A Probabilistic Framework for Edge Detection and Scale Selection

David H. Marimont
Image Understanding Area
Xerox Palo Alto Research Center
Palo Alto, Calif. 94304
marimont@parc.xerox.com

Yossi Rubner
Department of Computer Science
Stanford University
Stanford, Calif. 94305
rubner@cs.stanford.edu

Abstract

We devise a statistical framework for edge detection by performing a statistical analysis of zero crossings of the second derivative of an image. This analysis enables us to estimate at each pixel of an image the probability that an edge passes through the pixel. We present a statistical analysis of the Lindeberg operators that we use to compute image derivatives. We also introduce a confidence probability that tells us how reliable the edge probability is, given the image's noise level and the operator's scale. Combining the edge and confidence probabilities leads to a probabilistic scale selection algorithm. We present the results of experiments on natural images.

1 Introduction

A common approach to edge detection is to smooth the image and find maxima of the gradient magnitude in the gradient direction [9, 2, 4]. The maxima of the gradient magnitude are either found by explicit comparison with neighboring pixels, or equivalently, by computing the second derivative in the direction of the gradient and looking for zero crossings. Since image noise causes zero crossings where no edges exist, a threshold on the gradient magnitude is often used to eliminate the spurious zero crossings.

It is difficult to set a threshold for the gradient magnitude in a principled way, even if the threshold is based on a measure of image noise [2, 19]. The gradient magnitude can be large in regions of smooth shading where there are no edges – often larger than it is at low-contrast edges. It is sensitive to changes in illumination and difficult to estimate accurately [8]. As a consequence, detecting edges by thresholding the gradient magnitude leads to many errors.

Here we show that a statistical analysis of zero crossings that takes image noise into account suffices to detect and locate edges accurately. Our analysis is based on probabilistic models of both sensor noise and the responses of the operators that we use to detect edges. The algorithm that results is controlled not with *ad hoc* parameters but in terms of probabilities of error. Moreover, our analysis suggests that we should largely dispense with the use of the gradient

magnitude. Instead, we devise statistical tests to detect zero crossings that are more correct and more reliable than those that use the gradient magnitude. Our statistical approach enables us to estimate at each pixel the probability that an edge passes through the pixel.

An edge probability can be low for two reasons: either it is very unlikely that there is a edge, or the image measurements used to estimate the edge probability are so noisy that the estimate of the edge probability is not reliable. Distinguishing between these two cases is important: in the latter case, increasing the scale of the operator can result in a higher edge probability. We introduce a confidence probability that is high when the image measurements used to estimate the edge probability have a high signal-to-noise ratio and low when they do not. The confidence probability also tells us when taking higher derivatives of an image to compute the edge probabilities will not make the results unstable because of noise amplification.

We use the linear derivative operators proposed by Lindeberg [10] to smooth and differentiate the image. As Lindeberg shows, these operators are more accurate than sampled Gaussian derivatives, particularly at finer scales, and preserve scale-space properties. To use these operators in our statistical framework, we provide an analysis of their statistical properties.

Smoothing introduces the problem known as *scale selection*: choosing the scale of the smoothing operator. Recently Lindeberg [11] and Elder [5] have addressed this problem in edge detection. Edges in natural images range from sharp to diffuse, a property Elder calls the edge's *blur scale*. Elder argues that the optimal operator scale depends on blur scale, edge contrast, and sensor noise. He suggests increasing the scale of the derivative operators until the detected edges are reliable in a statistical sense. He calls this the *minimum reliable scale*.

The contribution of our work is the statistical framework for the edge and confidence probabilities. Like Elder, when confidence is low, we increase the scale of the operator until the confidence is high enough for us to accept the edge probability as reliable. Our framework is based

on a different and more thorough statistical analysis than Elder's.

The algorithm we propose here produces two *probability maps,* edge and confidence probabilities for every pixel in the image. Edges are not produced directly, although we show how to use the information we provide to detect edges. Our results provide the basis for improved edge detection, as well as scale selection, edge linking, and other low-level vision algorithms that depend on accurate statistical information about the presence of edges.

The outline of the paper is as follows. Section 2 reviews previous work. Section 3 develops the statistical framework for our approach in the one-dimensional case. Section 4 extends this framework to the two-dimensional case and presents experimental results on natural images. Section 5 presents an algorithm for detecting edges using edge probability maps. Section 6 applies our framework to the problem of scale selection. Section 7 summarizes our approach and results and discusses future work.

2 Previous Work

Haralick [9] was the first to use zero crossings of the second derivative in the direction of the gradient to detect edges. He used a statistical analysis to threshold the gradient magnitude based on an estimate of image noise, and proposed confidence intervals for the gradient direction. He did not, however, suggest a statistical test based on zero crossings themselves. Many other authors have proposed approaches to thresholding based on the gradient magnitude (*e.g.,* Voorhees and Poggio [19]). For a recent review of this literature, see Rosin [18].

