February 1994

A Novel Radial Intensity Based Edge Operator

Gregory Provan
University of Pennsylvania

Hany Farid
University of Pennsylvania

Eero Simoncelli
University of Pennsylvania

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-94-07.

A Novel Radial Intensity Based Edge Operator

Abstract

A novel edge operator is introduced based on *steerable asymmetric linear filters* consisting of *radial wedge segments*. An *intensity* profile is computed by averaging intensity values along a radial wedge segment as it "sweeps" about a small circular neighborhood. The "steerability" of the filters allows for interpolation of a continuous profile function for *n* discretely sampled postions of the radial wedge segments. Edge strength is then calculated as a simple difference of *conditinal means* of the resulting intensity profile. This paper introduces the basic paradigm of using asymmetric filters for low-level image processing tasks and shows how this approach is utilized to design a novel edge operator (the Radial InTensity Edge, or RITE, operator). Features of the RITE operator include: (1) a mathematically simple algorithm with comparable performance to the well-known gradient-based Deriche operator; (2) better performance at points where several edges intersect; (3) an average time complexity reduction by a factor of 1.7 to 2.1 over the Deriche operator.

Keywords

edge detection, steerable filters, asymmetric filters

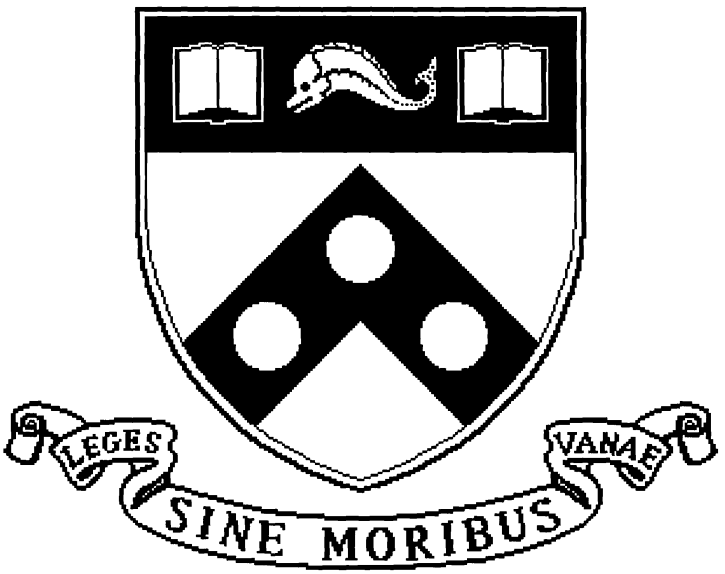
Comments

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-94-07.

A Novel Radial Intensity Based Edge Operator

MS-CIS-94-07
GRASP LAB 368

Gregory Provan
Hany Farid
Eero Simoncelli



University of Pennsylvania
School of Engineering and Applied Science
Computer and Information Science Department
Philadelphia, PA 19104-6389

February 1994

A Novel Radial Intensity Based Edge Operator

Gregory Provan, Hany Farid, Eero Simoncelli
(provan@central, farid@grip, eero@central).cis.upenn.edu
GRASP Laboratory
University of Pennsylvania
Philadelphia, PA 19104-6228

Abstract

A novel edge operator is introduced based on *steerable asymmetric linear filters* consisting of *radial wedge segments*. An *intensity profile* is computed by averaging intensity values along a radial wedge segment as it "sweeps" about a small circular neighborhood. The "steerability" of the filters allows for interpolation of a continuous profile function from n discretely sampled positions of the radial wedge segments. Edge strength is then calculated as a simple difference of *conditional means* of the resulting intensity profile. This paper introduces the basic paradigm of using asymmetric filters for low-level image processing tasks and shows how this approach is utilized to design a novel edge operator (the Radial InTensity Edge, or RITE, operator). Features of the RITE operator include: (1) a mathematically simple algorithm with comparable performance to the well-known gradient-based Deriche operator; (2) better performance at points where several edges intersect; (3) an average time complexity reduction by a factor of 1.7 to 2.1 over the Deriche operator.

Keywords: Edge Detection, Steerable Filters, Asymmetric Filters

1 Introduction

Edge detection is a crucial component of most machine vision systems. Although significant effort has been directed towards developing edge detection operators, there is still considerable room for improvement in the existing operators. The most widely-used approach to edge detection is the gradient-based approach (see [3, 1, 2, 7], among many others). Although this approach performs well in most circumstances, there are well-known problems, particularly, in regions of high curvature or where multiple edges intersect.

A completely new approach to edge detection is introduced here which performs at least as well as other operators based purely on bottom-up, local information. [1] The edge operator introduced in this paper is unique in its use of *steerable asymmetric linear filters* consisting of *weighted wedge segments.* An *intensity profile* is computed by averaging intensity values along the wedge as it "sweeps" around a small circular window. The "steerability" of the filters allows for interpolation of a continuous profile function from n discretely sampled positions of the radial wedge segments. The edge strength is then calculated as a simple difference of *conditional means* of the intensity profile.

This paper introduces the basic paradigm of using steerable asymmetric filters for low-level image processing tasks and shows how this approach is utilized to design a novel edge operator, which we refer to as the Radial InTensity Edge (RITE) operator. The RITE operator introduced here generates results comparable to the Deriche [3] operator, (which is based on the popular Canny operator [1]) and performs better in some circumstances. Specifically, the RITE operator offers:

1. a mathematically simple algorithm with comparable performance to other popular edge operators
2. better performance at points where several edges intersect
3. an average time complexity reduction by a factor of 1.7 to 2.1 over the Deriche operator

This paper is organized as follows, Section 2 introduces the intuition and mathematical notation behind the RITE operator. Section 3 relates our operator to other popular edge detection methods. Section 4 presents the algorithm for the RITE operator, an empirical comparison to the Deriche operator on several real-world images, and an analysis of the time complexity of the RITE and Deriche operators.

2 Theory

The process of detecting edges in grey-scale images consists of identifying edges via intensity differences in the image. Typically this involves (1) convolving the input image with filters, and then (2) applying a discriminant function to the convolved images in order to identify edges. The most common approach, the gradient-based approach, applies symmetric gaussian filters and then uses the magnitude of the intensity gradients as the discriminant for identifying edges. In this paper, an operator is described which applies *asymmetric* filters and then uses *intensity differences* as the discriminant for identifying edges. This section first presents the intuition behind this novel approach, and then outlines the mathematical basis of the operator.

