Optimal Local Weighted Averaging Methods in Contour Smoothing

R. Legault and C.Y. Suen

The authors are with the Centre for Pattern Recognition and Machine Intelligence, Concordia University, Montreal, Canada H3G 1M8

Abstract

In several applications where binary contours are used to represent and classify patterns, smoothing must be performed to attenuate noise and quantization error. This is often implemented with local weighted averaging of contour point coordinates, because of the simplicity, low-cost and effectiveness of such methods. Invoking the 'optimality' of the Gaussian filter, many authors will use Gaussian-derived weights. But generally these filters are not optimal, and there has been little theoretical investigation of local weighted averaging methods per se. This paper focusses on the direct derivation of optimal local weighted averaging methods *tailored towards specific computational goals* such as the accurate estimation of contour point positions, tangent slopes, or deviation angles. A new and simple digitization noise model is proposed to derive the best set of weights for different window sizes, *for each computational task.* Estimates of the fraction of the noise actually removed by these optimum weights are also obtained. Finally, the applicability of these findings for arbitrary curvature is verified, by numerically investigating equivalent problems for digital circles of various radii.

Keywords

Contour Smoothing; Optimal Local Weighted Averaging; Digitization Noise Modelling; Gaussian Smoothing.

I. Introduction

There are numerous applications involving the processing of 2-D images, and 2-D views of 3-D images, where binary contours are used to represent and classify patterns of interest. Measurements are then made using the contour information (e.g. perimeter, area, moments, slopes, curvature, deviation angles etc.). To obtain reliable estimates of these quantities, one must take into account the noisy nature of binary contours due to discrete sampling, binarization, and possibly the inherent fuzziness of the boundaries themselves[1]. In some cases, this can be done explicitly and exhaustively (see Worring & Smeulders [1] on curvature estimation). But more frequently it is done implicitly by smoothing. Following this operation, the measurements of interest can be obtained directly from the smoothed contour points, as in this paper, or from a curve fitted to these points. For a recent example of this last approach, see Tsai & Chen [2].

The *smoothing of binary contours or curves* for local feature extraction directly from

[1] For example, the "borders" of strokes in handwriting.

the discrete data is the focus of this article. More precisely, we will investigate *optimum local weighted averaging methods* for particular measurement purposes such as estimating point positions, derivatives (slopes of tangents), and deviation angles from point to point (see Fig. 2).

In this article, the following definitions will be used. Let $\mathbf{p}_i = (x_i, y_i)$, $i = 1, 2, \ldots N$, be the sequence of N points (4- or 8-connected) around the closed contour. Since the contour is cyclic, $\mathbf{p}_{N+i} = \mathbf{p}_i$ and $\mathbf{p}_{1-i} = \mathbf{p}_{N+1-i}$. Furthermore, let $\mathbf{v}_i = \mathbf{p}_i - \mathbf{p}_{i-1}$, and θ_i be the counter-clockwise elevation angle between $\mathbf{v}_i$ and the horizontal x-axis. We have $\theta_i = c_i \cdot \frac{\pi}{4}$ where c_i is the Freeman [3] chain code (see Fig. 1). For 4-connectivity, the values of c_i are limited to even values. We also define d_i, the differential chain code, as $d_i \equiv (c_{i+1} - c_i + 11) \bmod 8 \; - \; 3$.

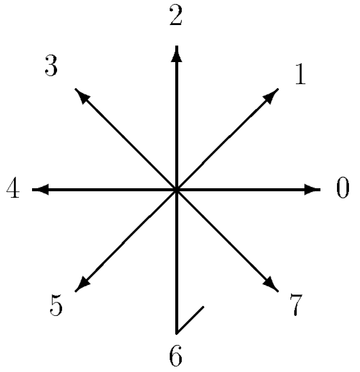


Figure 1: Freeman chain code

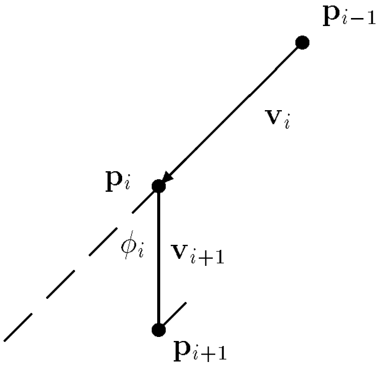


Figure 2: Deviation angle at $\mathbf{p}_i$

The *deviation angle* at point $\mathbf{p}_i$ will be denoted ϕ_i. It is the angle between the small vectors $\mathbf{v}_i$ and $\mathbf{v}_{i+1}$ (See Fig. 2). Of course we have $\phi_i = d_i \cdot \frac{\pi}{4}$.

Finally, the local weighted averaging method investigated in sections II, III, and IV is defined as:

$$\mathbf{p}_i^{(k)} = \sum_{j=-n}^{n} \alpha_j \, \mathbf{p}_{i+j}^{(k-1)}, \;\; k = 1, 2, \ldots, K \tag{1}$$

where $\mathbf{p}_i^{(k)}$ is the contour point i after k smoothing steps ($\mathbf{p}_i^{(0)} = \mathbf{p}_i$). The window size is $w = (2n + 1)$.

A. Variety of Approaches Used in Applications

Due to limited computing power, early methods were quite simple and found justification in their "good results". Thus we find schemes removing/filling one-pixel wide protrusions/intrusions based on templates, or replacing certain pairs in the chain code sequence by other pairs or by singletons ([4]). However, from early on, local weighted averaging methods are the most frequently used. They are applied to differential chain codes ([5], [6], [7]), possibly with compensation for the anisotropy of the square grid ([8]); they are applied to cartesian coordinates ([9]), possibly with weights depending on neighboring pixel configuration ([10]) or varying with successive iterations ([11]); they are applied to deviation angles ([12]).

With advances in computing power and insight into the smoothing problem, more complex methods were developed with more solid theoretical foundations. In this process, "Gaussian smoothing" has become very popular. One approach consists of applying local weighted averaging with Gaussian weights. Dill et al. [13] use normalized Gaussian-smoothed differential chain codes with fixed σ and window size. In Ansari & Huang [14], the weights and window size may vary from point to point based on

$$\mathbf{p}_i^{(1)} = \sum_{j=-n_i}^{n_i} \alpha_{ij}\mathbf{p}_{i+j}, \quad \text{with} \quad \alpha_{ij} = \frac{1}{\sqrt{2\pi}\sigma_i} e^{-\frac{j^2}{2\sigma_i^2}}, \tag{2}$$

where $\sigma_i \;=\; 0.33\,n_i$ and n_i is given by Teh & Chin's [15] *region of support.* See also Pei & Horng [16].

Variable amounts of smoothing can be applied to the entire curve, taking the overall behaviour of the smoothed curve across scale as its complete description. Witkin [17] convolves a signal $f(x)$ with Gaussian masks over a continuum of sizes:

$$F(x,\sigma) = f(x) * G(x,\sigma) = \int_{-\infty}^{\infty} f(u) \cdot \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-u)^2}{2\sigma^2}} du. \tag{3}$$

$F(x,\sigma)$ is called the *scale-space image* of $f(x)$ and it is analyzed in terms of its inflection points. This concept of scale-space was also originally explored by Koenderink [18].

Asada & Brady [19] analyze the convolution of simple shape primitives with the first and second derivatives of $G(x,\sigma)$ and then use the knowledge gained to extract these shape primitives from the contours of objects. Mokhtarian & Mackworth [20] compute the Gaussian-smoothed curvature using

$$\kappa(t,\sigma) = \frac{\dot{X}(t,\sigma)\ddot{Y}(t,\sigma) - \dot{Y}(t,\sigma)\ddot{X}(t,\sigma)}{(\dot{X}^2(t,\sigma) + \dot{Y}^2(t,\sigma))^{3/2}}, \tag{4}$$

where $X(t,\sigma)$ and $Y(t,\sigma)$ are the coordinates (x(t),y(t)) convolved with a Gaussian filter $G(t,\sigma)$. The locus of points where $\kappa(t,\sigma) = 0$ is called the *generalized scale-space image* of the curve which they use for image matching purposes. Wuescher & Boyer [21] also use Gaussian-smoothed curvature but with a single σ to extract regions of constant curvature from contours.

Multiscale shape representations are not necessarily associated with Gaussian smoothing. Saint-Marc et al. [22] propose scale-space representations based on *adaptive smoothing*, which is a local weighted averaging method where the smoothed signal $S^{(k+1)}(x)$ after $(k+1)$ smoothing steps is obtained as:

$$S^{(k+1)}(x) = \frac{1}{\Gamma(x)} \sum_{j=-1}^{1} \gamma^{(k)}(x+j)\, S^{(k)}(x+j), \quad k = 0, 1, \ldots, K, \quad \text{with} \tag{5}$$

$$\gamma^{(k)}(x+j) = e^{-\frac{\left|\frac{\partial S^{(k)}}{\partial x}(x+j)\right|^2}{2t^2}}, \quad \Gamma(x) = \sum_{j=-1}^{1} \gamma^{(k)}(x+j). \tag{6}$$

Several multiscale shape representations based on non-linear filters have also been used successfully. Maragos [23] investigated morphological opening/closing filters depending on a structuring element and a scale parameter; using successive applications of these operators and removing some redundancy, a collection of skeleton components (Reduced Skeleton Transform) can be generated which represents the original shape at various scales more compactly than multiscale filtered versions. See also Chen & Yan [24].

Recently Bangham et al. [25] used scale-space representations based on M- and N-sieves. For any 1D discrete signal $f : \mathcal{Z} \rightarrow \mathcal{Z}$, denoting as C_r the set of all intervals of r consecutive integers, they define:

$$\begin{aligned} \gamma_r f(x) &= \min_{\xi \in \mathrm{C_r} : \xi \ni \mathrm{x}} \; \max_{\mathrm{u} \in \xi} \; f(u) \\ \psi_r f(x) &= \max_{\xi \in \mathrm{C_r} : \xi \ni \mathrm{x}} \; \min_{\mathrm{u} \in \xi} \; f(u) \end{aligned} \tag{7}$$

Using $\mathcal{N}^r = \psi_r\gamma_r$ and $\mathcal{M}^r = \gamma_r\psi_r$, the N-sieves of f are the sequence: $f_1 = f$ and $f_{r+1} = \mathcal{N}^{r+1}f_r$, for $r \geq 1$. The M-sieves of f are defined similarly, using $\mathcal{M}$. Applying sieves horizontally and vertically to 2D images appears to preserve edges and reject impulsive noise better than Gaussian smoothing.

B. Theoretical Foundations

Regularization theory and the study of scale-space kernels are the two main areas which have provided insight into the special qualities of the Gaussian kernel for smoothing purposes. As will be seen, they do not warrant unqualified statements about the 'optimality' of Gaussian smoothing.

Consider a one-dimensional function $g(x)$, corrupted by noise $\eta(x)$. The observed signal is then $y(x) = g(x) + \eta(x)$. Assume that the information available is a sampling of this signal $y_1, y_2, \ldots, y_n$ obtained for $x = x_1, x_2, \ldots, x_n$. Here $x_i < x_{i+1}$. One approach to estimating $g(x)$ is to find $f(x)$ which minimizes

$$\frac{1}{n}\sum_{i=1}^{n}(f(x_i) - y_i)^2 \quad + \quad \lambda \int_{x_1}^{x_n}[f^{(m)}(u)]^2 du, \tag{8}$$

where λ is the *regularization parameter*. The solution is a *smoothing spline* of order $2m$ (see [26], [27]).

For *equally spaced data* and $m = 2$, Poggio et al. [28] have shown that the cubic spline solution is *very similar* to a Gaussian. Canny's paper on edge detection[29] is also cited to support the optimality of Gaussian filtering. But the Gaussian is only an *approximation* to his theoretically obtained optimal filter.

Babaud et al. [30] have considered the class of infinitely differentiable kernels $g(x, y)$ vanishing at infinity faster than any inverse of polynomial, and one-dimensional signals $f(x)$ that are continuous linear functionals on the space of these kernels. In this class, they have shown that only the Gaussian $g(x, y) = \frac{1}{\sqrt{2\pi}}ye^{-1/2(xy)^2}$ can guarantee that all first-order maxima (or minima) of the convolution

$$\phi(x, y) = f(x) * g(x, y) = \int_{-\infty}^{\infty} f(u)g(x - u, y)\, du \tag{9}$$

will increase (or decrease) monotonically as y increases.

Yuille & Poggio [31] extended the previous result by showing that, *in any dimension*, the Gaussian is the only *linear* filter that does not create generic zero crossings of the Laplacian as the scale increases. In the literature, such demonstrations are often designated as the scaling theorem. Mackworth & Mokhtarian [32] derived a similar result concerning their curvature scale-space representation. Wu & Xie [33] developed an elementary proof of the scaling theorem based on calculus, assuming signals represented by polynomials. Very recently, Anh et al. [34] provided another proof applicable to the broader class of *band-limited* signals and to a larger class of filtering kernels, by relaxing the smoothness constraint.

This property of the Gaussian filter was established for *continuous* signals. Lindeberg [35] has studied the similar problem for discrete signals. He postulated that scale-space should be generated by convolution with a one-parameter family of kernels and that the number of local extrema in the convolved signal $K * f$ should not exceed the number of local extrema in the original signal. Imposing a semigroup requirement, he found the unique one-parameter family of *scale-space kernels* $T(n;t)$ with a *continuous* scale parameter t; *as* t *increases*, it becomes less and less distinguishable from the discretized Gaussian.

Recently, Pauwels et al. [36] demonstrated that imposing recursivity and scale-invariance on linear, isotropic, convolution filters narrows down the class of scale-space kernels to a one-parameter family of filters *but is not restrictive enough to single out the Gaussian.* The latter results for a parameter value of 2; for higher values, the kernel has zero crossings. Pauwels et al. also derive Lindeberg's results as special cases of theirs.

