Unbiased extraction of lines with parabolic and Gaussian profiles[☆]

Carsten Steger

MVTec Software GmbH, Neherstraße 1, 81675 München, Germany

ARTICLE INFO

Article history:

Received 12 November 2010

Accepted 21 August 2012

Available online 8 September 2012

Keywords:

Feature extraction
Curvilinear structures
Lines
Scale-space
Low-level processing
Medical images

ABSTRACT

This paper presents an approach to extract curvilinear structures (lines) and their widths from two-dimensional images with high accuracy. Models for asymmetric parabolic and Gaussian line profiles are proposed. These types of lines occur frequently in applications. Scale-space descriptions of parabolic and Gaussian lines are derived in closed form. A detailed analysis of these scale-space descriptions shows that parabolic and Gaussian lines are biased more significantly than the well-known asymmetric bar-shaped lines by the partial derivatives of the Gaussian filters that are used to extract the lines. A bias function is constructed that relates the parameters of the lines to biased measurements that can be extracted from the image. It is shown that this bias function can be inverted. This is used to derive an algorithm to remove the bias from the line positions and widths. Examples on synthetic and real images show the high subpixel accuracy that can be achieved with the proposed algorithm. In particular, the line extractor is tested on a publicly available data set that includes manually labeled ground truth. The results on this data set show that very accurate results can be achieved on real data if the appropriate line model is used.

© 2012 Elsevier Inc. All rights reserved.

1. Introduction

The extraction of curvilinear structures, often simply called lines, has applications in many fields of science and technology. In the field of photogrammetry and remote sensing, it can be used in systems that extract roads from aerial and satellite images with different modalities, e.g., optical, synthetic aperture radar (SAR), and light detection and ranging (LIDAR) images [1–4], to extract road markings [3], to verify roads that are stored in geographic information systems (GISs) [5], to register maps to images [6], and even to extract buildings from SAR images, where lines occur as double reflections at concave dihedral corners on buildings [7]. Furthermore, line extraction can be used in the field of document analysis, e.g., to interpret engineering drawings [8]. Applications in the field of computer vision include 3D reconstruction using structured light [9–11] and stereo reconstruction [12,13]. In physics, line extraction can be used to detect gravitational waves in time–frequency diagrams [14,15]. The same techniques can also be used to detect sound events in pitch–time spectrograms of audio signals [16]. In the field of medical image analysis, applications of line extraction include the extraction of blood vessels for ophthalmic applications [17–19], for measurement of microcirculatory geometry [20], and for image registration [21,22], the detection of network patterns in skin lesions [23–25], the measurement

of neurites [26,27] and neurite growth [28,29], and the detection of vesicle movements [30].

In all of the above applications, it is necessary to extract the line positions and widths with high accuracy. However, it is well known that the smoothing that must be used to extract features like lines and edges from an image inherently leads to biased extraction results. This effect was first studied qualitatively in the context of edge extraction [31,32]. The bias inherent in line extraction is analyzed quantitatively in [33]: a line detection algorithm based on differential geometry is described. Furthermore, an asymmetric bar-shaped line profile is proposed and its scale-space behavior is analyzed in detail. The analysis shows that the line positions are severely biased whenever the line is asymmetric and that the line widths are always biased. An algorithm to remove the bias from the extraction results is proposed, which results in very accurate line positions and widths. The algorithm is extended in [34,35] to handle asymmetric staircase lines (lines for which the line has a gray value that lies between the gray values of the background on the left and right side of the line) and to complete missing junctions. Furthermore, [35] extends the algorithm to extract lines from multispectral images, e.g., RGB color images.

Several other approaches to line detection have been proposed. The algorithms that have been published until 1998 are reviewed in detail in [33–35]. Therefore, only approaches that have been published since 1998 will be reviewed here.

A line detection algorithm that uses multiple-orientation Gabor filtering is proposed in [36]. The approach assumes lines to have a Gaussian profile. Lines are extracted by convolving the image with

[☆] This paper has been recommended for acceptance by J.K. Udupa.
E-mail address: steger@mvttec.com

oriented filters that are applied in 1° steps. Because of the very large number of filters that are employed, the approach is very slow. Furthermore, the algorithm extracts a pixel-precise region that corresponds to the line and not the line position and width.

An algorithm that uses the same line extraction principle as [33] is proposed in [37,38] (an earlier version of the algorithm is described in [39]). The approach uses the function of the 1D Deriche smoothing filter [40] as the model line profile. It then uses the same principles that were used to derive the Canny edge detector [41] to compute an optimal line detection filter. Because of the choice of the model line profile, the optimal line detection filters turn out to be the first and second derivatives of the Deriche smoothing filter. The approach does not consider the effect of the Deriche filters on the line positions and widths. Furthermore, the approach does not consider asymmetrical line profiles. Since any kind of smoothing leads to biased extraction results, the output of this line detector is also biased, and thus not optimal.

A multi-scale ridge detector that is based on the smoothed structure tensor is described in [42]. This algorithm uses the same principles for line detection that were first described in [43] and extends them by a multi-scale approach. The lines extracted with this approach are only pixel-precise. Furthermore, the line width is not extracted. More importantly, the bias for asymmetrical lines is not modeled. A little experimentation shows that this kind of line detector will return significantly biased results for the line positions for asymmetric lines (and for the line widths if this kind of detector were extended to extract the line width).

A line extractor that defines the line by maximizing the dissimilarity between features within the line to features on both sides of the line is described in [13]. The criterion is that the line is different from both sides of the line in frequency space and that both sides of the line are similar in frequency space. This facilitates the extraction of textured lines. The approach is iterated for different distances between the line profiles, and the optimum response is used. Hence, lines of different widths can be extracted. The approach is very time-consuming. The accuracy of the line widths depends on the discretization of the optimization search, which is relatively coarse. Errors in the extracted line positions and widths of more than one pixel on average are reported in [13].

Finally, an approach to line extraction based on nonlinear filtering is proposed in [44]. The filter uses circular masks, within which the pixels are weighted according to how similar their gray value is to the gray value at the center of the mask. If the Gaussian-weighted sum of these evaluations is below a threshold, the pixel is labeled as a line pixel. This approach is basically a region segmentation algorithm, i.e., it returns the areas in an image that correspond to a line. The line position and width are not extracted explicitly.

While the asymmetric bar-shaped line profile proposed in [33] is the correct model for many applications, in some of the above applications, a more suitable line profile is desirable to achieve higher accuracies. For example, in applications in which radiation is transmitted through translucent tubular objects, the bar-shaped line model is suboptimal. This class of applications includes, for example, X-ray or optical images of blood vessels.

One type of line profile suitable for these applications is a parabolic line profile. This type of profile is appropriate if the lines themselves have crisp boundaries and the optics and sensor do not cause a significant blurring of the image. This kind of line profile has been used in several applications, e.g., in [45] to extract neurons in confocal microscopic images and in the structured light system described in [9]. The parabolic line profile is also an excellent approximation to an elliptical profile [46] and to the profile described in [47], which models the blood column and vessel wall explicitly, determines their absorption coefficients, and inserts them into the Beer–Lambert law.

A second type of line profile that can be used in these applications is a Gaussian line profile. This kind of profile is appropriate if the lines themselves appear blurred, e.g., because they are buried in a material that causes scatter [48] or because the optics or sensor cause significant blurring [46,49,48]. Gaussian profiles have been used in several applications, e.g., to extract blood vessels in retinal images [50] or in coronary angiograms [36]. The Gaussian line profile has been shown to be an appropriate model for extracting blood vessels from conjunctival images [48] and from retinal images [19]. It is also an excellent approximation to the smoothed elliptical profile used in [46,49,48].

Fig. 1 shows that the parabolic and Gaussian line models provide a very good match for the profiles that occur in several real applications.

This paper is organized as follows: Section 2 briefly describes the line detection algorithm. Section 3 defines the parabolic and Gaussian line models. The scale-space behavior of the line models is analyzed in Section 4. Section 5 describes the bias removal algorithm. Examples are given in Section 6. Finally, Section 7 concludes the paper.

2. Line detection algorithm

To make this paper self-contained, the underlying line detection algorithm will be described briefly. A detailed description can be found in [33,34].

For the derivation of the bias removal in this paper, it is useful to study the definition of lines in one-dimensional images, i.e., gray value profiles, first. As discussed below, lines in two-dimensional images are modeled as having one of the characteristic gray value profiles across the line, i.e., perpendicular to the line. Let us call the gray value profile $f(x)$. Let us assume that the profile is the parabolic or Gaussian profile shown in Fig. 1. Then, lines are defined as points where $f'(x) = 0$, i.e., local maxima (for bright lines) or local minima (for dark lines) of the gray value profile. For real images, which contain noise, this criterion must be augmented with a criterion to select salient lines. This can be achieved with a threshold on $|f''(x)|$, i.e., by requiring $f''(x) \ll 0$ for bright lines and $f''(x) \gg 0$ for dark lines. Furthermore, since for real images the determination of the image derivatives is an inherently ill-posed problem, the above derivatives are estimated by convolving the image with the derivatives of a Gaussian kernel. It is well known that, under very general assumptions, the Gaussian kernel is the only kernel that makes the problem of estimating the derivatives of a noisy function well-posed [51]. The Gaussian kernel and its derivatives are given by:

$$g(x, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}, \quad (1)$$

$$g'(x, \sigma) = \frac{-x}{\sqrt{2\pi}\sigma^3} e^{-\frac{x^2}{2\sigma^2}}, \quad (2)$$

$$g''(x, \sigma) = \frac{x^2 - \sigma^2}{\sqrt{2\pi}\sigma^5} e^{-\frac{x^2}{2\sigma^2}}. \quad (3)$$

In the analysis below, we will also need the integral of the Gaussian kernel:

$$\phi(x, \sigma) = \int_{-\infty}^x e^{-\frac{t^2}{2\sigma^2}} dt. \quad (4)$$

By convolving the gray value profile $f(x)$ with the derivatives of the Gaussian kernel, a scale-space description of the profile is obtained:

$$r(x, \sigma) = g(x, \sigma) * f(x), \quad (5)$$

$$r'(x, \sigma) = g'(x, \sigma) * f(x), \quad (6)$$

$$r''(x, \sigma) = g''(x, \sigma) * f(x). \quad (7)$$

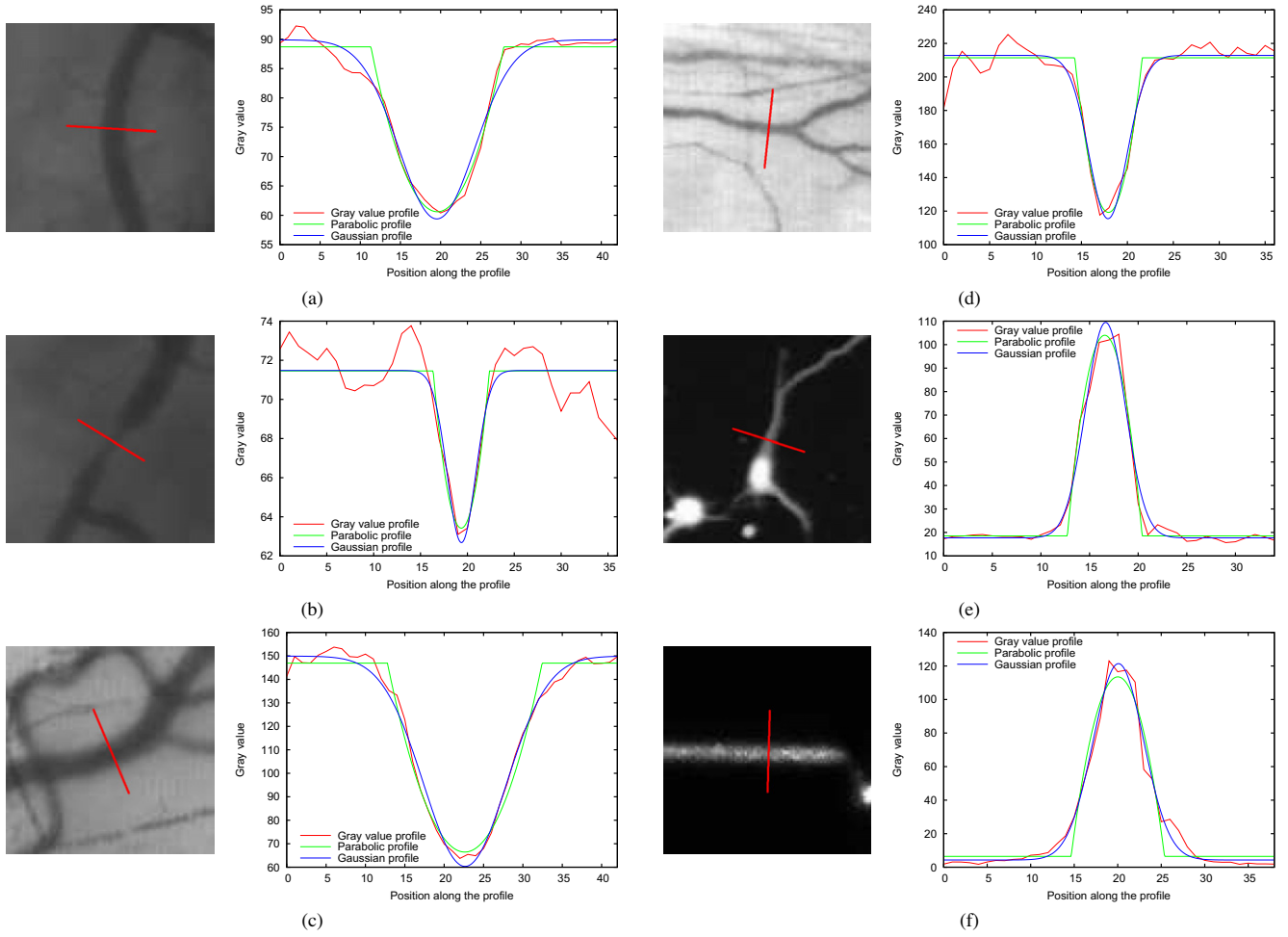


Fig. 1. Examples from different imaging modalities for gray value profiles and their approximations by parabolic and Gaussian line profiles. (a and b) Angiogram of a coronal artery with a stenosis. (c and d) Images of the conjunctiva. (e) Image of a neurite. (f) Image of a projected laser stripe. For images (a) and (b), the parabolic profile is a better fit, for images (c),(d), and (f), the Gaussian profile is a better fit, while for image (e) both profiles fit equally well.