2.1 Intuition

The intuition behind the use of asymmetric filters for feature extraction is relatively straightforward. Consider a circular window with radius r placed at a region on the image (Figure 1a). Possible intensity profiles computed by averaging along a *radial wedge segment* as the segment

[1] Some operators have been extended with top-down information, such as Zucker [6]

"sweeps" around the circular window, are shown in Figure 1b-d; these profiles correspond to regions b-d, respectively, in Figure 2. Distinguishing among different features is thus possible if

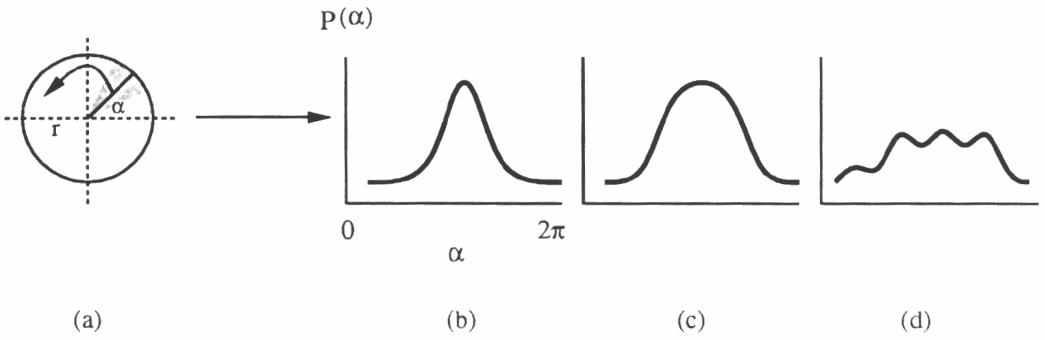


Figure 1: (a) Circular sampling window; (b-d) examples of possible profiles resulting from "sweeping" a wedge around the circular window.

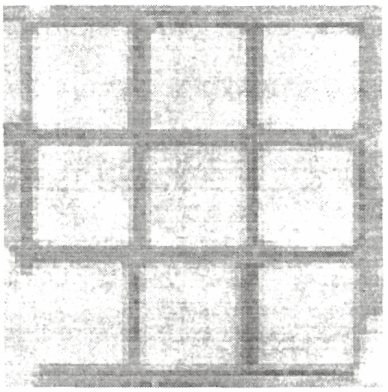


Figure 2: Sample image from which profiles in Figure 1 are taken.

the resulting intensity profiles are unique and can be differentiated. In particular, we show in this paper how a simple difference of *conditional means* can be used to detect edges. One of perhaps many other applications include using the kurtosis (ratio of higher order central moments) for vertex detection.[2]

Figure 3 depicts the intensity profiles of (a) an ideal edge; (b) a noisy edge; and (c) a mostly uniform area. The intensity difference is computed as the difference of conditional means (μ_+, μ_-), where μ_+ and μ_- are the mean of the values of the profile function P greater than and less than the mean μ, respectively, as depicted in Figure 3b.[3] The reason for using this difference of conditional means is that it provides a noise-insensitive measure of the *amplitude* of the profile function. Clearly, the intensity difference of a mostly uniform region (c) will be considerably smaller than that of an ideal or noisy edge. Thus, we propose that a simple difference of the conditional means of the intensity profile is sufficiently powerful to differentiate between edges and noise. The next section formalizes the intuition behind this proposed edge operator.

[2]Work in progress.

[3]Note that this difference of conditional means is invariant to the orientation of the edge.

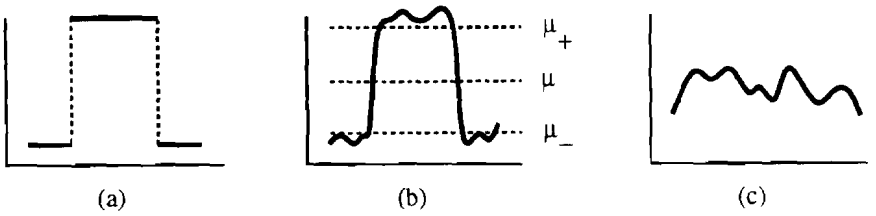


Figure 3: Intensity profiles from (a) an ideal edge; (b) a noisy edge; and (c) a mostly uniform area. In (b), μ is the mean intensity, μ_+ and μ_- are the conditional means, and $\mu_+ - \mu_-$ is the intensity difference.

2.2 Notation and Definitions

This section formalizes the concepts presented in the previous section. The ideas behind the computation of the intensity profiles are formalized followed by formal definitions of the difference of conditional means used by the RITE operator.

The processing is described for a single point in the image. Throughout, a polar image coordinate system (r, θ) is used with the origin at this point.

Definition 1 (Intensity Profile) *The intensity profile, $P(\cdot)$, is*

$$P(\alpha) = \int dr \int d\theta \; r \; w(r, \alpha - \theta) \; I(r, \theta) \tag{1}$$

where $I(r, \theta)$ is the image intensity in polar coordinates centered at a point in the image, and,

$$\begin{aligned} w(r,\theta) &= g(r)h(\theta) \\ g(r) &= \begin{cases} 1 & \text{if } r < r_0 \\ 0 & \text{otherwise} \end{cases} \\ h(r) &= \begin{cases} 1 & \text{if } |\theta| < \theta_0 \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

r_0 defines the radius of the wedge filter, and θ_0 defines its angular width.

Note, that equation (1) expresses a convolution (in the angular coordinate) of the image intensity with the wedge filter. Figure 4 shows $w(r, \theta - \alpha)$.
Three examples of various regions in an image and their corresponding profiles are shown in Figure 5.

Definition 2 (Difference of Conditional Means) *The measure of edge strength is given by the difference of conditional spatial means (δ).*

$$\delta = \mu_+ - \mu_-$$

Note that the function $P(\alpha)$ is an angular *profile function*. It is similar to the profile function computed by the RITE operator, although it is based on a different set of filters. Note also that this profile function is restricted to a very specific form: it is a single-cycle sinusoidal function of α.

Now reconsider the gradient edge-detection algorithm described above. The algorithm essentially computes $A(x, y)$, the amplitude of the profile function, and compares it to a threshold. Recall that the RITE algorithm performs an operation that is conceptually similar: it computes the profile function and then computes a metric that gives a measure of the amplitude of the function. Unlike the RITE algorithm, the gradient-based algorithm computes the amplitude *analytically* (although the computation is still nonlinear).

