Conformal Transplantation of Lightness to Varying Resolution Sensors

Brian V. Funt Michael Brockington Frank Tong
School of Computing Science, Simon Fraser University, Vancouver, B.C., Canada V5A 1S6
(604) 291-3126, e-mail funt@cs.sfu.ca

Abstract

Many standard computer vision algorithms depend upon the solution of Poisson equations, retinex lightness computation being one such algorithm. Using lightness as an example, we show how it can be computed on a varying resolution sensor so long as the sensor architecture is based on a conformal mapping. We develop the varying resolution lightness algorithm by the well known technique of using a conformal transplant. In the final analysis, very little of the algorithm needs to be changed, but a speed up in running time of roughly 50 times is obtained even though the total data is reduced by only a factor of 28.

1 Introduction

As part of the recent trend to "active vision," in the last few years there has been a drastic increase in interest in the area of camera sensors having varying resolution similar to that of the human eye. Such sensors, which are usually based on the complex-log transform, have nice properties in terms of image scaling and rotation. Much of the interest in such sensors ([17, 12, 11]) has been to exploit these properties to simplify particular vision problems (e.g., optical flow is simple when scaling is easy [7]); however, we can use these sensors to speed up significantly many basic vision algorithms directly. In this paper, we show how to do this for the particular case of lightness computation [5, 1] which is a fundamental component of the retinex [8] theory.

Retinex has always been developed in standard uniform-resolution, Cartesian coordinates; however, to be a reasonable theory of human color vision it clearly needs to be generalized to the varying resolution case since varying resolution is a fundamental property of the retina's sensor distribution. A serious question is whether or not the varying resolution will cause such differential blurring of the colors that retinex might not work.

Varying resolution sensors promise to provide very fast vision because processing power is concentrated in the high resolution area. It is well known that the complex-log mapping used in varying resolution sensors is conformal [17]; however, although it has been suggested that the conformality might be useful in the numerical solution of some physics problems, little has been done in exploiting the conformal properties in computer vision.

The importance of the conformal mapping lies in the fact that Poisson equations (and similar equations involving gradients) transplant under a conformal mapping with only a simple scaling adjustment. In addition to lightness algorithms, Poisson equations are at the very heart of many early-vision algorithms, including shape-from-shading, shape from motion, and surface interpolation. Since most such algorithms are $O(n^4)$ or at least $O(n^2 \log_2 n)$ for $n \times n$ images, given the fact that varying resolution sensors have many fewer pixels, there stand to be substantial savings in computation time when solved for in the transplanted, varying resolution case.

Our experiments with lightness computation show a roughly 48-fold speed up for a 128×65 pixel "retinal" image in comparison to a 512×512 pixel Cartesian image. For these results, we used the Fourier transform method of integration [13].

2 Complex log mapping for varying resolution

For the complex log mapping, the image plane is regarded as a complex plane with $z = x + iy$ ($i \equiv \sqrt{-1}$). Each location in the image z-plane is then mapped to the retinal[1] w-plane by

$$w = \ln z \tag{1}$$

Figure 1 shows the effect of this mapping on a rectangular grid. To avoid clutter, the figure shows fewer wedges and rings than are actually used. The actual mapping results in pixels on the retina being bounded by 65 concentric rings and 128 wedges.

Their pattern

[1] In our terminology we will exploit the analogy between the varying resolution image and the human retina and (shamelessly) use terms such as retina, fovea, periphery and fixation in referring to the computational model.

is best understood in terms of the polar form representation of complex numbers

$$z = r(\cos\theta + i\sin\theta) \quad ,$$

where $r = |z| = \sqrt{x^2+y^2}$ and $\theta = \tan^{-1}(y/x)$. After the $w = \ln z$ transform, $w = \ln r + i\theta$, so wedges correspond to discrete values of θ and rings to discrete values of $\ln r$.