Thus, line points l in gray value profiles are given by $r'(l, \sigma) = 0$ and $r''(l, \sigma) \ll 0$ or $r''(l, \sigma) \gg 0$. Furthermore, to determine the line width, the edges to the left and right of the line point will be used. Their location e is given by the maxima of $|r'(e, \sigma)|$ or by $r''(e, \sigma) = 0$ (both definitions are equivalent for one-dimensional profiles).

In two-dimensional images, lines are modeled as curves $s(t)$ that exhibit a characteristic 1D profile f perpendicular to the line, i.e., perpendicular to $s'(t)$. Let us call this direction n . Therefore, the first directional derivative in the direction n must vanish and the second directional derivative should have a large absolute value. The necessary partial derivatives are again computed by convolving the image with the partial derivatives of a 2D Gaussian kernel. Let us call these derivatives r_x, r_y, r_{xx}, r_{xy} , and r_{yy} . The direction n can be computed from the eigenvector that corresponds to the eigenvalue of largest magnitude of the Hessian matrix

$$H(x, y) = \begin{pmatrix} r_{xx} & r_{xy} \\ r_{xy} & r_{yy} \end{pmatrix}. \quad (8)$$

Line points are given by $r_n(l, \sigma) = 0$ and $|r_{nn}(l, \sigma)| \gg 0$, where the subscript n denotes the partial derivatives in the direction n and $l = (l_x, l_y)$ are the coordinates of the line point. As in the 1D case, the line width is determined based on the edge points that correspond to a line point. Their position is given by $r_{nn}(e, \sigma) = 0$, where the edge points $e = (e_x, e_y)$ must lie on a straight line from l in the direction $\pm n$.

In discrete 2D images, lines are detected by convolving the images with discrete versions of the Gaussian kernels (1)–(3). The responses r_x, r_y, r_{xx}, r_{xy} , and r_{yy} are used to determine n and a second-order Taylor polynomial for each pixel. To check whether a pixel contains a line point, the Taylor polynomial is used to test $r_n(l, \sigma) = 0$ for each pixel. Solving this equation returns the line point with subpixel accuracy. A pixel contains a line point if the point l lies within the pixel (where a pixel is assumed to be a 1×1 square).

The line points thus obtained are then linked into lines by an algorithm that takes the pixel neighborhood as well as the direction of the line into account. It returns a set of lines that is topologically sound, i.e., lines are split correctly at junctions where three or more lines meet. During the linking, a hysteresis threshold algorithm [41] based on $|r_{nn}(l, \sigma)|$ is used to select salient lines. Furthermore, the linking algorithm determines an orientation for each line point by consistently orienting the normal n in such a way that it always points to the right side of the line.

Next, the edge points on the right and left side of each line point are extracted. Line segments of length 2.5σ in the directions $\pm n$ are constructed for each line point. This is done by using the Bresenham algorithm [52], modified in a trivial manner in such a way that it returns each pixel that the above line intersects. For each pixel thus determined, the condition $r_{nn}(e, \sigma) = 0$ is checked. This is done by convolving the gradient image with 3×3 masks to deter-

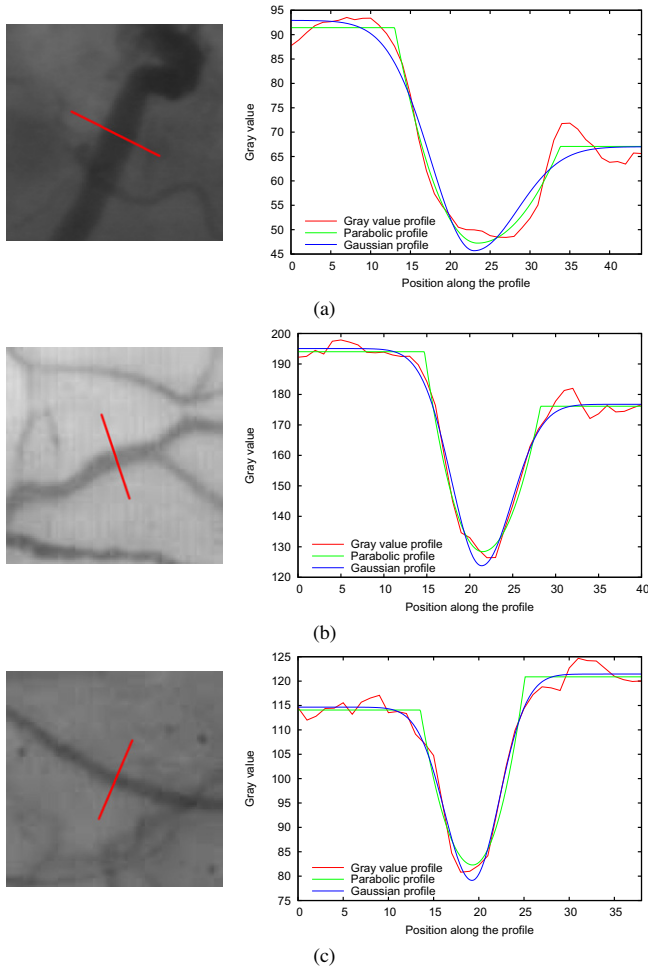


Fig. 2. Examples from different imaging modalities that show that the asymmetry of the lines can be substantial. (a) Coronary angiogram. (b) and (c) Images of the conjunctiva. For image (a), the asymmetric parabolic profile is a better fit, while for images (b) and (c), the asymmetric Gaussian profile is a better fit.

mine a second-order Taylor polynomial from the gradient magnitude image. This results in subpixel-accurate edge positions, which are used to compute the line width as the Euclidean distance between the edge points and the line point.

It should be noted that in discrete images there is a practical limit to the achievable accuracy of the line width extraction. As a limiting case, consider a line that is narrower than one pixel. In discrete images, this line simply will become a fainter line with a diameter of one pixel. Thus, the line widths of very narrow lines cannot be detected in discrete images. In addition, the discretization effects also affect lines that are slightly wider than one pixel. The practical limit for reliably extracting line width information seems to be around a diameter of two pixels.

In some cases, the edges of the line cannot be extracted, e.g., because the edges of the line are too faint or because the line is too wide. The latter case typically happens in junction areas, where the normal n of some of the lines meeting in the junction is oriented in such a way that it points along the line direction of another line that meets the junction, i.e., the edge point on one side of the line is effectively infinitely far away. Therefore, in junction areas, at least one edge of the line cannot be determined reliably. To remedy such cases, the line widths that could not be extracted are interpolated or extrapolated from adjacent line points for which the widths could be extracted.

Since the above algorithm returns biased widths if the line width is small compared to σ and biased line positions and widths if the line is asymmetric, the final step of the algorithm is to

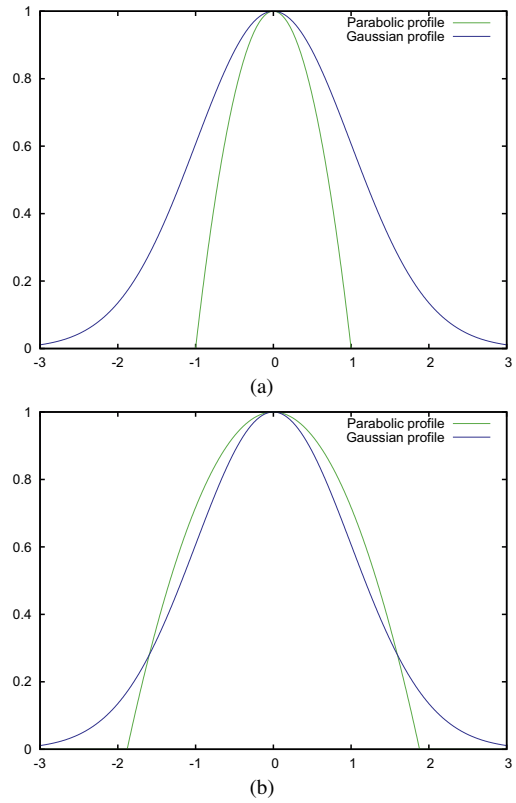


Fig. 3. Comparison of the parabolic and Gaussian line profiles (a) with equal widths ($w_p = w_g$) and (b) with equivalent widths ($w_p = \frac{3\sqrt{2}}{4} w_g$).

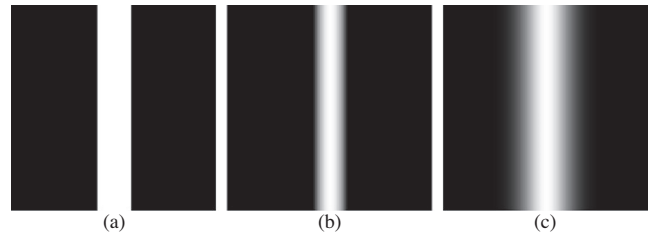


Fig. 4. Image of (a) a bar-shaped, (b) a parabolic, and (c) a Gaussian line with equal line widths.

remove the bias from the line positions and widths. The original algorithm uses a bar-shaped line model to predict and remove the bias. In the following sections, two alternative line models will be introduced and analyzed.

3. Models for line profiles

The original line detection algorithm uses an asymmetric bar-shaped line profile of width w and asymmetry $a \in [0, 1]$, defined by [33,34]:

$$f_b(x) = \begin{cases} 0, & x < -w, \\ 1, & |x| \leq w, \\ a, & x > w. \end{cases} \quad (9)$$

Note that the parameter w denotes the line width as measured from the center of the line, i.e., half of the line diameter. While this is a very good model for many applications, for some applications it is not the optimal model.

One alternative to the asymmetric bar-shaped line profile is the asymmetric parabolic line profile of width w and asymmetry $a \in [0, 1]$:

$$f_p(x) = \begin{cases} 0, & x < -w, \\ 1 - \left(\frac{x}{w}\right)^2, & -w \leq x \leq 0, \\ 1 - (1-a)\left(\frac{x}{w}\right)^2, & 0 < x \leq w, \\ a, & x > w. \end{cases} \quad (10)$$

A second alternative is the asymmetric Gaussian line profile of width w and asymmetry $a \in [0, 1]$:

$$f_g(x) = \begin{cases} e^{-\frac{x^2}{2w^2}}, & x \leq 0, \\ a + (1-a)e^{-\frac{x^2}{2w^2}}, & x > 0. \end{cases} \quad (11)$$

For all three model profile types, lines of arbitrary contrast h can be obtained by multiplying the respective profile by h .

Fig. 1 shows that parabolic and Gaussian line models provide a very good fit for many applications. In these examples, $a = 0$ was assumed. Furthermore, Fig. 2 provides several examples that show that significant asymmetries frequently occur in these applications. Sections 4 and 5 will show that asymmetric profiles lead to significantly biased line positions and widths for all three profile types.

The definitions of the three profile types have in common that the parameter w defines the location of the edges of the line if no smoothing is applied. However, these definitions lead to the fact that the lines thus defined have a significantly different appearance. This is most noticeable when comparing the parabolic and Gaussian profiles. Fig. 3a displays these line profiles with equal widths $w = 1$. The parabolic line profile has no energy (gray values) outside the interval $[-w, w]$. In contrast, the Gaussian profile has a substantial portion of its total energy outside this interval. Therefore, the Gaussian line will appear much wider than a parabolic line with the same width to the naked eye. This is also shown in Fig. 4, which displays images of the three line models with equal widths.

From the above example, it is clear that the widths w of the three line profile models are not directly comparable to each other when used on the same image. Furthermore, depending on the application, the width w returned by the line extractor may not be what is required by the application. In the discussion that follows, let us denote the widths of the three line profile models by w_b , w_p , and w_g , respectively.

To solve the first problem, a simple approach is to require that the profiles should have the same total energy, i.e., their integrals from $-\infty$ to ∞ should be equal. Let us denote these energies by E . We have $E_b = 2w_b$, $E_p = \frac{4}{3}w_p$, and $E_g = \sqrt{2\pi}w_g$. Thus, we have the following equivalent line widths: $w_p \equiv \frac{3\sqrt{2\pi}}{4}w_g \approx 1.8799w_g$, $w_p \equiv \frac{3}{2}w_b$, and $w_g \equiv \frac{\sqrt{2\pi}}{2}w_b \approx 0.7979w_b$. Fig. 3b shows that the appearance of the parabolic and Gaussian line profiles with equivalent line widths is much more similar than with equal line widths.

To approach the second problem, it can be noted that in many applications the desired line width is defined as the point where the gray value drops to a predefined fraction c of the contrast between the line and the background, e.g., 0.5, 0.25, or 0.1. For the bar-shaped line, this point is, of course, identical for all c . If we denote the width that corresponds to the fraction c by w_c , we have $w_c = w_b$ for the bar-shaped line. For the parabolic line, we have $w_c = \sqrt{1-c}w_p$. Finally, for the Gaussian line, we have $w_c = \sqrt{-2\ln c}w_g$. This also gives us a second way to define equivalent line widths. We have $w_p \equiv \sqrt{\frac{-2\ln c}{1-c}}w_g$, $w_p \equiv \sqrt{\frac{1}{1-c}}w_b$, and $w_g \equiv \sqrt{\frac{1}{-2\ln c}}w_b$. For example, for $c = 0.25$ we have $w_p \approx 1.9227w_g$, which corresponds quite closely to the equivalent line width computed with the first approach above.