Despite the similarity in the forms of the gradient and RITE operators, the linear filters used to gather the initial measurements are quite different. The gradient algorithm is based on directional derivatives of a radially symmetric function. These filters are anti-symmetric about the origin, whereas the RITE operator uses an *asymmetric* wedge function.

3.2 Steerable "Wedge" Filters

In this section, we show some examples of steerable asymmetric filters derived from derivative filters. The filters are first written in polar-separable form. Consider writing the $x-$derivative of a two-dimensional radially symmetric function, $\gamma(r)$, where $r = \sqrt{x^2 + y^2}$ is the radial coordinate:

$$\begin{aligned} \frac{\partial \gamma(r)}{\partial x} &= \frac{\partial r}{\partial x}\, \gamma'(r) \\ &= cos(\theta)\; r\gamma'(r) \end{aligned}$$

where θ is the angular coordinate and $\gamma'(r) = \frac{d}{dr}[\gamma(r)]$, the derivative of $\gamma(r)$ with respect to its argument. Note that this function is polar separable, a product of angular and radial component functions. Similarly, the $y-$derivative may be written in polar-separable form:

$$\begin{aligned} \frac{\partial \gamma(r)}{\partial y} &= sin(\theta)\; r\gamma'(r) \\ &= cos(\pi/2 - \theta)\; r\gamma'(r). \end{aligned}$$

Now, in polar coordinates, the steerability property is written as:

$$\begin{aligned} D_\alpha[\gamma(r)] = cos(\alpha - \theta) r\gamma'(r) &= [\cos(\alpha)\cos(\theta) + \sin(\alpha)\sin(\theta)]\, r\gamma'(r) \\ &= \cos(\alpha)\cos(0 - \theta) r\gamma'(r) + \sin(\alpha)\cos(\pi/2 - \theta) r\gamma'(r) \end{aligned}$$

This equation is a specific instance of the more general steerability condition described in [4]:

$$h(\alpha - \theta)g(r) = \sum_{n=0}^{N-1} k_n(\alpha) h(\alpha_n - \theta) g(r),$$

where $h(\cdot)$ is the angular portion steerable kernel, the $k_n(\cdot)$ are *interpolation functions*, the α_n are a fixed set of N orientations, and $g(r)$ is the radial portion of the kernel. The gradient example, with $N = 2$, is the simplest such filter set with $h(\theta) = \cos(\theta)$, $g(r) = r\gamma'(r)$, and $\alpha_n \in \{0, \pi/2\}$.

Now, the advantage of using steerable filters to compute our profile function should be clear. Because the filters are steerable, the full profile function can be *analytically* interpolated from a small number of samples:

$$\begin{aligned} P(\alpha) &= \int dr \int d\theta \; r \; w(r, \alpha - \theta) \; I(r, \theta) \\ &= \sum_{n=0}^{N-1} k_n(\alpha) \int dr \int d\theta \; r \; w(r, \alpha_n - \theta) \; I(r, \theta) \\ &= \sum_{n=0}^{N-1} k_n(\alpha) \; P(\alpha_n). \end{aligned}$$

The simplest example of an *asymmetric* steerable kernel is the following angular function:

$$h(\theta) = \frac{1}{2}\left[1 + \cos(\theta)\right].$$

It is easily verified that this function will be steerable with $N = 3$, and the "steering formula" is as follows:

$$h(\alpha - \theta) = \sum_{n=0}^{2} \left(\frac{1}{3} + \frac{2}{3}\cos(\alpha - \alpha_n)\right) h(\alpha_n - \theta) \tag{3}$$

where $\alpha_n = 2\pi n/3$. Combining this with an unspecified radial function $g(r)$ gives a functional form for the set of asymmetric steerable filters:

$$w_n(r, \theta) = h(\alpha_n - \theta) g(r).$$

where $\alpha_n \in \{0, 2\pi/3, 4\pi/3\}$. These three filter kernels are illustrated in figure 6. Note that

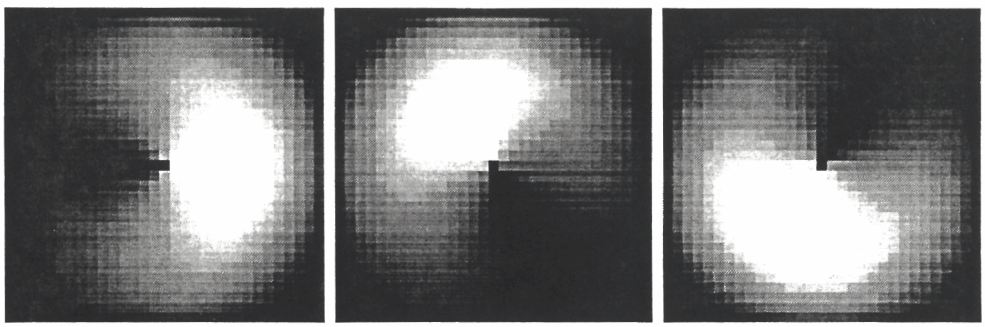


Figure 6: A simple set of steerable "wedge" filters.

these kernels are not very "wedge-like": the angular form of the function is too broad. Narrower steerable wedge filters can be created by raising the cosine function to a power:

$$w_n(r, \theta) = \left[1 + \cos(\alpha_n - \theta)\right]^N g(r)$$

where $\alpha_n = 2\pi n/(2N + 1)$, for $0 \leq n \leq 2N$. The interpolation functions may be derived using techniques described in [4, 8]. Some example filter kernels for $N = 8$ are shown in figure 7. Note that these are much narrower.

