(37)

The Scale-space Formulation of Pyramid Data Structures

by

Robert Hummel
Courant Institute of Mathematical Sciences

August, 1986

New York University
251 Mercer Street
New York, NY 10012

To appear as a chapter in a book edited by Len Uhr,
published by Academic Press, 1986

This research was supported by Office of Naval Research Grant N00014-85-K-0077 and NSF grant DCR-8403300.

The Scale-space Formulation of Pyramid Data Structures

Robert Hummel

Courant Institute of Mathematical Sciences
New York University
251 Mercer Street, New York, NY 10012 USA

Abstract

Pyramid data structures for image processing are usually defined using discrete grids and discrete levels. It has proven useful to formulate pyramids in terms of continuous variables. When the level of the pyramid is changed to be a continuous variable, we talk of the resulting domain as "scale-space." When both the image domain and level are treated as continuous, the resulting pyramid structures are most naturally viewed in terms of partial differential equations governing their formation. This viewpoint allows one to generalize to new kinds of pyramid data structures, analyze their information content, and develop rational methods for treating borders and other problems in the discrete construction of pyramids.

1. Scale-space

Pyramid data structures for signal and image processing usually are implemented as a stack of discretely sampled data. In this chapter, we formulate pyramid structures in terms of a collection, indexed by a continuous variable, of functions of continuous-domain variables. Thus the pixels and levels in a pyramid are replaced by position and scale attributes. The continuous variable replacing the notion of the level of the pyramid will be called the *scale parameter*, and will be denoted by t in this chapter. The domain of the resulting continuous pyramid, given by the variables (x,y,t), is called *scale-space*. We will study the scale-space representation of functions of two variables, and discuss methods for implementing the decomposition and representations in scale-space.

Why should one want to build a continuous formulation of pyramid data structures? Since implementations will almost always be discrete, and since there exist well-studied methods for building useful pyramid structures, a continuous theory might seem superfluous. However, as we will see, the discrete constructions can be seen to be discrete approximations to a continuous formulation, and therefore can be understood in greater generality from the standpoint of a continuous theory. Further, we can sometimes answer questions about details and difficulties in the construction of pyramids by appealing to the continuous theory — here we will especially consider the problem of handling borders. Finally, by connecting to theories of partial differential equations and other continuous formulations, we can often make use of a large body of results to facilitate observations and propositions about pyramids, to formulate variants and to know how to extract information from them. A program of exploiting these relationships for pyramid data structures is far from complete.

In the simplest pyramid data structure, the *Gaussian pyramid*, each level represents a coarser resolution version of the original data. The base of the pyramid contains the original data at full resolution, and higher levels typically contain blurred and subsampled versions of the immediately lower level. The subsampling that is most commonly applied is to select every other pixel on every other row, for a reduction factor of two in each dimension. Thus if the base level is, say, 512 by 512 pixels in extent, the next level will be 256 by 256, the next will be 128 by 128, etc. Other sampling methodologies are possible, but it is always true that each level contains fewer points than the preceding level.

In a continuous formulation, a level is represented by a function of continuous variables, and so each level is qualitatively the same. However, the essential information content can vary between levels, so that a discrete representation might require fewer data points when the information content is low. Specifically, let us focus on image data, and suppose that $f(x,y)$ represents the image data given as a real-valued function of two continuous variables. The continuous analog of the pyramid data will be a function

$$u(x,y,t)\ , \qquad (x,y) \in \mathbb{R}^2\ , \qquad t \geq 0\ .$$

The domain of the function will be called *scale space*, and the parameter t, which is the continuous analog to the pyramid level, is the scale. The value $t = 0$ represents the base of the pyramid, and for the Gaussian pyramid yields the condition

$$u(x,y,0) = f(x,y)\ .$$

The number of levels in a discrete pyramid is always finite, whereas for the continuous version we may have an unbounded scale $0 \leq t < \infty$, or a bounded scale $0 \leq t \leq T$.

Scale-space can also be defined for bounded image domains. In this case, the image is defined on a domain $D \subseteq \mathbb{R}^2$, so that $f(x,y)$ is given for $(x,y) \in D$. Then the pyramid data $u(x,y,t)$ will similarly be defined for $(x,y) \in D$, $t \geq 0$, and so scale-space in this case can be regarded as a cylinder.

The notion of scale-space as applied to multiresolution signal and image analysis is due to Witkin [1]. Pyramid data structures were common before then, so the essential new idea was the conversion of discrete levels into a continuum of scales. Analyses of pyramids using scale-space notions have been reported in [2,3,4,5,6,7]. Especially relevant to analysis in scale-space are zero-crossing and level-crossing features, inspired partly by the Marr-Hildreth edge operator [8].

2. The Gaussian Pyramid

As indicated in the previous section, the Gaussian pyramid consists of levels that contain blurred and subsampled versions of the original, base level. That is, each pixel in an upper level of the pyramid has a value given by a weighted sum of values in the base level. Gaussian pyramid data structures are discussed in [9,10,11]. In nearly all Gaussian pyramid structures, and the basis for the name is that, the weights have the form of a Gaussian distribution. A precise formulation is easier and more natural using the scale-space variables.