Finally, Bangham et al. [25] showed that discrete 1D M-sieves and N-sieves do not introduce new edges or create new extrema as the scale increases. In addition, Bangham et al. [37] have proven that when differences are taken between sieving operations applied recursively with increasing scale on 1D discrete signals, a set of granule functions are obtained which map back to the original signal. Furthermore, the amplitudes of the granules are, to a certain extent, independent of one another.

C. Practical Considerations

For practical applications, regardless of the smoothing method one decides to use, some concrete questions must eventually be answered. For the regularization approach, what value should be used for $\lambda\Gamma$ For Gaussian smoothing, what value of σ and what finite window sizeΓ When scale-space representation is used, if we say that significant features are those which survive over "a wide range of scale", we must eventually put some actual figures on this 'wide' range. These decisions can be entirely data-driven or based on prior experience, knowledge of particular applications etc. In the end, they may play a significant role in both the performance of the selected method and its implementation cost. We now briefly present some of these aspects.

In [38] [39], Shahraray & Anderson consider the regularization problem of equation 8, for $m = 4$, and they argue that finding the best value of λ is critical. For this purpose, they propose a technique based on minimizing the cross-validation mean square error (CVMSE):

$$\mathrm{CVMSE}(\lambda) = \frac{1}{n}\sum_{k=1}^{n}(g_{n,\lambda}^{[k]}(x_k) - y_k)^2, \tag{10}$$

where $g_{n,\lambda}^{[k]}$ is the smoothing spline constructed using all samples *except* y_k, and is then used to estimate y_k. The method is said to provide a very good estimate of the best λ, *for equally-spaced periodic data assuming only a global minimum.* Otherwise, a so-called generalized cross-validation function must be used.

The presence of discontinuities to be preserved in the contours of interest brings more complexity into the optimal smoothing problem. One possible solution was already mentioned: the adaptive smoothing of Saint-Marc et al. [22]. For one-dimensional regularization which preserves discontinuities, see Lee & Pavlidis [40]. For two-dimensional regularization which preserves discontinuities, see Chen & Chin [41]. For Gaussian smoothing which preserves discontinuities, see the methods of Ansari & Huang [14] and of Brady et al. [42].

Another problem is that repeated convolution of a closed curve with a kernel may not yield a closed contour or may cause shrinkage. For simple, *convex*, closed curves, Horn & Weldon [43] propose a new curve representation which guarantees closed curves. Mackworth & Mokhtarian [32] offer a different solution involving the reparametrization of the Gaussian-smoothed curve by its normalized arc length parameter. Lowe [44] attenuates shrinking by adding a correction to the Gaussian-smoothed curve. Oliensis [45] applies FFT to the signal and resets amplitudes to 0 only for frequencies larger than some threshold. Recently, Wheeler & Ikeuchi [46] present a new low-pass filter based on the idea of iteratively adding smoothed residuals to the smoothed signals. Results are said to be comparable to Oliensis' and somewhat better than Lowe's.

Li & Chen [47] investigate a possible solution to the high computation and storage requirements of generating "continuous" scale-space. They show that an optimal $\mathcal{L}_1$ approximation of the Gaussian can be obtained with a finite number of basis Gaussian filters:

$$G(x,\sigma) \;\approx\; \sum_{i=1}^{n} w_i(\sigma) G(x,\sigma_i) \tag{11}$$

from which scale-space can be constructed efficiently.

The above discussion exemplifies the potential complexity involved in implementing 'optimal' methods. Clearly, in practice, one should not lose track of the cost of these operations and how much smoothing is really required by the application of interest. It is not always necessary to attain the ultimate precision in every measurement. In many situations, simple and fast methods such as local weighted averaging with fixed weights and a small window size, will provide a very satisfactory solution in only 2 or 3 iterations (see [11], [5], [12]). Moreover, there is often little difference in the results obtained via different methods. Thus, Dill et al. [13] report similar results when a Gaussian filter and a triangular (Gallus-Neurath) filter of the same width are applied to differential chain codes; in Kasvand & Otsu [48], rectangular, triangular, and Gaussian kernels, with the same standard deviation, yield comparable outcomes (especially the latter two) for the smooth reconstruction of planar curves from piecewise linear approximations.

D. Present Work

Our interest in contour smoothing stems from practical work in handwriting recognition. In this and other applications, the discrimination of meaningful information from 'noise' is a complex problem which often plays a critical role in the overall success of the system. This filtering process can be handled across several stages (preprocessing, feature extraction, even classification), with different methods. In the preprocessing stage of one of our recognition schemes [49], a triangular filter with $w = 5$ was first applied to contours of characters before deviation angles ϕ_i were computed. Satisfactory results were obtained. Nevertheless, we were curious about the optimality of our choice of local weights.

Initial review revealed that, in many practical applications, the smoothing operation is still performed by some local weighted averaging schemes[2] because they are simple, fast, and effective (see for example [14], [50], [51], [52], [21]). However, little theoretical investigation of these methods *per se* has been conducted. Some authors rapidly invoke the 'optimality' of Gaussian filtering and use Gaussian-derived weights. Their results may be satisfactory as the Gaussian may be a good *approximation* to the 'optimum' filter, but its discretization and truncation may cause it to further depart from 'optimum' behaviour...

In this paper, we assume local weighted averaging with constant weights *as a starting point* and we investigate how these smoothing methods handle small random noise. To this end, we propose a simple model of a noisy horizontal border. The simplicity of the model allows a very pointed analysis of these smoothing methods. More precisely, for specific computational goals such as estimating contour point positions, derivatives (slopes of tangents), or deviation angles from the pixels of binary contours, we answer the following questions: what are the optimum fixed weights for a given window sizeΓ and what fraction of the noise is actually removed by these optimum weightsΓ

After deriving these results, we offer experimental evidence that their validity is not restricted to the limited case of noisy horizontal borders. This is done by considering digital circles. For each particular computational task, we find very close agreement between the optimum weights derived from our simple model and the ones derived numerically for circles over a wide range of radii.

An important side-result concerns the great caution which should be exercised in speaking of 'optimal' smoothing. Even for our simple idealized model, we find that the smoothing coefficients which best restore the original noise-free pixel *positions* are not the same which best restore the original local *slope*, or the original local *deviation angles*; furthermore, the best smoothing coefficients even depend on the specific difference method used to numerically estimate the slope. Hence, in choosing smoothing methods, researchers should probably first consider *what* it is they intend to measure after smoothing and *in what manner*.

In relation to this, we point out the work of Worring & Smeulders [1]. They analyze

[2] Once the Gaussian is discretized and truncated, it also simply amounts to a local weighted averaging method with particular weight values.

noise-free digitized circular arcs and exhaustively characterize *all centers and radii* which yield a given digitization pattern; by averaging over all these, an optimum measure of *radius* or *curvature* can be obtained. If radius or curvature is the measurement of interest and if utmost precision is required (with the associated computing cost to be paid), then their approach is most suitable.

Our work in contrast is not oriented towards measuring a single attribute. We focus on measurements such as position, slope and deviation angles because they are often of interest. But our model and approach can be used to investigate other quantities or other numerical estimates of the same quantities. The methods may be less accurate but they will be much less costly, and optimum in the category of local weighted averaging methods. The requirements of specific applications should dictate what is the best trade-off.

The rest of the article is organized as follows. The next section briefly describes the methods investigated and provides a geometric interpretation for them. In section III, the simple model of a noisy horizontal border is used to derive *optimal* values of the smoothing parameters, in view of the above-mentioned computational goals. Finally, the applicability of our findings for varying curvature is explored experimentally in section IV.

II. Local Weighted Averaging

We begin our study of local weighted averaging, as defined by Eq. 1, with window size $w = 2n + 1$. Of course, the smoothed contour points $\mathbf{p}_i^{(k)}$, after k smoothing iterations, can be obtained directly from the original points $\mathbf{p}_i$ as

$$\mathbf{p}_i^{(k)} = \sum_{j=-n'}^{n'} \beta_j \, \mathbf{p}_{i+j}, \tag{12}$$

where $n' = k\,n$, corresponding to a window size $w' = k(w-1)+1$, and the β's are functions of the α's and of k. The form of Eq. 1 is often computationnally more convenient. However, as long as k and n are finite, the study of local weighted averaging need only consider the case of a single iteration with *finite* width filters. When this is done, we will use the simpler notation $\mathbf{p}'_i$ instead of $\mathbf{p}_i^{(1)}$:

We now impose a simple requirement to this large family of methods. Since our goal is to smooth the small 'wiggles' along boundaries of binary images, it seems reasonable to require that when $\mathbf{p}_i$ and its neighbouring contour pixels are perfectly aligned, the smoothing operation should leave $\mathbf{p}_i$ unchanged. In particular, consider the x-coordinates of consecutive horizontally-aligned pixels from $\mathbf{p}_{i-n}$ to $\mathbf{p}_{i+n}$. For $j \in [-n, n]$, we have $x_{i+j} = x_i + j$. Our requirement that $x'_i = x_i$ then becomes

$$x'_i = \sum_{j=-n}^{n} \alpha_j (x_i + j) \;=\; x_i \sum_{j=-n}^{n} \alpha_j \;+\; \sum_{j=1}^{n} (\alpha_j - \alpha_{-j})\, j \;=\; x_i. \tag{13}$$

For this to hold whatever the value of x_i, we must have

$$\sum_{j=-n}^{n} \alpha_j = 1; \qquad \text{and} \qquad \alpha_{-j} = \alpha_j, \quad j = 1, 2, \ldots, n \tag{14}$$

Thus our requirement is equivalent to a normalization condition and a symmetry constraint on the α's.

A. Geometric Interpretation

It is a simple matter to find a geometric interpretation for local weighted averaging. Using the above conditions, Eq. 1 can be rewritten as:

$$\mathbf{p}_i^{(k)} = \mathbf{p}_i^{(k-1)} + \sum_{j=1}^{n} 2\alpha_j [\frac{\mathbf{p}_{i-j}^{(k-1)} + \mathbf{p}_{i+j}^{(k-1)}}{2} - \mathbf{p}_i^{(k-1)}]. \tag{15}$$

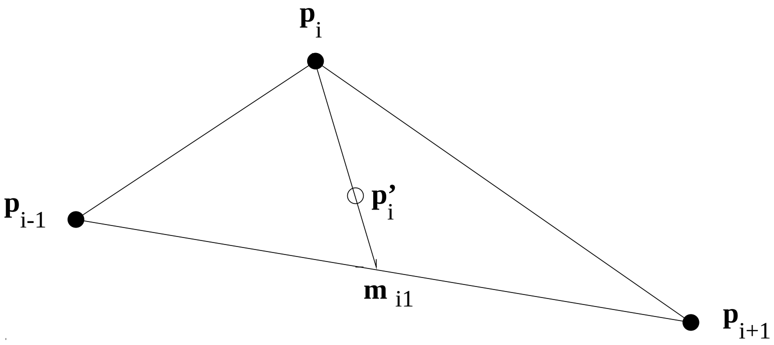


Fig. 3. Geometric interpretation for $w = 3$.

For a single iteration ($k = 1$) of the simplest method ($n = 1$), the last Eq. reduces to

$$\mathbf{p}'_i \;=\; \mathbf{p}_i \;+\; 2\alpha[\frac{\mathbf{p}_{i-1} + \mathbf{p}_{i+1}}{2} - \mathbf{p}_i]. \tag{16}$$

The points $\mathbf{p}_{i-1}, \mathbf{p}_i, \text{and} \mathbf{p}_{i+1}$ are generally not aligned and the situation is illustrated in Fig. 3, where $\mathbf{m}_{i1} = (\mathbf{p}_{i-1} + \mathbf{p}_{i+1})/2$ is the middle of the base of the triangle. Eq. 16

implies that the smoothed point $\mathbf{p}'_i$ is always on the *median* of the triangle from point $\mathbf{p}_i$. Furthermore, the effect of the unique coefficient α is clear since $|\mathbf{p}_i\mathbf{p}'_i|/|\mathbf{p}_i\mathbf{m}_{i1}| = 2\alpha$. As α varies continuously from 0 to 0.5, $\mathbf{p}'_i$ 'slides' from $\mathbf{p}_i$ to $\mathbf{m}_{i1}$.

Similarly, in the more general situation, the vectors $[\frac{\mathbf{p}_{i-j}^{(k-1)} + \mathbf{p}_{i+j}^{(k-1)}}{2} - \mathbf{p}_i^{(k-1)}]$ are the *medians* from $\mathbf{p}_i^{(k-1)}$ of the triangles $\Delta\mathbf{p}_{i-j}^{(k-1)}\mathbf{p}_i^{(k-1)}\mathbf{p}_{i+j}^{(k-1)}$. Eq. 15 indicates that the smoothed point $\mathbf{p}_i^{(k)}$ is obtained by adding to $\mathbf{p}_i^{(k-1)}$ a weighted sum of the medians of these triangles, using $2\alpha_j$ as weights. Thus, in a *geometric* sense, local weighted averaging as a contour smoothing method could be renamed *median smoothing*.

III. Optimum Results from a Simple Digitization Noise Model

This section addresses the question “If local weighted averaging is considered, what constant coefficients α_j should be used for smoothing binary contours in view of specific computational goalsΓ”. We develop an answer to this question, based on a simple model: *an infinite horizontal border with random 1-pixel noise.*

Why use this modelΓ Of course, we do not consider the horizontal line to be a very general object. Nor do we think that noise on any particular binary contour is a random phenomenom. We have noticed in our work that binary contours often bear small noise, commonly “1-pixel wiggles”. Our goal is to perform an *analytical study* of the ability of local weighted averaging smoothing methods to remove such noise. Since the filters are meant to be used with arbitrary binary contours, it seems reasonable to consider that over a large set of images noise can be considered random.