4. Scale-space description of the line models

As discussed in Section 2, the scale-space description of a line model can be obtained by substituting the corresponding line profile into (5)–(7). For all three line models considered in this paper, the convolutions can be calculated analytically.

The scale-space description of the bar-shaped line profile is given by [33,34]:

$$r_b(x, \sigma, w, a) = \phi(x+w, \sigma) + (a-1)\phi(x-w, \sigma), \quad (12)$$

$$r'_b(x, \sigma, w, a) = g(x+w, \sigma) + (a-1)g(x-w, \sigma), \quad (13)$$

$$r''_b(x, \sigma, w, a) = g'(x+w, \sigma) + (a-1)g'(x-w, \sigma). \quad (14)$$

The scale-space description of the parabolic line profile is given by:

$$\begin{aligned} r_p(x, \sigma, w, a) = & \frac{1}{w^2} \left((w^2 - (x^2 + \sigma^2))\phi(x+w, \sigma) \right. \\ & + a(x^2 + \sigma^2)\phi(x, \sigma) \\ & \left. - (1-a)(w^2 - (x^2 + \sigma^2))\phi(x-w, \sigma) \right) \\ & + \frac{\sigma^2}{w^2} \left(-(x-w)g(x+w, \sigma) \right. \\ & + axg(x, \sigma) \\ & \left. + (1-a)(x+w)g(x-w, \sigma) \right) \end{aligned} \quad (15)$$

$$\begin{aligned} r'_p(x, \sigma, w, a) = & \frac{2x}{w^2} \left(-\phi(x+w, \sigma) \right. \\ & + a\phi(x, \sigma) \\ & \left. + (1-a)\phi(x-w, \sigma) \right) \\ & + \frac{2\sigma^2}{w^2} \left(-g(x+w, \sigma) \right. \\ & + ag(x, \sigma) \\ & \left. + (1-a)g(x-w, \sigma) \right) \end{aligned} \quad (16)$$

$$\begin{aligned} r''_p(x, \sigma, w, a) = & \frac{2}{w^2} \left(-\phi(x+w, \sigma) \right. \\ & + a\phi(x, \sigma) \\ & \left. + (1-a)\phi(x-w, \sigma) \right) \\ & + \frac{2}{w} \left(g(x+w, \sigma) \right. \\ & \left. + (1-a)g(x-w, \sigma) \right). \end{aligned} \quad (17)$$

Finally, the scale-space description of the Gaussian line profile is given by:

$$\begin{aligned} r_g(x, \sigma, w, a) = & a\phi(x, \sigma) \\ & + \sqrt{2\pi}w \left(g\left(x, \sqrt{w^2 + \sigma^2}\right) \right. \\ & \left. \times \left(1 - a\phi\left(\frac{w^2x}{w^2 + \sigma^2}, \frac{w\sigma}{\sqrt{w^2 + \sigma^2}}\right) \right) \right) \end{aligned} \quad (18)$$

$$\begin{aligned} r'_g(x, \sigma, w, a) = & ag(x, \sigma) \\ & + \sqrt{2\pi}w \left(g'\left(x, \sqrt{w^2 + \sigma^2}\right) \right. \\ & \left. \times \left(1 - a\phi\left(\frac{w^2x}{w^2 + \sigma^2}, \frac{w\sigma}{\sqrt{w^2 + \sigma^2}}\right) \right) \right) \\ & - a\frac{w^2}{w^2 + \sigma^2}g\left(x, \sqrt{w^2 + \sigma^2}\right) \\ & \times g\left(\frac{w^2x}{w^2 + \sigma^2}, \frac{w\sigma}{\sqrt{w^2 + \sigma^2}}\right) \end{aligned} \quad (19)$$

$$\begin{aligned}
r_g''(x, \sigma, w, a) = & \\
& ag'(x, \sigma) \\
& + \sqrt{2\pi}w \left(g''(x, \sqrt{w^2 + \sigma^2}) \right. \\
& \times \left(1 - a\phi\left(\frac{w^2x}{w^2 + \sigma^2}, \frac{w\sigma}{\sqrt{w^2 + \sigma^2}}\right) \right) \\
& - 2a\frac{w^2}{w^2 + \sigma^2}g'(x, \sqrt{w^2 + \sigma^2}) \\
& \times g\left(\frac{w^2x}{w^2 + \sigma^2}, \frac{w\sigma}{\sqrt{w^2 + \sigma^2}}\right) \\
& - a\frac{w^4}{(w^2 + \sigma^2)^2}g(x, \sqrt{w^2 + \sigma^2}) \\
& \left. \times g'\left(\frac{w^2x}{w^2 + \sigma^2}, \frac{w\sigma}{\sqrt{w^2 + \sigma^2}}\right) \right). \quad (20)
\end{aligned}$$

As described in Section 2, the line positions l are given by $r'(l, \sigma, w, a) = 0$ (the condition $|r''(l, \sigma, w, a)| \gg 0$ can be disregarded for the model lines). For bar-shaped lines, the line position can be computed analytically as $l = -\frac{\sigma^2}{2w} \ln(1 - a)$. For the other two line models, the line position can only be computed by using a numerical root finding algorithm. The edge positions e corresponding to the line point are given by $r''(e, \sigma, w, a) = 0$. They must be computed using a numerical root finding algorithm for all three line models. To compute the line and edge positions, we use the Van Wijngaarden–Dekker–Brent method [53, Chapter 9.3] for the bar-shaped line profile and bisection [53, Chapter 9.1] for the parabolic and Gaussian line profiles.

Fig. 5 shows the line and edge positions for $\sigma = 1, a = 0$, and $w \in [0, 4]$ for all three line profiles. For $a = 0$, the line position is correct for all w . However, the line widths are biased for all three line models. For bar-shaped lines (Fig. 5a), the bias is significant for small line widths. Indeed, the extracted line width can never be smaller than σ . For large line widths, the line width will be asymptotically correct. Parabolic lines (Fig. 5b) exhibit a similar behavior for small w : the extracted line width can never be smaller than σ . However, for larger line widths, the extracted line width will be significantly too small. Finally, the extracted line width of Gaussian lines (Fig. 5c) also cannot be smaller than σ . Like for bar-shaped lines, the extracted line width approaches the true line width asymptotically. The bias of the line widths is, however, significantly larger for Gaussian lines than for bar-shaped lines.

The line and edge positions for asymmetrical lines are shown in Fig. 6 for the three line models for $w = 1, \sigma = 1$, and $a \in [0, 1)$. The line position is only correct for $a = 0$ for all models. Furthermore, the line width is biased for all a . The bar-shaped line exhibits the smallest bias of the line positions, while the parabolic line exhibits the largest bias. Furthermore, the bar-shaped line exhibits the smallest bias of the line widths.

The two examples above show that it is even more important to model and remove the bias for parabolic and Gaussian lines than for bar-shaped lines.

5. Removing the bias

The examples in the previous section have shown that the line positions and widths are biased for almost all combinations of w, a , and σ . To remove the bias, a complete description of the bias is required. Furthermore, the relation of the model parameters w, a , and σ to quantities that can actually be extracted from the image must be derived. These quantities are the total line width v and the ratio r of the gradient magnitudes at the edges that correspond to a line point.

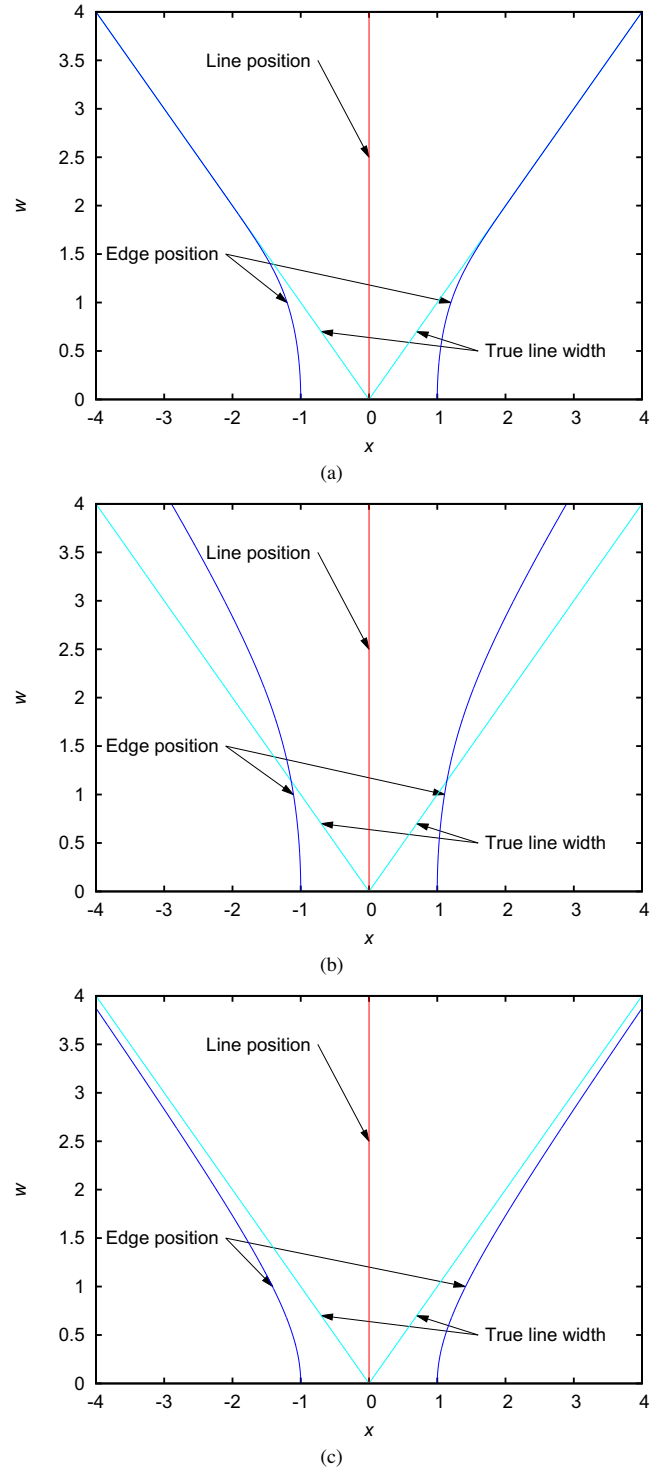


Fig. 5. Location of (a) a bar-shaped line, (b) a parabolic line, and (c) a Gaussian line and their corresponding edge positions (line widths) with $w \in [0, 4]$ for $\sigma = 1$ and $a = 0$.

As discussed in Section 4, (12)–(20) give an explicit scale-space description of the three proposed line models. The line position l is determined by the zero-crossings of $r'(x, \sigma, w, a)$. The total width of the line, as measured from the left to right edge, is given by the zero-crossings of $r''(x, \sigma, w, a)$. Let us call these positions e_l and e_r . Then the width to the left and right of the line is given by $v_l = l - e_l$ and $v_r = e_r - l$. The total line width

is $v = v_l + v_r = e_r - e_l$. The quantities l, e_l , and e_r have the following useful property:

Proposition 1. *The values of l, e_l , and e_r form a scale-invariant system. This means that if both σ and w are scaled by the same constant factor s , the line and edge locations will be sl, se_l , and se_r .*

Proof. For all three line models, it can be shown that

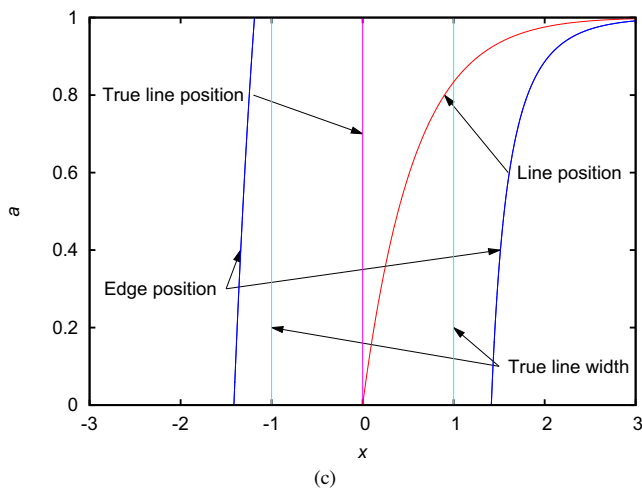
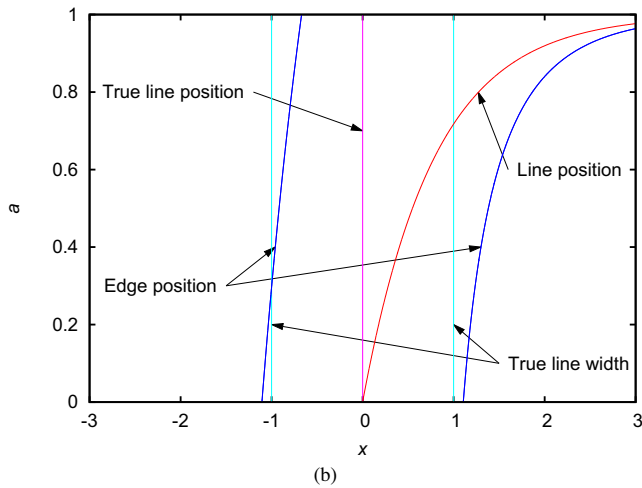
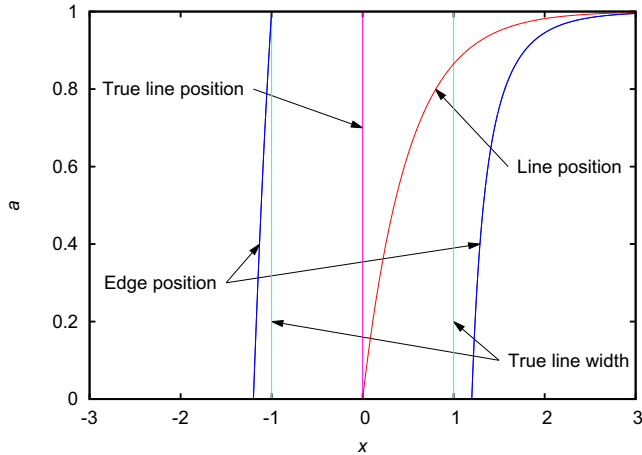


Fig. 6. Location of (a) a bar-shaped line, (b) a parabolic line, and (c) a Gaussian line and their corresponding edge positions (line widths) with $w = 1$, $\sigma = 1$, and $a \in [0, 1]$.

$$r'(x, \sigma, w, a) = sr'(sx, s\sigma, sw, a) \quad (21)$$

$$r''(x, \sigma, w, a) = s^2 r''(sx, s\sigma, sw, a). \quad (22)$$

Since the line and edge positions are given by the zero crossings of $r'(x, \sigma, w, a)$ and $r''(x, \sigma, w, a)$, respectively, the proposition follows. \square

This shows that w and σ are not independent. In fact, only scale-normalized quantities, e.g., $w_\sigma = w/\sigma$, $v_\sigma = v/\sigma$, need to be considered. Thus, the bias analysis can be performed for $\sigma = 1$. All other values can be obtained by a simple multiplication by the actual scale σ .