For the scale-space Gaussian pyramid, we define

$$u(x,y,t) = \int\int_{\mathbb{R}^2} G(x',y',t) f(x-x',y-y')dx'dy' , \tag{2.1}$$

where

$$G(x,y,t) = \frac{1}{4\pi t} e^{-(x^2+y^2)/4t} . \tag{2.2}$$

For any fixed value of $t>0$, $G(x,y,t)$ is a Gaussian distribution centered at (0,0) with standard deviation $\sigma = \sqrt{2t}$. For $t=0$, $G(x,y,t)$ is (in a sense that can be made rigorous) a delta-function centered at (0,0). As a result, we have

$$u(x,y,0) = f(x,y) , \tag{2.3}$$

and for $t>0$, $u(\cdot,\cdot,t) = G(\cdot,\cdot,t) * f$, where by $*$ we denote convolution.

Suppose that we have a uniformly conducting infinite plane, and that at time $t=0$ a unit impulse of heat is placed at position $(x,y) = (0,0)$. Then, as time progresses, the impulse will diffuse to a symmetrical heat distribution centered at (0,0). If heat diffuses according to the Heat Equation

$$\frac{\partial u}{\partial t} = \Delta u , \tag{2.4}$$

where the Laplacian of u is

$$\Delta u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} ,$$

then after t units of time, the initial impulse will have diffused to the Gaussian distribution $G(x,y,t)$. In general, if the initial distribution of heat over the flat plate is given by $f(x,y)$ instead of having an impulse, then we can apply a superposition principle to deduce that after t units of time, the heat distribution will be $u(x,y,t)$, with u given by Equation (2.1).

Accordingly, the natural framework with which to study Gaussian pyramids on a continuous domain is by means of the Heat Equation. The Heat Equation is an example of a parabolic partial differential equation, and is a classical topic of study in mathematical analysis. Any standard text on partial differential equations will contain a treatment of this problem; examples are [12,13,14]. Questions such as existence, uniqueness, and dependence of the solution on the initial data are common topics in these treatments. These questions are not necessarily so clear-cut — for example, without certain growth conditions and regularity assumptions, uniqueness of a solution to (2.3) and (2.4) is not guaranteed.

Equations (2.3) and (2.4) constitute the initial value Heat Equation problem on the unbounded and unrestricted domain $\mathbb{R}^2$. Typically, the initial data $f(x,y)$ will be zero outside of a bounded domain (i.e., $f(x,y)$ will have compact support), or satisfy

$$f \in L^p(\mathbb{R}^2), \quad \text{some } p \geq 1. \tag{2.5}$$

In these cases, with the unrestricted domain $\mathbb{R}^2$, the initial data will spread by the diffusion process over the entire domain, so that $u(x,y,t) \to 0$ as $t \to \infty$. On a bounded domain $D \subseteq \mathbb{R}^2$, we can pose a boundary-value version of the Heat

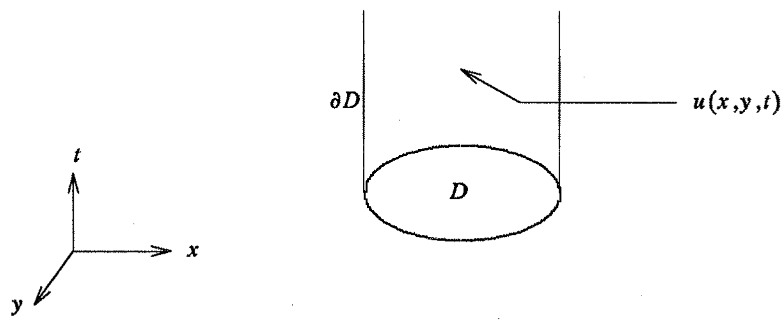

Figure 1. Scale-space for a bounded domain D.

Equation (see Figure 1). One such formulation is given by

$$\frac{\partial u}{\partial t} = \triangle u \quad \text{in } D \times (0,\infty) , \tag{2.6}$$

$$u(x,y,0) = f(x,y) , \quad (x,y) \in D ,$$

$$u(x,y,t) = f(x,y) , \quad (x,y) \in \partial D , \quad t \geq 0 .$$

Another formulation is applicable only if

$$\frac{\partial f}{\partial \nu}(x,y) = 0 , \quad (x,y) \in \partial D ,$$

where $\partial / \partial \nu$ denotes the normal derivative to the domain D. This formulation is given by

$$\frac{\partial u}{\partial t} = \triangle u \quad \text{in } D \times (0,\infty) , \tag{2.7}$$

$$u(x,y,0) = f(x,y) , \quad (x,y) \in D ,$$

$$\frac{\partial u}{\partial \nu}(x,y,t) = 0 , \quad (x,y) \in \partial D , \quad t \geq 0 .$$

Other formulations are possible. Each alternative boundary value formulation has to be studied separately in terms of existence, uniqueness, and dependence of the solutions on the boundary data. However, a well-posed boundary value problem can generally be transformed into a discrete pyramid construction procedure. We convert the three formulations above into discrete constructive methods in Section 5.1 .