Furthermore, we do not make the very frequent implicit assumption that a smoothing filter can be optimal independently of the specific attributes one intends to measure or even the specific numerical estimation method used. For specific measurements and computation methods, we would like to find the best choice of smoothing coefficients for a given window size and an estimate of how much noise these coefficients remove; if the window size is increased [3], what are then the best coefficients and how much more is gained compared to the smaller window sizeΓ

These questions are very pointed and we have no workable expression for small random

[3] assuming the feature structure scale allows this.

noise on an *arbitrary* binary contour which would allow to derive answers *analytically*. Thus we choose to look at an ideal object for which we can easily model random 1-pixel noise and our study can be carried out. Similar approaches are often followed. For example, in studying optimal edge detectors, Canny [29] considers the ideal step edge. There is no implication that this is a common object to detect in practice; simply it makes the analytical investigation easier and can still allow to gain insight into the edge detection problem more generally. The practicality of our own findings concerning optimal local weighted averaging will be verified in section IV. We now give a definition for our simple model.

The infinite horizontal border with random 1-pixel noise consists of all points (x_i, y_i), $i \in \mathcal{Z}$, satisfying

$$x_i = i, \quad \forall i \in \mathcal{Z} \tag{17}$$

$$y_i = \begin{cases} y_o, & \text{with probability } (1-p) \\ y_o + 1, & \text{with probability } p \end{cases} \tag{18}$$

An example of such a simple noisy boundary is shown in Fig. 4.

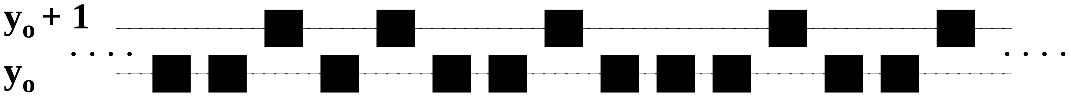


Fig. 4. Noisy Horizontal Border

With this model, $x_{i-j} = x_i - j$ and $x_{i+j} = x_i + j$. It then follows from Eq. 15 that the smoothing operation will not change the x-coordinates and the local weighted averaging will only affect the y-coordinates. The best fitting straight line through this initial data is easy to obtain since it must be of the form $y = \tilde{y}$. It is obtained by minimizing the mean square distance

$$\overline{d^2} = (1-p)(\tilde{y} - y_o)^2 \; + \; p(\tilde{y} - y_o - 1)^2 \tag{19}$$

with respect to $\tilde{y}$. The best fitting line is simply $y = y_o + p$. We will consider this to be the Eq. of the ideal border which has been corrupted by the digitization process, yielding the situation of Eq. 18.

We now examine the problem of applying "optimal" local weighted averaging to the data of our simple model. Our aim is to eliminate the 'wiggles' along the noisy horizontal border as much as possible. An alternate formulation is that we would want the border, after smoothing, to be "as straight as possible" and as close as possible to $y = y_o + p$.

Several criteria can be used to assess the straightness of the border and optimize the smoothing process:

- Minimize the mean square *distance* to the best fitting line after the data has been smoothed.
- Minimize the mean square *slope* along the smoothed data points (ideally, the border is straight and its slope should be 0 everywhere).
- Minimize the mean square *deviation angle* ϕ_i (see Fig. 2) along the smoothed data (ideally, ϕ_i should also be 0 everywhere).

Each of the above criteria is sound and none can be said to be the *best* without considering the particular situation further. The first criterion is the most commonly used in the curve fitting literature. In this paper however, we want to derive optimal smoothing methods *tailored for specific computational tasks*; hence, we will consider each of the above criteria in turn. If our interest is simply to obtain numerical estimates of the *slopes* at contour points, the optimal α_j's derived based on the second criterion should be preferred. And for estimating *deviation angles* ϕ_i, the optimal coefficients derived from the third criterion would be better.

For the first criterion, we will use d_{rms}, the root mean square (r.m.s.) distance to the best fitting line, as our measure of noise before and after the smoothing step; for the second criterion, m_{rms}, the r.m.s. slope along the border; with the third criterion, ϕ_{rms}, the r.m.s. deviation angle along the border. For the original *unsmoothed* data, these noise measures can be computed using the probabilities of the possible configurations of 2 or 3 consecutive pixels. Thus for the original, unsmoothed data we have:

$$d_{rms} = \sqrt{p(1-p)} \tag{20}$$

$$m_{rms} = \begin{cases} \sqrt{2p(1-p)} & \text{using } \; m_i = y_{i+1} - y_i \\ \sqrt{\dfrac{p(1-p)}{2}} & \text{using } \; m_i = (y_{i+1} - y_{i-1})/2 \end{cases} \tag{21}$$

$$\phi_{rms} = \frac{\pi}{2}\sqrt{\frac{3p(1-p)}{2}} \tag{22}$$

Based on our simple model, we now derive the best smoothing parameters for each of the three criteria mentioned above. Once obtained, we will compute the corresponding noise measures for the *smoothed data* which we will denote by ${}_{[w]}d'_{rms}$, ${}_{[w]}m'_{rms}$, and${}_{[w]}\phi'_{rms}$ respectively. In this notation the 'prime' indicates a single smoothing step and w is the window size used.

A. Best Parameters to Minimize d'_{rms}

The unsmoothed y-coordinate of our border points is a discrete random variable following a simple Bernouilli distribution for which the expected value is $y_o + p$ and the variance is $p(1-p)$. Obviously, the best fitting line is simply the expected value and d_{rms} in Eq. 20 is the standard deviation of y_i. After smoothing, the expected value of y'_i will remain $y_o + p$ for *any* (finite) window size w. Denoting the expected value by E, we have from elementary probability theory:

$$E(y'_i) = E(\sum_{j=-n}^{n} \alpha_j y_{i+j}) = \sum_{j=-n}^{n} \alpha_j E(y_{i+j}) = \sum_{j=-n}^{n} \alpha_j E(y_i) = E(y_i). \tag{23}$$

Now we find the best choice of smoothing parameters and the corresponding noise measure ${}_{[w]}d'_{rms}$. Denoting variance by $\sigma^2()$, we have:

$$\overline{d_i'^2} = \sigma^2(y'_i) = \sum_{j=-n}^{n} \sigma^2(\alpha_j y_{i+j}). \tag{24}$$

Each $\alpha_j y_{i+j}$ is a discrete random variable (with two possible values) and its variance is $p(1-p)\alpha_j^2$. Thus

$$\overline{d_i'^2} = p(1-p) \sum_{j=-n}^{n} \alpha_j^2. \tag{25}$$

We must now minimize $\sum\limits_{j=-n}^{n} \alpha_j^2$, subject to the constraint $\alpha_0 + 2\sum_{j=1}^{n} \alpha_j - 1 = 0$. This problem is typically solved using the Lagrange multipliers method (see [53], page 182). from which we obtain the simple result $\alpha_k = \alpha_0$, for each k. All coefficients are equal, hence of value $1/(2n+1)$ [4]. Substituting this value into Eq. 25, we obtain the corresponding

[4] As expected, straight averaging reduces the variance of a collection of random variables faster than any other weighted average.

$\overline{d_i'^2}$. Our findings can be summarized as follows:

- For a single smoothing iteration with arbitrary window size w, for *any* value of p, the best choice of parameters to minimize the mean square distance is to set all α_j's to $1/w$, resulting in ${}_{[w]}d'_{rms} = \sqrt{\frac{p(1-p)}{w}}$.

The fraction of the noise which is removed by the smoothing operation is $1 - \sqrt{\frac{1}{w}}$. Hence, for $w = 3$, the noise is reduced by 42.3%; for $w = 5$, by 55.3%; for $w = 7$, by 62.2%. Finally, contrary to what one might expect, we note that the optimum 5-point smoothing operation is *not* to apply the optimum 3-point operation twice. The latter is equivalent to a 5-point window with $\alpha_j = (3 - |j|)/9$ which gives $d'_{rms} = \frac{1}{9}\sqrt{19p(1-p)}$. This would remove approximately 51.6% of the noise.

B. Best Parameters to Minimize m'_{rms}

In this section, we apply our second criterion for straightness and minimize the root mean square value of the *slope* after smoothing. We consider two different ways of computing the slope from contour points.

B.1 Based on $m'_i = y'_{i+1} - y'_i$

The simplest numerical estimate of the slope is given by the forward difference formula $m'_i = y'_{i+1} - y'_i$. This gives

$$\begin{aligned} m_i'^2 &= (y'_{i+1} - y'_i)^2 \\ &= \left(\sum_{j=-n+1}^{n+1} \alpha_{j-1} y_{i+j} - \sum_{j=-n}^{n} \alpha_j y_{i+j} \right)^2 \\ &= (\alpha_n y_{i+1+n} - \alpha_{-n} y_{i-n})^2 + 2 \sum_{j=-n+1}^{n} \alpha_n (\alpha_{j-1} - \alpha_j) y_{i+1+n} y_{i+j} \\ &\qquad - 2 \sum_{j=-n+1}^{n} \alpha_{-n} (\alpha_{j-1} - \alpha_j) y_{i-n} y_{i+j} + \left(\sum_{j=-n+1}^{n} (\alpha_{j-1} - \alpha_j) y_{i+j} \right)^2 . (26) \end{aligned}$$

Expanding the last squared summation and taking the mean, we obtain the following expression:

$$\overline{m_i'^2} = \alpha_n^2 \overline{y_{i+1+n}^2} - 2\alpha_{-n}\alpha_n \overline{y_{i+1+n} y_{i-n}} + \alpha_{-n}^2 \overline{y_{i-n}^2}$$

$$+\,2\sum_{j=-n+1}^{n}\alpha_n(\alpha_{j-1}-\alpha_j)\overline{y_{i+1+n}y_{i+j}}\;-\;2\sum_{j=-n+1}^{n}\alpha_{-n}(\alpha_{j-1}-\alpha_j)\overline{y_{i-n}y_{i+j}}$$
$$+\sum_{j=-n+1}^{n}(\alpha_{j-1}-\alpha_j)^2\overline{y_{i+j}^2}\;+\;2\sum_{\substack{-n+1\\ j<k}}^{n}(\alpha_{j-1}-\alpha_j)(\alpha_{k-1}-\alpha_k)\overline{y_{i+j}y_{i+k}}. \qquad (27)$$

For any $s,t\in\mathcal{Z}$, $\overline{y_sy_t}\;=\;(y_o+p)^2$ and $\overline{y_s^2}\;=\;(y_o+p)^2\;+\;p(1-p)$. Substituting these results, Eq. 27 can be further simplified by making use of Eq. 14. We obtain:

$$\overline{m_i'^2}=2p(1-p)\left[\sum_{j=-n}^{n}\alpha_j^2+\sum_{j=-n+1}^{n}\alpha_{j-1}\alpha_j\right]. \qquad (28)$$

Using Eq. 14 again, some algebraic manipulation leads to an expression for $\overline{m_i'^2}$ involving only the n independent parameters $\alpha_1,\alpha_2,\ldots,\alpha_n$:

$$\begin{aligned}\overline{m_i'^2}\;&=\;2p(1-p)\Bigg[1-2\alpha_1+4(\alpha_1-1)\sum_{j=1}^{n}\alpha_j\\&\qquad+2\sum_{j=1}^{n}\alpha_j^2+4\left(\sum_{j=1}^{n}\alpha_j\right)^2\;-\;2\sum_{j=1}^{n-1}\alpha_j\alpha_{j+1}\Bigg].\end{aligned} \qquad (29)$$

Our task now is to minimize the mean squared slope. Differentiating this with respect to α_k, for $1\le k\le n$, we obtain a system of n linear equations as shown below.

$$\begin{pmatrix}10&5&6&6&6&6&6&\cdots&6&6&6&6\\5&6&3&4&4&4&4&\cdots&4&4&4&4\\6&3&6&3&4&4&4&\cdots&4&4&4&4\\6&4&3&6&3&4&4&\cdots&4&4&4&4\\6&4&4&3&6&3&4&\cdots&4&4&4&4\\6&4&4&4&3&6&3&\cdots&4&4&4&4\\&&\vdots&&&\vdots&&&&&\vdots&\\6&4&4&4&4&4&4&\cdots&4&3&6&3\\6&4&4&4&4&4&4&\cdots&4&4&3&6\end{pmatrix}\begin{pmatrix}\alpha_1\\\alpha_2\\\alpha_3\\\alpha_4\\\alpha_5\\\alpha_6\\\vdots\\\alpha_{n-1}\\\alpha_n\end{pmatrix}=\begin{pmatrix}3\\2\\2\\2\\2\\2\\\vdots\\2\\2\end{pmatrix} \qquad (30)$$

Solutions are given in Table I for $1\le n\le 6$. The column before last gives the fraction of the noise which is removed by the optimum smoothing method. For comparison purposes, the last column provides the equivalent result when all weights are set equal to $\frac{1}{w}$. Finally we note that for window size $w=5$, the triangular filter using $\alpha_j=(3-|j|)/9$ results in a noise reduction of 80.8%, slightly better than the equal-weights method.

n	α_0	$\alpha_{\pm1}$	$\alpha_{\pm2}$	$\alpha_{\pm3}$	$\alpha_{\pm4}$	$\alpha_{\pm5}$	$\alpha_{\pm6}$	$1-\frac{m'_{rms}}{m_{rms}}$	$1-\frac{m'_{rms}}{m_{rms}}$
								optimum	$\alpha_j=\frac{1}{w}$
1	$\frac{2}{5}$	$\frac{3}{10}$						0.684	0.667
2	$\frac{9}{35}$	$\frac{8}{35}$	$\frac{1}{7}$					0.831	0.800
3	$\frac{4}{21}$	$\frac{5}{28}$	$\frac{1}{7}$	$\frac{1}{12}$				0.891	0.857
4	$\frac{5}{33}$	$\frac{8}{55}$	$\frac{7}{55}$	$\frac{16}{165}$	$\frac{3}{55}$			0.922	0.889
5	$\frac{18}{143}$	$\frac{35}{286}$	$\frac{16}{143}$	$\frac{27}{286}$	$\frac{10}{143}$	$\frac{1}{26}$		0.941	0.909
6	$\frac{7}{65}$	$\frac{48}{455}$	$\frac{9}{91}$	$\frac{8}{91}$	$\frac{33}{455}$	$\frac{24}{455}$	$\frac{1}{35}$	0.953	0.923

TABLE I

Best parameters to minimize $(y'_{i+1}-y'_i)^2$ and fraction of noise removed

B.2 Based on $m'_i = \frac{1}{2}(y'_{i+1} - y'_{i-1})$

A more accurate[5] estimate of the slope is given by $m'_i = \frac{1}{2}(y'_{i+1} - y'_{i-1})$. Expanding it in terms of the original coordinates and following the same approach as in the preceding section, we arrive at:

$$\overline{m_i'^2} = \frac{p(1-p)}{2}\left[\sum_{j=-n}^{n}\alpha_j^2 + \sum_{j=-n+1}^{n-1}\alpha_{j-1}\alpha_{j+1}\right]. \tag{31}$$

When expressed in terms of the n independent α's, this becomes:

$$\begin{aligned}\overline{m_i'^2} \;&=\; \frac{p(1-p)}{2}\Bigg[1 \,-\, 2\alpha_2 \,-\, \alpha_1^2 \,+\, 4(\alpha_2-1)\sum_{j=1}^{n}\alpha_j \\ &\qquad + 2\sum_{j=1}^{n}\alpha_j^2 \,+\, 4\left(\sum_{j=1}^{n}\alpha_j\right)^2 \,-\, 2\sum_{j=1}^{n-2}\alpha_j\alpha_{j+2}\Bigg].\end{aligned} \tag{32}$$

Minimizing with respect to α_k, for $1 \le k \le n$, we obtain another set of n linear equations for which the solutions are listed in Table II for $1 \le n \le 6$.