As mentioned above, both Lindeberg and Elder have proposed solutions to the problem of scale selection. Lindeberg's approach does not use any statistical notions [11]. Instead, he detects edges by thresholding a measure of edge strength integrated along a curve of edge points. Our main interest in his work is the operators he uses for smoothing and differentiation. We discuss this further in Section 3.2.

Elder's statistical approach to scale selection [5] inspired many of our ideas. The edges he detects must meet two probabilistic criteria: the gradient magnitude must be significantly greater than zero, and there must a significant zero crossing in the second derivative in the direction of the gradient. He tries to find three minimum reliable scales for each pixel, one for the gradient magnitude and two for the second derivative (corresponding to its values to either side of a pixel in the gradient direction). He finds the three scales when the two probabilistic criteria are met, and only then is an edge declared to pass through the pixel.

Our analysis differs from Elder's in the following ways. First, we do not test for the gradient magnitude for the reasons mentioned earlier. Instead, our confidence probability uses a combination of second and third derivatives,
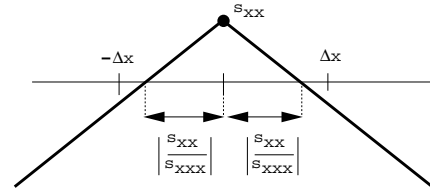

Figure 1: The zero-crossing condition: $|s_{xx}/s_{xxx}| < \Delta x$.

as explained in Section 3.5. Second, we perform a more accurate analysis of the statistical properties of the second derivative. This enables us to estimate the *probability* of an edge where Elder makes a hard decision about the existence of an edge. Third, his notion of minimum reliable scale applies to the scales of the operators used for the first and two second derivatives used in his test for an edge. Thus an edge may have three different minimum reliable scales associated with it. In our analysis, the minimum reliable scale is the *single* scale of smoothing used to compute all the derivatives. Thus we associate the minimum reliable scale with the edge itself, not with the different derivatives involved in its detection.

3 The one-dimensional case

Here we develop our theory of edge probabilities for one-dimensional signals. We show how to compute the probability that an edge passes through a pixel, and a confidence probability in the measurements underlying the edge probability.

3.1 A model for edges based on zero crossings

We model the sensor noise in the signal as a stationary, additive, zero-mean white-noise process with standard deviation σ_n. The observed signal is thus $r(x) = s(x) + n(x)$, where $s(x)$ is the "true" signal, and $n(x)$ is the noise. Usually we will drop the positional parameter x and simply refer to r, s and n.

We define an edge as a location where s_{xx}, the second derivative of the signal, has a zero crossing. To decide whether the pixel centered at x has a zero crossing in the second derivative, we use the condition

$$|s_{xx}/s_{xxx}| < \Delta x \, , \qquad (1)$$

where Δx is half the pixel width. Condition 1 is true when the linear approximation to the second derivative at x, the line through s_{xx} with slope s_{xxx}, crosses the x-axis within Δx of x (*i.e.,* within the pixel), as depicted in Fig. 1. To avoid "phantom edges" (see Clark [3]), we also require that $s_x s_{xxx} < 0$. While it is commonly thought impractical to use the third derivative of a noisy signal because of noise amplification, we use the confidence probability that we propose in Section 3.5 to determine when the value of third derivative is too low to be reliable.

We note that while the standard deviation of the noise σ_n is an input parameter to all of the algorithms developed below, it is not trivial to estimate [2, 19].

3.2 Signal derivatives

To use Condition 1 to detect zero crossings, we compute the second and third derivatives of the signal. We use the Lindeberg smoothing and derivative operators [10], which have scale-space properties that hold in the discrete domain, just as continuous Gaussian derivatives do in the continuous domain. Like derivatives of Gaussians in the continuous case, they introduce no new zero crossings of the second derivative as the scale of smoothing increases. Sampled Gaussian derivatives do not have this property.

Another important advantage of the Lindeberg operators is that smoothing and differentiation commute. This is important computationally, since it makes it possible to smooth a signal once, which is expensive, and then cheaply differentiate multiple times by applying derivative operators with small support to the smoothed signal. This is not true of sampled Gaussian derivatives. Moreover, Lindeberg shows that at small scales, sampled Gaussian second derivatives sometimes give the wrong sign, which makes them particularly unsuitable for detecting zero crossings.