The fact that we later apply fast Fourier (FFT) and fast cosine transforms to the retina governs the choice of the number of wedges and rings in that they require inputs of size 2^n and 2^n+1. In contrast, the new retinal RVS-1 CCD chip of the Sandini consortium [10, 16] has 30 rings and 64 wedges, plus 104 additional "foveal" pixels.

Our choice of a 128×65 retina leaves a central hole of radius 10 measured in image pixels. The value assigned to each retinal pixel is the average of the image pixels it covers. Some of the retinal pixels are in fact smaller than a single image pixel, in which case the image is over sampled. Oversampling occurs for all retinal pixels within radius 20.

Since the center of the fovea corresponds to $z = 0$, it presents a problem because the mapping $w = \ln z$ becomes undefined. Sandini's CCD overcomes the problem by providing a standard rectilinear grid of 104 pixels within the fovea. For our purposes, this solution has the drawback of creating a mapping that is no longer conformal throughout the retinal image. Schwartz [12] instead defines the mapping as $w = \ln(z+a)$, where a is fixed and entirely real. This too has drawbacks because of nonuniformities in the number of pixels per row and column in the mapping, which makes it impossible to apply the Fast Fourier Transform directly.

To cover the fovea, we have experimented with two alternatives. The first is a variant of the Sandini approach in which the fovea is defined by a cartesian grid of 33 by 33 pixels. This foveal patch, as shown in Figure 2, is the smallest patch of size 2^n+1 that is large enough to overlap the complex-log periphery as shown in Figure 2. The second approach is to fill the hole with 64 extra rings so that the central hole becomes less than one pixel wide.

Figure 3 shows the red band of a synthetic color mondrian image along with a fixation at the center of the image. For display purposes, the inverse transform of the retinal image is reproduced and the foveal hole becomes clearly visible. In the corresponding full-color image, the colors become blurred towards the periphery.

3 Lightness computation: cartesian case

Since image intensity is formed as the product of the intensity of the incident illumination and the surface albedo (reflectance), image intensity itself does not provide a direct indicator of intrinsic surface properties. Of particular, concern are variations in incident illumination across the scene. In a color image, the same issue arises, but in each of the three color bands. Land's retinex theory is based on comparison of the the estimate of surface albedo, called the "lightness," obtained separately for each band.

Horn [5, 6] extended Land's original one-dimensional, path-following lightness method to two-dimensions. Blake [1] then overcame the Dirichlet boundary condition restrictions inherent in Horn's method.

The basic idea of Horn-Blake lightness computation is to integrate the thresholded derivative of the logarithm of image intensity. This works because the world is assumed to be "mondrian" (i.e., a planar world covered with patches of uniform color and Lambertian reflectance). Figure 2a shows a mondrian. In principle, the mondrian assumptions severely limit the applicability of the method; however, Moore [9] obtains interesting results on unconstrained three-dimensional scenes by simply ignoring the restrictions.

In a mondrian world all transitions between patches of differing reflectance occur abruptly and all illumination changes occur gradually. For illumination $E(\lambda)$, surface reflectance $S(\lambda)$ and sensor sensitivity $R(\lambda)$, the sensor response $B(x,y)$ for image location (x,y) is given by:

$$B = \int_{\text{visible}} E(\lambda)\ S(\lambda)\ R(\lambda)\ d\lambda \qquad (2)$$

Lightness theories assume that the wavelength interdependence of illumination and reflectance can be ignored and equation (2) safely rewritten in terms of albedo A as:

$$B(x,y) = E(x,y)\ A(x,y) \qquad (3)$$

where

$$A = \int_{\text{visible}} S(\lambda)\ R(\lambda)\ d\lambda \qquad (4)$$

Applying logarithms to both sides of equation (3) turns the product into a sum: $b(x,y) = e(x,y) + a(x,y)$ ($b = \log B$ etc.). Taking the gradient of this sum yields the sum of two gradients, one corresponding to the illumination and the other to the albedo. Since the illumination varies smoothly, its gradient is small everywhere. The albedo gradient is zero within each color patch and very large (infinite in the continuous case) at patch boundaries. Thresholding $b(x,y)$ to zero whenever it is small eliminates the illumination component of the gradient so that only the albedo component remains.