Note that the distribution of coefficients from α_{-n} to α_n is no longer unimodal. Furthermore, $\alpha_0 < \alpha_1$ for odd values of n. Finally we note that for window size $w = 5$, the

[5] Provided the data resolution is fine enough.

n	α_0	$\alpha_{\pm 1}$	$\alpha_{\pm 2}$	$\alpha_{\pm 3}$	$\alpha_{\pm 4}$	$\alpha_{\pm 5}$	$\alpha_{\pm 6}$	$1-\frac{m'_{rms}}{m_{rms}}$	$1-\frac{m'_{rms}}{m_{rms}}$
								optimum	$\alpha_j = \frac{1}{w}$
1	$\frac{1}{5}$	$\frac{2}{5}$						0.553	0.529
2	$\frac{2}{7}$	$\frac{1}{7}$	$\frac{3}{14}$					0.733	0.717
3	$\frac{2}{15}$	$\frac{1}{5}$	$\frac{1}{10}$	$\frac{2}{15}$				0.817	0.798
4	$\frac{9}{55}$	$\frac{6}{55}$	$\frac{8}{55}$	$\frac{4}{55}$	$\frac{1}{11}$			0.865	0.843
5	$\frac{9}{91}$	$\frac{12}{91}$	$\frac{8}{91}$	$\frac{10}{91}$	$\frac{5}{91}$	$\frac{6}{91}$		0.895	0.871
6	$\frac{4}{35}$	$\frac{3}{35}$	$\frac{3}{28}$	$\frac{1}{14}$	$\frac{3}{35}$	$\frac{3}{70}$	$\frac{1}{20}$	0.915	0.891

TABLE II

Best parameters to minimize $\left(\frac{y'_{i+1}-y'_{i-1}}{2}\right)^2$ and fraction of noise removed

triangular filter using $\alpha_j = (3-|j|)/9$ results in a noise reduction of 66.7%, notably less than the equal-weights method.

C. Best Parameters to Minimize Deviation Angles

In this section, we examine the smoothing problem based on minimizing the deviation angles ϕ'_i. Here the problem is more complex and we will *not* obtain general expressions of the optimum smoothing parameters which are independant of the probability p involved in our model. We restrict our study to the cases $w = 3$ and $w = 5$.

Our definitions of section I imply that $\phi'_i = \theta'_{i+1} - \theta'_i$. From trigonometry, we have

$$\tan \phi'_i = \frac{\tan \theta'_{i+1} - \tan \theta'_i}{1 + \tan \theta'_i \tan \theta'_{i+1}}. \tag{33}$$

Now $\tan \theta'_i = y'_i - y'_{i-1}$ and $\tan \theta'_{i+1} = y'_{i+1} - y'_i$. Thus we can obtain $\tan \phi'_i$ from the smoothed coordinates and then ϕ'_i from the value of the tangent.

C.1 For $w = 3$

For $w = 3$, ϕ'_i at $\mathbf{p}'_i$ will depend on the *original* contour points in a 5-point neighbourhood around $\mathbf{p}_i$. For our model of Eq. 18, there are $2^5 = 32$ possible configurations for such a neighbourhood, which must be examined for their corresponding ϕ'_i. Of course, these computations need not be performed manually; they can be carried out using a language for symbolic mathematical calculation.

Adding together the contributions from the 32 possible configurations, weighted by the respective probabilities of these configurations, results in the following expression for $\overline{\tan^2 \phi_i'}$:

$$\begin{aligned}\overline{\tan^2 \phi_i'} &= 2p(1-p)(1-3p+3p^2)\left(\alpha^2 + \frac{(1-4\alpha)^2}{(1+\alpha-3\alpha^2)^2} + \frac{2(1-3\alpha)^2}{9\alpha^2(2-3\alpha)^2}\right) \\ &\quad + 2p^2(1-p)^2\left((1-3\alpha)^2 + (1-2\alpha)^2 + \frac{(1-3\alpha)^2}{(1+\alpha-2\alpha^2)^2}\right. \\ &\qquad \left. + \frac{(2-7\alpha)^2}{\alpha^2(7-12\alpha)^2} + \frac{2\alpha^2}{(1-\alpha^2)^2} + \frac{(1-4\alpha)^2}{32\alpha^2(1-2\alpha)^2}\right) \qquad (34)\end{aligned}$$

For simplicity we have dropped the subscript on the unique parameter α_1. Numerical optimization was performed to find the value of α which minimizes Eq. 34. No single value of α will minimize $\overline{\tan^2 \phi_i'}$ for all values of p. The results are shown in Fig. 5(a). The best value of α is now a smooth function of p. However we note that the domain of variation is very little.

We cannot compare the results obtained minimizing the mean squared tangent of ϕ_i' to the situation without smoothing, since $\overline{\tan^2 \phi_i}$ is infinite. By taking the *arc tangent* function of Eq. 33, we can obtain the values of the angles ϕ_i' themselves and we can derive an expression for $\overline{\phi_i'^2}$ in the same manner. Numerical optimization of this expression yields the results shown in Fig. 5(b). As can be seen, they are almost the same as those of Fig. 5(a). In a similar fashion, we can generate expressions for $\overline{|\tan \phi_i'|}$ and $\overline{|\phi_i'|}$, for which the best smoothing parameters are shown in Fig. 5(c) and 5(d) respectively. Here there seems to be one predominant best parameter over a wide range of values for p.

All the results shown in Fig. 5 were obtained numerically, for values of p ranging from 0.005 to 0.995, in steps of 0.005. As expected, all these curves are symmetric about $p = 0.5$, so we will limit our discussion to $p < 0.5$. In Fig. 5(c), the best value of α for $p \in (0.005, 0.095)$ is $\alpha = 0.25$; then, for $p \in (0.145, 0.495)$, the best value is $\alpha = 0.2857$. Between these two intervals, p increases almost linearly. In Fig. 5(d), the same values of α are found: $\alpha = 0.25$ is the best choice for $p \in (0.005, 0.145)$ and $\alpha = 0.2857$ is the best choice for $p \in (0.150, 0.495)$.

In Fig. 5(a) and 5(b), the best value of α is a smoothly varying function of p. But we notice that $\alpha = 0.2857$ is an intermediate value of α in the narrow range of best values. In fact, choosing $\alpha = 0.2857$ for *all* values of p, the value of $\overline{\tan^2 \phi'_i}$ is always within 0.2% of the minimum possible.

Despite the differences in the actual curves of Fig. 5, the corresponding ranges of best α's are always quite narrow and very similar, independently of the exact criterion chosen. From now on, to maintain uniformity with the treatment of sections III-A and III-B, we will restrict ourselves to minimizing the mean squared angle.

Eq. 22 provided a measure of the noise before smoothing: $\phi_{rms} = \frac{\pi}{2}\sqrt{\frac{3p(1-p)}{2}}$. The fraction of this noise $(1 - \phi'_{rms}/\phi_{rms})$ which is removed by a simple smoothing operation with $w = 3$ was computed for different values of α. The results are displayed in Fig. 6. The solid line represents the best case and we see that approximately 73% of the r.m.s. noise is removed. The dashed line, representing the case where $\alpha = 0.2857$ is used for *all* values of p, is *not* distinguishable from the best case at this scale. The dash-dotted and the dotted lines represent the fraction of noise removed for $\alpha = 0.25$ and $\alpha = \frac{1}{3}$ respectively. In this last case, this fraction is a constant equal to 0.6655.

C.2 For $w = 5$

For $w = 5$, $2^7 = 128$ possible configurations of a 7-point neighbourhood centered on $\mathbf{p}'_i$ must be considered to obtain the values of ϕ'^2_i after smoothing. In Eq. 34, there were 9 distinct terms involving p and α. Now the computation of $\overline{\tan^2 \phi'_i}$ results in 35 distinct terms in p, α_1, and α_2. We will not reproduce this lengthy expression here...

Fig. 7(a) presents the best choice of parameters to minimize $\overline{\phi'^2_i}$. We notice that there is very little variation in their values over the range of values of p. The optimum parameters are approximately $\alpha_{\pm 1} = 0.2381$ and $\alpha_{\pm 2} = 0.1189$. These values are close to $\frac{2}{9}$ and $\frac{1}{9}$, the values for the triangular 5-point filter. Fig. 7(b) shows the fraction of ϕ_{rms} removed by the smoothing operation. The solid line represents the case where the optimum parameters are used for each value of p. In this situation, approximately 88.75% of the noise is removed. The dashed line represents the case $\alpha_{\pm 1} = \frac{2}{9}$ and $\alpha_{\pm 2} = \frac{1}{9}$, for which 86% of the noise is removed approximately. We see that these results are close to the ideal situation.

IV. Verifying Results for Varying Curvature

In the preceding section, we have studied optimum local weighted averaging extensively, based on a model of a horizontal border with random 1-pixel noise. Particular solutions were derived based on error criteria chosen in light of specific computational tasks to be performed after the smoothing operation. But can these results be relied upon to handle digitization noise along *arbitrary* contoursΓ

Our results were obtained for a *straight, horizontal* border, i.e. a line of curvature 0. But for arbitrary contours, curvature may vary from point to point. Should optimum smoothing parameters vary with curvature and, if so, in what mannerΓ For a given window size, can a fixed set of smoothing parameters be found which will give optimum (or near optimum) results across a wide range of curvature valuesΓ If so, how does this set of parameters compare with the one we have derived using our simple modelΓ

In this section, we try to answer these above questions by performing some experiments with digital circles. It should be clear that our interest is not with digital circles per se but rather, as explained above, with the variation of optimum smoothing parameters with curvature. The approach will be to examine, for digital circles of various radii, situations which are equivalent to the ones studied for the horizontal straight border in sections III-A, III-B, and III-C. Using numerical optimization, we will find the best choice of smoothing parameters for each situation, over a wide range of curvature values, and compare them with the values obtained previously. An example of a digital circle is shown in Fig. 8 for a radius $R = 7$.

A. Minimizing Error on Distances to Center

In this section, we consider the distances d_i from the center to each pixel P_i as approximations to the radius R. See Fig. 9(a). After smoothing, pixel P_i is replaced by pixel P_i' which is at a distance d_i' from the center of the circle. Our aim is to find the values of the smoothing parameters, for $w = 3$ and $w = 5$, which will minimize $\overline{(R - d_i')^2}$. Of course, these parameters might vary depending on the radius of the circles.

For reasons of symmetry, it is only necessary to consider one quadrant; with special attention to the main diagonal, we can restrict our attention to the first octant of each circle. Let $N_{1/8}$ be the number of pixels which are *strictly* within the first octant. The mean value of $(R - d'_i)^2$ is obtained by adding twice the sum of $(R - d'_i)^2$ for these $N_{1/8}$ points, plus the value for the pixel at coordinates $(R, 0)$, plus the value for the pixel on the main diagonal (if present). This sum is then divided by $2N_{1/8} + 1$ (or $2N_{1/8} + 2$, if there is a pixel on the main diagonal).

We now give a simple example, for $R = 4$ and $w = 3$. First we consider the situation before any smoothing is applied. For the point on the x-axis, the value of $(R - d_0)^2$ is always 0. For the next point $(4, 1)$, $(R - d_1)^2 = (\sqrt{17} - 4)^2$. For the next point $(3, 2)$, $(R - d_2)^2 = (\sqrt{13} - 4)^2$. Finally, for the point on the main diagonal $(3, 3)$, $(R - d_3)^2 = (\sqrt{18} - 4)^2$. The contributions for points $(4, 1)$ and $(3, 2)$ are counted twice and added to the contributions for the diagonal point $(3, 3)$. This sum is then divided by 6. Taking the square root of the result, we obtain $(R - d_i)_{rms} = 0.25832$.

After smoothing with the smallest window size, $w = 3$, we have the following values for $(R - d'_i)^2$:

$$\begin{aligned}(R - d'_0)^2 &= 0 \\ (R - d'_1)^2 &= (\sqrt{(4-\alpha)^2 + 1} - 4)^2 \\ (R - d'_2)^2 &= (\sqrt{(3+\alpha)^2 + 4} - 4)^2 \\ (R - d'_3)^2 &= (\sqrt{2(3-\alpha)^2} - 4)^2 .\end{aligned} \tag{35}$$

and we must minimize the expression $\frac{1}{6}\left(2(R - d'_1)^2 + 2(R - d'_2)^2 + (R - d'_3)^2\right)$. Thus the best smoothing parameter for $R = 4$ and $w = 3$ is found to be $\alpha = 0.23610$.