Lindeberg's smoothing operator has the form

$$\mathcal{T}(x; \sigma^2) = e^{-\sigma^2} I_x(\sigma^2) ,$$

where σ is the scale parameter, and I_x is the modified Bessel function of integer order x. To simplify the notation, we define $T_a(x) \equiv \mathcal{T}(x; a\sigma^2)$. Because smoothing and differentiation commute, the various derivatives can be computed after smoothing, which leads to the following linear operators for smoothing and differentiation [10]:

$$
\begin{aligned}
(\mathcal{D}_x T_1)(x) &= \frac{1}{2}\left(T_1(x+1) - T_1(x-1)\right), \\
(\mathcal{D}_{xx} T_1)(x) &= T_1(x+1) - 2T_1(x) + T_1(x-1), \quad (2) \\
(\mathcal{D}_{xxx} T_1)(x) &= \frac{1}{2}\left((\mathcal{D}_{xx}T_1)(x+1) - (\mathcal{D}_{xx}T_1)(x-1)\right).
\end{aligned}
$$

Since we assume the noise is Gaussian, the vector of filtered noise derivatives $\mathbf{n}$ is zero-mean Gaussian. The covariance matrix $K(\mathbf{n})$ completely specifies their distribution, and is computed by standard techniques from statistical signal processing; see Papoulis [15] for details. Since it is symmetric, we show only its lower half:

$$
\mathbf{n} = \begin{bmatrix} n_x \\ n_{xx} \\ n_{xxx} \end{bmatrix} , \quad
K(\mathbf{n}) = \sigma_n^2 \begin{bmatrix} B & & \\ 0 & D & \\ F & 0 & E \end{bmatrix} ,
$$

where

$$B = \tfrac{1}{2}T_2(0) - \tfrac{1}{2}T_2(2) ,$$

$$
\begin{aligned}
D &= 2T_2(2) - 8T_2(1) + 6T_2(0) , \\
E &= -\tfrac{1}{2}T_2(4) + 2T_2(3) - 2T_2(2) - 2T_2(1) + \tfrac{5}{2}T_2(0) , \\
F &= -\tfrac{1}{2}T_2(3) + T_2(2) + \tfrac{1}{2}T_2(1) - T_2(0) .
\end{aligned}
$$

3.3 The probability of a zero crossing

Now we derive the probability that there is a zero crossing within the interval $(x - \Delta x, x + \Delta x)$. We use Condition 1 to get the probability of a zero crossing (zc) in the second derivative in a pixel centered at x:

$$P\{\text{zc} \mid r_{xx}, r_{xxx}\} = P\left\{|s_{xx}/s_{xxx}| < \Delta x \mid r_{xx}, r_{xxx}\right\} .$$

The definition $r = s + n$ implies that the conditional distribution of s_{xx} given r_{xx} is $\mathcal{N}\left(r_{xx}, \sigma_{xx}^2\right)$ (*i.e.*, normal with mean r_{xx} and variance σ_{xx}^2). Similarly, the distribution of s_{xxx} given r_{xxx} is $\mathcal{N}\left(r_{xxx}, \sigma_{xxx}^2\right)$. Here σ_{xx}^2 and σ_{xxx}^2 are elements $(2, 2)$ and $(3, 3)$ of the covariance matrix $K(\mathbf{n})$ derived in the previous section.

Fieller [6, 16] derived the cumulative distribution function (c.d.f) of the quotient of two normal random variables, which we use to obtain

$$P\left\{|s_{xx}/s_{xxx}| < \Delta x \mid r_{xx}, r_{xxx}\right\} = \quad (3)$$
$$G\left(\frac{\sigma_{xxx}}{\sigma_{xx}}\Delta x\right) - G\left(-\frac{\sigma_{xxx}}{\sigma_{xx}}\Delta x\right) ,$$

where

$$
\begin{aligned}
G(t) &= L\left(\frac{a - bt}{\sqrt{1+t^2}}, -b, \frac{t}{\sqrt{1+t^2}}\right) \quad (4) \\
&+ L\left(\frac{-a + bt}{\sqrt{1+t^2}}, b, \frac{t}{\sqrt{1+t^2}}\right) ,
\end{aligned}
$$

$a = r_{xx}/\sigma_{xx}$, and $b = r_{xxx}/\sigma_{xxx}$. $L(\cdot, \cdot, \cdot)$ is often used with bivariate normal distributions and is defined for the standardized bivariate normal distribution $\psi(x, y; \rho)$ as

$$L(h, k, \rho) = \int_h^\infty \int_k^\infty \psi(x, y; \rho)\, dy\, dx .$$

$L(h, k, \rho)$ has no closed-form solution. However, good approximations exist [16], and precomputed tables can be used for efficiency.

