:645–669.

Shah, J. (1996). A common framework for curve evolution, segmentation and anisotropic diffusion. In *Proceedings IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 136–142, San Francisco, CA.

Tootell, R., Hamilton, S., Silverman, M., and Switkes, E. (1988). Functional anatomy of macaque striate cortex. II. retinotopic organization. *Journal of Neuroscience*, 8:1531–1568.

Van Essen, D. C., Newsome, W. t., and Maunsell, J. H. r. (1984). The visual field representation in striate cortex of the macaque monkey: asymmetries, anisotropies, and individual variability. *Vision Research*, 24(5):429–448.

Wässle, H., Grünert, U., Röhrenbeck, J., and Boycott, B. B. (1990). Retinal ganglion cell density and cortical magnification factor in the primate. *Vision Research*, 30(11):1897–1911.

Weiman, C. (1988). 3-d sensing with polar exponential sensory arrays. *SPIE Conference on digital and optical shape representation and pattern recognition*.

Wertheim, T. (1894). Uber die indirekte sehscharfe. *Zeitschrift fur Psychologie und Physiologie der Sinnesorgane*, 7:172–187.

Whitaker, R. T. (1993). Geometry-limited diffusion in the characterization of geometric patches in images. *CVGIP: Image Understanding.*, 57(1):111–120.

Whitaker, R. T. and Pizer, S. M. (1991). A multi-scale approach to nonuniform diffusion. *Computer Vision, Graphics and Image Processing*, 57:99–110.

Williams, D. and Coletta, N. (1987). Cone spacing and the visual resolution limit. *Journal of the Optical Society of America*, 4:1514–1523.

Wilson, H. R., Levi, D., Maffei, L., Rovamo, J., and DeValois, R. (1990). The perception of form: retina to striate cortex. In Spillmann, L. and Werner, J. S., editors, *Visual Perception: The Neurophysiological Foundations*, chapter 10, pages 231–272. Academic Press.

FIGURE 2.1. Example of an image (left), and its complex log transformation (right), for various values of the map parameter *a*. Note that decreasing *a* increases the representation of the foveal region in the log plane. Dark areas correspond to regions outside the domain of the mapping.

FIGURE 3.1. Representation of the relationship between the basis vectors and the gradient in the two spaces. The mapping preserves the angles between the vectors, but not their lengths.

FIGURE 4.1. Graphical representation of the different parameterizations of the same surface S.

FIGURE 5.1. Example of an application of equation (3.11) for calculating the gradient magnitude in the log plane. From left to right: input image with intensity values $I(x,y)=x$; log map of the input image; gradient magnitude calculated treating the image plane as Cartesian; and far right: gradient magnitude calculated using equation (3.11).

FIGURE 5.2. Example of an application of equation (3.15b) for calculating the Laplacian in the log plane. From left to right: input image with intensity values $I(x,y)=r^2$; log map of the input image; Laplacian calculated treating the image plane as a Cartesian; and far right: Laplacian calculated using equation (3.15b).

FIGURE 5.3. Top row: corner (left) and the determinant of the Hessian (right) in Cartesian coordinates. Bottom row: complex log transform of the corner (left), Laplacian of the corner computed directly in the log domain (middle), and the determinant of the Hessian computed directly in the log plane. The actual corner location is a zero of the log domain Laplacian at $(\log(a),0)$.

FIGURE 5.4. Anisotropic diffusion (*k*=0.00075) in the Cartesian and log domains. From left to right: original images, diffusion in Cartesian domain (50 iterations), complex log transform of the original images, and images after 5 iterations of equation (5.11) directly in the log plane.

Figure 2.1

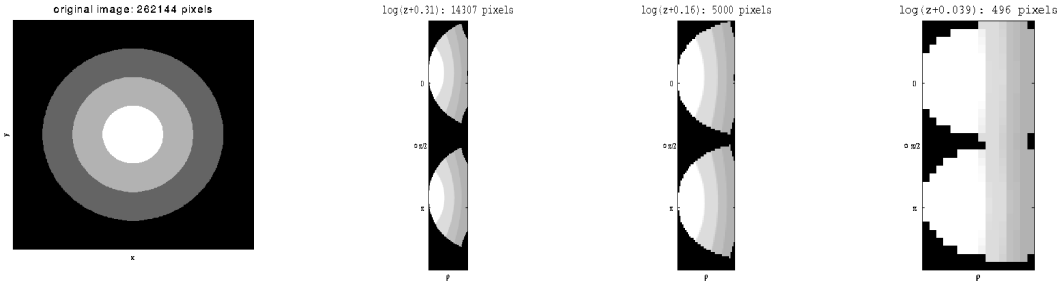
original image: 262144 pixels
log(z+0.31): 14307 pixels
log(z+0.16): 5000 pixels
log(z+0.039): 496 pixels