In the numerical computations it is possible to take advantage of the fact that, for small window sizes, the smoothing rarely affects the y-coordinates in the first octant; exceptions occur occasionally for the last pixel in the first octant (not on the diagonal) and for the pixel on the diagonal when the preceding pixel has the same x-coordinate. This last condition is found only for radii values of 1, 4, 11, 134, 373, 4552 etc... (see Kulpa [54]).

The coordinates of the pixels for the first octant of the digital circles were generated using the simple procedure presented in Horn [55], with a small correction pointed out by Kulpa [56] (see also Doros [57]). The best smoothing parameters were obtained for integer radii values ranging from 2 pixels to 99 pixels, in steps of 1. The results are presented in

Fig. 9(b) for $w = 3$ and 9(c) for $w = 5$. For comparison, the values derived in section III-A from our model of a noisy horizontal edge are shown with dashed lines.

For $w = 3$, we see that for radii values larger than 20 pixels the best $\alpha_{\pm 1}$ oscillates around $\frac{1}{3}$, as derived from our model. Similarly, for $w = 5$, the best values of $\alpha_{\pm 1}$ and $\alpha_{\pm 2}$ are close to the predicted value of 0.2. For small radii values however, the optimum $\alpha_{\pm 2}$-values are much lower than this value and the optimum $\alpha_{\pm 1}$-values are correspondingly higher. This is easily understood since a 5-pixel neighbourhood covers a relatively large portion of the circumference in these cases (as much as one eighth of the total circumference for a radius of 6 pixels, one fourth for a radius of 3 pixels). In fact, for radii values of 2, 3, 4, 6, and 8 pixels, it is best to use $\alpha_{\pm 2} = 0.0$ and smooth using only the nearest neighbour.

Fig. 9(d) compares the r.m.s. values of the errors on the radii without smoothing (solid line) to the best values possible after smoothing with window sizes of $w = 3$ (dashed line) and $w = 5$ (dotted line) respectively.

For each value of the radius, we have also compared the noise reduction achieved using the optimum parameters to that achieved with the constant values $\frac{1}{3}$ and $\frac{1}{5}$. The results are presented in Fig. 10(a) and 10(b), for $w = 3$ and $w = 5$ respectively. For $w = 3$, the 2 curves are indistinguishable for $R > 18$ pixels, and they are very close for $R \geq 10$. For $w = 5$ and radii values 2, 3, 4, 6, and 8 pixels, smoothing with $\alpha_{\pm 1} = \alpha_{\pm 2} = 0.2$ is actually *worse* than no smoothing at all. But, for $R > 18$, the best curve and that obtained with these fixed values are very close.

Finally, for $R \geq 10$ and window sizes $w = 3$ and $w = 5$, Table III compares the mean noise reduction of 3 methods: the optimum method, corresponding to the variable parameters of Fig. 9(b) and Fig. 9(c); the fixed parameter method derived from our model of section III; and the best fixed parameter method obtained from numerical estimates. We see that the results are very close and that the method derived from our simple noise model compares very well with the numerically determined best fixed parameter method.

B. Minimizing Error on Tangent Directions

In this section, we compare the direction of the tangent to a circle at a given point to the numerical estimate of that direction, obtained for digital circles. The situation is

Window	Method	$\alpha_{\pm 1}$	$\alpha_{\pm 2}$	Mean noise reduction
w = 3	optimum	variable		0.484317
	model	$\frac{1}{3}$		0.482672
	best fixed	0.3329		0.482673
w = 5	optimum	variable	variable	0.6184
	model	$\frac{1}{5}$	$\frac{1}{5}$	0.6068
	best fixed	0.2148	0.1703	0.6117

TABLE III

MEAN NOISE REDUCTION FOR $10 \leq R \leq 99$

illustrated in Fig. 11(a).

Since the *slope* of the tangent is infinite at pixel $(R, 0)$ of the digital circle, we will consider instead the *angle* which the tangent line makes with the x-axis. The radius from the center of the circle to pixel P_i makes an angle θ_i with the positive x-axis. Now consider the point where this radius intersects the continuous circle. Theoretically, the angle between the tangent to the circle at that point and the x-axis is $\frac{\pi}{2} + \theta_i$. On the other hand, the numerical estimate of this angle is given by $\frac{\pi}{2} + \varphi_i$, where φ_i is the angle between the horizontal axis and the perpendicular bisector of the segment from $P_{i-1}$to P_{i+1} (see Fig. 11(a)). The difference between these angles, $(\varphi_i - \theta_i)$ is the error on the elevation of the tangent to the circle at the point of interest. Our goal is to minimize the r.m.s. value of $(\varphi_i' - \theta_i')$, where the primes refer to the quantities after smoothing.

The values of φ_i and θ_i are readily computed in terms of the original coordinates of the digital circle as follows:

$$\theta_i = \tan^{-1}\left(\frac{y_i}{x_i}\right); \quad \text{and} \quad \varphi_i = \tan^{-1}\left(\frac{x_{i-1} - x_{i+1}}{y_{i+1} - y_{i-1}}\right). \tag{36}$$

The values of φ_i' and θ_i' are obtained similarly, in terms of the coordinates *after smoothing.*

Once again, the r.m.s. error for the entire circle can be computed by considering only the first octant; and the best smoothing parameters were obtained for integer radii values ranging from 4 pixels to 99 pixels, in steps of 1. The results are presented respectively in Fig. 11(b) for $w = 3$ and 11(c) for $w = 5$.

In section III-B.2, for $w = 3$, the optimum value derived for $\alpha_{\pm 1}$ was $\frac{2}{5}$; for $w = 5$, the optimum values derived for $\alpha_{\pm 1}$ and $\alpha_{\pm 2}$ were $\frac{1}{7}$ and $\frac{3}{14}$ respectively. These values are shown with horizontal dashed lines in Fig. 11(b) and 11(c). The best smoothing parameters vary with the values of R. However, when we compute their means for $4 \leq R \leq 99$, the results obtained are very close to the predicted values. Thus, for $w = 3$, $\overline{\alpha_{\pm 1}} = 0.408$ (compared to 0.4); for $w = 5$, $\overline{\alpha_{\pm 1}} = 0.1415$ (compared to 0.1429) and $\overline{\alpha_{\pm 2}} = 0.2129$ (compared to 0.2143).

Fig. 11(d) compares the r.m.s. values of the errors on the elevation of the tangents to a circle without smoothing (solid line) to the best values possible after smoothing with window sizes $w = 3$ (dashed line) and $w = 5$ (dotted line).

The noise reduction produced by smoothing is equal to $1.0 - \frac{(\varphi'_i - \theta'_i)_{rms}}{(\varphi_i - \theta_i)_{rms}}$. For each value of the radius, we have compared the noise reduction achieved using the optimum parameters to that achieved with the constant values $\alpha_{\pm 1} = \frac{2}{5}$, for $w = 3$, and $\alpha_{\pm 1} = \frac{1}{7}$, $\alpha_{\pm 2} = \frac{3}{14}$, for $w = 5$. The results are presented in Fig. 12(a) and 12(b) respectively.

As can be seen, the constant values predicted by our simple model yield noise reduction results which are very close to optimum.

Finally, for $4 \leq R \leq 99$ and window sizes $w = 3$ and $w = 5$, Table IV compares the mean noise reduction of 3 methods as explained previously. The best smoothing parameters derived from our simple noise model and the numerically determined best fixed parameters are almost the same and their performance is nearly optimal.

C. Minimizing Error on Deviation Angles

In this section, we compare the deviation angles along the circumference of a circle to the numerical estimates obtained for digital circles. The situation is illustrated in Fig. 13(a).

Consider 3 consecutive pixels P_{i-1}, P_i and P_{i+1} on the circumference of a digital circle. The deviation angle at P_i is denoted by ϕ_i. Now the line segments from the center of the circle to these 3 pixels (partly represented by dashed lines in the figure) have elevation angles of θ_{i-1}, θ_i, and θ_{i+1} respectively. The intersection of these line segments with the circle are the true circle points Q_{i-1}, Q_i, and Q_{i+1} with these elevations. Connecting these

Window	Method	$\alpha_{\pm 1}$	$\alpha_{\pm 2}$	Mean noise reduction
w = 3	optimum	variable		0.58084
	model	$\frac{2}{5}$		0.57474
	best fixed	0.4026		0.57478
w = 5	optimum	variable	variable	0.7658
	model	$\frac{1}{7}$	$\frac{3}{14}$	0.7572
	best fixed	0.1445	0.2088	0.7576

TABLE IV

Mean noise reduction for $4 \leq R \leq 99$

points by line segments defines a deviation angle δ_i at Q_i, for which ϕ_i is a numerical estimate.

The difference between δ_i and ϕ_i is the error on the deviation angle at the point of interest. Our goal in this section is to find the optimum parameters which will minimize the r.m.s. value of this error, after smoothing.

In terms of the pixel coordinates (x_i, y_i), the deviation angle ϕ_i is equal to

$$\phi_i = \tan^{-1}\left(\frac{(y_{i+1} - y_i)(x_i - x_{i-1}) - (x_{i+1} - x_i)(y_i - y_{i-1})}{(x_{i+1} - x_i)(x_i - x_{i-1}) + (y_{i+1} - y_i)(y_i - y_{i-1})}\right). \tag{37}$$

To compute δ_i, we first obtain the elevation angles as $\theta_i = \tan^{-1}(y_i/x_i)$ and then the coordinates of the circle points Q_i as

$$(\tilde{x}_i, \tilde{y}_i) = (R\ cos\theta_i, R \sin\theta_i). \tag{38}$$

Finally, δ_i is computed as in Eq. 37, using $(\tilde{x}, \tilde{y})$ instead of (x, y).

The best smoothing parameters were obtained for integer radii ranging from 4 pixels to 99 pixels, in steps of 1. The results are presented in Fig. 13(b) for $w = 3$ and 13(c) for $w = 5$.

In section III-C, for $w = 3$, the optimum value derived for $\alpha_{\pm 1}$ was 0.2857; for $w = 5$, the optimum values derived for $\alpha_{\pm 1}$ and $\alpha_{\pm 2}$ were 0.2381 and 0.1189 respectively. These values are shown with dashed lines in Fig. 13(b) and 13(c). Again, the best smoothing parameters vary with the values of R. However, when we compute their means for $4 \leq R \leq 99$, the results obtained are very close to the predicted values. Thus, for $w = 3$, $\overline{\alpha_{\pm 1}} = 0.2886$; for $w = 5$, $\overline{\alpha_{\pm 1}} = 0.2363$ and $\overline{\alpha_{\pm 2}} = 0.1232$.

Fig. 13(d) compares the r.m.s. values of the errors on the deviation angles without smoothing (solid line) to the best values possible after smoothing with window sizes $w = 3$ (dashed line) and $w = 5$ (dotted line).

The noise reduction produced by smoothing is equal to $1.0 - \dfrac{(\delta_i' - \phi_i')_{rms}}{(\delta_i - \phi_i)_{rms}}$. For each value of the radius, we have compared the noise reduction achieved using the optimum parameters to that achieved with the constant values $\alpha_{\pm 1} = 0.2857$, for $w = 3$, and $\alpha_{\pm 1} = 0.2381$, $\alpha_{\pm 2} = 0.1189$, for $w = 5$. The results are presented in Fig. 14(a) and 14(b) respectively.

As can be seen, the constant values predicted by our simple model yield noise reduction results which are very close to optimum.

For $4 \leq R \leq 99$ and window sizes $w = 3$ and $w = 5$, Table V compares the mean noise reduction of 3 methods as explained previously. The best smoothing parameters derived from our simple noise model and the numerically determined best fixed parameters are even closer than in the 2 previous cases and their performance is very nearly optimal.

Window	Method	$\alpha_{\pm 1}$	$\alpha_{\pm 2}$	Mean noise reduction
w = 3	optimum	variable		0.770903
	model	0.2857		0.765716
	best fixed	0.2865		0.765732
w = 5	optimum	variable	variable	0.910833
	model	0.2381	0.1189	0.906895
	best fixed	0.2386	0.1190	0.906917

TABLE V

Mean noise reduction for $4 \leq R \leq 99$

Finally, for $w = 5$, if one prefers to use $\alpha_{\pm 1} = \frac{2}{9}$ and $\alpha_{\pm 2} = \frac{1}{9}$ (computationnally convenient for deviation angle measurements), the mean error reduction level is 0.8832. This is very good but not quite as effective as the optimum methods previously discussed.

A comparison of the error reduction with the best possible solution, for every value of R, appears in Fig. 15.

V. Conclusion

This paper has presented a different avenue to solve the problem of optimum smoothing of 2-D binary contours. Several approaches were reviewed with particular emphasis on their theoretical merits and implementation difficulties. It was argued that most methods are eventually *implemented* as a local weighted average with particular weight values. Hence we adopted this scheme as the starting point of our investigation into optimum methods.

Furthermore, there are many applications where smoothing is performed to improve the precision of *specific measurements* to be computed from the contour points. In such cases, the smoothing parameters should be chosen based on the nature of the computations intended, instead of relying on a single, general 'optimality' criterion. Thus our work was focused on optimum local weighted averaging methods tailored for specific computational goals. In the present article, we have considered three such goals: obtaining reliable estimates of point positions, of slopes, and of deviation angles along the contours.

To study the problem, a simple model was defined to represent 1-pixel random noise along a straight horizontal border. Based on this simple model, an in-depth analytical investigation of the problem was carried out, from which precise answers were derived for the 3 chosen criteria.

Despite its simplicity, this model captures well the kind of perturbations which digitization noise causes in the numerical estimation of various quantities along 2-D binary contours *even with arbitrary curvature.* This was indeed verified, for window sizes of $w = 3$ and $w = 5$, by finding the best smoothing parameters, using equivalent criteria, for digital circles over a wide range of radii.