3.4 The probability of an edge

As we mentioned in Section 3.1, to avoid phantom edges, we require that $s_x s_{xxx} < 0$. This implies the following relation between zero-crossing and edge probabilities:

$$P\{\text{edge}\} = P\{\text{zc} \mid s_x s_{xxx} < 0\} .$$

We separate the edge probability into two cases depending on the sign of s_x:

$$
\begin{aligned}
P\{\text{edge}\} &= P\{\text{edge} \mid s_x > 0\}P\{s_x > 0\} \quad (5) \\
&+ P\{\text{edge} \mid s_x < 0\}P\{s_x < 0\} ,
\end{aligned}
$$

where $P\{s_x < 0\}$ and $P\{s_x > 0\}$ are easy to compute because s_x has a known Gaussian distribution, and

$$P\{\text{edge} \mid s_x > 0\} = P\{\text{zc} \mid s_x > 0, s_{xxx} < 0\} \quad (6)$$
$$P\{\text{edge} \mid s_x < 0\} = P\{\text{zc} \mid s_x < 0, s_{xxx} > 0\}.$$

Unfortunately, because s_x and s_{xxx} are not independent, it is difficult to obtain simple expressions for the right-hand sides of Eq. 6. However, assuming independence greatly simplifies the analysis, and our experimental results suggest that this is a good approximation. Moreover, as we will see in Section 4.2, this simplification is unnecessary in the two-dimensional case, the main goal of our analysis.

When s_x and s_{xxx} are independent, the first term on the r.h.s. of Eq. 4 corresponds to the case $s_{xxx} < 0$ and the second to the case $s_{xxx} > 0$ (see Fieller [6]). Therefore, by defining $G^+(t)$ and $G^-(t)$ accordingly, we can write

$$P\{\text{edge} \mid s_x > 0\} = G^+\left(\frac{\sigma_{xxx}}{\sigma_{xx}}\Delta x\right) - G^+\left(-\frac{\sigma_{xxx}}{\sigma_{xx}}\Delta x\right),$$
$$(7)$$

and similarly for $P\{\text{edge} \mid s_x < 0\}$ using $G^-(t)$.

Now we have defined all the components of the r.h.s. of Eq. 5, and we can compute the edge probability.

At very fine scales, the discrete approximation to the third derivative defined by Eq. 2 is inaccurate, which affects the edge probabilities. Instead we use backward and forward differences of the second derivative. The edge probabilities are then computed separately for the backward and forward differences and combined.

We can localize edges to subpixel precision by adjusting Δx in Eq. 7 to integrate the probability over a region smaller than a pixel. The region can be as small as desired and positioned anywhere between two adjacent pixels.

3.5 The confidence probability

Edge probabilities alone can be misleading. For example, the edge probability at a pixel can be low because there is indeed no edge near the pixel, or because noise is masking the true signal and it is difficult to determine if an edge exists. We seek some measure of confidence that the values of the second and third derivatives used to compute the edge probabilities are not due to noise alone.

The test that we use is explained by Muirhead [14], and we follow his discussion here. As we showed in Section 3.3, the distribution of $\mathbf{X} = \begin{bmatrix} s_{xx} & s_{xxx} \end{bmatrix}^t$ is $\mathcal{N}(\mu, \Sigma)$, where t denotes matrix transpose, and

$$\mu = \begin{bmatrix} r_{xx} \\ r_{xxx} \end{bmatrix}, \quad \Sigma = \begin{bmatrix} \sigma_{xx}^2 & 0 \\ 0 & \sigma_{xxx}^2 \end{bmatrix}.$$

The p.d.f. of the quadratic form $(\mathbf{X} - \mu)^t \Sigma^{-1} (\mathbf{X} - \mu)$ is chi-squared with two degrees of freedom, denoted χ_2^2.

To test the null hypothesis $H_0 : \mu = \mathbf{0}$ against alternatives $\mu \neq \mathbf{0}$, a test of size α is to reject H_0 if

$$Q \equiv \mathbf{X}^t \Sigma^{-1} \mathbf{X} > c_m(\alpha),$$

where $c_m(\alpha)$ is the upper $100\alpha\%$ point of the χ_2^2 distribution. When H_0 is not true, Q has a non-central χ_2^2 distribution with noncentrality parameter δ, denoted $\chi_2^2(\delta)$, where $\delta = \mu^t \Sigma^{-1} \mu$. The power of this test is

$$\beta(\delta) = P\{\chi_2^2(\delta) > c_m(\alpha)\}.$$

To compute the confidence probability, we perform this test at every pixel of the image. The size α is defined as the probability of incorrectly deciding $\mu \neq \mathbf{0}$. As is standard in such testing, we set α to be very low, since this is the error we most want to avoid. Rather than making a hard decision at every pixel based on whether $Q > c_m(\alpha)$, we use as our confidence probability β, the power of the test. The power is defined as the probability of correctly deciding that $\mu \neq \mathbf{0}$ when $Q > c_m(\alpha)$.