Figure 3.1

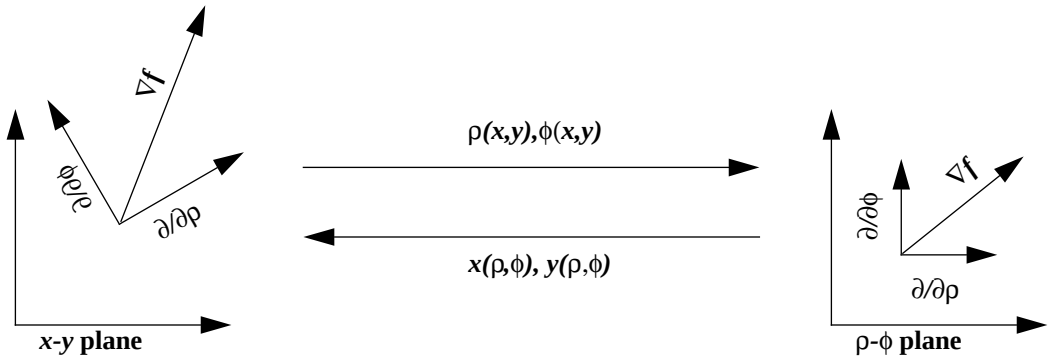

Figure 4.1

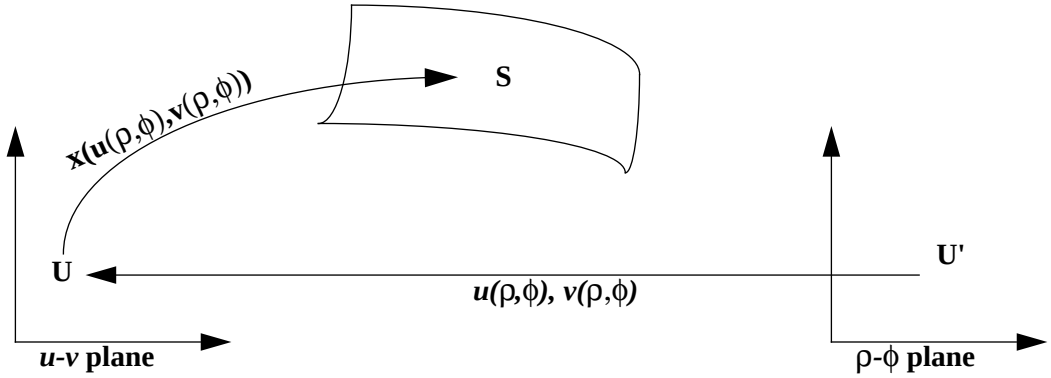

Figure 5.1

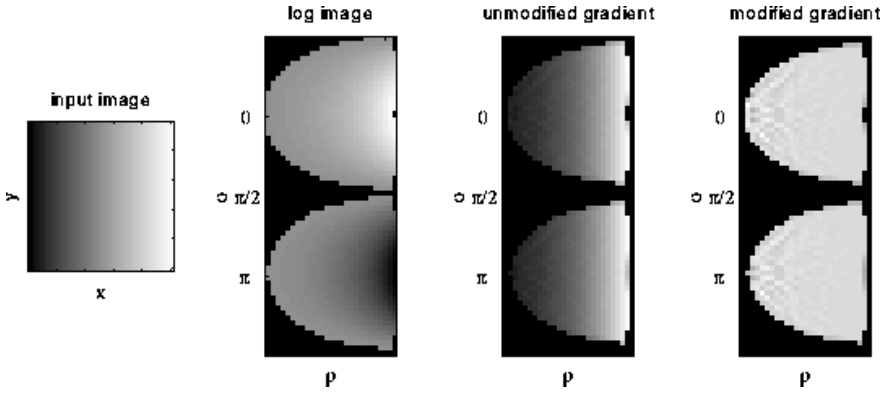
input image
y
x
log image
0
θ π/2
π
ρ
unmodified gradient
0
θ π/2
π
ρ
modified gradient
0
θ π/2
π
ρ