The boundary formulation (2.6) given above, where the data is specified on the sides of the cylinder, is an example of a Dirichlet-type boundary value problem. The next formulation (Equations (2.7)), is closer to a Neumann-type problem, since the normal derivative data is specified on the sides. Accordingly, we have three boundary formulations: Equation (2.5), corresponding to the lack of boundaries and embedding in an infinite plane, (2.6) corresponding to fixed boundaries and physically equivalent to diffusion of heat on a plate in contact with heat reservoirs on the boundary clamping the temperature, and (2.7) corresponding to adiabatic diffusion of heat in an insulated plate.

3. The Laplacian Pyramid

The Laplacian pyramid data structure has proven, in many respects, to be more useful than the Gaussian pyramid. In discrete settings, the Laplacian pyramid is obtained by taking a difference between adjacent layers in the Gaussian pyramid. Of course, since the levels have different sizes, they must first be made commensurate. In the Burt version of the Laplacian pyramid, this is done by expanding the smaller level by an interpolation procedure (see Figure 2). The result is a pyramid structure in which each level contains something approximating a difference-of-Gaussian filter of the original data. Accordingly, each level can be regarded as a band-pass filter on the data; further, the original data can be reconstructed by essentially adding together all levels, from small levels to larger, expanding the result at each stage.

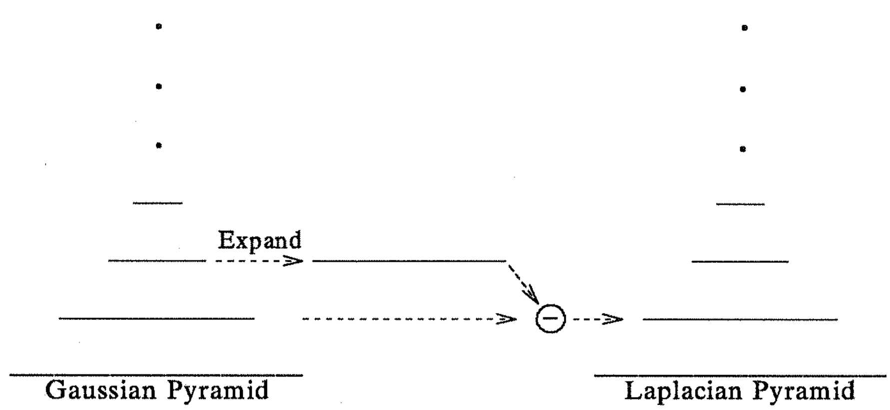

Figure 2. The Burt construction of the Laplacian pyramid.

The key to the continuous analog is the substitution of a difference quotient for a difference of levels. Thus if $u(x,y,t)$ represents a Gaussian pyramid, and $u(x,y,t_1) - u(x,y,t_2)$ is therefore a difference of levels, the Laplacian pyramid will be formed from

$$v(x,y,t) = \lim_{t_2 \to t_1} \frac{u(x,y,t_2) - u(x,y,t_1)}{t_2 - t_1} .$$

Thus each level of the Laplacian pyramid is a scaled infinitesimal difference of levels of the continuous Gaussian pyramid. More precisely, $v(x,y,t) = \partial u(x,y,t)/\partial t$.

Recall, however, that the Gaussian pyramid function $u(x,y,t)$ is a solution to the Heat Equation. Thus

$$v(x,y,t) = \frac{\partial u}{\partial t}(x,y,t) = \triangle u(x,y,t) .$$

So the Laplacian pyramid contains data that is simply the Laplacian of values in the Gaussian pyramid. Moreover, u is obtained by convolution of the original data f with the Heat kernel K: $u = K*f$. We may use properties of convolutions to write $v(x,y,t)$ in three forms:

$$v(x,y,t) = \triangle u(x,y,t) = \triangle K * f = K * \triangle f .$$

Thus the Laplacian pyramid data can be formed from the Laplacian of the Gaussian pyramid, or by filtering the original data f with the Laplacian-of-Gaussian kernel $\triangle K$, or by filtering the Laplacian of the original data by a Gaussian.

The last equation shows that v itself is a solution to the Heat Equation:

$$\frac{\partial v}{\partial t} = \triangle v ,$$

with initial data $\triangle f$. This implies a rapid method of constructing v, by blurring $\triangle f$ data. However, in practice, blurring $\triangle f$ can lead to numerical precision difficulties, and so one of the other forms is generally used for construction of v. Use of the Laplacian of the Gaussian (the "Mexican Hat" or "Sombrero" operator), $\triangle K$, is possible (see Figure 3), and made much less expensive by methods of Huertas and Medioni [15]. However, much simpler is to simply evaluate the Laplacian of the data in the Gaussian pyramid, or, equivalently, difference adjacent levels. Note, however, that high numerical precision (floating point or large integer representation) is needed for construction of the Gaussian data in order to accurately compute v.