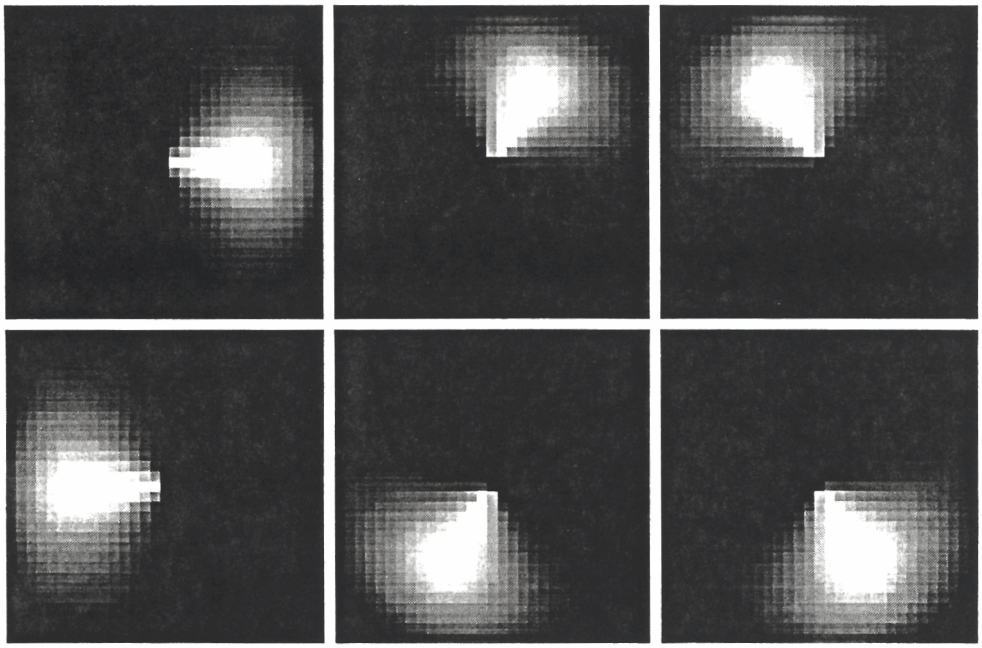


Figure 7: Six of the set of 17 steerable wedge filters for $N = 8$.

3.3 Signal Processing

The RITE operator is also related to some approaches that have been used in the signal processing and pattern-classification communities, in that they all use statistical classification techniques (see [5] for a review of signal classification measures).

Cast in the terminology of signal classification, the RITE operator performs binary classification on the profile function: if an edge is present, the profile function will have two intensity levels, or classes, C_1 and C_2, and the goal is to classify C_1 and C_2. Given the profile function P with mean $m = E(P)$, this classification is done by computing the difference between the two conditional means $\mu_+(P) \equiv E(P|P \geq m)$ and $\mu_-(P) \equiv E(P|P < m)$, where E means expected value.[4]

Related binary classification approaches are widely used in signal processing. Several measures related to the measure used in the RITE operator have been used; one of the closest is the deflection $D \equiv (1/\sigma_2^2(P))[m_1(P) - m_2(P)]^2$, where $\sigma_2^2(P) = [P - \mu_-(P)|P > m]^2$ (cf. [5]). A full description of the relationships between the classification measures used in the RITE operator and in the signal processing community is given in the full paper.

4 Edge Detection

Outlined below are two algorithms for edge detection based on calculating intensity differences from intensity profiles. The intensity profiles are computed by "sweeping" a *weighted radial wedge segment* about small local neighborhoods.

[4]We assume that class C_1 is defined by P where $P \leq m$, and class C_2 is defined by P where $P > m$.

The first algorithm (Algorithm 1) computes the intensity profile by "sweeping" a *weighted wedge* around a 5×5 neighborhood. The "sweeping" of a wedge is simulated by convolving each point on the image with nine *wedge filters* (See Appendix B). The edge strength is then simply computed as the intensity difference of the resulting profile (See Section 2). This implementation of the RITE operator was not designed for efficiency, but rather, to show the feasibility and power of the radial intensity approach to edge detection. A simpler, faster version of this algorithm is described below.

The second algorithm (Algorithm 2) computes the intensity profile by sampling eight neighbors about a 3×3 neighborhood. Thus, the weighted wedge described in Algorithm 1 is approximated here as an *un-weighted radial segment.* The edge strength is then computed as the intensity difference of the resulting profile. Although more sensitive to noisy images (due to the smaller sampling window), this algorithm is extremely fast, running four times as fast as Algorithm 1, and, in general, generating comparable results.

Both Algorithms are outlined below. The code for both algorithms is written in 'C' and implemented on a Sparc architecture. The code is extremely straight-forward, requiring only 320 and 240 lines for Algorithms 1 and 2, respectively.

Algorithm 1:

- *Edge Detection:*
  For each pixel in the image:

  1. Perform convolution with each of nine 5×5 wedge filters (See Appendix B).
  2. Compute the intensity profile (P).
  3. Compute the intensity difference (δ) of the profile (P).
  4. Set the output pixel value to δ.

- *Post Processing:*
  For each pixel in the image:

  1. Convolve image with a 5×5 "smoothing" filter.
  2. Perform post-processing to "thin" detected edges. The post-processing requires two passes through the image, where, at each point in the image, a pixel whose intensity value is less than the average of its eight neighbors is trimmed.

Algorithm 2:

- *Gaussian Smoothing (optional):*
  Convolve with a 3×3 Gaussian filter for noise suppression.

- *Edge Detection:*
  For each pixel in the image:

  1. Sample eight neighbors about a 3×3 neighborhood.
  2. Compute the intensity profile (P).
  3. Compute the intensity difference (δ) of the profile (P).
  4. Set the output pixel value to δ.

- *Post Processing:*
  For each pixel in the image:

  1. Convolve image with a 3×3 "smoothing" filter.
  2. Perform post-processing to "thin" detected edges. The post-processing requires two passes through the image, where, at each point in the image, a pixel whose intensity value is less than the average of its eight neighbors is trimmed.

Results from several real world images for both implementations of the RITE operator and, for comparison, the Deriche operator [3] are presented in the next section. Run-time benchmarks and a formal time complexity analysis of the RITE and Deriche operators are also presented.

4.1 Results

Results from both implementations of the RITE operator, and for comparison, the Deriche operator on real-world images are presented in this section. The first test image, the Hieroglyphs image [5] is shown in Figure 8.

Figure 8: Test image (612×176): Hieroglyphs (transliterated - *gemeh*) means *perception*.

Results, prior to post-processing, for the Deriche operator and both implementations of the RITE operator, are shown in Figure 11 (See Appendix A). Figure 12 depicts a magnified view of the *eye* from the Hieroglyphs image (See Appendix A). Results, after post-processing, for the Deriche operator and both implementations of the RITE operator, are shown in Figure 13 (See Appendix A). Throughout, the Deriche operator was run with an α value equal to 1.75.

Empirical comparisons of edge operators are frequently quite subjective. In the images presented in Appendix A, we hope only to show that the RITE operator compares favorably to the Deriche operator. Prior to the application of a post-processing stage, the RITE operator produces results nearly identical to that of the Deriche operator. Although the post-processing stage of the RITE operator is very simple in comparison to the Deriche operator, the resulting outputs are comparable. However, the RITE operator outperforms the Deriche in certain areas. Examples of such areas are given in the next section.

[5] The hieroglyphs in Figure 8 (transliterated - *gemeh*) means *perception*.