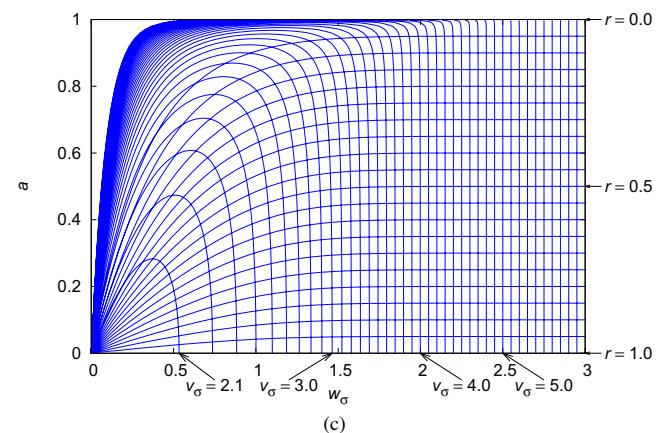
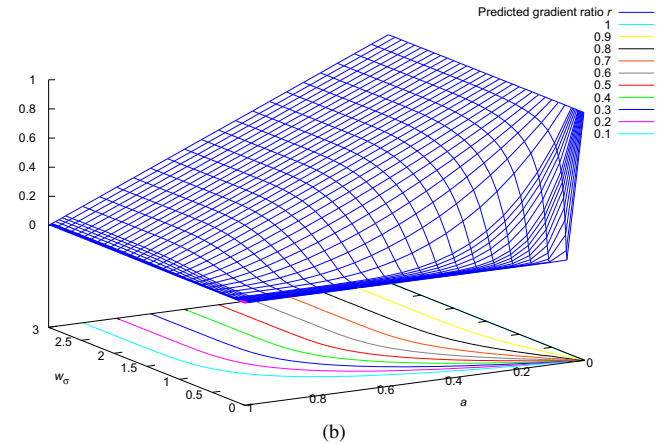
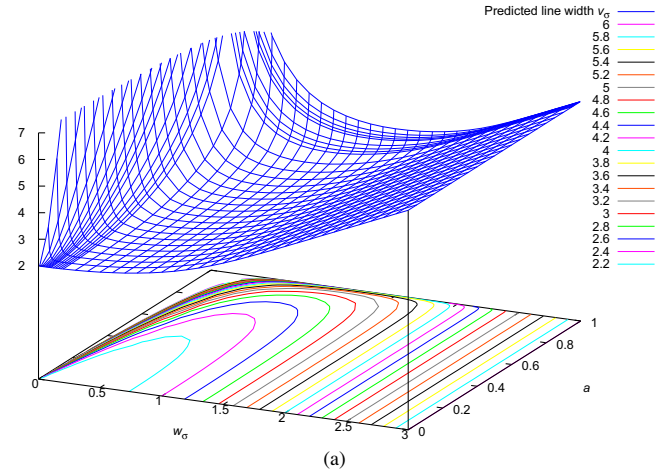


Fig. 7. Predicted behavior of the bar-shaped line f_b for $w_\sigma \in [0, 3]$ and $a \in [0, 1]$. (a) The predicted line width v_σ . (b) The predicted gradient ratio r . (c) The contour lines of v_σ and r .

Based on the above results, the predicted total line width v_σ can be calculated for all relevant w_σ and $a \in [0, 1)$. Figs. 7a, 8a, and 9a display v_σ for $w_\sigma \in [0, 3]$. Note that the functions have been sampled more densely in areas where they are very steep to show that v_σ can grow without bounds for $w_\sigma \downarrow 0$ or $a \uparrow 1$. Furthermore, note that all three graphs use the same scale to facilitate the comparison of the different amounts of bias. The behavior discussed in the examples in Section 4 holds in general. In particular, for larger w_σ , the extracted line width v_σ is significantly smaller for the parabolic line profile than for the bar-shaped line profile. On the other hand, v_σ is significantly larger for the Gaussian line profile than for the bar-shaped line

profile for larger w_σ . This again shows that the bias is more significant for parabolic and Gaussian lines than for bar-shaped lines.

Note that v_σ can be extracted from the image as described in Section 2. To remove the bias, which depends on the two parameters w_σ and a , a second observation is required. Since the true contrast h of the line profile is unknown, this quantity must be independent of h . A quantity that can be extracted easily from the image is the ratio of the gradient magnitude at e_r and e_l , i.e., $r = |r'(e_r, \sigma, w, a)| / |r'(e_l, \sigma, w, a)|$. It is obvious that h cancels out. Furthermore, r is invariant to simultaneous scalings of σ and w . Note that, by definition, e_l lies on

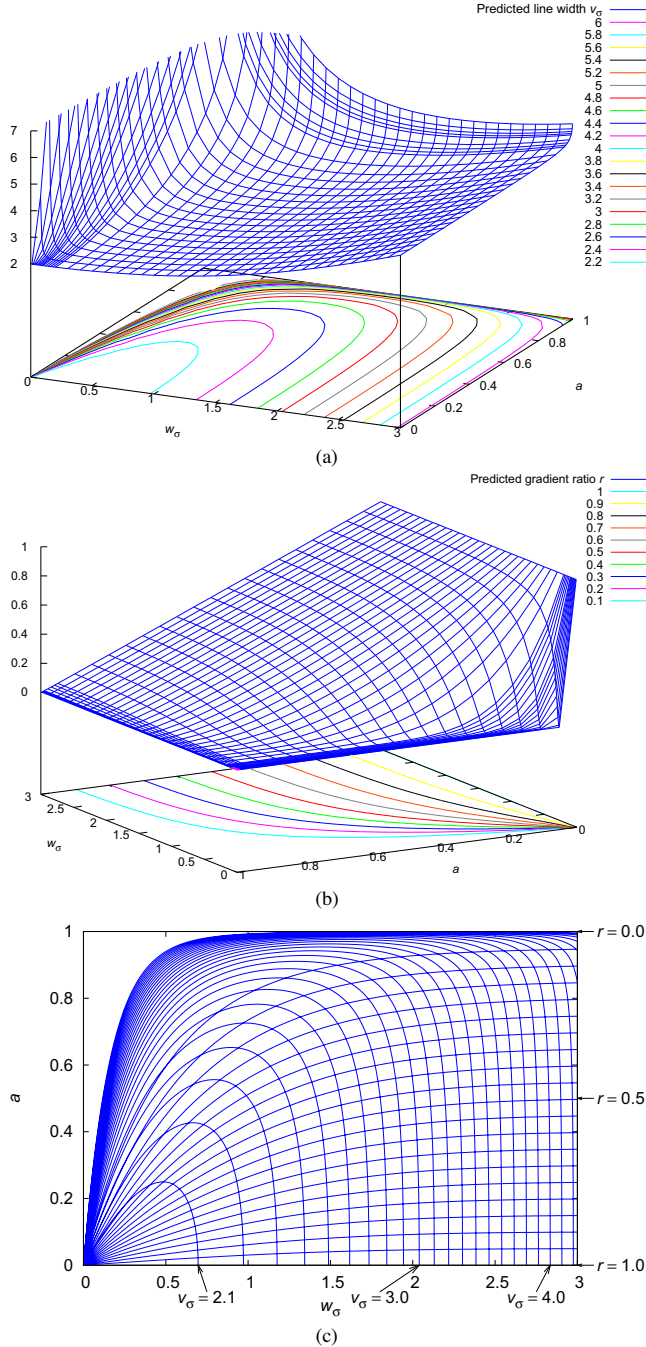


Fig. 8. Predicted behavior of the parabolic line f_p for $w_\sigma \in [0, 3]$ and $a \in [0, 1)$. (a) The predicted line width v_σ . (b) The predicted gradient ratio r . (c) The contour lines of v_σ and r .

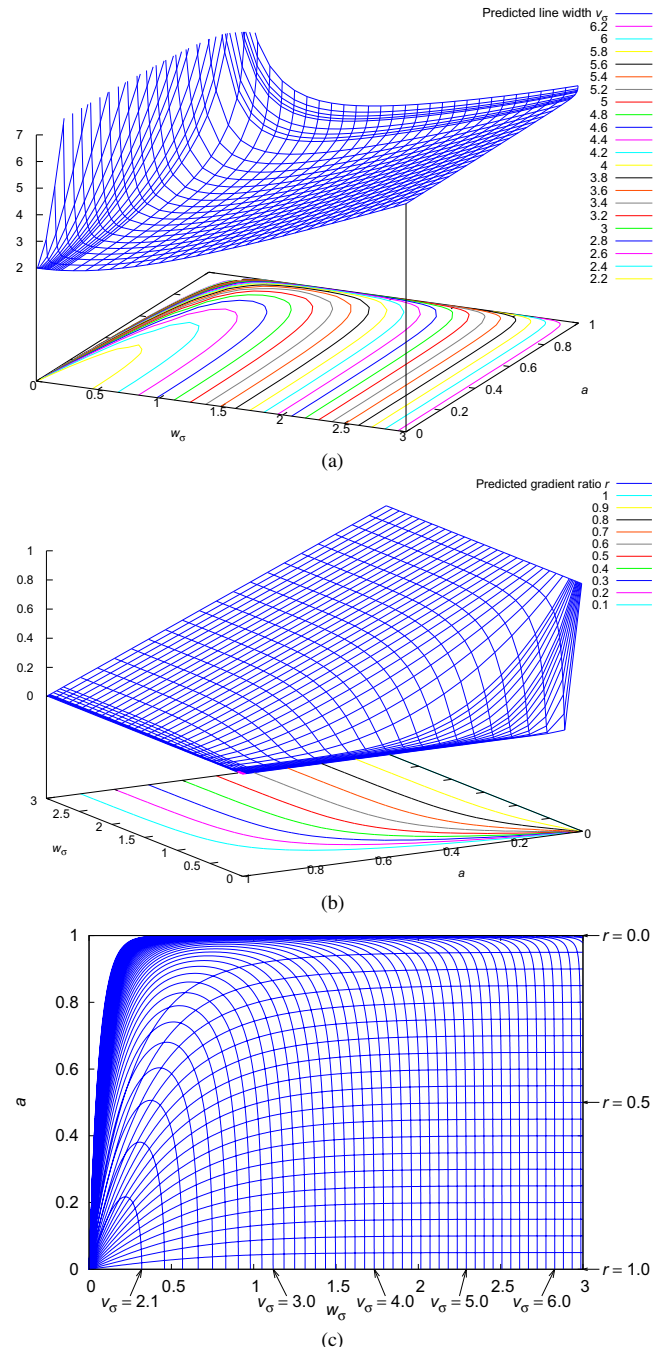


Fig. 9. Predicted behavior of the Gaussian line f_g for $w_\sigma \in [0, 3]$ and $a \in [0, 1)$. (a) The predicted line width v_σ . (b) The predicted gradient ratio r . (c) The contour lines of v_σ and r .

the side of the line with the stronger contrast. Therefore, $r \in (0, 1]$. Figs. 7b, 8b, and 9b display the predicted r for $w_\sigma \in [0, 3]$. For large w_σ , r is close to $1 - a$ for all line models. For small w_σ , it drops to near-zero for all a . The differences of the three line models in the predicted r are less dramatic than for v_σ .

From the above discussion, it can be seen that the true values of w_σ and a are mapped to the biased quantities v_σ and r , which can be extracted from the image. More formally, there is a bias function $b : (w_\sigma, a) \in [0, \infty] \times [0, 1] \mapsto (v_\sigma, r) \in [2, \infty] \times [0, 1]$. Since the algorithm must determine the true values (w_σ, a) from the observed (v_σ, r) , the inverse b^{-1} of the bias function b must be determined. Figs. 7c, 8c, and 9c display the contour lines of b for $v_\sigma \geq 2$ and $r \in (0, 1]$. For all three line models, the contour lines of v_σ are U-shaped with the tightest visible U corresponding to $v_\sigma = 2.1$. The contour line corresponding to $v_\sigma = 2$ is actually only the point $(0, 0)$. The contour lines for r run across with the lowermost visible contour line corresponding to $r = 0.95$. The contour line for $r = 1$ lies completely on the w_σ -axis. It can be seen that for any pair of contour lines from v_σ and r , there is only one intersection point. Hence, b is invertible for all three line models.