We can also see, from the definition of v, the reconstructibility of f. We have

$$f(x,y) = -\int_0^T v(x,y,t)dt + u(x,y,T) ,$$

where either $T=\infty$, or T is the height of the finite scale-space cylinder. The equation follows from the fact that $v = \partial u/\partial t$, and the fundamental theorem of calculus. In the case $T=\infty$, the function $u(x,y,t)$ means $\lim_{t \to \infty} u(x,y,t)$, and is generally known. For the case $f \in L^P$ and domain $\mathbb{R}^2$ (formulation 2.5), we obtain $u(x,y,T) \equiv 0$. Dirichlet boundary conditions (formulation 2.6) lead to $u(x,y,T)$ a solution to the boundary value Laplacian equation:

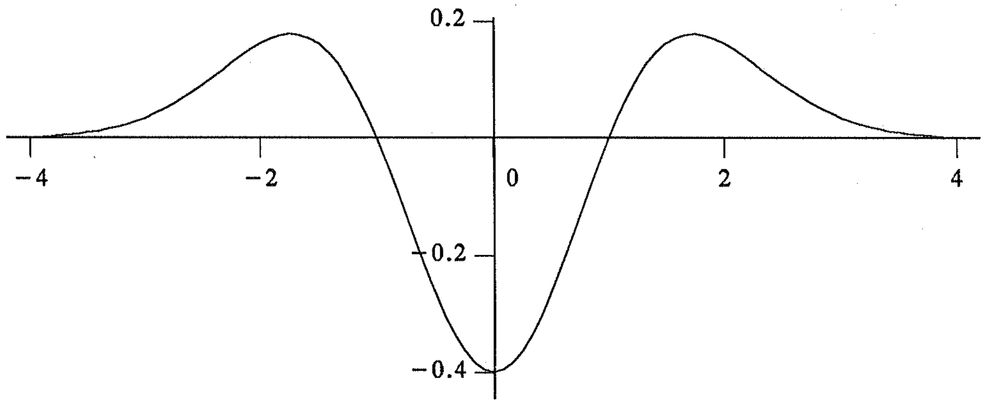

Figure 3. The Laplacian (second derivative) of a Gaussian in one dimension.

$$\Delta u(x,y,T) = 0, \quad (x,y) \in D,$$
$$u(x,y,T) = f(x,y), \quad (x,y) \in \partial D .$$

Finally, Neumann boundary conditions (formulation 2.7) lead to $u(x,y,T) \equiv c$, where c is the mean value of $f(x,y)$. In all cases, we see that $f(x,y)$ can be reconstructed from v by adding together all levels, and then adding in a simple function. Thus as long as the Gaussian pyramid data $v(x,y,t)$ is supplemented with the data $u(x,y,T)$, (which may be zero, harmonic, or a constant), the information supplies a complete representation of the original data $f(x,y)$.

4. Zero-crossings

The Marr-Hildreth edge operator is defined as the zero-crossings of the filtered image data, where the filter used is the Laplacian of a Gaussian [8]. On a pyramid data structure, the zero-crossings at each level of the Laplacian pyramid reveal structure characteristic of the image at a specific scale of resolution. Strong, clear edges in the image data will frequently show zero-crossings at many levels near the relevant locations, but there will also be zero-crossings at other locations.

In scale-space, the zero-crossings are the common borders between regions where the data is positive and regions where the data is negative. This definition differs only slightly from the zero set, since the zero set might contain isolated zeros and other kinds of pathologies. If we restrict scale-space to one level by fixing a value for t, the portions of the zero-crossings within that level give the Marr-Hildreth edges at that scale, using a Laplacian-of-Gaussian filter. When implemented discretely, the levels are more typically a difference-of-Gaussian representation of the data. But in continuous variables, we can define the zero-crossings mathematically as

$$Z = \partial\{ (x,y,t) \mid v(x,y,t) > 0\} \cap \partial\{(x,y,t) \mid v(x,y,t) < 0\} .$$

The interest in zero-crossings is heightened by an analysis of their evolution as the scale varies. Because $v(x,y,t)$ is a continuous function for $t>0$, the zero-crossings form sheets in scale-space that cut through the scales. If a zero-crossing sheet persists for a large range in scale, then there is very likely an associated prominent feature in the image data, such as a strong edge. Even though the zero-crossings are part of the zero set of a smooth function, they can have cusps, irregularities, and all kinds of pathological properties. However, since $v(x,y,t)$ is an analytic function, these pathologies will be isolated.