4.2 On the Accuracy of Edge Operators

Two criteria proposed to evaluate edge operators are signal-to-noise ratio and "accuracy". Although much work has been devoted to edge detection, the notion of what an edge is, and more importantly, what an accurate edge operator is, remains ill-defined. In this paper we do not propose to clarify this topic. Instead, we introduce qualitative arguments as to why the RITE operator is more accurate than Deriche; where by accurate, we mean the identification of edges true to the original image both in locality and edge strength.

One well-known problem with gradient-based operators is their failure to accurately identify edges along high-curvature regions or places where multiple edges intersect. In such regions, operators like Deriche produce imprecise edges. In contrast, the RITE operator performs well in such regions. Figure 9 depicts an example where the Deriche operator produces discontinuous edges at the neck region in the Venus image, while the RITE operator produce continuous edges in this region. It is important to note that prior to post-processing, both operators produce continuous edges at the neck region. However, application of the post-processing stage in the Deriche operator results in a discontinuity of five pixels (not due to thresholding), possibly due to a lack of consistent structure in these regions. In contrast, even the simplistic post-processing stage of the RITE operator is able to preserve regions where several edges intersect. We believe that the use of asymmetric filters in the RITE operator results in better "edge structure" thus yielding higher accuracy at such regions; current work involves further study in this area.

4.3 Performance in the Presence of Noise

The images presented in the previous sections have some amounts of noise, but, a more thorough investigation of the performance of edge detection in the presence of noise is warranted. Results from several experiments suggest that in the presence of *low-intensity noise*, the RITE operator performs extremely well. This is due to the fact that the intensity difference for low intensity noise tends to be small (while intensity differences of edges tends to be high), and thus few false positive edges are detected by the operator. On the other hand, *high intensity* noise tends to have high intensity differences in the same range as edges. The RITE operator which uses 5×5 sampling performs "smoothing" by virtue of the weighted wedge, however, in the case of the 3×3 version of the RITE operator, it is necessary to perform Gaussian smoothing prior to edge detection. Results from the RITE (5×5 and 3×3) operator and the Deriche operator in the presence of Gaussian noise are shown in Figure 10.

4.4 Run-Time Analysis

Both implementations of the RITE operator run in linear time with respect to the image size (i.e. $O(n)$, where n is the size of the image). An empirical comparison of run-times of both versions of the RITE operator and the Deriche operator is shown in Table 1. All times (given in seconds) are averaged over ten separate runs on five different 256×256. All runs were performed on a Sun Sparc 10. With post-processing, the fast version (i.e. 3×3 sampling neighborhood) of the RITE operator is 1.16 times faster than the Deriche operator, and without post-processing, 1.5 times faster.

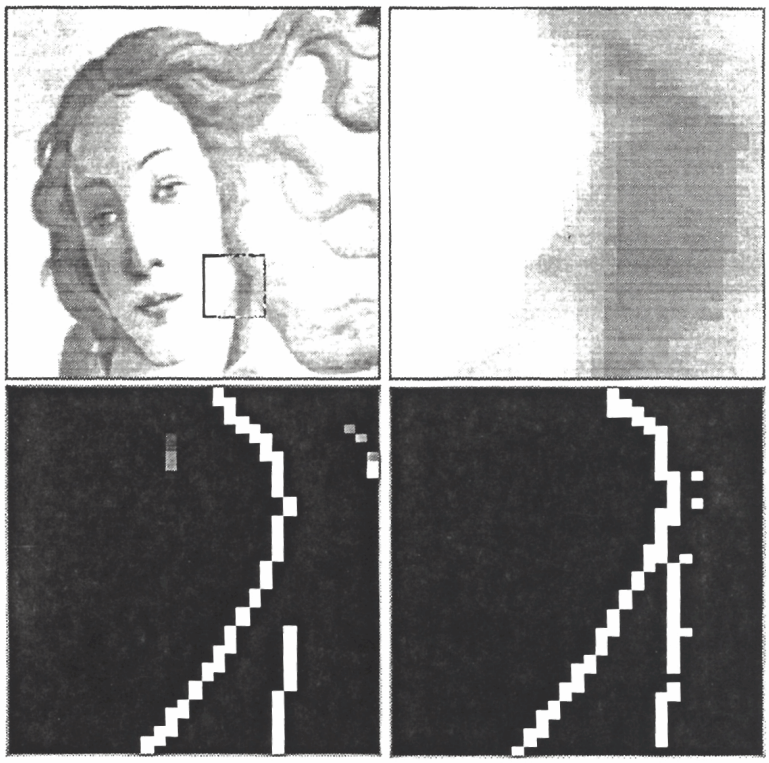


Figure 9: Venus image (top-left): Magnification of neck (top-right); Results from the Deriche operator (bottom-left); Results from the RITE operator (bottom-right). Note that, unlike the Deriche operator, the RITE operator produces continuous edges at the neck region.

Since comparisons of run-time are highly dependent on the particular implementation of each algorithm, an analysis of the computational complexity (i.e. number of mathematical operations) of all three algorithms was performed (Table 2). On average, running on a Sparc 10, an addition/subtraction takes 7.46×10^{-7} seconds, a multiplication 8.32×10^{-7} seconds, and division 8.68×10^{-7} seconds. Thus, the Deriche operator requires 6.9718×10^{-7} seconds of computation at each point on the image, while the RITE operator requires only 4.014×10^{-7} seconds (5×5) and 3.182×10^{-7} (3×3) seconds of computation. Both implementations of the RITE operator, 5×5 and 3×3, result in a theoretical speed up factor of 1.74 and 2.19, respectively. These values do *not* include the post-processing stage of either operator; we simply note here that the post-processing stage of the RITE operator is considerably simpler than that of the Deriche operator.

Prior to post-processing, both versions of the RITE operator require only local information from a single pass through the image. As such, the algorithm is well suited for a parallel implementation. Although the post-processing stage of the RITE operator requires two passes through the image, only local information is needed during each pass and thus this stage could also be easily implemented in parallel. Currently, the RITE operator has only been implemented on a sequential architecture.