To integrate the resulting gradient field thresholded by T, Blake differentiates again and obtains a Poisson equation to be solved for the lightness ϕ:

$$\nabla^2 \phi = \nabla \cdot T \nabla b \qquad (5)$$

where the boundary conditions normal to the boundary are $\vec{n} \cdot \nabla\phi = \vec{n} \cdot T\nabla b$. Blake solves this equation iteratively. In contrast, we follow the direct, transform method of Simchony *et al.* [13] and Swarztrauber [15].

In our implementation, the discrete gradient is computed by convolution with the two directional-derivative kernels:

$$\boxed{0.5 \mid -0.5} \quad \text{and} \quad \begin{array}{|c|}\hline 0.5 \\ \hline -0.5 \\ \hline \end{array}$$

A second application of these kernels (with due attention paid to eliminating shifts caused by using even kernels), as required by equation (5), means that the discrete Laplacian is effectively computed by convolution with the kernel:

	1/4	
1/4	−1	1/4
	1/4	

By the convolution theorem therefore, the Fourier transform of $\nabla^2\phi$ is simply the Fourier transform of ϕ multiplied by the Fourier transform of this Laplacian kernel. Dividing by the transformed kernel and taking the inverse transform, therefore, yields the lightness ϕ. The transform of the discrete approximation to the Laplacian is simply

$$\frac{i^2}{2}\left(2 - \cos\left(\frac{2\pi u}{M}\right) - \cos\left(\frac{2\pi v}{N}\right)\right)$$

where $i = \sqrt{-1}$, $u = 0, 1, 2, ..., M$ and $v = 0, 1, 2, ..., N$.

The Fourier transform imposes periodic boundary conditions; whereas, Blake's method requires Neumann conditions. Neumann conditions can be accommodated by using the discrete cosine transform instead.

4 Conformal mapping

Under a conformal mapping between coordinate systems, the corresponding Poisson equations in the two domains are very similar. For a Poisson equation of the form:

$$\nabla^2 \phi \;=\; \rho(x, y) \tag{6}$$

The corresponding Poisson equation to be solved under the conformal mapping is obtained from the conformal transplant of the forcing function ρ. Complex variables $z = x + iy$ are used to represent the (x, y)-plane. For completeness, we quote Henrici [4]:

> Let $f : z \rightarrow f(z)$ map D conformally onto some model region E and let $g : w \rightarrow f^{-1}(w)$ denote the inverse mapping. Let $\sigma(w) := \rho(g(w))$ for w in E; *i.e.*, let σ be the conformal transplant of ρ under the mapping f. If ϕ satisfies equation (5) its conformal transplant ψ satisfies[2]
>
> $$\nabla^2 \psi \;=\; \sigma(w)\ |g'(w)|^2$$

4.1 Transplanting lightness computation

To compute lightness on the retina, we must transplant equation (5). The mapping $f(z)$ from image to retina is $f(z) = \ln z$ and its inverse $g(w) = e^w$. It is trivial to show that $w = u + iv$ under this mapping satisfies the Cauchy-Riemann equations ($u_x = v_y$; $u_y = -v_x$) and therefore is conformal.

The transplant $\sigma(w)$ of the forcing function, $\rho = \nabla \cdot T\nabla b$, from the right-hand side of equation (5) is $\sigma(w) = \nabla \cdot T\nabla b(e^w)$.