To remove the bias, b^{-1} must be calculated using a multi-dimensional root finding algorithm. As described above, the data for b is computed numerically. We use Broyden's method [53, Chapter 9.7] as the root finding algorithm since it does not require derivative information. Because this is a computationally expensive and numerically complex operation, b^{-1} is precomputed for selected values of v_σ and r and the true values are obtained by interpolation. Suitable step sizes are given by 0.1 for v_σ and 0.05 for r . Hence, the intersection points of the contour lines in Figs. 7c, 8c, and 9c are the entries in the table of b^{-1} . Figs. 10a, 11a, and 12a show the true values of w_σ for any given v_σ and r for the three line models, while Figs. 10b, 11b, and 12b show the true values of a . It can be seen that despite the fact that b is very ill-behaved for small w_σ , b^{-1} is actually quite well-behaved. Therefore, bilinear interpolation can be used to obtain accurate values for w_σ and a . Finally, Figs. 10c, 11c, and 12c show the corrections c_σ that must be applied to the line position to remove the bias. It can be seen that the correction is significantly larger for the parabolic and Gaussian profiles than for the bar-shaped profile for all parameter combinations. This shows once more that bias removal is even more important for parabolic and Gaussian lines than for bar-shaped lines.

6. Examples

6.1. Synthetic images

To assess the accuracy of the line positions and widths that can be obtained by the proposed line extractor, it is imperative to have ground truth data available. Therefore, this section reports the results of using the line extractor on images that were synthetically generated according to the three line models proposed in Section 3. In all of the experiments, all three line models will be used to correct the extracted positions and widths for all types of input images. This will establish whether it is important to use the appropriate line model to correct the results or whether the residual errors are insignificant when an incorrect line model is used.

In the first test, images that contain a single line of constant width without any asymmetry ($a = 0$) were created. These images are similar to the images shown in Fig. 4. The line width was varied between 2 and 12 in steps of 0.1. To test the accuracy of the widths in nearly the full range of the bias correction tables that are de-

scribed in Section 5, $\sigma = 5$ was used to extract the lines. Fig. 13 shows the line width errors that resulted from these experiments. The errors are displayed as a percentage of the true line width w . The results show that the extracted line width is very accurate if the line model that matches the line type in the image is used to remove the bias. Here, the errors are smaller than 10% everywhere, and much smaller if the line width is larger than approximately 0.75σ . This shows that the line width is extracted with a very high subpixel accuracy for all line widths. On the other hand, Fig. 13 shows that a significant bias remains if the line model that is used for the correction does not adequately match the line type in the image. The residual bias in the line widths is frequently larger than 30% or, in absolute terms, frequently larger than one pixel.

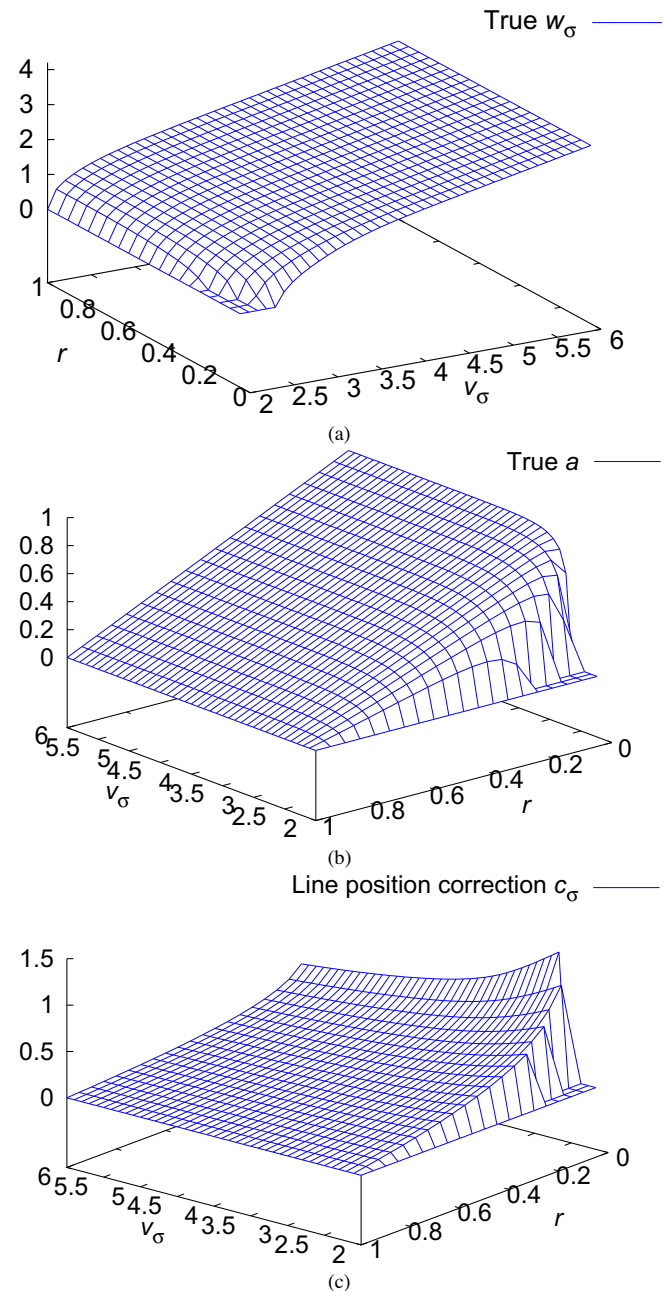


Fig. 10. The inverted bias function for the bar-shaped line f_b for $v_\sigma \in [2, 6]$ and $r \in (0, 1]$. (a) The true line width w_σ . (b) The true asymmetry a . (c) The correction c_σ that must be applied to the line position.

To test the accuracy of the line positions, images that contain a single line of constant width $w = 6.5$ were created. The asymmetry was varied between 0 and 0.8 in steps of 0.02. Since the bias is larger if σ is in the order of the line width or larger, the line extraction was performed with $\sigma = 6.5$. The resulting position errors are shown in Fig. 14. The errors are displayed as a percentage of the true line width w . Like in the previous experiment, the results show that the line position is very accurate if the line model that is used to remove the bias matches the line type in the image. The errors are smaller than 2% everywhere, and much smaller than that almost everywhere. The errors that are close to 2% occur only for parabolic lines with very high asymmetry ($a \geq 0.75$; see Fig. 14b). Note that a 2% error corresponds to an error of 0.13 pixels with the selected line width ($w = 6.5$). This shows that the line po-

sition is extracted with a very high subpixel accuracy for all asymmetries. As in the previous experiment, a significant bias remains if the line model used to correct the bias does not adequately match the line type in the image. The residual bias is not as dramatic as the bias of the line widths, but is significant nevertheless.

The residual errors that occur in the above examples even if the correct line model is used to remove the bias are caused by two effects. First, an error is introduced by the bilinear interpolation of the tabulated inverted bias function (see Section 5). This effect could be reduced by sampling the table more finely. Second, as described in Section 2, the edges that correspond to a line point are extracted by convolving the gradient image with 3×3 filter masks. This introduces an additional, albeit small, smoothing that is not modeled in the bias removal. A possible solution for this effect is

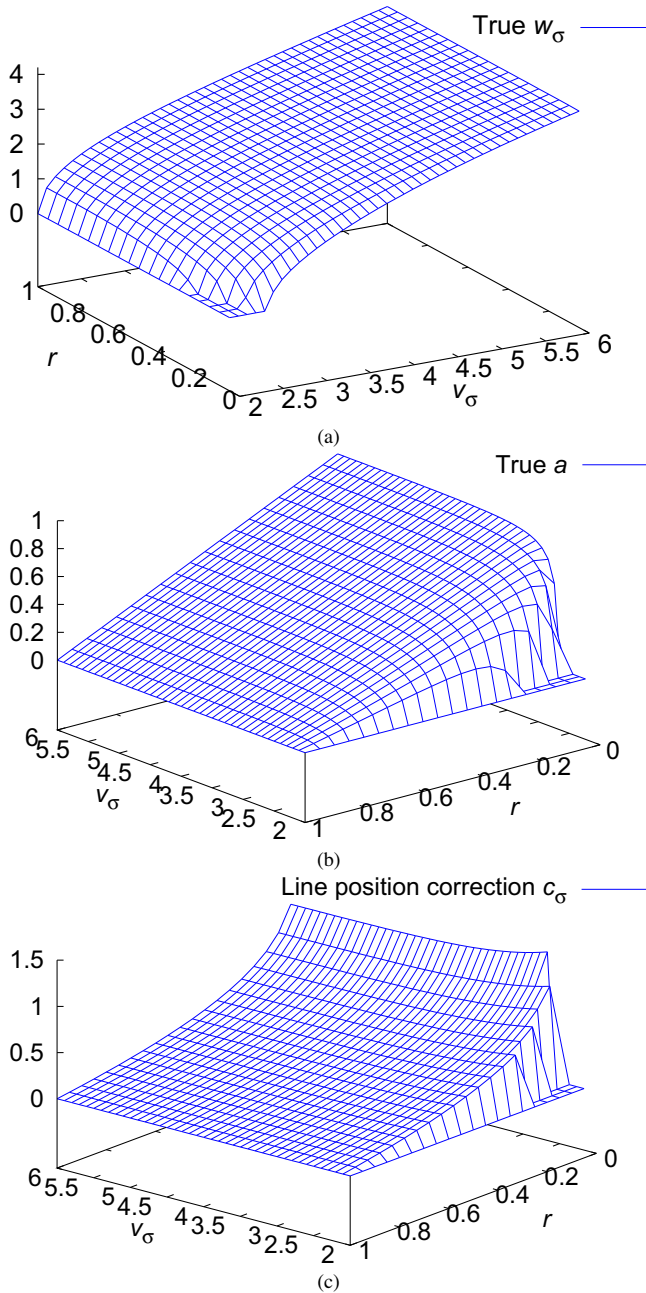


Fig. 11. The inverted bias function for the parabolic line f_p for $v_\sigma \in [2, 6]$ and $r \in (0, 1)$. (a) The true line width w_σ . (b) The true asymmetry a . (c) The correction c_σ that must be applied to the line position.

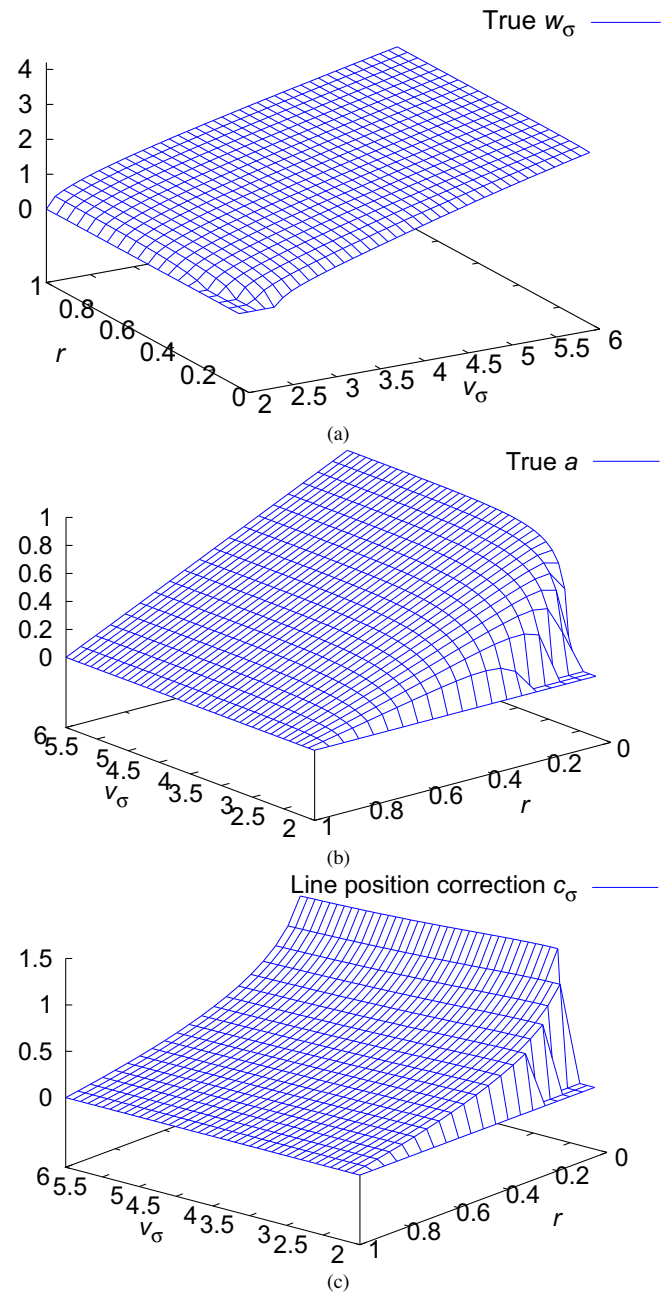


Fig. 12. The inverted bias function for the Gaussian line f_g for $v_\sigma \in [2, 6]$ and $r \in (0, 1)$. (a) The true line width w_σ . (b) The true asymmetry a . (c) The correction c_σ that must be applied to the line position.