Figure 5.2

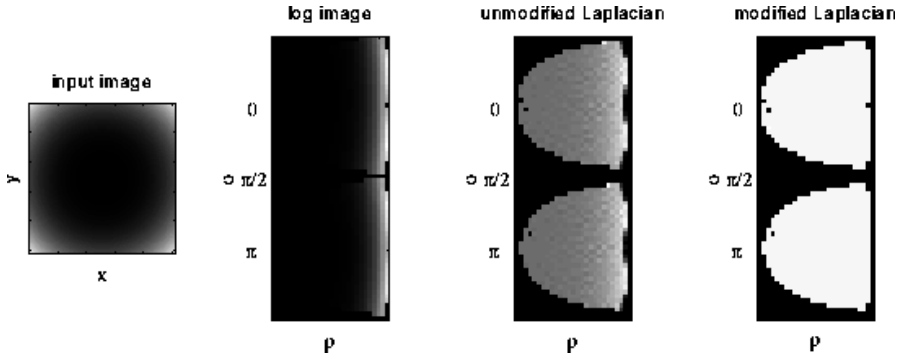
input image
y
x
log image
0
θ π/2
π
ρ
unmodified Laplacian
0
θ π/2
π
ρ
modified Laplacian
0
θ π/2
π
ρ

Figure 5.3

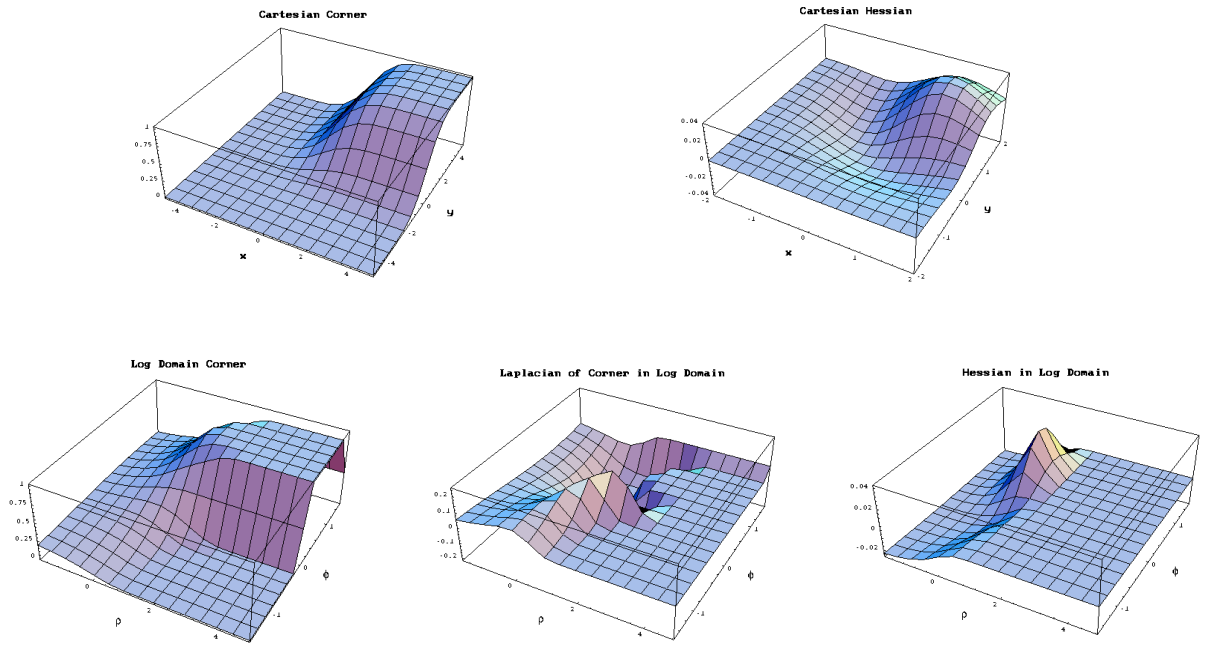
Cartesian Corner
Cartesian Hessian
Log Domain Corner
Laplacian of Corner in Log Domain
Hessian in Log Domain

Figure 5.4