Since $g'(w) = e^w$, the transplant of equation (5) to the retina is:

$$\nabla_w^2 \psi(w) \;=\; [\nabla_z T \nabla_z b(e^w)]|e^w|^2 \tag{7}$$

It is important to note that the threshold is defined in terms of the magnitude (which is approximated by the sum of the absolute values of its components) of the log-intensity gradient $|\nabla b|$ on the image z-plane. However, gradients will be computed directly on the retinal image since that is what the input is. Denoting the retinal log-intensity as $c(w)$, then under conformal transplantation, we can obtain the corresponding gradient in the image plane as

$$\nabla b(z) \;=\; \nabla c(w)\overline{f'(z)} \tag{8}$$

[2] pp.371-372 of reference [4].

The gradient magnitudes relate by a simple scaling, which using the polar form of a complex number becomes simply $|f'(z)| \;=\; 1/r$. In the discrete case, the scale factor can be thought of as accounting for the fact that the distance between each retinal pixel corresponds to a distance between image pixels that increases linearly with the distance from the retinal origin. Convolution across the retina corresponds to "convolution" in the image by a mask whose scale varies with image location.

When equation (8) is substituted into equation (7) and the second ∇ applied, the scalings cancel, so that in fact the resulting equation to be solved in terms of gradients calculated on the retina is simply

$$\nabla_w^2 \psi \;=\; \nabla_w T \nabla_w c(w) \tag{9}$$

As a result of the above considerations, the algorithm for lightness computation directly on the retinal image becomes:

(1) Compute $|\nabla c|$

(2) Compute $T\nabla c$ with
$$T \;=\; \begin{cases} 0 & \text{when } |\nabla c|/r \text{ is small;} \\ 1 & \text{otherwise.} \end{cases}$$

(3) Compute the forcing function $\sigma = \nabla T \nabla c$ by applying ∇ to the result of step 2.

(4) Solve the transplanted Poisson problem, equation (9), via the transform method.

4.2 Boundary conditions

Boundary conditions play a crucial role in determining the overall solution that will be obtained. For lightness computation in the image plane, it is preferable to constrain the normal derivative at the boundary rather than to fix values at the boundary. Blake constraints the derivative normal to the boundary for the recovered log-lightness to be the same as that of the log-intensity of the image after thresholding.

In order to calculate the gradient at the boundary, however, some assumptions need to be made about the derivative there. We assume that the image reflects about the boundary which leads to the Neumann condition of the normal derivative being zero, $\vec{n} \cdot \nabla c = 0$.

The boundary conditions on the retina are slightly different in that there are two types. The normal derivative at the periphery is constrained to zero as in the image-plane case and similarly for the interior boundary between the retina and the central foveal region. These boundaries correspond to $r = r_{min}$ and

$r = r_{max}$. The direction of the normal changes with location along the periphery, but conveniently it always coincides with the r-direction.

For $\theta = \theta_{min}$ and $\theta = \theta_{max}$ the situation is different in that the boundaries correspond to neighboring wedges on the retina. As such, the derivative can be computed by wrapping around from one boundary to the other via circular convolution with the derivative mask.

For the transform method of solving the Poisson equation, these different boundary conditions require different transforms. The circular convolution case corresponds directly to using a Fourier transform because the boundary conditions are periodic.

For the Neumann conditions in which the image is treated as reflecting about the boundary, this leads directly to the cosine transform, since the cosine transform can be derived directly from the Fourier transform of the image combined with its reflection. A rigorous development expressed in terms of the eigenvector decomposition of the convolution matrix can be found in Simchony *et al.* [13]

5 Thresholding

As one might expect, thresholding leads to problems. The need for thresholds is a well known disadvantage of the lightness method and we have not attempted to remedy the problem here. Varying resolution sensors exacerbate the problem, however.

While the above theoretical development shows how to transplant the Horn-Blake lightness computation to a varying resolution retina, it does not work well in that form and the failures primarily arise as problems with the threshold. The threshold is intended to separate illumination variations from albedo variations but these become muddied by the smoothing that occurs by the reduction in resolution towards the periphery. The problem stems from the fact that albedo and illumination changes are supposed to differ in their spatial frequency; however, the smoothing effect filters out much of the high frequency information making the separation of illumination and albedo more difficult.