Zero-crossing sheets enjoy a property that we will call the "evolution property." This property says, informally, that zero-crossing sheets are never created at intermediate values of scale t, but rather evolve and can only disappear as t increases. Thus zero-crossing contours at $t=0$ lead to sheets that evolve as t increases, but no new sheets appear for $t>0$ (see Figure 4). In particular, sheets are nested one within another. Thus the entire collection of zero-crossing surfaces form a structure that can be extracted and used as a structural description of the original image. This property, which holds for level-crossings as well as zero-crossings, was noticed by Witkin in numerical experiments, and proven by various authors under

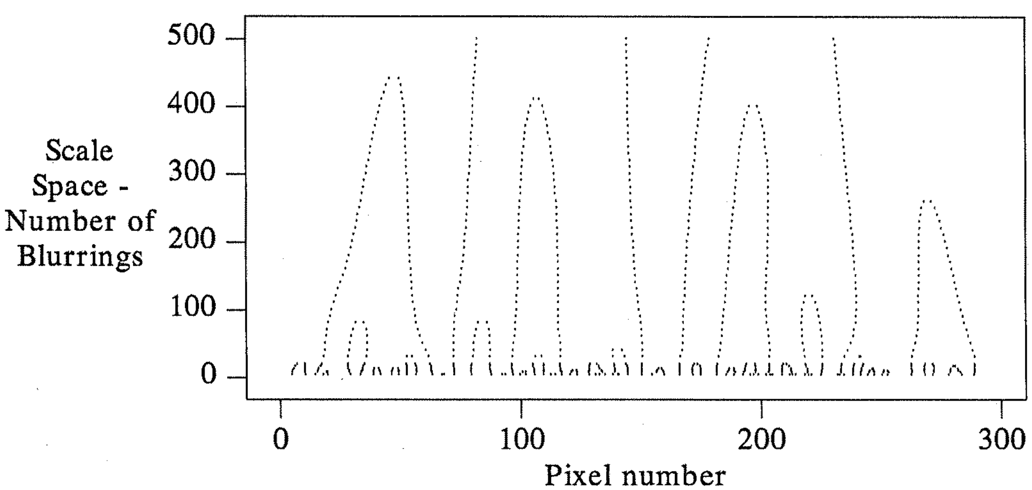

Figure 4. An example of zero-crossings in scale space. The initial data is a scan line of a real image, and the zero-crossings are formed from the Laplacian of the Gaussian. Above 20 blurring steps, zero-crossings are shown only on every tenth row. Note that zero-crossings are never created at intermediate values of t.

assumptions of regularity of the zero-crossing surfaces.

Haralick suggested [16] that the ability of zero-crossings to localize edges could be improved by using a second directional derivative in the direction of the gradient, instead of using the Laplacian operator. That is, instead of using $v=\Delta u$, we set

$$v = \frac{\left(\frac{\partial u}{\partial x}\right)^2 \frac{\partial^2 u}{\partial x^2} + 2\left(\frac{\partial u}{\partial x}\right)\left(\frac{\partial u}{\partial y}\right)\frac{\partial^2 u}{\partial x \partial y} + \left(\frac{\partial u}{\partial y}\right)^2 \frac{\partial^2 u}{\partial y^2}}{|\nabla u|^2} .$$

It is not hard to show that the right hand side represents a second directional derivative $D_\theta^2 u$ of u, taken in the direction θ that is the same as the direction ∇u. Haralick uses the "facet model" to obtain u, instead of Gaussian convolution. Preliminary experiments with zero-crossings of this operator and related operators suggest that edges are in fact more accurately found by this method.

However, introducing an operator other than the Laplacian opens up a host of new issues. Recall that in the unrestricted domain $\mathbb{R}^2$, $u(x,y,t)$ is obtained by Gaussian convolution against the image data $f(x,y)$, and that Gaussian convolution arises due to the presence of the Heat Equation. We can replace the Heat Equation with the following nonlinear partial differential equation for u to form a new kind of pyramid:

$$\frac{\partial u}{\partial t} = \frac{\left(\frac{\partial u}{\partial x}\right)^2 \frac{\partial^2 u}{\partial x^2} + 2\left(\frac{\partial u}{\partial x}\right)\left(\frac{\partial u}{\partial y}\right)\frac{\partial^2 u}{\partial x \partial y} + \left(\frac{\partial u}{\partial y}\right)^2 \frac{\partial^2 u}{\partial y^2}}{|\nabla u|^2} .$$

If u is constructed in this manner, using some appropriate version of the boundary conditions, then we would set

$$v = \frac{\partial u}{\partial t}$$

to obtain $v(x,y,t)$ as a scaled second directional derivative function. In this case, as well as when u is the Gaussian pyramid data and v is the second directional derivative of u, the evolution property is no longer guaranteed to hold.

5. Some results using scale-space

We can use scale-space notions to analyze properties of pyramid data structures, and to suggest new construction methods. We focus on three areas here. First, we discuss methods for implementing the Dirichlet and Neumann boundary formulations for construction of the Gaussian pyramid on a bounded domain. Next, we discuss theorems involving zero-crossings. Finally, we mention possible alternative features in scale space that might be used for image representation.