In this general case, the best smoothing parameters were found to vary according to the length of the radius. Thus, in order to take full advantage of these optimum filters, it would be necessary to compute local estimates of the radius of curvature for groups of consecutive pixels along the contour, and then apply the best parameters found for these radii. This would significantly reduce the efficiency of the smoothing operation. However, it is not really necessary to go to that extent since the performance of these varying-weight optimum filters can be very nearly approached by methods with a *fixed* set of parameters. The latter were derived by numerical computation, for a wide range of radii. And it turned out that their values were very close to those predicted using our simple digitization noise model.

These numerical computations with varying radius of curvature validate our proposed model and confer added confidence to the results obtained from it. Researchers requiring simple and effective local weighted averaging filters before making numerical estimates of specific quantities can thus rely on this model to derive optimum methods tailored to their particular needs.

VI. Acknowledgements

The authors would like to thank Dr. Louisa Lam for helpful comments and suggestions made on an earlier draft of this paper.

This work was supported by the National Networks of Centres of Excellence research program of Canada, as well as research grants awarded by the Natural Sciences and Engineering Research Council of Canada and by an FCAR team research grant awarded by the Ministry of Education of Quebec.

References

[1] M. Worring and A.W.M. Smeulders, "Digitized circular arcs: Characterization and parameter estimation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-17, no. 6, pp. 587–598, 1995.

[2] D.-M. Tsai and M.-F. Chen, "Curve fitting approach for tangent angle and curvature measurements," *Pattern Recognition*, vol. 27, no. 5, pp. 699–711, 1994.

[3] H. Freeman, "On the encoding of arbitrary geometric configurations," *IRE Trans. Electron. Comput.*, vol. EC-10, pp. 260–268, 1961.

[4] T. Sakai, M. Nagao, and H. Matsushima, "Extraction of invariant picture sub-structures by computer," *Comp. Graphics and Image Proc.*, vol. 1, no. 1, pp. 81–96, 1972.

[5] M. J. Eccles, M. P. C. McQueen, and D. Rosen, "Analysis of the digitized boundaries of planar objects," *Pattern Recognition*, vol. 9, pp. 31–42, 1977.

[6] H. Freeman, "On the digital computer classification of geometric line patterns," in *Proc. Nat. Electronics Conf.*, 1962, vol. 18, pp. 312–324.

[7] G. Gallus and P. W. Neurath, "Improved computer chromosome analysis incorporating preprocessing and boundary analysis," *Physics in Medecine and Biology*, vol. 15, no. 3, pp. 435–445, 1970.

[8] J. W. McKee and J. K. Aggarwal, "Computer recognition of partial views of curved objects," *IEEE Transactions on Computers*, vol. C-26, pp. 790–800, 1977.

[9] N. I. Badler and C. Dane, "The medial axis of a coarse binary image using boundary smoothing," in *Proc. IEEE Conf. on Pattern Recognition and Image Processing*, 1979, pp. 286–291.

[10] P. Kammenos, "Performances of polar coding for visual localisation of planar objects," in *Proc. 8th Int. Symposium on Industrial Robots*, Stuttgart, Germany, 1978, pp. 143–149.

[11] J. D. Dessimoz, "Curve smoothing for improved feature extraction from digitized pictures," *Signal Processing*, vol. 1, pp. 205–210, 1979.

[12] T. J. Ellis, D. Proffitt, D. Rosen, and W. Rutkowski, "Measurements of the lengths of digitized curved lines," *Comp. Vision, Graphics and Image Proc.*, vol. 10, pp. 333–347, 1979.

[13] A. R. Dill, M. D. Levine, and P. B. Noble, "Multiple resolution skeletons," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-9, no. 4, pp. 495–503, 1987.

[14] N. Ansari and K.-W. Huang, "Non-parametric dominant point detection," *Pattern Recognition*, vol. 24, no. 9, pp. 849–862, 1991.

[15] C.-H. Teh and R. T. Chin, "On the detection of dominant points on digital curves," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-11, no. 8, pp. 859–872, 1989.

[16] S.-C. Pei and J.-H. Horng, "Fitting digital curve using circular arcs," *Pattern Recognition*, vol. 28, no. 1, pp. 107–116, 1995.

[17] A. Witkin, "Scale-space filtering," in *Proc. Int. Joint Conf. on AI*, 1983, pp. 1019–1022.

[18] J. J. Koenderink, "The structure of images," *Biological Cybernetics*, vol. 50, pp. 363–370, 1984.

[19] H. Asada and M. Brady, "The curvature primal sketch," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 1, pp. 2–14, 1986.

[20] F. Mokhtarian and A. Mackworth, "Scale-based description and recognition of planar curves and two-dimensional shapes," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 1, pp. 34–43, 1986.

[21] D. M. Wuescher and K. L. Boyer, "Robust contour decomposition using a constant curvature criterion," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-13, no. 1, pp. 41–51, 1991.

[22] P. Saint-Marc, J. S. Chen, and G. Medioni, "Adaptive smoothing: A general tool for early vision," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-13, no. 6, pp. 514–529, 1991.

[23] P. Maragos, "Pattern spectrum and multiscale shape representation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-11, no. 1, pp. 701–716, 1989.

[24] M. H. Chen and P. F. Yan, "A multiscale approach based upon morphological filtering," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-11, pp. 694–700, 1989.

[25] J. A. Bangham, P. D. Ling, and R. Harvey, "Scale-space from nonlinear filters," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-18, pp. 520–528, 1996.

[26] C. M. Reinsch, "Smoothing by spline functions," *Numerische Mathematik*, vol. 10, pp. 177–183, 1967.

[27] I. J. Schoenberg, "Spline functions and the problem of graduation," *Proc. Nat. Acad. Sci.*, vol. 52, pp. 947–950, 1964.

[28] T. Poggio, H. Voorhees, and A. Yuille, "A regularized solution to edge detection," Tech. Rep., M.I.T. AI Laboratory, AI Memo 776, 1985.

[29] J. F. Canny, "A computational approach to edge detection," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986.

[30] J. Babaud, A. P. Witkin, M. Baudin, and R. O. Duda, "Uniqueness of the gaussian kernel for scale-space filtering," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 1, pp. 26–33, 1986.

[31] A. L. Yuille and T. A. Poggio, "Scaling theorems for zero crossings," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 1, pp. 15–25, 1986.

[32] A. K. Mackworth and F. Mokhtarian, "The renormalized curvature scale space and the evolution properties of planar curves," in *Proc. Comp. Vision and Pattern Recognition*, Ann Arbor, Michigan, June 1988, pp. 318–326.

[33] L. Wu and Z. Xie, "Scaling theorems for zero-crossings," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-12, no. 1, pp. 46–54, 1990.

[34] V. Anh, J. Y. Shi, and H. T. Tsui, "Scaling theorems for zero crossings of bandlimited signals," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-18, pp. 309–320, 1996.

[35] T. Lindeberg, "Scale-space for discrete signals," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-12, no. 3, pp. 234–254, 1990.

[36] E. J. Pauwels, L. J. Van Gool, P. Fiddelaers, and T. Moons, "An extended class of scale-invariant and recursive scale space filters," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-17, no. 7, pp. 691–701, 1995.

[37] J. A. Bangham, P. Chardaire, C. J. Pye, and P. D. Ling, "Multiscale nonlinear decomposition: The sieve decomposition theorem," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-18, pp. 529–539, 1996.

[38] B. Shaharay and D. J. Anderson, "Optimal estimation of contour properties by cross-validated regularization," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-11, no. 6, pp. 600–610, 1989.

[39] B. Shaharay and D. J. Anderson, "Optimal smoothing of digitized contours," in *Proc. Comp. Vision and Pattern Recognition*, Miami Beach, Florida, June 1986, pp. 210–218.

[40] D. Lee and T. Pavlidis, "One-dimensional regularization with discontinuities," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-10, no. 6, pp. 822–829, 1988.

[41] M.-H. Chen and R.T. Chin, "Partial smoothing splines for noisy boundaries with corners," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-15, no. 11, pp. 1208–1216, 1993.

[42] M. Brady, J. Ponce, A. Yuille, and H. Asada, "Describing surfaces," *Comp. Vision, Graphics and Image Proc.*, vol. 32, pp. 1–28, 1985.

[43] B. K. P. Horn and E. J. Weldon Jr., "Filtering closed curves," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 5, pp. 665–668, 1986.

[44] D. G. Lowe, "Organization of smooth image curves at multiple scales," *Int. Journal on Computer Vision*, vol. 3, pp. 119–130, 1989.

[45] J. Oliensis, "Local reproducible smoothing without shrinkage," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 15, no. 3, pp. 307–312, 1993.

[46] M. D. Wheeler and K. Ikeuchi, "Iterative smoothed residuals: A low-pass filter for smoothing with controlled shrinkage," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-18, no. 3, pp. 334–337, 1996.

[47] X. Li and T. Chen, "Optimal $\mathcal{L}_1$ approximation of the gaussian kernel with application to scale-space construction," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-17, no. 10, pp. 1015–1019, 1995.

[48] T. Kasvand and N. Otsu, "Regularization of digitized plane curves for shape analysis and recognition," in *Proc. SPIE Conf. 'Architecture and Algorithms for Digital Image Processing*, San Diego, August 1983, pp. 44–52.

[49] R. Legault and C. Y. Suen, "Refining curvature feature extraction to improve handwriting recognition," in *Proc. 3rd Int. Workshop on Frontiers in Handwriting Recognition*, Buffalo, N.Y., May 1993, pp. 31–40.

[50] B. Blesser, "Multistage digital filtering utilizing several criteria," *U.S. Patent 4 375 081*, February 1983.

[51] C. Y. Suen, C. Nadal, R. Legault, T. A. Mai, and L. Lam, "Computer recognition of unconstrained handwritten numerals," *Proceedings of the IEEE, Special issue on Optical Character Recognition*, vol. 80, no. 7, pp. 1162–1180, July 1992.

[52] C. C. Tappert, "Speed, accuracy, flexibility trade-offs in on-line character recognition," Tech. Rep., IBM Research Report RC13228 (#59158), T. J. Watson Research Center, Yorktown Heights, 1987.

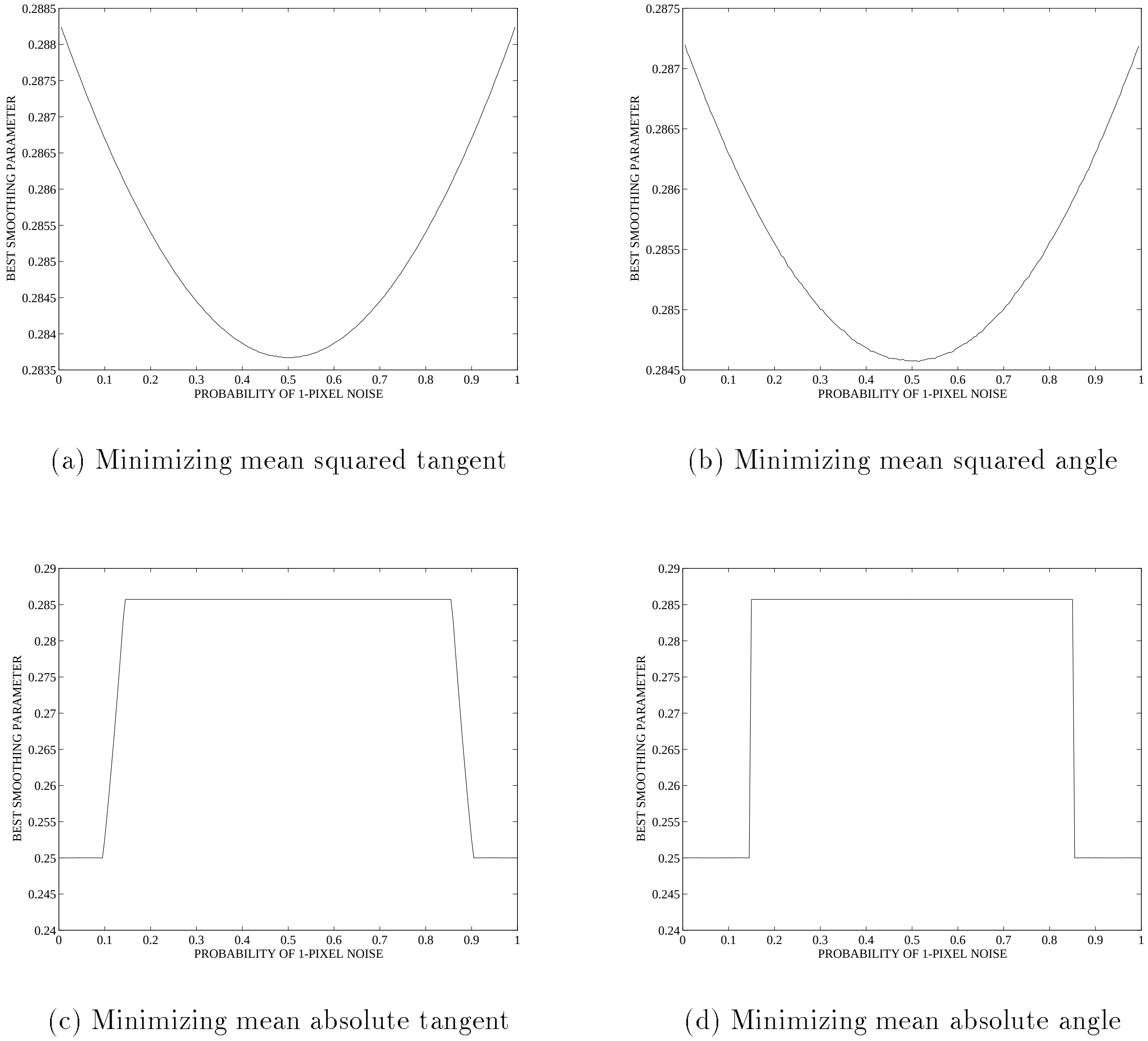
[53] A. E. Taylor and W. R. Mann, *Advanced Calculus*, John Wiley & Sons, 1983.