4 The two-dimensional case

To extend our results on edge probabilities to the two-dimensional case, we now look for zero crossings of the second derivative of the signal in the direction of its gradient. This requires that we know the distributions of the first, second, and third directional derivatives of the noise, and the distribution of the gradient direction. Computing the probability of an edge in a given direction is similar to the one-dimensional case, once we have the distributions of the directional derivatives. But the gradient direction is noisy as well. Therefore we use conditional probabilities and integrate the product of the probability of an edge in a given direction and the probability that that direction is the gradient.

Let $s(x, y)$, $r(x, y)$, and $n(x, y)$ be the true image, the observed image and the noise respectively. As before, we drop the positional parameters when it is clear that we are referring to values at a single pixel. Let $\mathbf{r}$ denote the vector of first, second and third derivatives of r with respect to x and y. From this vector the gradient direction and directional derivatives can be estimated. Let $\theta_g = \tan^{-1}(s_y/s_x)$ be the gradient direction of the signal. The probability of an edge given $\mathbf{r}$ can be written

$$P\{\text{edge} \mid \mathbf{r}\} = \int_0^{2\pi} P\{\text{edge} \mid \theta = \theta_g, \mathbf{r}\} P\{\theta = \theta_g \mid \mathbf{r}\} \, d\theta_g.$$
$$(8)$$

The sections that follow provide the details of the right-hand size of this equation. We will use the following additional notation. Let $g = (s_x^2 + s_y^2)^{1/2}$ be the gradient magnitude of the signal, and let $\hat{g} = (r_x^2 + r_y^2)^{1/2}$ and

$\hat{\theta}_g = \tan^{-1}(r_y/r_x)$ be the gradient magnitude and direction of the observations. We denote the ith derivatives of r in direction θ by r_i^θ, and similarly for s and n.

4.1 Image derivatives

Here we present the distributions of the first, second and third derivatives of the noise in the x and y directions, and the distribution of the gradient direction.

As in the one-dimensional case, the derivatives are jointly Gaussian with zero means, and the covariance matrix completely specifies their distribution. We again use the discrete Lindeberg operators [10]. Using autocorrelation functions and power spectral densities as before, we find the covariance matrix. Since it is symmetric, we show only its lower half:

$$K(\mathbf{n}) = \sigma_n^2 \begin{bmatrix} AB & & & & & & & & \\ 0 & AB & & & & & & & \\ 0 & 0 & AD & & & & & & \\ 0 & 0 & 0 & BB & & & & & \\ 0 & 0 & CC & 0 & AD & & & & \\ AF & 0 & 0 & 0 & 0 & AE & & & \\ 0 & BC & 0 & 0 & 0 & 0 & BD & & \\ BC & 0 & 0 & 0 & 0 & CF & 0 & BD & \\ 0 & AF & 0 & 0 & 0 & 0 & CF & 0 & AE \end{bmatrix},$$

where
$\mathbf{n} = \begin{bmatrix} n_x & n_y & n_{xx} & n_{xy} & n_{yy} & n_{xxx} & n_{xxy} & n_{xyy} & n_{yyy} \end{bmatrix}^t$, $A = T_2(0)$, $C = 2T_2(1) - 2T_2(0)$ and B, D, E, and F are as in Section 3.2.

To compute directional derivatives, we use separable two-dimensional kernels based on derivatives of the Lindeberg one-dimensional kernels. The directional derivative kernels can be steered to any direction by taking an appropriate linear combination of basis kernels, the derivatives in the x and y directions. For details on steering, see Freeman and Adelson [7]. Using linearity, we can steer the results of convolving the image with the basis kernels.

The distribution of the gradient direction has been derived by Lyvers and Mitchell [12], Gregson [8], and Ramesh and Haralick [17]. Given the observations r_x and r_y, s_x and s_y have Gaussian distributions with means r_x and r_y and standard deviation $\sigma_1 = \sigma_x = \sigma_y$. Because of the structure of $K(\mathbf{n})$, n_x and n_y are independent, and thus s_x given r_x and s_y given r_y are as well.

The distribution for θ_g given $\hat{g}$ and $\hat{\theta}_g$ is the offset normal distribution [13],

$$f(\theta_g \mid \hat{g}, \hat{\theta}_g) = \frac{\phi(z)}{\sqrt{2\pi}} + (z \cos \delta_g)\, \phi(z \sin \delta_g)\, \Phi(z \cos \delta_g), \tag{9}$$

where $z = \hat{g}/\sigma_1$, $\delta_g = \theta_g - \hat{\theta}_g$, $\phi(z)$ is the p.d.f. of the standard univariate normal distribution, and $\Phi(z)$ its c.d.f. Note the distribution is over δ_g but parameterized by z.

Figure 2: Pixel width as a function of direction.