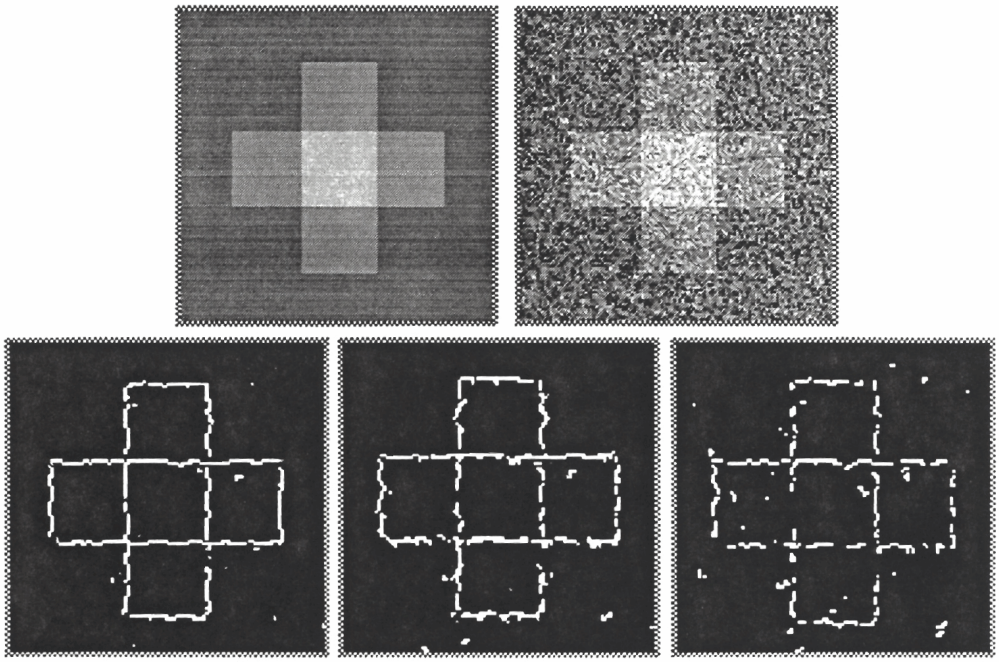


Figure 10: Cross image: Original 128 × 128 image (top-left); Cross with additive Gaussian noise (σ=32) (top-right); Results from Deriche operator (α = 1.75) (bottom-left), left). Results from RITE operator with 5 × 5 sampling and 3 × 3 (bottom-middle, bottom-right).

Table 1: Time benchmarks for the Deriche and RITE operator.

Operator	Average run-time (sec)	Standard Deviation
Deriche $\alpha = 1.75$ (post)	1.745	0.112
RITE 5 × 5 (post)	5.547	1.031
RITE 3 × 3 (post)	1.506	0.121
Deriche $\alpha = 1.75$ (no post)	1.449	0.098
RITE 5 × 5 (no post)	4.422	0.079
RITE 3 × 3 (no post)	0.962	0.195

5 Discussion

This paper introduces a novel approach to general low level image processing. In particular, the use of *asymmetric linear filters* and the edge strength metric of computing the *difference of conditional means* is, to our knowledge, novel.

The application of the general approach to image processing introduced in this paper is an edge operator (the RITE operator) based on asymmetric linear filters consisting of *radial wedge segments*. The edge operator computes an intensity profile by averaging intensity values along the wedge as it "sweeps" around a small local neighborhood. Edge strength is then measured as

Table 2: Arithmetic operations performed at each point in image for Deriche and RITE operator.

Operator	Add/Sub	Mul	Div	Total
Deriche $\alpha = 1.0$	31	56	0	87
RITE 5×5	38	10	4	52
RITE 3×3	29	0	3	32

a simple difference of conditional means.

Empirically, results from the RITE operator are comparable to the well-known gradient-based Deriche operator. However, in some respects the RITE operator outperforms the Deriche operator. In particular, unlike the Deriche operator, the RITE operator generates continuous edges at points where several edges intersect. Note that although prior to post-processing, both operators produce continuous edges at points where several edges intersect, the Deriche operator is unable to preserve these regions after applying its post-processing stage. In contrast, even the simple post-processing stage of the RITE operator is able to successfully preserve such regions. This type of behavior is especially attractive since these regions generally contain important contour information.

Further improvements over the Deriche operator include a reduction in time complexity by a factor of 1.7 to 2.1. The RITE operator is also ideal for a parallel implementation since the operator requires only local information from a single pass through the image.

The use of asymmetric linear filters has been shown to be a sufficiently powerful image processing tool; further applications to a variety of low-level image processing tasks are currently being investigated.

Appendix A

Results from Deriche, and RITE operator on Hieroglyphs image:

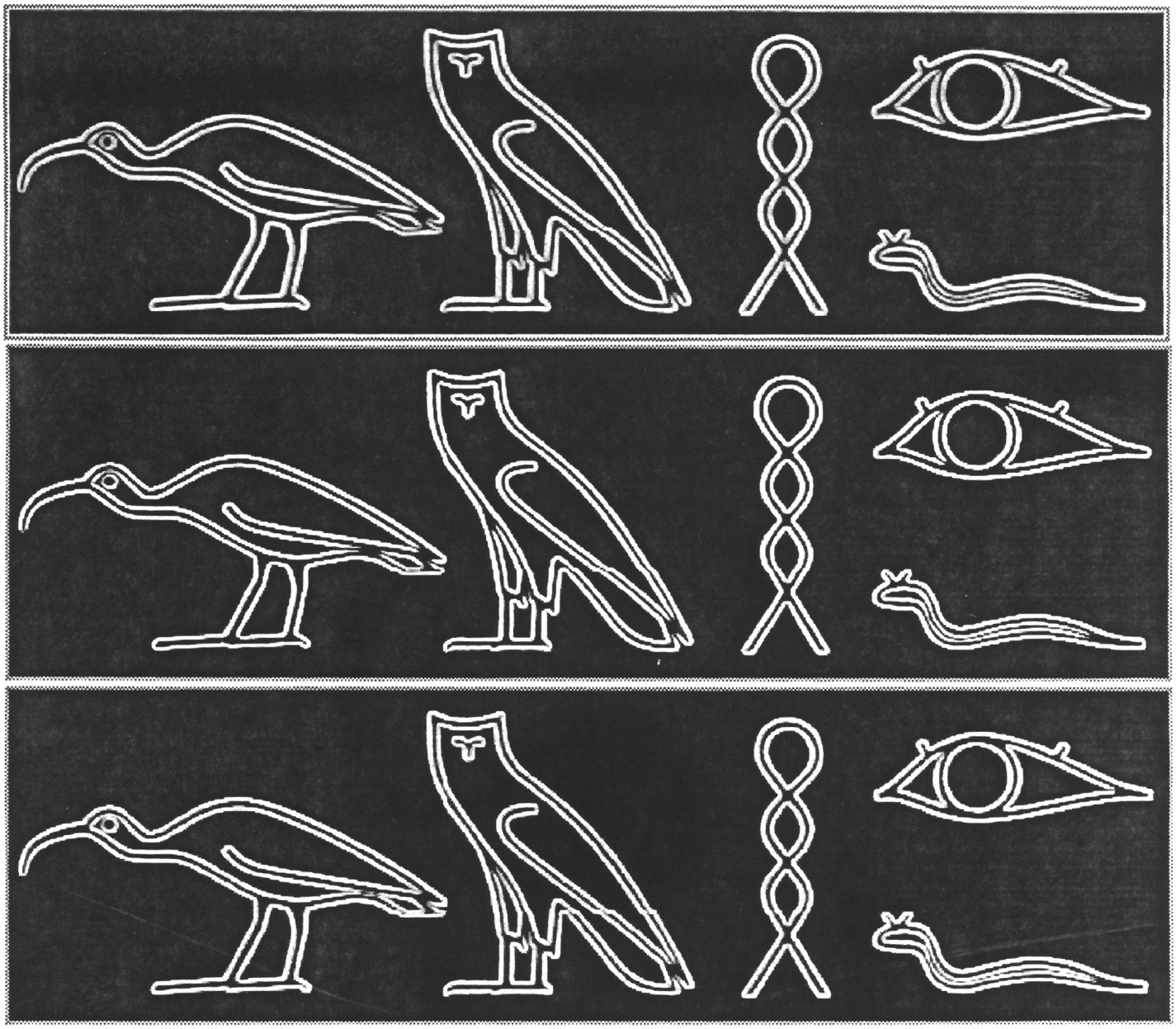


Figure 11: Results from edge detection on Hieroglyphs image (prior to post-processing). Results from Deriche operator with $\alpha = 1.75$ (top); Results from RITE operator with 5×5 sampling (middle), and 3×3 sampling (bottom).

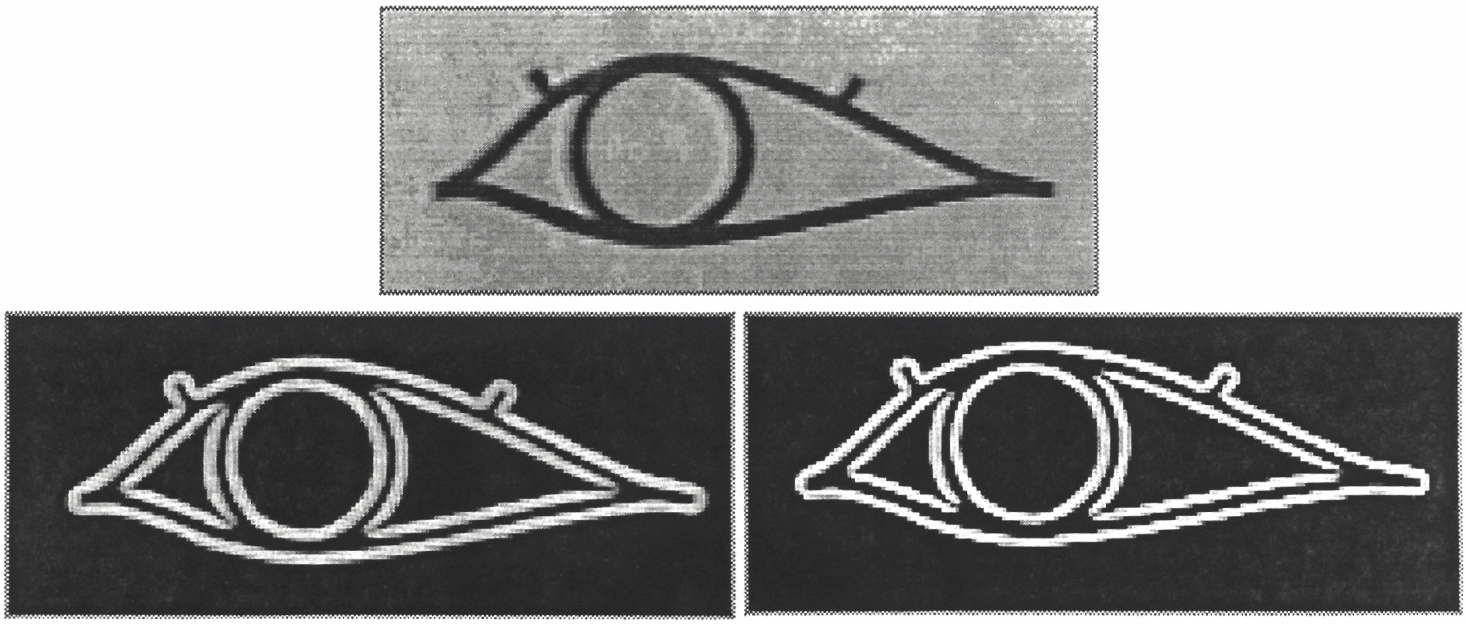


Figure 12: Magnification of eye from Hieroglyphs image (top); Results from Deriche operator ($\alpha = 1.75$) and from RITE operator (5×5) bottom left and right, respectively.