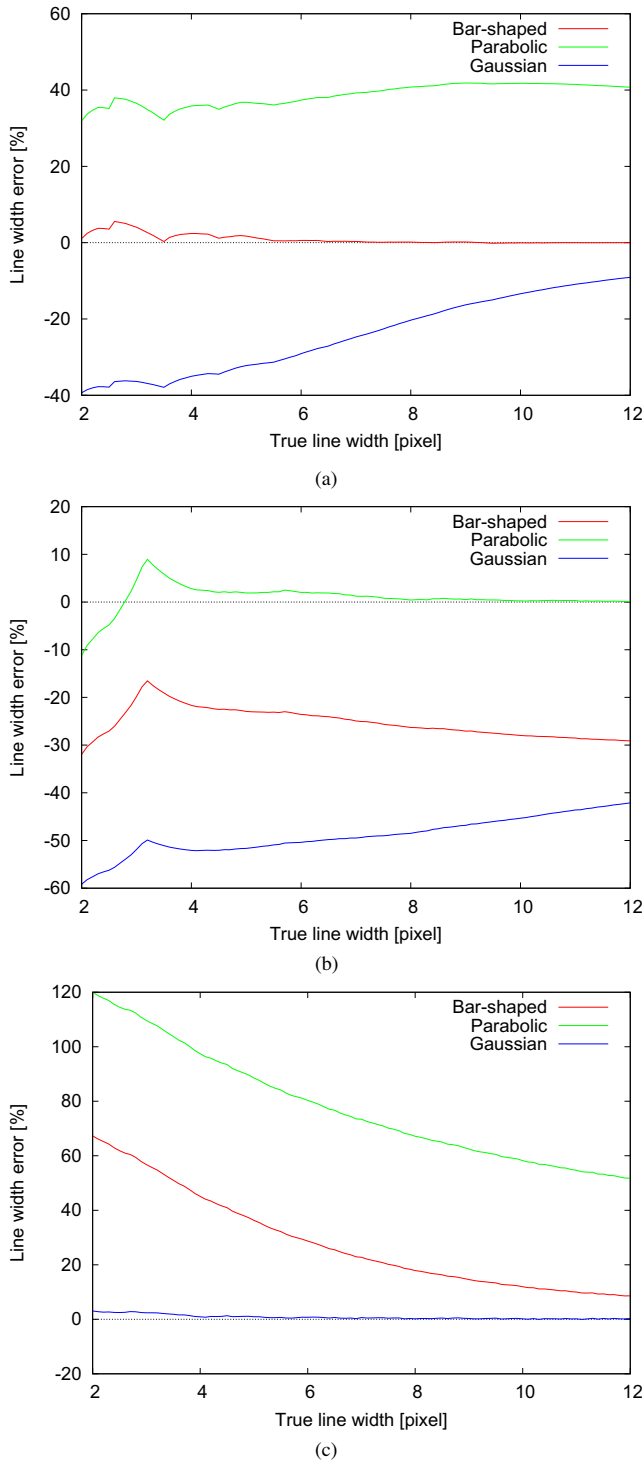


Fig. 13. Line width error in percent of the true line width after correction with the different line models for lines extracted from images containing (a) a bar-shaped line, (b) a parabolic line, and (c) a Gaussian line with $w \in [2, 12]$, $\sigma = 5$, and $a = 0$.

discussed in [33]. Both effects are most noticeable for very small line widths.

The above results show that it is essential that a line model that adequately matches the line type in the image is used to remove the bias. If this is observed, very high subpixel accuracies can be achieved.

In addition, the fact that the line positions and widths are corrected to their true values if the correct line model is selected also provides evidence that the scale-space descriptions of the line pro-

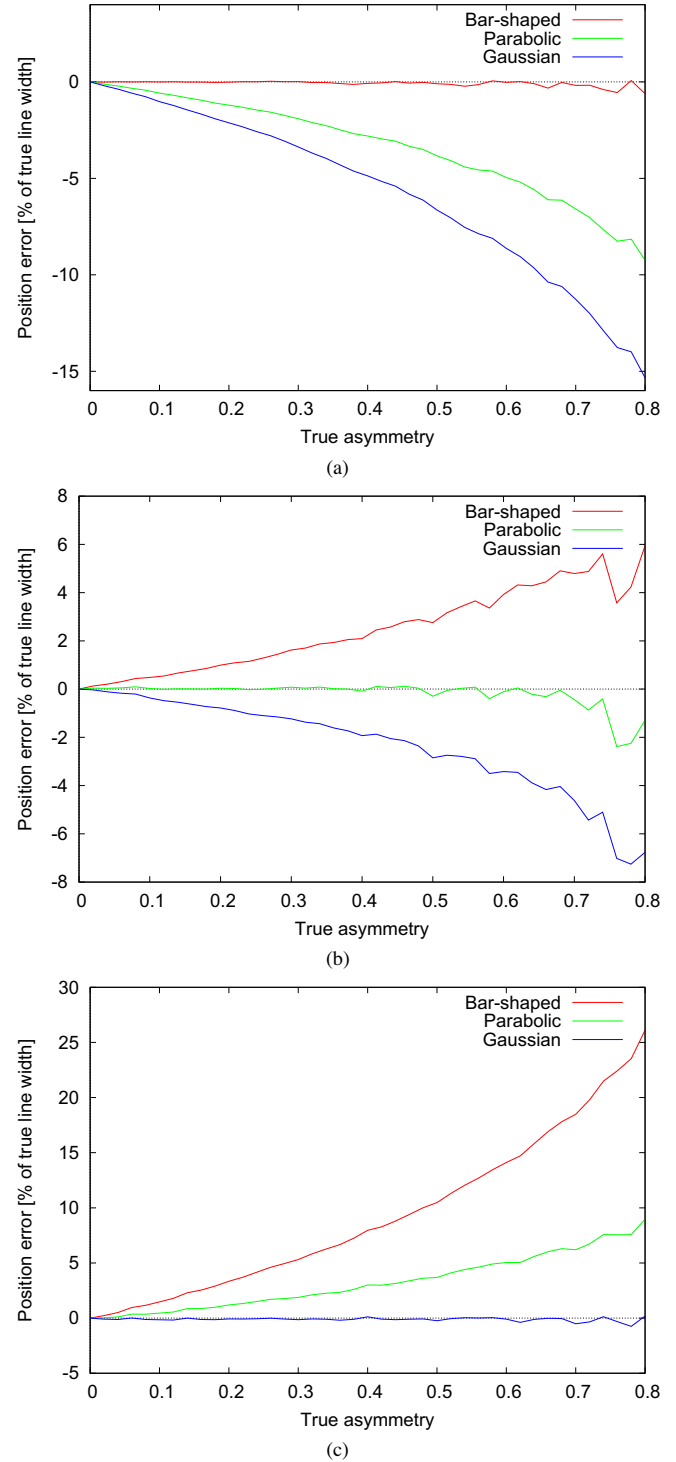


Fig. 14. Line position error in percent of the true line width after correction with the different line models for lines extracted from images containing (a) a bar-shaped line, (b) a parabolic line, and (c) a Gaussian line with $w = 6.5$, $\sigma = 6.5$, and $a \in [0, 0.8]$.

files (12)–(20) are correct, since the bias removal functions b^{-1} are directly derived from them.

6.2. Real images

This section discusses the results of the line extraction algorithm on several real images. The parabolic and Gaussian line mod-

els were used to remove the bias. The results are compared to the results of the original algorithm [33,34], which uses the bar-shaped line model. The examples cover two of the applications that were shown in Figs. 1 and 2. As was noted in these figures, the parabolic and Gaussian line models provide suitable models for different applications. Furthermore, as was discussed at the end of Section 3, the line widths defined by the parabolic and Gaussian line models have somewhat different semantics. On the other hand, all applications that will be shown deal with the extraction of the width of tubular objects. Therefore, to make the results comparable to each other, the line width returned by the bias correction will be used to determine the point at which the gray value drops (for a bright line) or rises (for a dark line) to a fraction $c = 0.25$ of the contrast between the line and the background. Thus, the line widths visualized in the examples are given by $\sqrt{0.75} w_p \approx 0.866 w_p$ for the parabolic profile and $\sqrt{-2 \ln 0.25} w_g \approx 1.665 w_g$ for the Gaussian profile.

Fig. 15 displays an example of the results that are obtained when using the parabolic line model to correct the line positions and widths that are extracted from a coronary angiogram. As can be seen from Fig. 15b and d, using the bar-shaped line model to remove the bias results in line widths that are significantly too small. A significant portion of the vessel is outside the extracted line width. The results using the parabolic line model in Fig. 15c and e provide a significantly more accurate extraction of the line width. In particular, it can be seen from Fig. 15e that the extracted line widths correspond very closely to the points where the gray values rise to 25% of the contrast of the line (i.e., to $c = 0.25$).

Similar results can be seen in Fig. 16, which shows an application that extracts lines from images of blood vessels in the conjunctiva. Here, the Gaussian line model provides significantly more accurate results than the bar-shaped model. In both images, it can be seen that the bar-shaped line model results in lines that are extracted significantly too narrow. Again, a significant portion of the blood vessel lies outside the extracted line width. In contrast, using the Gaussian line model with $c = 0.25$ yields significantly more accurate line widths. Note that the algorithm handles the asymmetric line in Fig. 16f–j correctly.

6.3. Real images with ground truth

To conclude the discussion of the performance of the algorithm, an evaluation of the accuracy of the line widths was carried out based on the data set described in [54]. This data set consists of 20 retinal fundus images, four of which are shown in Fig. 17, along with two sets of 20 images with ground truth pixels that correspond to the blood vessels. The two sets of ground truth were labeled by two different persons. The ground truth differs substantially between the two sets: “review indicated that the first person took a more conservative view of the boundaries of vessels and in the identification of small vessels than the second person” [54]. Fig. 18 displays the actual distribution of line widths in the two data sets. In terms of Section 3, the second person generally seems to have used a much smaller value of c to label the ground truth than the first person, resulting in significantly wider lines.

As noted above, the ground truth data sets of [54] consist of binary images in which the pixels that belong to blood vessels have been labeled manually. To assess the accuracy of the line widths, this data must be converted into ground truth line widths. This was done by extracting bar-shaped lines from the ground truth images with the line extractor proposed in this paper. This is a feasible approach since Section 6.1 has shown that the correct line widths are extracted if the model matches the line type in the image, which is obviously the case for the binary ground truth images. Next, lines were extracted from the green channel of the original retinal fundus images (as in [54]) with $\sigma = 3$, and the line

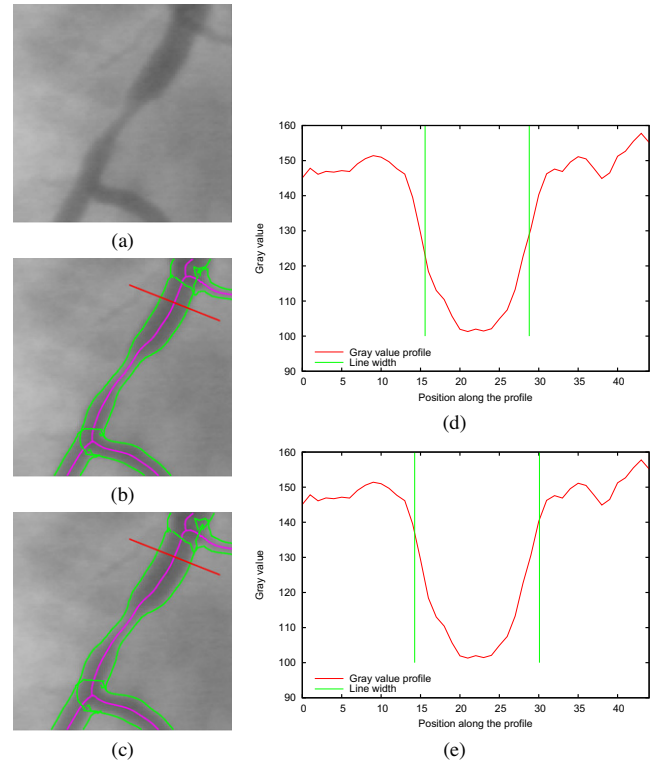


Fig. 15. Example for the extraction of lines from a coronary angiogram. For better visibility, the image contrast has been enhanced by a factor of 2. (a) Image part. (b) Lines and their widths extracted using the bar-shaped line model for bias removal. (c) Lines and their widths extracted using the parabolic line model for bias removal (with $c = 0.25$). (d) Gray value profile along the straight line in (b) and (c). The vertical lines visualize the extracted line width using the bar-shaped line model. (e) Gray value profile along the straight line in (b) and (c). The vertical lines visualize the extracted line width using the parabolic line model. Note that the bar-shaped line model leads to line widths that are significantly too small. A significant portion of the blood vessel lies outside the extracted line width. In contrast, the line widths extracted with the parabolic model with $c = 0.25$ capture the correct extent of the blood vessel.

width and position correction of Section 5 was applied for the three different line models. The line positions were then used to find the closest line point in the ground truth data. If the two line points were closer than a threshold (two pixels), the line width error was computed by subtracting the extracted line width from the ground truth line width. As in Section 3, to enable a meaningful evaluation of the line widths for the parabolic and Gaussian models, the actual location of the line width must be defined based on the model line width w . In this case, the contrast-based definition of Section 3 is appropriate. As noted above, the ground truth data apparently was determined based on significantly different values of c . Empirically, the Gaussian line profile with $c = 0.37$ and $c = 0.27$ resulted in the best agreement of the extraction results with the two sets of ground truth. This method of estimating c had to be adopted because it has not been published what criteria the two persons that created the two ground truth sets used to determine the line widths. Ideally, of course, the ground truth should have been generated by specifying c before labeling the ground truth.