4.2 The probability of an edge given θ_g

In the one-dimensional case, the probability of an edge is given by Eq. 5. In the two-dimensional case, given that the edge is in the gradient direction θ_g, we replace the one-dimensional derivatives implicit in that equation by directional derivatives steered to the θ_g direction. We need also to account for the fact that the "width" of a two-dimensional pixel varies with θ_g, as depicted in Fig. 2. We now write the half-width of a pixel as

$$\Delta x(\theta_g) = \frac{\Delta x}{\max(|\cos \theta_g|, |\sin \theta_g|)}. \tag{10}$$

In one dimension, we computed the edge probability in two parts depending on the sign of s_x. In two dimensions, because θ_g is the gradient direction, we need only consider the $s_x > 0$ case because $P\{s_x > 0\} = 1$. Modifying Eq. 7 accordingly, we obtain

$$P\{\text{edge} \mid r_2^{\theta_g}, r_3^{\theta_g}, \theta_g\} = \tag{11}$$
$$G^+\left(\frac{\sigma_3}{\sigma_2}\Delta x(\theta_g)\right) - G^+\left(-\frac{\sigma_3}{\sigma_2}\Delta x(\theta_g)\right),$$

where σ_2 and σ_3 are the standard deviations of the second and third noise derivatives in direction θ_g. In practice, they depend on θ_g so little that we approximate them by σ_{xx} and σ_{xxx} respectively. Here, $r_2^{\theta_g}$ and $r_3^{\theta_g}$ are part of the definition of G^+, just as r_{xx} and r_{xxx} were in Section 3.3.

4.3 The probability of an edge

Now we can fill in the details of the the two probabilities in the integrand of the right-hand side of Eq. 8, the probability of an edge given the observations. The first probability, $P\{\text{edge} \mid \theta = \theta_g, \mathbf{r}\}$, is given by Eq. 11. The second probability, $P\{\theta = \theta_g \mid \mathbf{r}\}$, is the offset normal distribution as given by Eq. 9.

We have not been able to evaluate the integral in Eq. 8 in closed form. Indeed, it follows from the discussion of Eq. 3 in Section 3.3 that Eq. 11 itself involves integrals that have no closed-form solution. We evaluate the integral numerically using precomputed tables for the offset normal distribution.

Integrating Eq. 11 over θ corresponds to evaluating the edge probability in the θ direction and weighting that probability by the probability that $\theta = \theta_g$, which we compute

(a)

(b)

(c)

(d)

Figure 3: (a) The flower image; (b) detail; (c) edge probability map: dark values indicate high probabilities; (d) detail.

with the offset normal distribution. The observed gradient magnitude $\hat{g}$ influences the edge probabilities because the offset normal distribution depends on $\hat{g}$ through $z = \hat{g}/\sigma_1$ (see Eq. 9). When z is low, the offset normal approaches the uniform distribution, although its maximum is still at the observed gradient direction; when z is high, the offset normal is very concentrated near the observed gradient direction. Because σ_1 is constant within an image, this lowers edge probabilities when $\hat{g}$ is small but not (or only slightly) when it is large. The overall effect is to reduce background noise. We have observed, however, that evaluating Eq. 11 in the observed gradient direction is a reasonable approxi-

mation to Eq. 8.

4.4 The confidence probability

The confidence probability in two dimensions is computed the same way as in one dimension, except now there are seven degrees of freedom in the chi-squared distribution: the second derivative has three components and the third derivative four. The covariance matrix of these derivatives is a submatrix of $K(\mathbf{n})$ in Section 4.1.

4.5 Examples

We have tested the computation of probability maps on a variety of synthetic and natural images. Fig. 3(a) shows an image of a flower in front of a building, and Fig. 3(c)
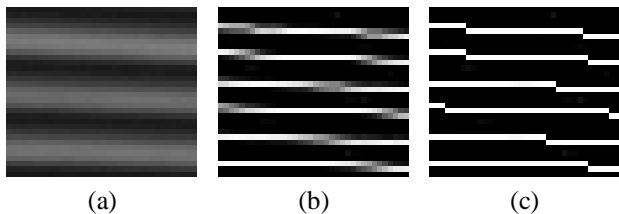
Figure 4: (a) Detail of window in flower image; (b) in flower probability map; (c) edge detection.

its probability map, using $\sigma = 1$. Fig. 3(b) magnifies a small region of the image, and Fig. 3(d) the corresponding region of the edge probability map. Note that even though the two edges of the stem on the right side of Fig. 3(b) are only separated by one pixel and have low contrast in the original image, the high-probability edges in the probability map are clearly visible. This is a consequence of both the accuracy of the Lindeberg operators and the ability of the zero-crossing probabilities to localize edges. Also in Fig. 3(d), the vertical edges of the brick on the left side of the map are blurred because the scale of smoothing is too fine to localize these edges, which have a relatively large blur scale, and therefore a low confidence probability.