The problem is compounded by the fact that gradient magnitudes should be scaled as in equation (8). In principle, the step edges in albedo between patches should lead to gradients of infinite magnitude but in practice they of course are finite. While infinite gradients would remain infinite, scaling the finite ones reduces them so much that a very low threshold must be used to prevent them being filtered out as illumination changes. This of course leads to problems with too much of the illumination change being retained in high resolution areas of the retina. Carrying over the threshold method directly from the image plane to the retinal plane thus does not work very well.

By experiment, what we found to work best was to threshold on the *uncorrected* gradient magnitude. This circumvents the problem of scaling the finite approximations of infinite gradients, but does not address the problem of smoothing at the periphery. Of course, it would be unreasonable not to expect the results to degrade as the resolution decreases.

Separating the large intensity gradients from the small ones amounts to an edge-detection process. The Canny operator [3] has been suggested for use in lightness computation [2], but in our tests it did not lead to any overall improvement in the lightness recovery. Of course, the Canny operator is gradient based and as such is not very different from the straight thresholding except for some Gaussian smoothing which helps overcome problems with noise.

6 Fovea

The hole at the center of the retina presents a serious problem because that is exactly where it should have the best resolution, not no resolution. As mentioned above, we have tried two approaches to the problem. The more straightforward, but less efficient, approach is to fill the hole with more retinal pixels. Unfortunately this leads to severe oversampling of the image so that many of these extra retinal pixels are redundant.

The more efficient approach is to cover the hole with a rectangular grid of pixels directly from the image. This means that the lightness computation must be solved in two parts—one for the complex-log periphery and a second for the central patch—which are then merged. For the central patch we computed the lightness using Neumann boundary conditions as Blake suggests. The solutions from patch and periphery both have an undetermined constant of integration, so to merge them requires adding a constant to one so that it matches the other. This match is done on the basis of a least-squares fit of the overlapping pixels.

Input	Time	Speed Up
128x65 periphery	1.5 s	n/a
64x33 periphery	0.4 s	n/a
33x33 fovea	0.3 s	n/a
128x65 periphery & fovea	1.8 s	47.8
64x33 periphery & fovea	0.7 s	122.9
128x129 retina	3.2 s	26.9
64x65 retina	0.8 s	107.5
5 separate fixations	9.0 s	9.56
512x512 cartesian	86.0 s	1.0

Table 1: **Comparative Timings**

7 Results

As can be seen from Figure 4 the recovered lightness has eliminated most of the illumination gradient. The two different methods of handling the fovea produce very similar results. As should be expected, the accuracy of the results declines towards the extreme periphery where the resolution becomes poor.

The cpu times (Sun 4/300 running at 16 MIPS, 2.6 MFlOPS) and speed up relative to computing lightness directly on the 512 by 512 image using the same algorithm are tabulated in Table 1. The timings do not include the time to convert between the cartesian and complex-log coordinate systems since it is assumed that any system actually using a space-variant sensor would rely entirely on the complex-log coordinate system alone.

For the retina with the cartesian patch over the fovea, the total number of retinal pixels is $(65 \times 128 + 33^2) = 9409$ which means there is a data reduction factor of 27.9 relative to the 512^2 image pixels. The amount of speed up therefore exceeds the amount of data reduction by almost a factor of two.

The algorithms have not been optimized for speed and so there are factors such as file input and output overheads that effect the timings in Table 1. The speed comparisons are also somewhat inequitable in that the retina only covers a circle of diameter 512 pixels, whereas the image covers a full 512×512 square. On the other hand, the part of the retina which is oversampled need not be except to make the retina's field of view match the width of the image. The FFT restriction to data with dimensions that are powers of two makes the match awkward.

Further speed increases can be achieved by using a retina of still coarser resolution. In tests with a 64×65 periphery, however, we found that the extreme blurring of the mondrian edges near the periphery made them difficult to detect. Figure 4e shows a sample result which is still pretty reasonable.