5.1. Boundaries

We will discuss methods for discretizing the boundary formulations (2.6) and (2.7). The discussion will be mostly confined to one space dimension, although the results extend easily. More details can be found in [17]. In a discrete form, we are given data $f(i)$, $i=-N, \ldots, N$, and we wish to construct

$$u(i,k),\ i= -N, \ldots, N,\ k \geq 0\ ,$$

with

$$u(i,0) = f(i)\ ,$$

and u a solution to the "discretized Heat Equation:"

$$u(i,k+1) - u(i,k) = \frac{1}{4}u(i-1,k) - \frac{1}{2}u(i,k) + \frac{1}{4}u(i+1,k)\ .$$

Disregarding borders, the construction is clear. First we set $u(i,0)=f(i)$, as required, and then recursively for $k=1,2, \cdots$, define

$$u(i,k+1) = \frac{1}{4}u(i-1,k) + \frac{1}{2}u(i,k) + \frac{1}{4}u(i-1,k)$$

for $i= -N+1, \ldots, N-1$.

However, special treatment is needed on the borders. To implement Dirichlet boundary conditions, we set

$$u(N,k+1) = u(N,k)$$

$$u(-N,k+1) = u(-N,k)\ .$$

The result is that the boundary data at $i=-N$ and $i=N$ remains fixed at $f(-N)$ and $f(N)$ respectively.

For Neumann-like boundary conditions, we set

$$u(N,k+1) = \frac{1}{4}u(N-1,k) + \frac{3}{4}u(N,k)\ ,$$

$$\text{and } u(-N,k+1) = \frac{3}{4}u(-N,k) + \frac{1}{4}u(-N+1,k)\ .$$

These formulas arise if we imagine $u(-N-1,k)=u(-N,k)$ and $u(N+1,k)=u(N,k)$. The advantage of the Neumann-type conditions is that the mean value

$$\frac{1}{2N+1}\sum_{i=-N}^{N} u(i,k)$$

will remain constant in k. Thus as $k\rightarrow\infty$, $u(i,k)$ will converge to a constant value, which is the average value of $f(i)$.

In higher dimensions, Neumann-type boundary conditions are slightly tricky for non-convex domains. The best bet for image data is to apply the one-dimensional procedure first on the rows, and then on the columns of the result.

Having constructed the discrete Gaussian pyramid data, Laplacian pyramid data can be obtained from $v(i,k)=u(i,k+1)-u(i,k)$. Note that if Dirichlet-type boundary conditions are used with fixed boundary data then the Laplacian data will be zero on the borders.

5.2. Results about zero-crossings

It turns out that the evolution property for level-crossings of the Gaussian pyramid data in scale-space, stated in Section 4, is completely equivalent to the maximum principle for parabolic partial differential equations [7]. The maximum principle applies for solutions to the Heat Equation, and states that inside any bounded cylinder generated by a finite subdomain and finite interval in t, the maximum will occur either on the bottom or the sides of the cylinder. Consequently, we know that the evolution property holds for each of the boundary formulations stated in Section 2, namely (2.5), (2.6), and (2.7).

It is easy to observe that the edges in an image carry a lot of information about the image. An accurate edge image will generally allow identification of the objects, and an artist can reconstruct an approximation to the original image by coloring in regions surrounded by edges in the edge image. Accordingly, there is some reason to hope that the edge data, especially given edge data at a variety of scales, completely represents the original data. The scale-space formulation of this hypothesis would assume that the given data is the location of all zero-crossings in all of scale-space. We can then ask whether the original data, $f(x,y)$, is completely characterized, or characterized to within a class of transformations, by that data. If the original data is completely characterized, we can further ask whether reconstruction is in practice possible.

The answer to these questions are not known. There are some hints, although it is entirely possible that the hints are misleading. For example, if the original data $f(x,y)$ is a polynomial in x and y, then the Laplacian pyramid data $v(x,y,t)$ will be a polynomial in x, y, and t, and the zero-crossings are part of the *real analytic varieties* of v as studied in algebraic geometry. It can be shown that under certain assumptions, the real analytic varieties determine the polynomial, and thus can determine the initial data [18,3]. However, the result for polynomial data implies nothing for nonpolynomial data. Even though a general function can be closely approximated on a bounded domain by a polynomial, the fact that the representation is unstable means that approximations to the representation can lead to arbitrarily large inaccuracies in the reconstruction.

As an example of another result, suppose that the information about the locations of the zero-crossings in scale space are supplemented with information about the magnitude of the gradient of the Laplacian pyramid data at the zero-crossings. That is, we are given the zero-crossing set Z, and the information $|\nabla v(x,y,t)|$, for $(x,y,t) \in Z$. Then theoretically reconstruction of $f(x,y)$ is possible, at least in regions enclosed by bounded zero-crossing surfaces in scale space (see Figure 5). The reconstruction method requires that data at a level $t=T$ above the top of the zero-crossing contour be computed, and deblurred to give data on the initial surface $t=0$. Details of this method are given in [7], although it should be noted that the deblurring process is necessarily ill-conditioned ([19]).