5 Edge Detection

Fig. 4 illustrates the computation of edges from the edge probability map. Fig. 4(a) shows a detail on the building's window from the image in Fig. 3. Fig. 4(b) shows the edge probability map. The edges are nearly horizontal, transitioning from one scan line to the next every few pixels. Note that the probabilities in the transitional areas are lower than when the edge runs along a scan line. This is because an edge in a transitional region is close to the boundary between two pixels, and its probability is split between them. It is interesting to note that this results in a natural anti-aliasing in the computer graphics sense.

Detecting edges by thresholding the edge probabilities directly will leave gaps where an edge passes near the boundary between two pixels. The following scheme avoids this problem. When an edge passes between two pixels, the pixels will lie approximately along the line in the gradient direction at the pixels. We visit every pixel and check that its probability is greater than those of its neighbors one pixel away in the gradient direction (and its opposite). If so, we sum that pixel's probability with the larger of its two neighbors. We can do this because we defined Δx in Eq. 10 carefully to cover exactly one pixel. When an edge passes through the center of the pixel, the contribution of its larger neighbor will usually be negligible. When an edge passes between two pixels, the contribution of its larger neighbor will be significant. Fig. 4(c) shows the result of using this procedure on the probability map in

Fig. 4(b).

6 Scale Selection

Once we have computed the edge and confidence probabilities at different scales, we select the appropriate scale for each pixel by using Elder's observation that the best scale is the minimum reliable scale [5]. We choose the minimum reliable scale by thresholding the pixel's confidence probability at some high value and choosing the minimal scale that exceeds that threshold. Once the minimum reliable scale is found, the edge probability of that pixel is copied to an image that stores the final result.

An example for the scale selection can be seen in Fig. 5, which shows an image of a mannequin and its shadow, and edge and confidence probability maps at three scales. Elder used this image to illustrate his approach to scale selection [5]. The image contains edges at a variety of scales, from sharp edges on the mannequin itself to edges that are increasingly diffuse along the shadow from its feet to head.

Fig. 5(c) shows edge and confidence probability maps computed with three different scales of smoothing, using $\sigma_n = 2$. In the finer-scale edge probability maps, the details of the mannequin itself are captured precisely, but as the edges become more diffuse on the shadow, the probabilities become lower, reaching background levels on the shadow's thighs. As scale increases, the diffuse edges now have higher probabilities, but the sharp ones are smeared out, and some details are lost entirely. The confidence maps were computed with size α at the 5σ level (where the σ here is the standard deviation of the normal distribution). Fig. 5(b) is the result of applying the edge detection algorithm from the previous section to the edge probability map after scale selection, i.e., a map that combines the results from scales $1, 2,$ and 4. We used a threshold of 0.9 on the resulting probability map to get the edges.

7 Conclusions and future work

In this paper, we have analyzed the behavior of zero crossings of the second derivative from a statistical point of view. We have shown how to compute the probability that an edge passes through a pixel, and a probability that the measurements underlying the edge probability are reliable. We used the Lindeberg operators for their accuracy and scale-space properties, and analyzed their statistical properties. We argued that testing for edges with the gradient magnitude was prone to errors, and showed that the proper role for the gradient magnitude is through the offset normal distribution, which in our approach assigns different weights to the probability of an edge in different directions. Our results on natural images demonstrate the promise of this approach.
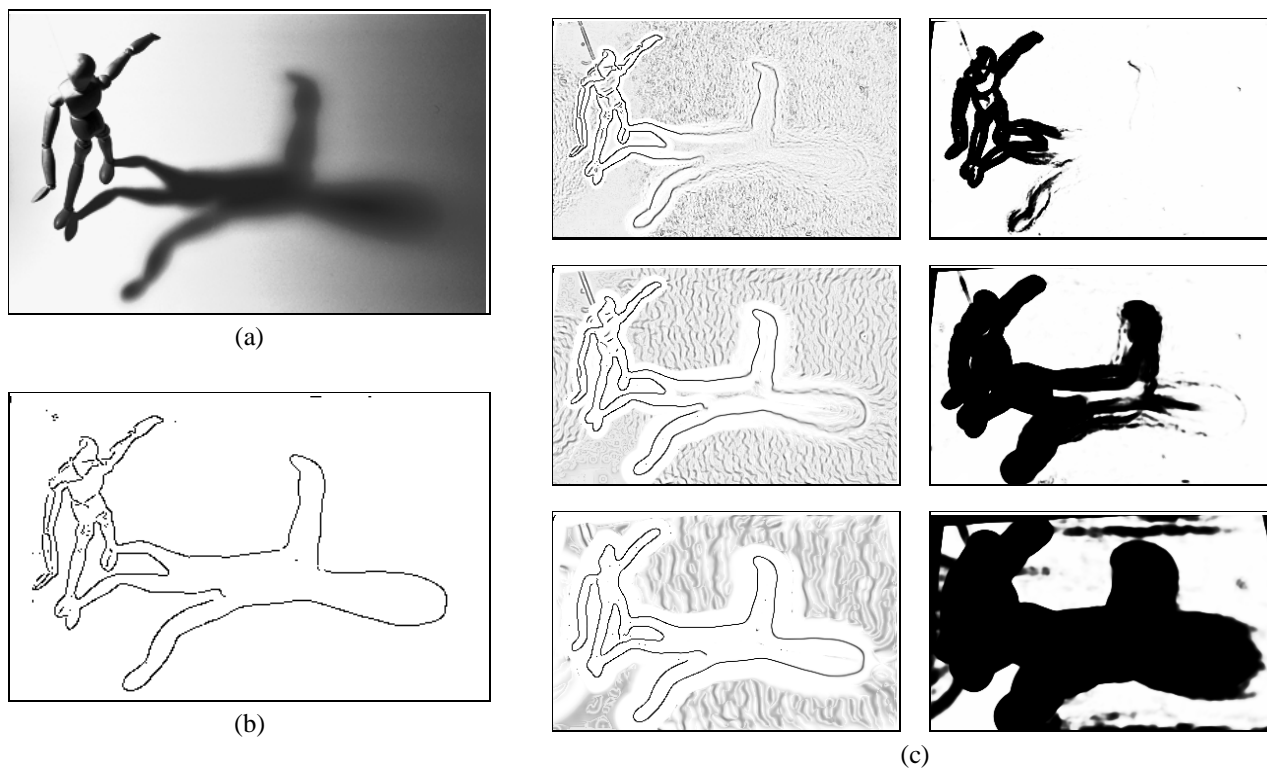
(a)