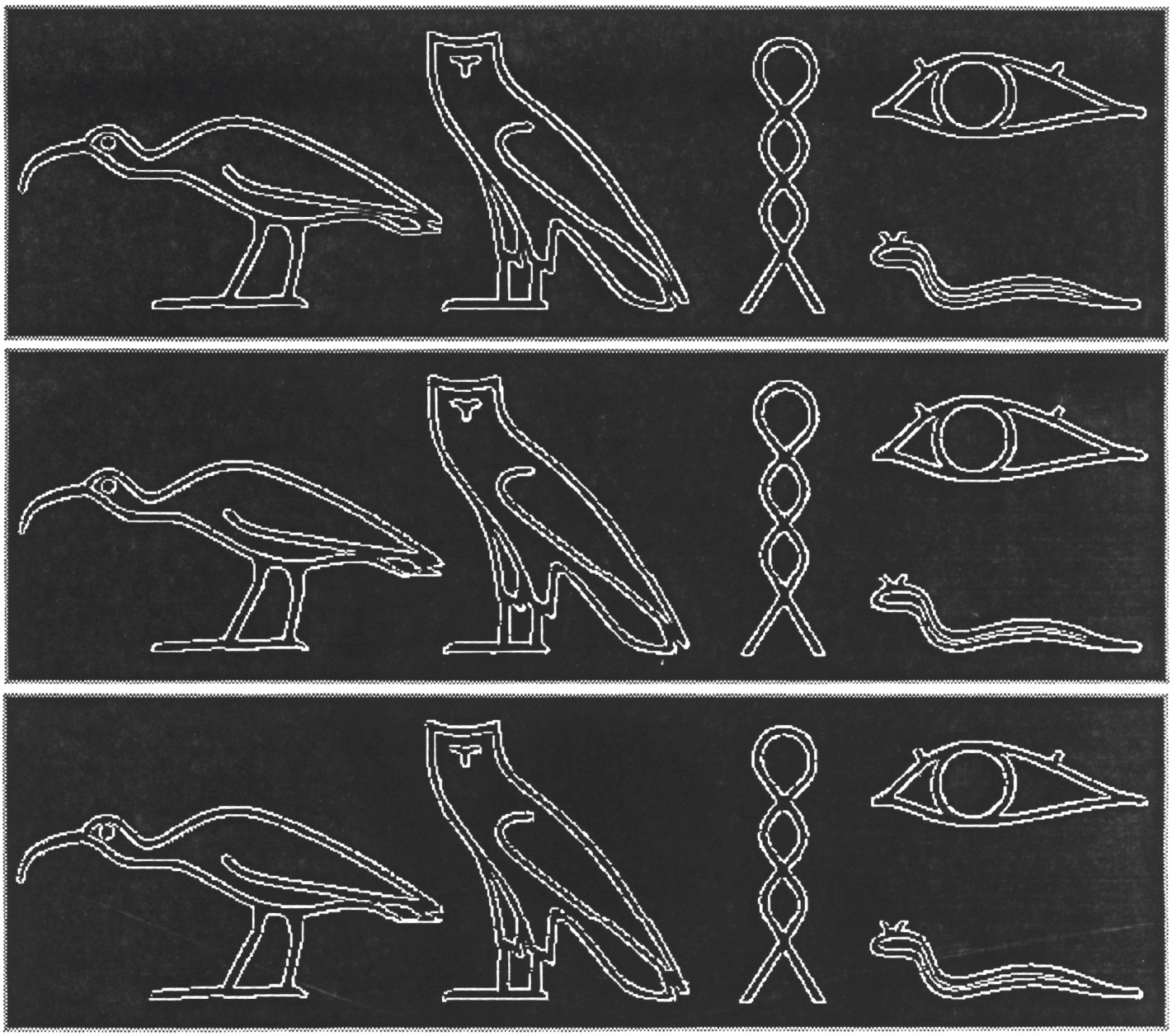


Figure 13: Results from edge detection on Hieroglyphs image (after to post-processing). Results from Deriche operator with $\alpha = 1.75$ (top); Results from RITE operator with 5×5 sampling (middle), and 3×3 sampling (bottom).

Appendix B

Below are the nine 5×5 wedge filters used by the RITE operator (Algorithm 1).

	0	1	2	3	4
0	0.0	0.1	2.2	5.0	0.0
1	0.0	0.0	6.1	41.7	14.8
2	0.0	0.0	0.0	97.5	34.6
3	0.0	0.0	6.1	41.7	14.8
4	0.0	0.1	2.2	5.0	0.0

	0	1	2	3	4
0	0.0	1.8	0.4	0.0	0.0
1	9.3	20.1	1.1	0.0	0.0
2	30.6	86.3	0.0	0.0	0.0
3	18.1	64.8	19.8	0.5	0.0
4	0.0	10.2	7.0	0.7	0.0

	0	1	2	3	4
0	0.0	2.6	15.7	15.5	0.0
1	0.2	3.4	44.4	77.9	17.4
2	0.0	0.0	0.0	59.3	21.0
3	0.0	0.0	0.1	6.9	4.4
4	0.0	0.0	0.0	0.4	0.0

	0	1	2	3	4
0	0.0	0.0	0.0	0.0	0.0
1	1.4	1.5	0.0	0.0	0.0
2	10.9	30.9	0.0	0.4	0.1
3	13.0	73.3	73.9	12.3	0.9
4	0.0	18.3	26.2	6.6	0.0

	0	1	2	3	4
0	0.0	12.1	33.5	16.9	0.0
1	3.1	30.1	94.6	53.7	7.4
2	1.0	2.8	0.0	11.6	4.1
3	0.0	0.0	0.0	0.2	0.3
4	0.0	0.0	0.0	0.0	0.0

	0	1	2	3	4
0	0.0	0.0	0.0	0.0	0.0
1	0.0	0.0	0.0	0.2	0.3
2	1.0	2.8	0.0	11.6	4.1
3	3.1	30.1	94.6	53.7	7.4
4	0.0	12.1	33.5	16.9	0.0

	0	1	2	3	4
0	0.0	18.3	26.2	6.6	0.0
1	13.0	73.3	73.9	12.3	0.9
2	10.9	30.9	0.0	0.4	0.1
3	1.4	1.5	0.0	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0

	0	1	2	3	4
0	0.0	0.0	0.0	0.4	0.0
1	0.0	0.0	0.1	6.9	4.4
2	0.0	0.0	0.0	59.3	21.0
3	0.2	3.4	44.4	77.9	17.4
4	0.0	2.6	15.7	15.5	0.0

	0	1	2	3	4
0	0.0	10.2	7.0	0.7	0.0
1	18.1	64.8	19.8	0.5	0.0
2	30.6	86.3	0.0	0.0	0.0
3	9.3	20.1	1.1	0.0	0.0
4	0.0	1.8	0.4	0.0	0.0

References

[1] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–697, 1986.

[2] L. S. Davis. A survey of edge detection techniques. *Computer Graphics and Image Processing*, 4:248–270, 1975.

[3] R. Deriche. Using canny's criteria to derive a recursively implemented optimal edge detector. *International Journal of Computer Vision*, 1(1), 1987.

[4] W. T. Freeman and E. H. Adelson. The design and use of steerable filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(9):891–906, 1991.

[5] W. Gardner. A unifying view of second-order measures for quality for signal classification. *IEEE Trans. Comm.*, COM-28:807–816, 1980.

[6] L. A. Iverson and S. W. Zucker. *Logical/linear operators for image curves.* Technical Report TR-CIM-92-12, McGill University, 1992.

[7] L. G. Roberts. Machine perception of three-dimensional solids. In a. a. J P Tippett, editor, *Optical and Electro-optical Information Processing*, MIT Press, Cambridge, MA:, 1965.

[8] E. P. Simoncelli. *Distributed Analysis and Representation of Visual Motion.* PhD thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, Cambridge, MA, January 1993. Also available as MIT Media Laboratory Vision and Modeling Technical Report #209.