[54] Z. Kulpa, "On the properties of discrete circles, rings, and disks," *Comp. Graphics and Image Proc.*, vol. 10, pp. 348–365, 1979.

[55] B. K. P. Horn, "Circle generators for display devices," *Comp. Graphics and Image Proc.*, vol. 5, pp. 280–288, 1976.

[56] Z. Kulpa, "A note on the paper by B.K.P. Horn: Circle generators for display devices," *Comp. Graphics and Image Proc.*, vol. 9, pp. 102–103, 1979.

[57] M. Doros, "Algorithms for generation of discrete circles, rings, and disks," *Comp. Graphics and Image Proc.*, vol. 10, pp. 366–371, 1979.



(a) Minimizing mean squared tangent

(b) Minimizing mean squared angle

(c) Minimizing mean absolute tangent

(d) Minimizing mean absolute angle

Fig. 5. Best α to minimize deviation angles for $w = 3$.

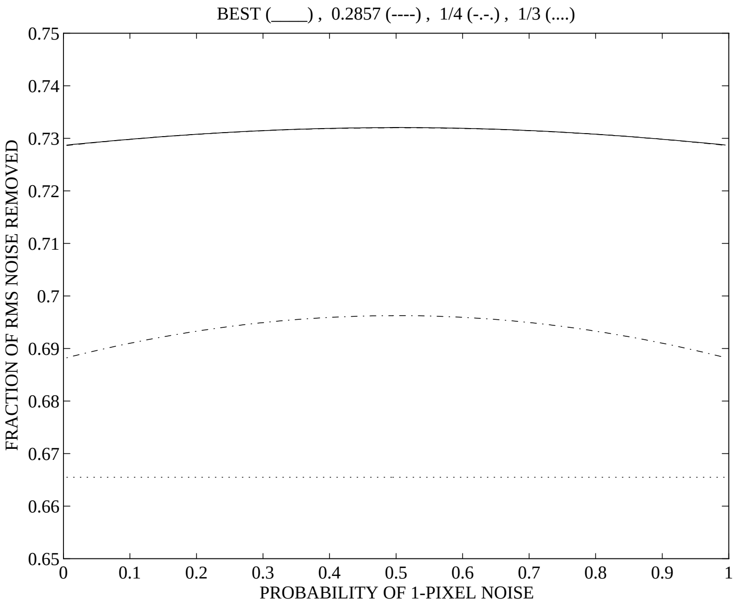
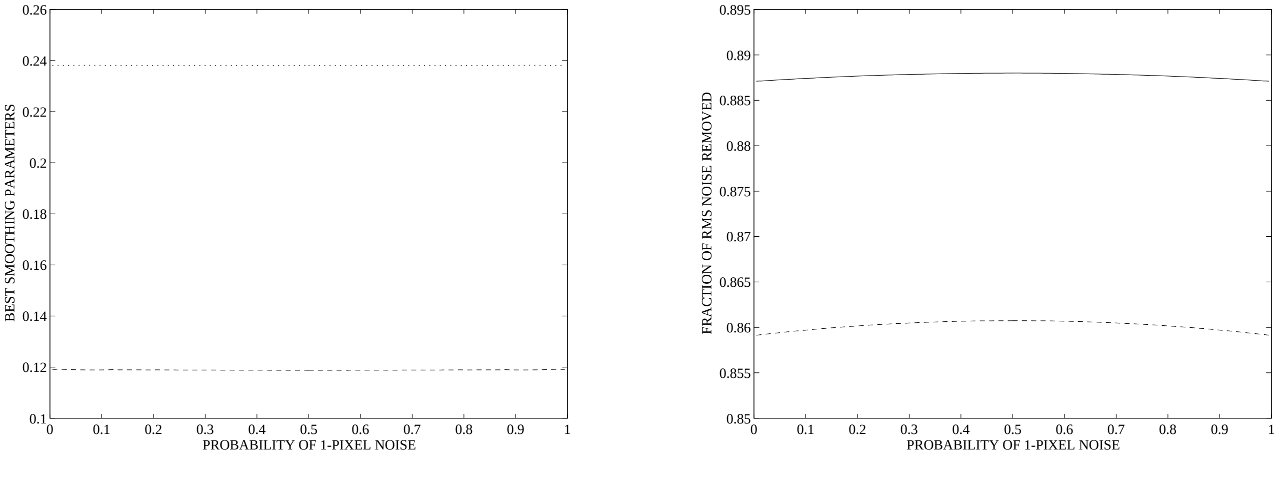


Fig. 6. Fraction of ϕ_{rms} removed for $w = 3$.



(a) Optimum parameters

$\alpha_{\pm 1}$ (dotted); $\alpha_{\pm 2}$ (dashed)

(b) Fraction of ϕ_{rms} removed

Best (solid); $\alpha_{\pm 1} = \frac{2}{9}$ & $\alpha_{\pm 2} = \frac{1}{9}$ (dashed)

Fig. 7. Minimizing $\overline{\phi_i'^2}$ for $w = 5$.

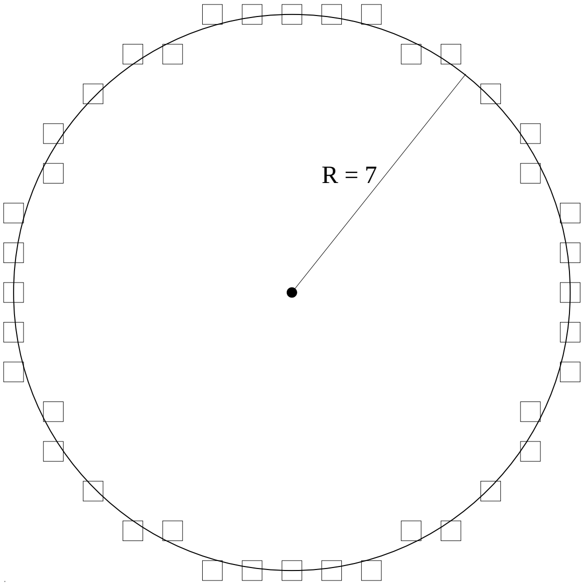
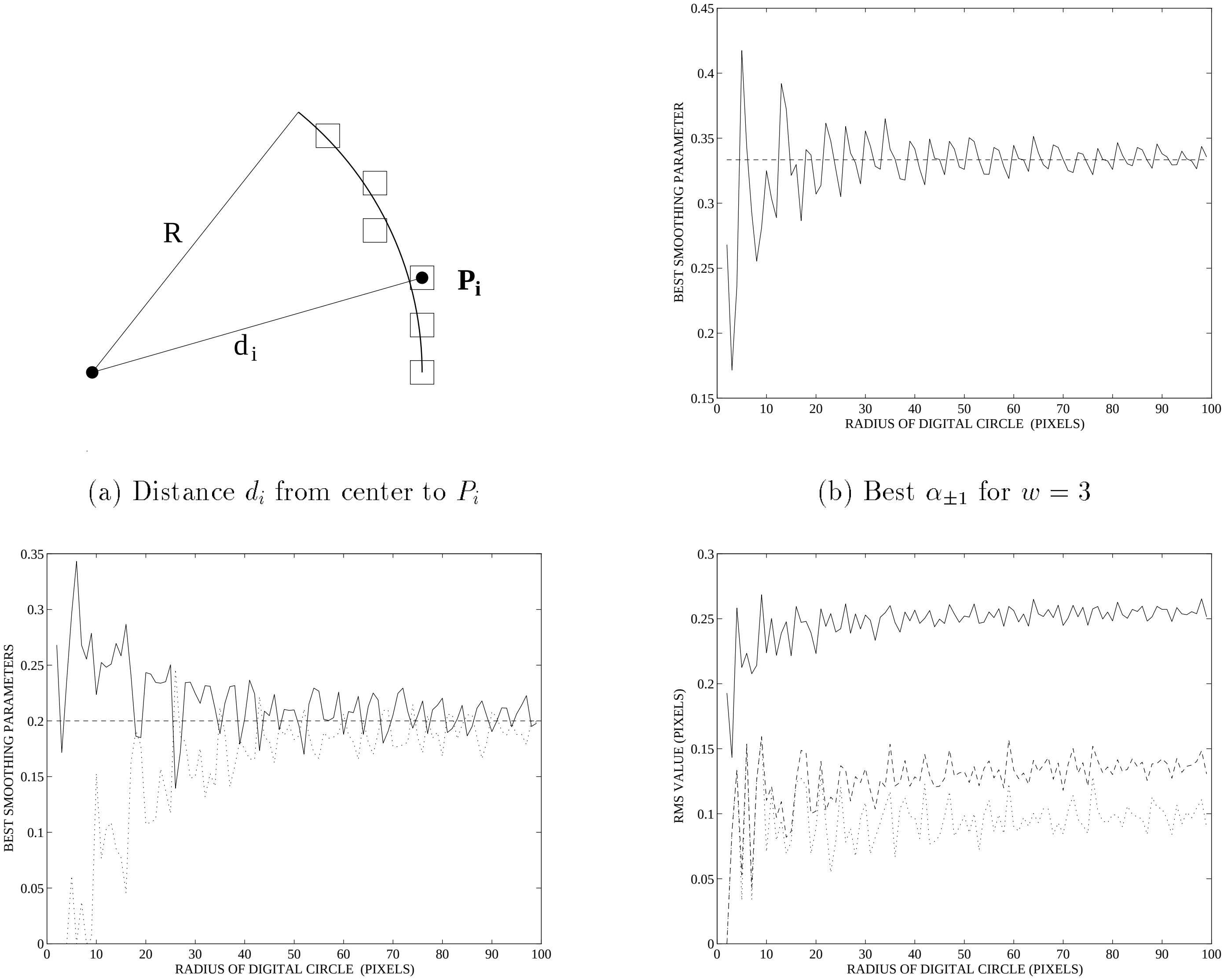


Fig. 8. Digital circle of radius $R = 7$.



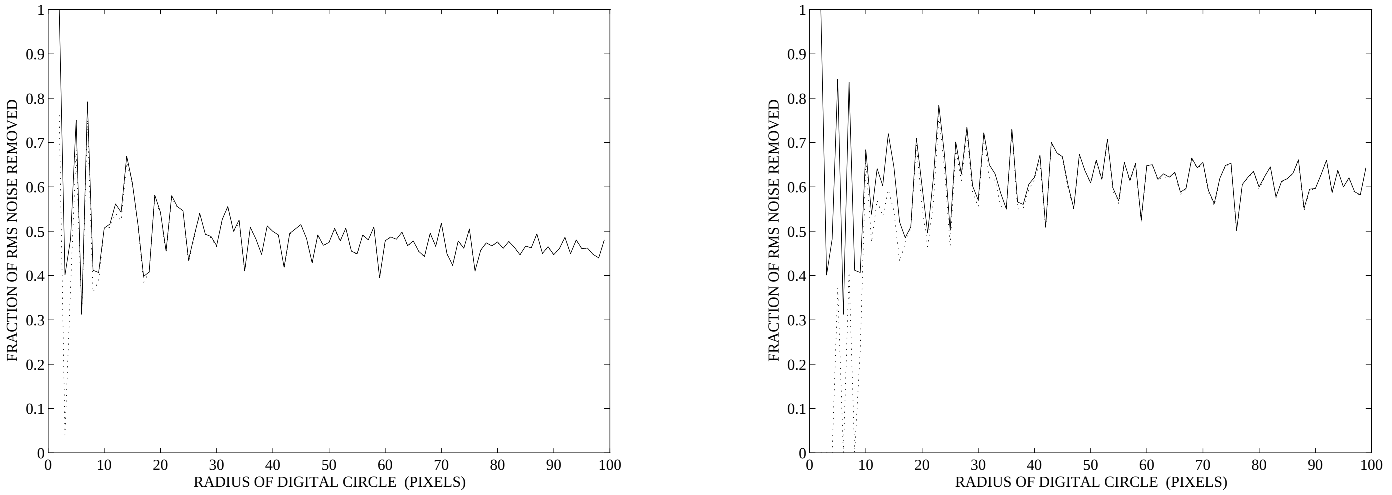
(a) Distance d_i from center to P_i

(b) Best $\alpha_{\pm 1}$ for $w = 3$

(c) Best $\alpha_{\pm 1}$ (solid), $\alpha_{\pm 2}$ (dotted) for $w = 5$

(d) RMS error: $w = 3$ (dashed); $w = 5$ (dotted)