We can also attempt to reconstruct the data $f(x,y)$ given only the zero-crossings (and just a little bit more) of $v(x,y,t)$ by making use of the sgn-function, defined by

$$\operatorname{sgn}(x) = \begin{cases} -1 & x<0 \\ 0 & x=0 \\ 0 & x>0 \end{cases} .$$

Knowledge of the location of the zero-crossings is essentially equivalent to knowledge of

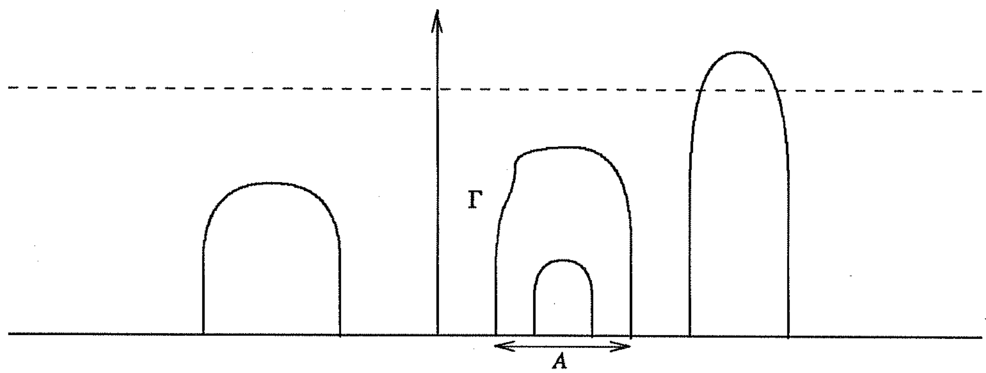

Figure 5. Reconstruction of data on segment A is theoretically possible given the gradient data on the zero-crossing curve Γ. The reconstruction requires deblurring data that is computed along the dashed line.

$$s(x,y,t) = \operatorname{sgn}(v(x,y,t)) \ .$$

Then by defining a differentiable approximation to the sgn-function, say $\phi_c(x) = \arctan(x/c)$, we can seek to find a function $\tilde{f}$ minimizing

$$\int\int\int \left(\phi_c(\triangle G(\cdot,\cdot,t) * \tilde{f}) - s(x,y,t)\right)^2 dxdydt$$

Use of a related method in one-dimension has been reported with good results, and even applied to rows and columns to reconstruct image data [20]. However, preliminary experiments by the author on full two-dimensional reconstructions seem to imply that the zero-crossings contain much information about typical images, but not enought for accurate, sharp reconstruction.

5.3. Alternative representation

The Laplacian pyramid data and the zero-crossings of the Laplacian data in scale-space are two possible bases for representations of image data. The resulting discretizations, when formed into pyramid data structures, have to be evaluated in terms of their utility as well as their (mathematical) information content. There are many other possible representations, and we briefly mention a couple of possibilities.

As noted above (Section 5.2), the data $s(x,y,t) = \operatorname{sgn}(v(x,y,t))$ may be more useful than the zero-crossings alone. However, the data $\phi_c(v(x,y,t))$, while much more complex and equivalent to the full Laplacian data $v(x,y,t)$, might have favorable properties when quantized and sampled. Similarly, Zucker and Hummel

[21] suggest the data

$$G(\cdot,\cdot,t) * \phi_+(\triangle f(x,y)) ,$$
$$G(\cdot,\cdot,t) * \phi_-(\triangle f(x,y)) ,$$

where ϕ_+ and ϕ_- are approximate positive-part and negative-part functions with saturation (Figure 6).

As an alternative to zero-crossings, Koenderink suggests level-crossings that surround blobs [4], defined as follows. First, the Gaussian pyramid data $u(x,y,t)$ is constructed from initial data $f(x,y)$. It will be observed that relative maxima in $f(x,y)$ give rise to relative maxima (in x and y) of $u(x,y,t)$ as t increases, forming curves in (x,y,t) space. Similarly, relative minima track to relative minima as t increases. These curves terminate at intermediate values of t, where the tracking of a relative extremum terminates (the curve will coalesce with the curve for a saddle point), and there are no corresponding extrema for larger values of t. Denote such a terminal position by (x_0,y_0,t_0) and form the level-crossings in scale-space of points having the value $v(x_0,y_0,t_0)$, and consider the component containing the point (x_0,y_0,t_0). The collection of all such level-crossing components will form tubes surrounding blobs (both bright and dark) in the image, and forms the representation. This representation is complete, may (in practice) depend continuously on the image data (at least, better than do the zero-crossings), but is not likely to permit stable reconstructions. Nonetheless, a representation based on

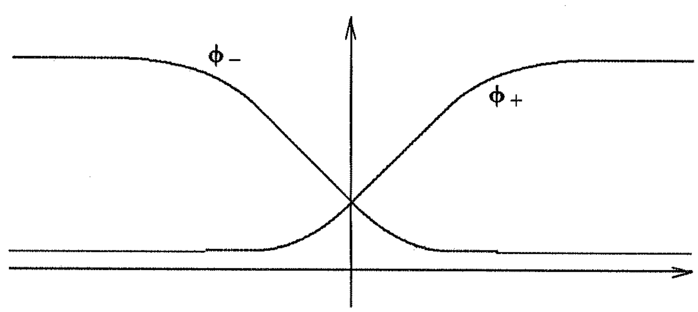

Figure 6. Approximate positive and negative part functions, with saturation for large magnitude inputs. These curves are supposed to model simple kinds of neural response functions, where the rest state of gives a low output response, one kind of input causes inhibition of the normal rest response, and a different kind of input causes excitation through a linear region, before a saturation level is reached.

tracking blobs in scale-space makes good intuitive sense.