Fig. 19 displays the results of this evaluation. The left column displays the line width errors with respect to the ground truth labeled by the first person, while the right column shows the line width errors with respect to the ground truth labeled by the second person. Each graph shows the median error and the interquartile range of the errors. To derive these statistics, the errors were grouped into ranges of line widths in steps of 0.5, i.e., the error bars

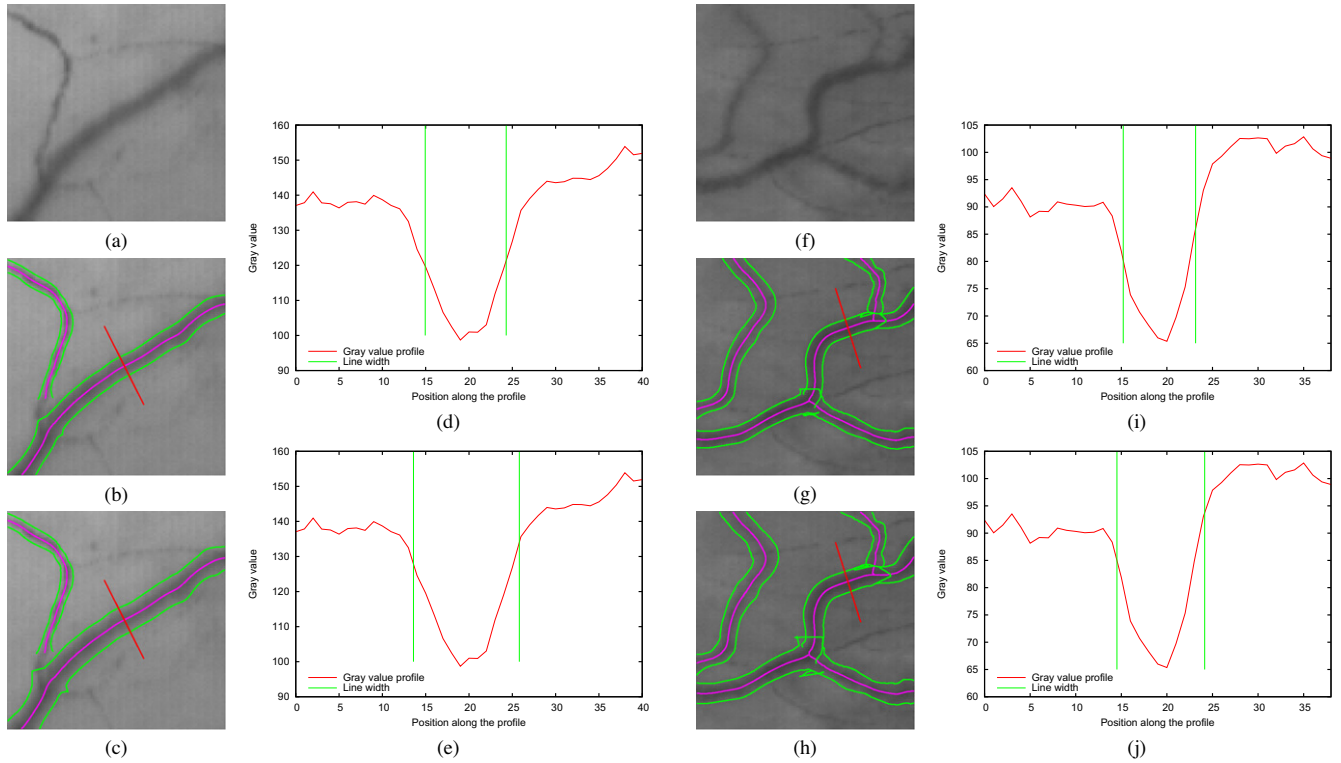


Fig. 16. Example for the extraction of lines from images of blood vessels in the conjunctiva. (a and f) Image parts. (b and g) Lines and their widths extracted using the bar-shaped line model for bias removal. (c and h) Lines and their widths extracted using the Gaussian line model for bias removal (with $c = 0.25$). (d and i) Gray value profile along the straight lines in (b), (c), (g), and (h), respectively. The vertical lines visualize the extracted line width using the bar-shaped line model. (e), (j) Gray value profile along the straight lines in (b), (c) and (g), (h), respectively. The vertical lines visualize the extracted line width using the Gaussian line model. Note that the bar-shaped line model leads to line widths that are significantly too small. A significant portion of the blood vessels lies outside the extracted line width. In contrast, the line widths extracted with the Gaussian model with $c = 0.25$ capture the correct extent of the blood vessels.

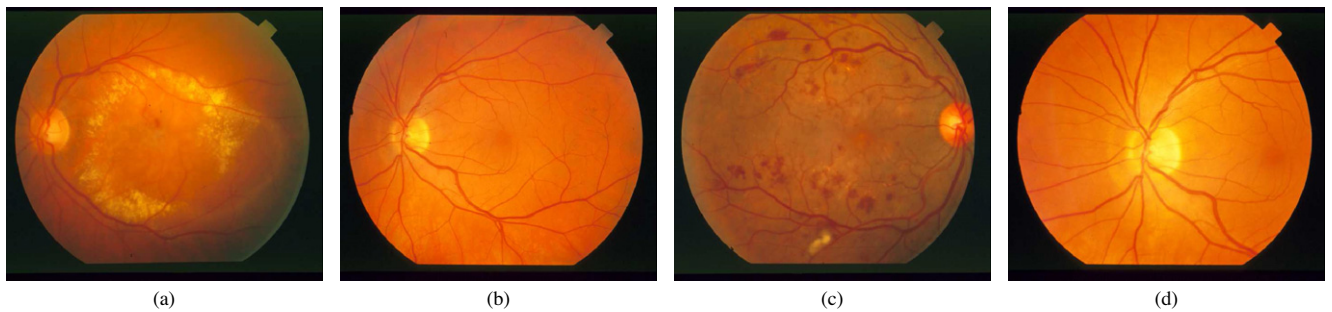


Fig. 17. Four of the 20 retinal fundus images in the data set in [54].

correspond to ground truth line widths in the intervals $[i/2, (i+1)/2)$, $i = 0, \dots, 16$. The error bars in the graphs in Fig. 19 are drawn at the midpoints of these intervals. The first row displays the line width errors that are obtained with the bar-shaped line model. Since the bar-shaped model does not match the line profile of the blood vessels, the line widths are significantly biased. Small line widths are extracted too large, while large line widths are extracted too small. This effect was also observed in [17]. The second row exhibits the results of using the parabolic line model. This results in a significantly smaller bias for large line widths, which, however, are still extracted too small. Finally, the third row shows the results of using the Gaussian line model. This results in very accurate line widths if the true line width is greater than 2. There is still some remaining bias for smaller line widths. Note that for very small line widths (smaller than 1), the line and its corresponding edge will have to lie in almost the same pixel. Therefore, it is likely infeasible to extract these line widths correctly. The remaining bias for line widths between 1 and 2 may

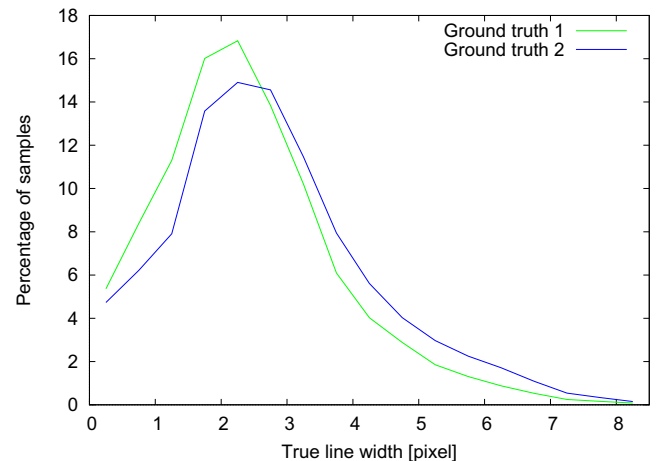


Fig. 18. The distribution of line widths for the two sets of ground truth in [54].

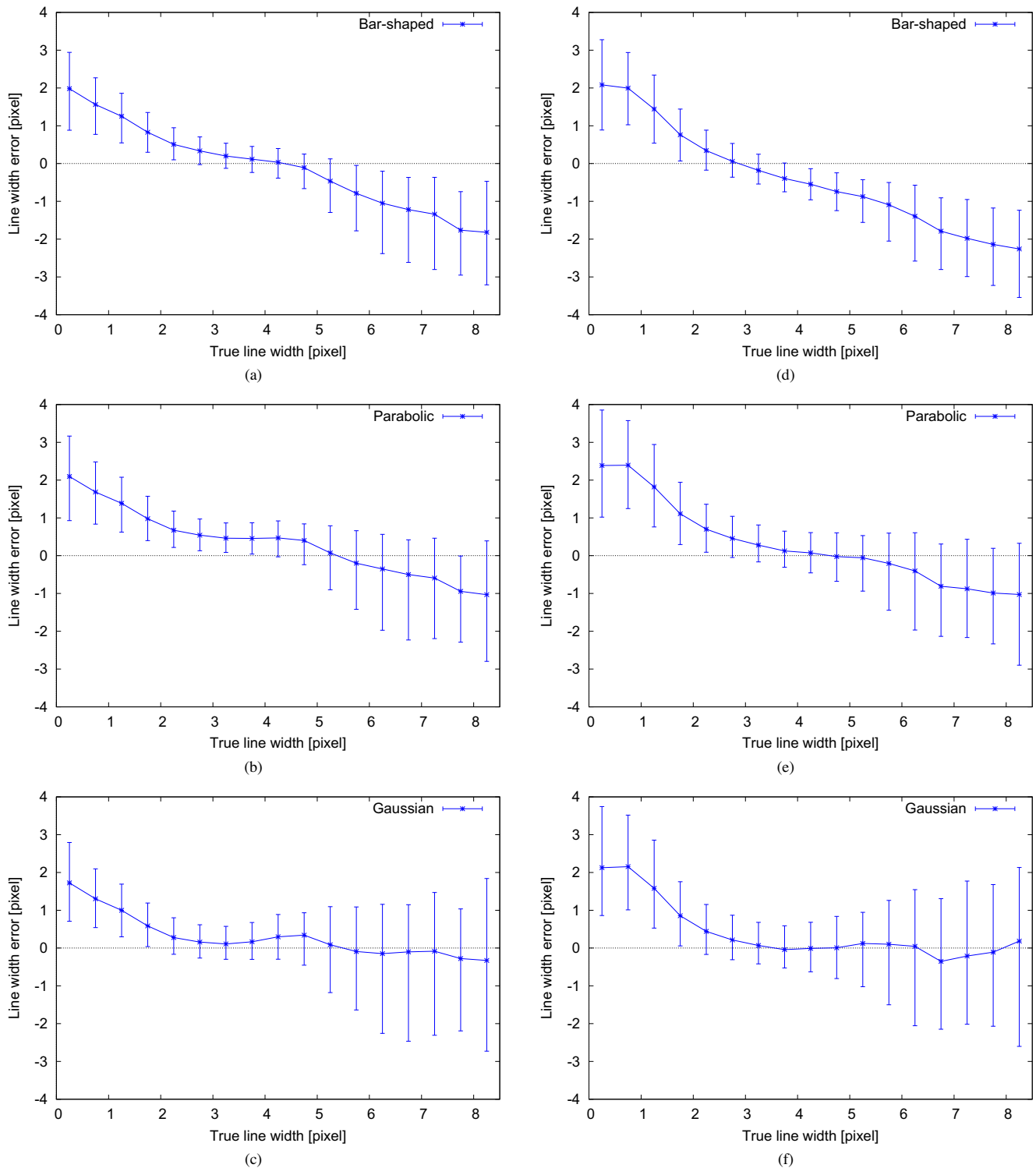


Fig. 19. The line width error of the three different line models based on the ground truth of [54]. The left column (a)–(c) displays the line width errors with respect to the ground truth labeled by the first person in [54], while the right column shows the line width errors with respect to the ground truth labeled by the second person. The first row (a and d) exhibits the line width errors of the bar-shaped line, the second row (b and e) the line width errors of the parabolic line, and the third row (c and f) the line width errors of the Gaussian line. Each graph shows the median error and the interquartile range of the errors.

be caused by the superposition of two Gaussians—one from the line itself and one from the blurring of the optics and sensor with which the images have been acquired. This is currently not modeled.

The interquartile ranges in the graphs show that line widths close to the value of σ are extracted with the highest precision.

In fact, for these line widths the interquartile range of errors roughly corresponds to the errors that are inherent in the ground truth, which is only pixel-precise. For large line widths, the interquartile range is larger because in the test set the large line widths occur predominantly in line junction areas. As described in Section 2, none of the line profiles can model the geometry of the line

adequately in junction areas, which leads to a wider distribution of errors. For small line widths, the lines in the test set correspond to narrow blood vessels, which often have a very low contrast and therefore are influenced by background structures to a much larger extent. The larger interquartile range is caused by the fact that the edges of the lines are sometimes extracted at background structures in the image that have a larger contrast than the actual line. To put this into perspective, Fig. 18 shows that approximately 1% of the lines have a ground truth line width greater than 6 pixels and approximately 10% of the lines have a ground truth line width smaller than 1 pixel. Therefore, approximately 89% of the line widths are extracted very accurately by the proposed algorithm.

7. Conclusions

This paper has presented an approach to extract line positions and line widths with high accuracy. It has extended the approach proposed in [33,34], which uses asymmetric bar-shaped lines as the line model, by two line models that occur frequently in practice: asymmetric parabolic and Gaussian lines. A detailed scale-space analysis has shown that the bias for parabolic and Gaussian lines is even more significant than for bar-shaped lines. An algorithm to remove the bias from the extracted line positions and widths has been described. Experiments on synthetic images have shown that the line position and width can be extracted with sub-pixel accuracy if the line model used to remove the bias matches the line type in the image well. Experiments on real images have shown that the accuracy of the extracted line widths improve significantly if the parabolic or Gaussian line model is used in applications that use images of tubular objects, such as images of blood vessels in different modalities or images of neurites.

The approach to model and remove the bias proposed in this paper could be extended in various directions. For example, it seems possible to extend the approach to line profiles for which the integrals that are required to predict the bias cannot be computed analytically. In this case, the integrals must be evaluated numerically [53, Chapter 4]. An example for this kind of profile is the elliptical profile proposed in [46]. This profile can easily be extended to handle asymmetries. On the other hand, the parabolic profile proposed in this paper already provides an excellent approximation of an elliptical profile. Therefore, the accuracy gained by deriving a model for elliptical profiles may not prove worth the effort. Another extension might be to model lines as a symmetric profile plus a linear background gradient, e.g., $f_g(x) + bx$, where $a = 0$ in $f_g(x)$. Here, $b \neq 0$ causes an asymmetry. This might prove to be useful in some medical applications of line extraction. With this kind of model, the integrals required to predict the bias are tractable for all three line models. Finally, it might be interesting to develop an algorithm that automatically selects the appropriate line model based on the image data itself. Whether this might be useful depends on the number of applications where lines of different types can occur simultaneously in an image.