(b)

(c)

Figure 5: (a) The shadow image; (b) detected edges after scale selection; (c) edge (left) and confidence (right) probability maps for scales $\sigma = 1, 2, 4$.

We are currently working on computing edge probabilities to subpixel accuracy, applying our techniques to detecting other features like ridges and junctions, and using the probabilistic information we compute in linking and grouping edges and segmenting curves.

Acknowledgments

References

[1] A. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions*, Applied Mathematics Series, National Bureau of Standards, 1964.

[2] John F. Canny, "A Computational Approach to Edge Detection," *IEEE PAMI* 8:6, 679–700, 1986.

[3] James Clark, "Authenticating Edges Produced by Zero-Crossing Algorithms," *IEEE PAMI* 11:1, 43–57, 1989.

[4] Rachid Deriche, "Using Canny's Criteria to Derive a Recursively Implemented Optimal Edge Detector," *IJCV* 1:2, 167-187, 1987.

[5] James H. Elder and Steven W. Zucker, "Local Scale Control for Edge Detection and Blur Estimation," *ECCV'96*, 57–69, 1996.

[6] E. C. Fieller, "The Distribution of the Index in a Normal Bivariate Population," *Biometrika* 24, 428–440, 1932.

[7] W. T. Freeman, and E. H. Adelson, "The design and use of steerable filters," *IEEE PAMI* 13:9, 891–906, 1995.

[8] P. H. Gregson, "Using Angular Dispersion of Gradient Direction for Detecting Edge Ribbons," *IEEE PAMI* 15:7, 682–697, 1993.

[9] Robert M. Haralick, "Digital step edges from zero crossings of the second directional derivative," *IEEE PAMI* 6:1, 58–68, 1984.

[10] Tony Lindeberg, *Scale-Space Theory in Computer Vision*, Kluwer Academic Publishers, 1994.

[11] Tony Lindeberg, "Edge Detection and Ridge Detection with Automatic Scale Selection," *CVPR'96*, 465–470, 1996.

[12] Edward P. Lyvers and Robert O. Mitchell, "Precision Edge Contrast and Orientation Estimation," *IEEE PAMI* 10:6, 927–937, 1988.

[13] K. V. Mardia, *Statistics of directional data*, Academic Press, 1972.

[14] Robb J. Muirhead, *Aspects of Multivariate Statistical Theory,*, John Wiley & Sons, 1982.

[15] A. Papoulis. *Probability, Random Variables and Stochastic Processes*, 2nd Ed., McGraw-Hill, 1972.

[16] Jagdish K. Patel and Campbell B. Read, *Handbook of the Normal Distribution,* 2nd Ed., Marcel Dekker, 1996.

[17] Visvanathan Ramesh and Robert M. Haralick, "Random perturbation models for boundary extraction sequence," *Machine Vision and Applications* 9, 229-239, 1997.

[18] Paul L. Rosin, "Edges: saliency measures and automatic thresholding," *Machine Vision and Applications* 9, 139–159, 1997.

[19] Harry Voorhees and Tomaso Poggio, "Detecting Textons and Texture Boundaries in Natural Images," *ICCV'87,* 250–258, 1987.