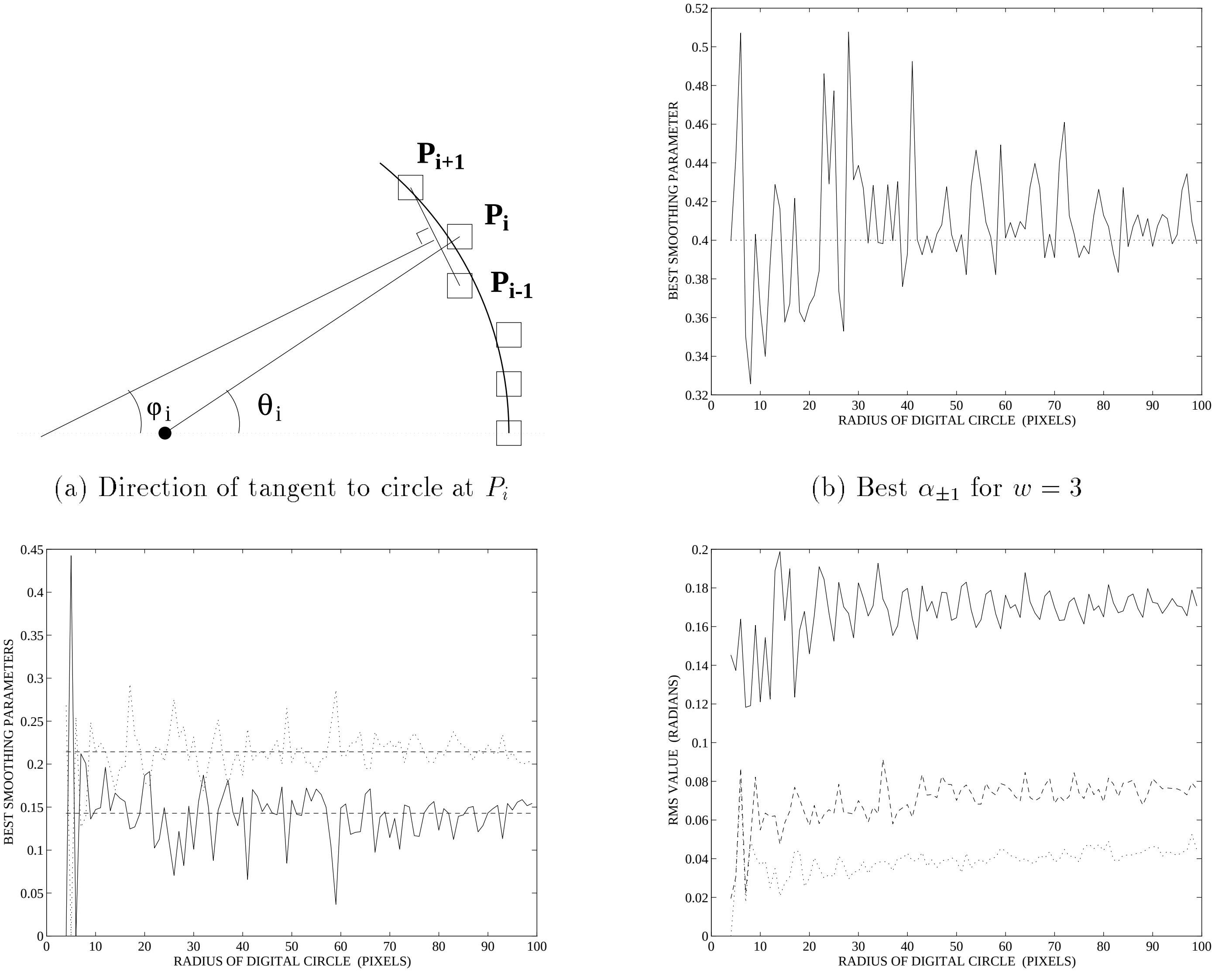
Fig. 9. Best Smoothing to Minimize $\overline{(R - d_i')^2}$.



(a) Using best $\alpha_{\pm 1}$ (solid); using $\frac{1}{3}$ (dotted)

(b) Using best $\alpha_{\pm 1}, \alpha_{\pm 2}$ (solid); using $\frac{1}{5}$ (dotted)

Fig. 10. Fraction of RMS noise removed: (a) $w = 3$; (b) $w = 5$.



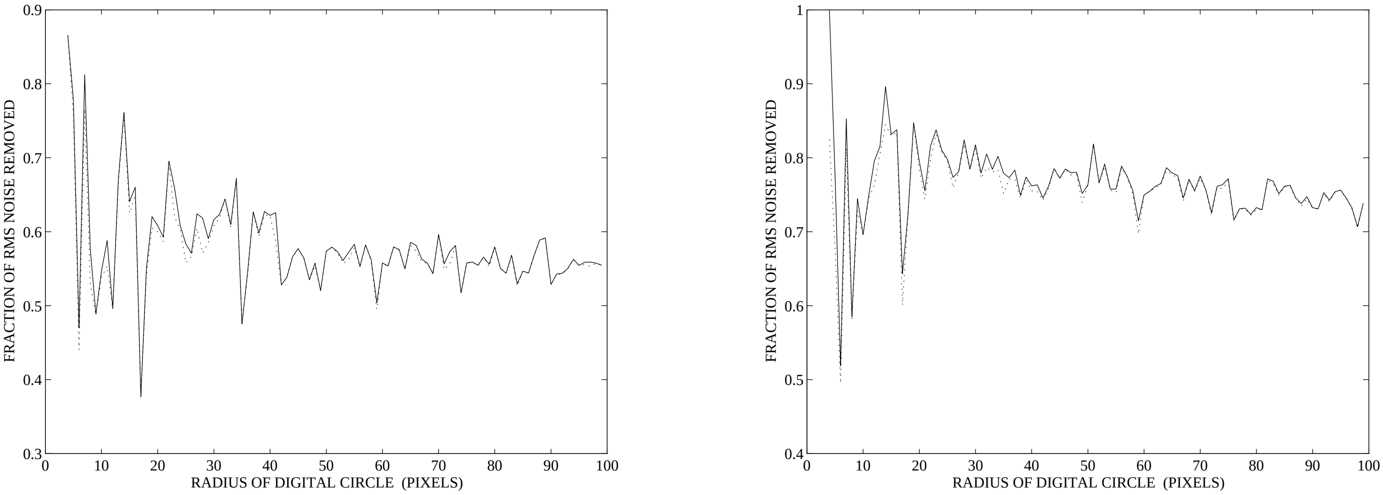
(a) Direction of tangent to circle at P_i

(b) Best $\alpha_{\pm 1}$ for $w = 3$

(c) Best $\alpha_{\pm 1}$ (solid), $\alpha_{\pm 2}$ (dotted) for $w = 5$

(d) RMS error: $w = 3$ (dashed); $w = 5$ (dotted)

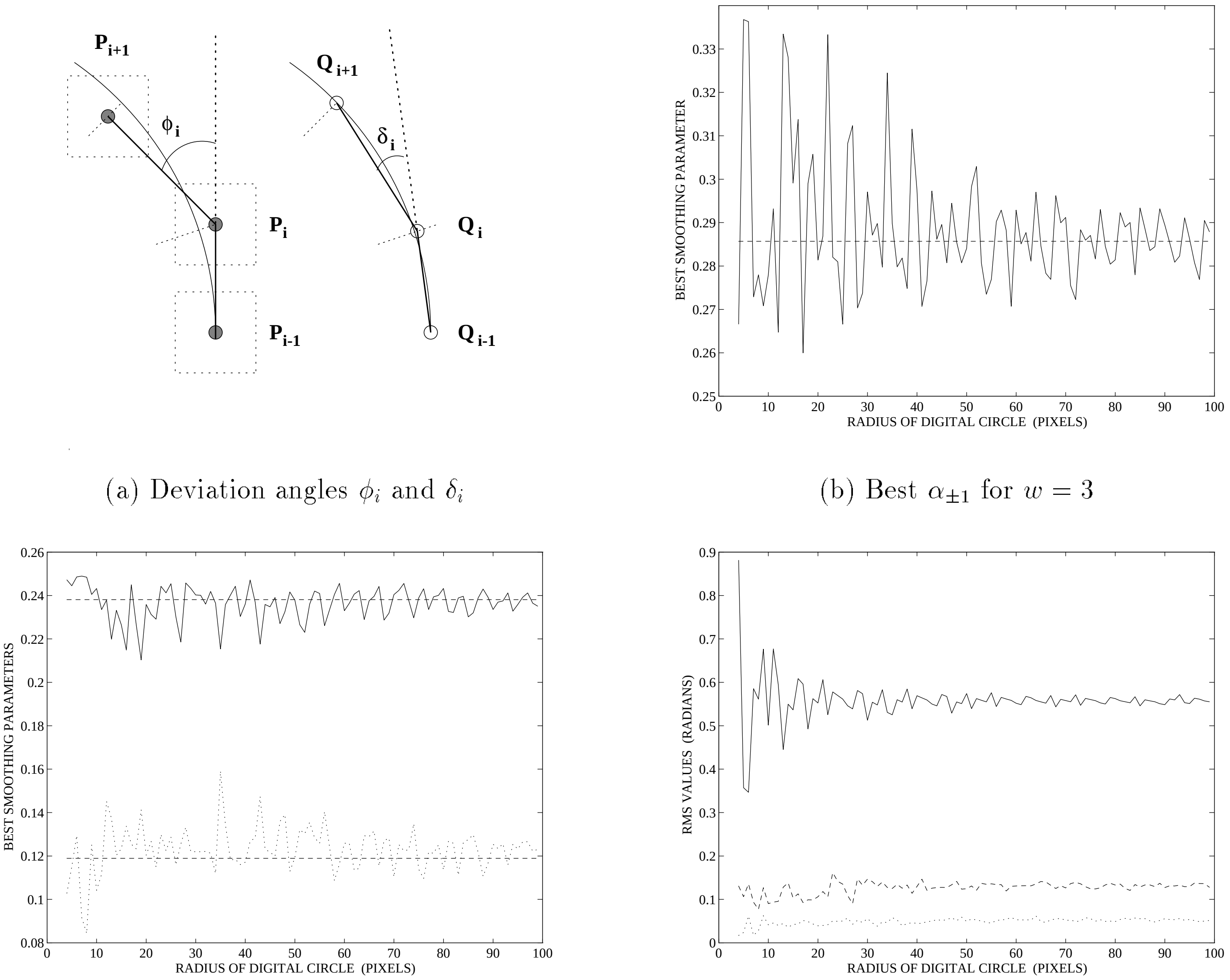
Fig. 11. Best Smoothing to Minimize $\overline{(\varphi_i' - \theta_i')^2}$.



(a) Using best $\alpha_{\pm 1}$ (solid); using $\frac{2}{5}$ (dotted)

(b) Using best $\alpha_{\pm 1}, \alpha_{\pm 2}$ (solid); using $\frac{1}{7}, \frac{3}{14}$ (dotted)

Fig. 12. Fraction of RMS noise removed: (a) $w = 3$; (b) $w = 5$.



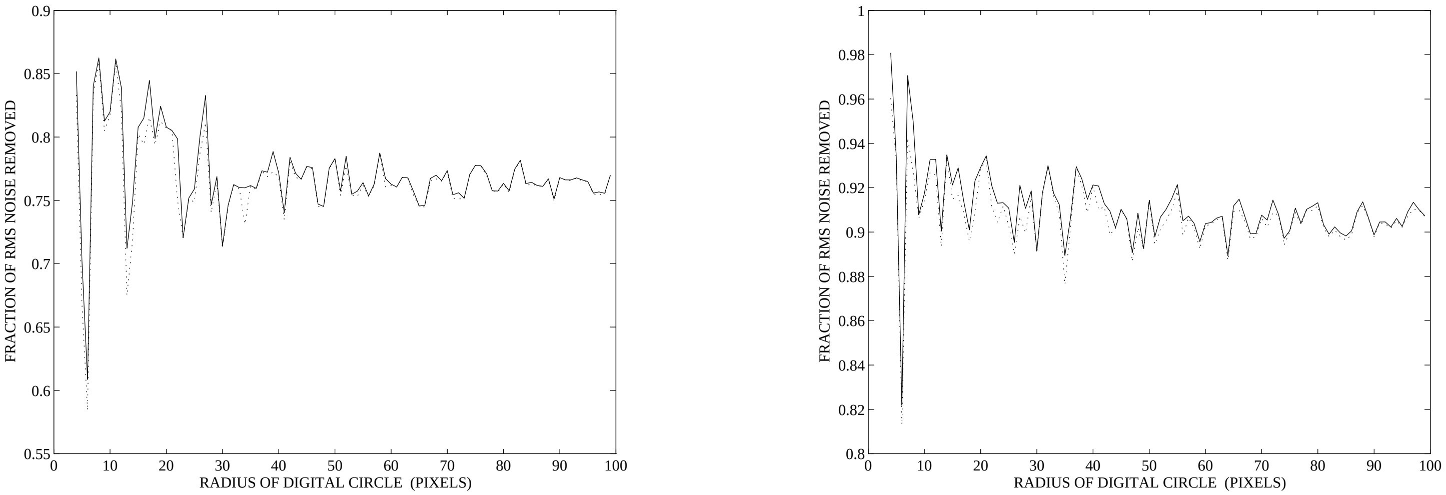
(a) Deviation angles ϕ_i and δ_i

(b) Best $\alpha_{\pm 1}$ for $w = 3$

(c) Best $\alpha_{\pm 1}$ (solid), $\alpha_{\pm 2}$ (dotted) for $w = 5$

(d) RMS error: $w = 3$ (dashed); $w = 5$ (dotted)

Fig. 13. Best Smoothing to Minimize $\overline{(\delta_i' - \phi_i')^2}$.



(a) Using best $\alpha_{\pm 1}$ (solid); using 0.2857 (dotted)

(b) Using best $\alpha_{\pm 1}, \alpha_{\pm 2}$ (solid); using 0.2381, 0.1189 (dotted)

Fig. 14. Fraction of RMS noise removed: (a) $w = 3$; (b) $w = 5$.

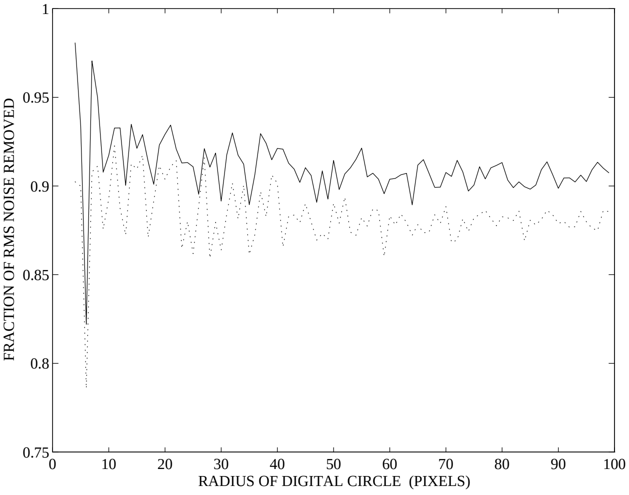


Fig. 15. RMS noise removed for $w = 5$: best case (solid); $\frac{2}{9}, \frac{1}{9}$ (dotted)