Figure 5 compares the lightness results for a composite of 5 fixations to lightness results computed directly on the original image. The fixations were merged by setting each image pixel to the result obtained from the nearest fixation. Using just 5 fixations yields a result relatively comparable to the lightness calculated on the entire original image. Of course, since the speed up is a factor of 48, it is possible to do 48 fixations before consuming the same amount of time as processing the image directly.

8 Conclusions

Lightness recovery is simply *one example* of a whole host of early vision algorithms that depend on the solution of Poisson equations. By exploiting the mapping properties of complex-log mapping architectures, it is quite simple to transplant these algorithms (we have begun a transplant of shape from shading) to the case of a varying resolution sensor architecture. In so doing, we obtain a roughly 50-fold decrease in running time, although at the expense of obtaining a result that has much lower peripheral resolution.

Transplanting existing algorithms to the space-variant case can be advantageous in comparison to algorithms that depend on the scale and rotation invariance of the complex-log mapping because the images do not have to be obtained under special constraints (see [7] page 359 for a discussion of these constraints).

For some algorithms, a sensor with conformal properties is not necessary. For example, Swain's [14] color indexing algorithm can be generalized to almost any type of sensor because it relies on calculating of color areas only. In general, though, a sensor layout based on a conformal mapping will have significant advantages for most vision applications. We conclude that while much attention has been paid to the rotation and scale invariance of complex-log mappings, perhaps not enough has been paid to the fact that most early vision algorithms can be implemented on them quite straightforwardly using the conformal transplant technique.

References

[1] A. Blake. Boundary conditions for lightness computation in Mondrian world. *Computer Vi-*

sion, Graphics, and Image Processing, 32:314–327, 1985.

[2] G. Brelstaff and A. Blake. Computing lightness. *Pattern Recognition Letters*, 5:129–138, 1987.

[3] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.

[4] P. Henrici. *Applied and Computational Complex Analysis*. Wiley, New York, 1974.

[5] B. K. P. Horn. Determining lightness from an image. *Computer Vision, Graphics, and Image Processing*, 3:277–299, 1974.

[6] B. K. P. Horn. *Robot Vision*. The MIT Press, 1986.

[7] R. Jain. Complex logarithm mapping and the focus of expansion. In *Proc. SIGGRAPH/SIGART Workshop Motion: Representation and Perception*, Toronto, April 1983.

[8] E. H. Land and J. J. McCann. Lightness and retinex theory. *J. Opt. Soc. Amer.*, 61:1–11, 1971.

[9] A. Moore, J. Allman, and R. M. Goodman. A real-time neural system for color constancy. *IEEE Trans. Neural Networks*, 2(2), March 1991.

[10] Retina Vision System 1 (RVS-1), 1991. Contact L. Hermans and G. Sandini.

[11] G. Sandini and V. Tagliasco. An antrhopomorphic retina-like structure for scene analysis. *Computer Graphics Image Processing*, 14:365–372, 1980.

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

[13] T. Simchony, R. Chellappa, and M. Shao. Direct analytical methods for solving Poisson equations in computer vision problems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:435–445, 1990.

[14] M. Swain and D. Ballard. Low resolution cues for guiding saccadic eye movements. In *Proc. 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, June 1992.

[15] Paul N. Swartztrauber. The methods of cyclic reduction, Fourier analysis and the FACR algorithm for the discrete solution of Posson's equation of a rectangle. *SIAM Review*, 19(3):490–501, 1977.

[16] J. van der Spiegel et al. A foveated retina-like sensor using CCD technology. In C. Mead and M. Ismail, editors, *Analog VLSI Implementations of Neural Networks*. Kluwer, Boston, 1989.

[17] C. Weiman and G. Chaikin. Logarithmic spiral grids for image processing and display. *Computer Graphics and Image Processing*, 11:197–226, 1979.