Another possible avenue for future research might be to try to model and remove the bias for other kinds of line detectors as well. For example, it seems to be possible to perform the bias prediction for the line detector proposed in [37,38]. The convolutions with the Deriche filters [40] seem to be tractable at least for the bar-shaped line profile. Note, however, that the modeling is performed in 1D, and therefore requires that the filters that are used to extract lines in 2D are isotropic. However, it is well known that the Deriche filters are anisotropic [55]. Therefore, it might turn out that the analysis is intractable in 2D. Furthermore, even if the analysis is tractable, the bias correction might require a 3D table indexed by ν_σ , r , and the angle of the line in the image, and might thus be infeasible because of large memory consumption (code size). An-

other class of line extractors for which it might be interesting at least to model the bias is the class of extractors that are based on the structure tensor [43,42]. This kind of extractor will also return significantly biased results. In contrast to the line extractor proposed in this paper, in which the line position is biased towards the side of the line with the weaker gradient, the bias in the structure tensor approaches is towards the side of the line with the stronger gradient. Since the structure tensor approaches involve two smoothing operations, one for the calculation of the gradient and one for the calculation of the structure tensor itself, the analysis might turn out to be intractable analytically.

Acknowledgments

The author wishes to express his gratitude towards the following persons for providing some of the images that were used in this paper: Dr. Udo Bürgel of the University of Mainz for providing the coronary angiogram in Figs. 1, 2, and 15; Dr. Sarah Barman of Kingston University, London, for providing the images of the conjunctival vessels in Figs. 1, 2, and 16; Dr. Gerald Birk of Boehringer Ingelheim Pharma GmbH & Co. KG, Biberach, for providing the neurite images in Fig. 1; and Prof. Adam Hoover of Clemson University for providing the retinal fundus images in Fig. 17.

References

- [1] B. Wessel, Automatische Extraktion von Straßen aus SAR-Bilddaten, Deutsche Geodätische Kommission, Reihe C, Heft 600, München, 2006.
- [2] K. Hedman, Statistical Fusion of Multi-aspect Synthetic Aperture Radar Data for Automatic Road Extraction, Deutsche Geodätische Kommission, Reihe C, Heft 654, München, 2010.
- [3] S. Hinz, A. Baumgartner, Automatic extraction of urban road networks from multi-view aerial imagery, ISPRS J. Photogramm. Remote Sens. 58 (2003) 83–98.
- [4] S. Hinz, C. Wiedemann, Increasing efficiency of road extraction by self-diagnosis, Photogramm. Eng. Remote Sens. 70 (2004) 1457–1466.
- [5] M. Gerke, M. Butenuth, C. Heipke, F. Willrich, Graph-supported verification of road databases, ISPRS J. Photogramm. Remote Sens. 58 (2004) 152–165.
- [6] W. Krüger, Robust and efficient map-to-image registration with line segments, Mach. Vision Appl. 13 (2001) 38–50.
- [7] A. Thiele, E. Cadario, K. Schulz, U. Thönnessen, U. Soergel, Building recognition from multi-aspect high-resolution InSAR data in urban areas, IEEE Trans. Geosci. Remote Sens. 45 (2007) 3583–3593.
- [8] A. Jonk, R. van den Boomgaard, A. Smeulders, Grammatical inference of dashed lines, Comput. Vision Image Understand. 74 (1999) 212–226.
- [9] Z. Wei, F. Zhou, G. Zhang, 3d coordinates measurement based on structured light sensor, Sensor Actuat. A: Phys. 120 (2005) 527–535.
- [10] A.K. Wong, P. Niu, X. He, Fast acquisition of dense depth data by a new structured light scheme, Comput. Vision Image Understand. 98 (2005) 398–422.
- [11] J. Sun, G. Zhang, Z. Wei, F. Zhou, Large 3d free surface measurement using a mobile coded light-based stereo vision system, Sensor Actuat. A: Phys. 132 (2006) 460–471.
- [12] S. Hinz, M. Stephani, L. Schiemann, K. Zeller, An image engineering system for the inspection of transparent construction materials, ISPRS J. Photogramm. Remote Sens. 64 (2009) 297–307.
- [13] C. Lemaître, J. Miteran, J. Matas, Definition of a model-based detector of curvilinear regions, in: W.G. Kropatsch, M. Kampel, A. Hanbury (Eds.), Computer Analysis of Images and Patterns, Lecture Notes in Computer Science, vol. 4673, Springer-Verlag, Berlin, 2007, pp. 686–693.
- [14] W.G. Anderson, R. Balasubramanian, Time-frequency detection of gravitational waves, Phys. Rev. D 60 (1999).
- [15] C. Torres, W.G. Anderson, Progress on a detection algorithm for longer lived gravitational wave bursts, Classical Quant. Grav. 22 (2005) S1169–S1178.
- [16] S. Ravulapalli, S. Sarkar, Association of sound to motion in video using perceptual organization, 18th International Conference on Pattern Recognition, vol. 1, IEEE Computer Society Press, Los Alamitos, 2006, pp. 1216–1219.
- [17] C.G. Owen, T.J. Ellis, E.G. Woodward, A comparison of manual and automated methods of measuring conjunctival vessel widths from photographic and digital images, Ophthalm. Physiol. Opt. 24 (2004) 74–81.
- [18] C.G. Owen, R.S.B. Newsom, A.R. Rudnicka, T.J. Ellis, E.G. Woodward, Vascular response of the bulbar conjunctiva to diabetes and elevated blood pressure, Ophthalmology 112 (2005) 1801–1808.
- [19] A.A. Vlachokosta, P.A. Asvestas, G.K. Matsopoulos, N. Uzunoglu, T.G. Zeyen, Preliminary study on the association of vessel diameter variation and glaucoma, in: 29th Annual International Conference of the IEEE Engineering

- in Medicine and Biology Society, IEEE Computer Society Press, Los Alamitos, pp. 888–891.
- [20] J.G.G. Dobbe, G.J. Streekstra, B. Atasever, R. van Zijderveld, C. Ince, Measurement of functional microcirculatory geometry and velocity distributions using automated image analysis, *Med. Biol. Eng. Comput.* 46 (2008) 659–670.
 - [21] P.L. Rosin, D. Marshall, J.E. Morgan, Multimodal retinal imaging: new strategies for the detection of glaucoma, *International Conference on Image Processing*, vol. 3, IEEE Computer Society Press, Los Alamitos, 2002, pp. 137–140.
 - [22] G.K. Matsopoulos, P.A. Asvestas, N.A. Mouravliansky, K.K. Delibasis, Multimodal registration of retinal images using self organizing maps, *IEEE Trans. Med. Imaging* 23 (2004) 1557–1563.
 - [23] M.G. Fleming, C. Steger, J. Zhang, J. Gao, A.B. Cognetta, I. Pollak, C.R. Dyer, Techniques for a structural analysis of dermatoscopic imagery, *Comput. Med. Imaging Graph.* 22 (1998) 375–389.
 - [24] M.G. Fleming, C. Steger, A.B. Cognetta, J. Zhang, Analysis of the network pattern in dermatoscopic images, *Skin Res. Technol.* (1999) 42–48.
 - [25] C. Grana, R. Cucchiara, G. Pellacani, S. Seidenari, Line detection and texture characterization of network patterns, *18th International Conference on Pattern Recognition*, vol. 2, IEEE Computer Society Press, Los Alamitos, 2006, pp. 275–278.
 - [26] G. Xiong, X. Zhou, A. Degterev, L. Ji, S.T.C. Wong, Automated neurite labeling and analysis in fluorescence microscopy images, *Cytometry Part A* 69A (2006) 494–505.
 - [27] Y. Zhang, X. Zhou, R.M. Witt, B.L. Sabatini, D. Adjero, S.T.C. Wong, Dendritic spine detection using curvilinear structure detector and LDA classifier, *NeuroImage* 36 (2007) 346–360.
 - [28] S. Masure, H. Geerts, M. Cik, E. Hoefnagel, G.V.D. Kieboom, A. Tuytelaars, S.H.A.S.J. Lesage, J.E. Leysen, L. van der Helm, P. Verhasselt, J. Yon, R.D. Gordon, Enovin, a member of the glial cell-line-derived neurotrophic factor (GDNF) family with growth promoting activity on neuronal cells, *Eur. J. Biochem.* 266 (1999) 892–902.
 - [29] R. Nuydens, G. Dispersyn, G.V.D. Kieboom, M. de Jong, R. Connors, F. Ramaekers, M. Borgers, H. Geerts, Bcl-2 protects against apoptosis-related microtubule alterations in neuronal cells, *Apoptosis* 5 (2000) 43–51.
 - [30] V. Racine, M. Sachse, J. Salamer, V. Fraiser, A. Trubuil, J.-B. Sibarita, Visualization and quantification of vesicle trafficking on a three-dimensional cytoskeleton network in living cells, *J. Microsc.* 225 (2007) 214–228.
 - [31] M. Shah, A. Sood, R. Jain, Pulse and staircase edge models, *Comput. Vision Graph. Image Process.* 34 (1986) 321–343.
 - [32] J.S. Chen, G. Medioni, Detection, localization, and estimation of edges, *IEEE Trans. Pattern Anal. Mach. Intell.* 11 (1989) 191–198.
 - [33] C. Steger, An unbiased detector of curvilinear structures, *IEEE Trans. Pattern Anal. Mach. Intell.* 20 (1998) 113–125.
 - [34] C. Steger, Unbiased Extraction of Curvilinear Structures from 2D and 3D Images, Dissertation, Fakultät für Informatik, Technische Universität München, 1998.
 - [35] C. Steger, Subpixel-precise extraction of lines and edges, in: *International Archives of Photogrammetry and Remote Sensing*, volume XXXIII, part B3, International Society for Photogrammetry and Remote Sensing, 2000, pp. 141–156.
 - [36] J. Chen, Y. Sato, S. Tamura, Orientation space filtering for multiple orientation line segmentation, *IEEE Trans. Pattern Anal. Mach. Intell.* 22 (2000) 417–429.
 - [37] D. Ziou, Optimal line detector, *15th International Conference on Pattern Recognition*, vol. 3, IEEE Computer Society Press, Los Alamitos, 2000, pp. 534–537.
 - [38] F. Deschênes, D. Ziou, M.-F. Auclair-Fortier, Detection of lines, line junctions and line terminations, *Int. J. Remote Sens.* 25 (2004) 511–535.
 - [39] D. Ziou, Line detection using an optimal IIR filter, *Pattern Recogn.* 24 (1991) 465–478.
 - [40] R. Deriche, Fast algorithms for low-level vision, *IEEE Trans. Pattern Anal. Mach. Intell.* 12 (1990) 78–87.
 - [41] J. Canny, A computational approach to edge detection, *IEEE Trans. Pattern Anal. Mach. Intell.* 8 (1986) 679–698.
 - [42] K. Rothaus, X. Jiang, Multi-scale midline extraction using creaseness, in: S. Singh, M. Singh, C. Apte, P. Perner (Eds.), *Third International Conference on Advances in Pattern Recognition*, Lecture Notes in Computer Science, vol. 3687, Springer-Verlag, Berlin, 2005, pp. 502–511.
 - [43] W. Förstner, A framework for low level feature extraction, in: J.-O. Eklundh (Ed.), *Third European Conference on Computer Vision*, Lecture Notes in Computer Science, vol. 801, Springer-Verlag, Berlin, 1994, pp. 383–394.
 - [44] L. Liu, D. Zhang, J. You, Detecting wide lines using isotropic nonlinear filtering, *IEEE Trans. Image Process.* 16 (2007) 1584–1595.
 - [45] G.J. Streekstra, J. van Pelt, Analysis of tubular structures in three-dimensional confocal images, *Network: Comput. Neural Syst.* 13 (2002) 381–395.
 - [46] T.N. Pappas, J.S. Lim, A new method for estimation of coronary artery dimensions in angiograms, *IEEE Trans. Acoust. Speech Signal Process.* 36 (1988) 1501–1513.
 - [47] S.M.B. Rassam, V. Patel, O. Brinchmann-Hansen, O. Engvold, Accurate vessel width measurement from fundus photographs: a new concept, *Brit. J. Ophthalmol.* 78 (1994) 24–29.
 - [48] C.G. Owen, T.J. Ellis, A.R. Rudnicka, E.G. Woodward, Optimal green (red-free) digital imaging of conjunctival vasculature, *Ophthalm. Physiol. Opt.* 22 (2002) 234–243.
 - [49] F.P. Miles, A.L. Nuttall, Matched filter estimation of serial blood vessel diameters from video images, *IEEE Trans. Med. Imaging* 12 (1993) 147–152.
 - [50] S. Chaudhuri, S. Chatterjee, N. Katz, M. Nelson, M. Goldbaum, Detection of blood vessels in retinal images using two-dimensional matched filters, *IEEE Trans. Med. Imaging* 8 (1989) 263–269.
 - [51] L.M.J. Florack, B.M. ter Haar Romeny, J.J. Koenderink, M.A. Viergever, Scale and the differential structure of images, *Image Vision Comput.* 10 (1992) 376–388.
 - [52] D.F. Rogers, *Procedural Elements for Computer Graphics*, McGraw-Hill, Inc., New York, 1985.
 - [53] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*, second ed., Cambridge University Press, Cambridge, 1992.
 - [54] A. Hoover, V. Kouznetsova, M. Goldbaum, Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response, *IEEE Trans. Med. Imaging* 19 (2000) 203–210.
 - [55] S. Lanser, W. Eckstein, A modification of Deriche's approach to edge detection, *11th International Conference on Pattern Recognition*, vol. III, IEEE Computer Society Press, Los Alamitos, 1992, pp. 633–637.