6. Summary

Pyramid data structures can be analyzed in an analytic formulation based on notions of scale-space and partial differential operators. We've seen that the Gaussian pyramid can be viewed as a method of solving the Heat Equation using the image intensity values for the initial data. The Laplacian pyramid can be viewed as a partial derivative, in the scale parameter, of the Gaussian pyramid data, from the standpoint of this continuous formulation. We are also able to use the continuous formulation to define and study zero-crossings in scale-space, particularly of the Laplacian pyramid data.

We've given three examples of how the continuous formulation assists in our understanding of pyramid data structures. The first example concerned border affects, and we discussed three ways of handling borders when constructing pyramids of images defined on a bounded domain. Each of these methods is motivated by a different formulation of the Heat Equation problem: namely, (1) embedding in the infinite domain; (2) fixing the border values in a Dirichlet problem; and (3), setting boundary normals to zero on the cylinder sides, in a Neumann-type boundary formulation.

In the second example of assistance yielded by the continuous treatment of pyramids, we remarked on some of the theorems and representations possible based on zero-crossings of the Laplacian pyramid data. In particular, we are concerned with the information content in the zero-crossings, and the reconstructibility of the initial data given zero-crossing information.

Finally, we looked at some of the alternate representations that have been posed in continuous scale-space. These alternate forms might lead to useful discrete pyramid structures with different construction procedures than commonly used.

Acknowledgements

This research was supported by Office of Naval Research Grant N00014-85-K-0077 and NSF grant DCR-8403300. Shmuel Peleg gave assistance in many discussions.

References

[1] Witkin, A., "Scale space filtering," *Proceedings of the 8th International Joint Conference on Artificial Intelligence*, p. 1019 (1983).

[2] Yuille, A. L. and T. A. Poggio, "Scaling theorems for zero crossings," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **8**, pp. 15-25 (1986).

[3] Yuille, A. L. and T. Poggio, "Fingerprints theorems for zero crossings," *J. Optical Society of America* **2**, pp. 683-692 (1985).

[4] Koenderink, Jan J., "The structure of images," *Biological Cybernetics* **50**, pp. 363-370 (1984).

[5] Babaud, J., A. P. Witkin, M. Baudin, and R. O. Duda, "Uniqueness of the Gaussian kernel for scale-space filtering," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **8**, pp. 26-33 (1986).

[6] Hummel, Robert A. and Basilis C. Gidas, "Zero crossings and the heat equation," NYU Robotics Report 18 (March, 1984).

[7] Hummel, Robert, "Representations based on zero-crossings in scale-space," *Proceedings of the IEEE Computer Vision and Pattern Recognition Conference*, pp. 204-209 (June, 1986).

[8] Marr, D. and E. Hildreth, "Theory of edge detection," *Proceedings Royal Society London (B)*, p. 187 (1980).

[9] Burt, P. and T. Adelson, "The laplacian pyramid as a compact image code," *IEEE Trans. on Communications*, p. 532 (1983).

[10] Crowley, J., "A representation for visual information," CMU Robotics Institute, Ph.D. Thesis (1982).

[11] Rosenfeld, A., *Multiresolution Image Processing and Applications*, Springer Verlag (1984).

[12] Bers, L., F. John, and M. Schechter, *Partial Differential Equations*, American Mathematical Society, Providence, RI (1964).

[13] John, F., *Partial Differential Equations*, Springer-Verlag, New York (1975).

[14] Widder, D., *The Heat Equation*, Academic Press, New York (1975).

[15] Huertas, A. and G. Medioni, "Detection of intensity changes with subpixel accuracy using Laplacian-of-Gaussian masks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (1986). To appear.

[16] Haralick, R., "Digital step edges for zero crossings of second directional derivatives," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, p. 58 (1984).

[17] Hummel, Robert A. and David Lowe, "Computing Gaussian Blur," *International Conference on Pattern Recognition*, (October, 1986).

[18] Mumford,, Personal communication; A copy of Mumford's proof is available from the author.

[19] Hummel, Robert A. and B. Kimia, "Deblurring gaussian blur," *Computer Vision, Graphics, and Image Processing*, (1985). To Appear. Also appears in *IEEE Conference on Computer Vision and Pattern Recognition*, San Francisco, 1985

[20] Zeevi, Y. Y. and D. Rotem, "Image reconstruction from zero crossings," *IEEE ASSP Magazine*, (1986). To appear.

[21] Zucker, Steven W. and Robert A. Hummel, "Receptive fields and the reconstruction of visual information," *Human Neurobiology* **5**, pp. 121-128 (1986).