Extracting Curvilinear Structures: A Differential Geometric Approach

Carsten Steger

Forschungsgruppe Bildverstehen (FG BV)
Informatik IX, Technische Universität München
Orleansstr. 34, 81667 München, Germany
E-mail: stegerc@informatik.tu-muenchen.de

Abstract. In this paper a method to extract curvilinear structures from digital images is presented. The approach is based on differential geometric properties of the image function. For each pixel, the second order Taylor polynomial is computed by convolving the image with the derivatives of a Gaussian smoothing kernel. Line points are required to have a vanishing gradient and a high curvature in the direction perpendicular to the line. The use of the Taylor polynomial and the Gaussian kernels leads to a single response of the filter to each line. Furthermore, the line position can be determined with sub-pixel accuracy. Finally, the algorithm scales to lines of arbitrary width. An analysis about the scale-space behaviour of two typical line types (parabolic and bar-shaped) is given. From this analysis, requirements and useful values for the parameters of the filter can be derived. Additionally, an algorithm to link the individual line points into lines and junctions that preserves the maximum number of line points is given. Examples on aerial images of different resolution illustrate the versatility of the presented approach.

1 Introduction

Extracting lines in digital images is an important low-level operation in computer vision that has many applications, especially in photogrammetric and remote sensing tasks. There it can be used to extract linear features, like roads, railroads, or rivers, from satellite or low resolution aerial imagery.

The published schemes to line detection can be classified into three categories. The first approach detects lines by only considering the gray values of the image [4, 8]. Line points are extracted by using purely local criteria, e.g., local gray value differences. Since this will generate a lot of false hypotheses for line points, elaborate and computationally expensive perceptual grouping schemes have to be used to select salient lines in the image [5, 8]. Furthermore, lines cannot be extracted with sub-pixel accuracy.

The second approach is to regard lines as objects having parallel edges [9, 11]. In a first step, the local direction of a line is determined for each pixel. Then two edge detection filters are applied in the direction perpendicular to the line. Each edge detection filter is tuned to detect either the left or right edge of the line. The responses of each filter are combined in a non-linear way to yield the final response of the operator [9]. The advantage of this approach is that since the edge detection filters are based on the derivatives of Gaussian kernels, the procedure can be iterated over the scale-space parameter σ to detect lines of arbitrary widths. However, because special directional

edge detection filters have to be constructed that are not separable, the approach is computationally expensive.

In the third approach, the image is regarded as a function $z(x, y)$ and lines are detected as ridges and ravines in this function by locally approximating the image function by its second or third order Taylor polynomial. The coefficients of this polynomial are usually determined by using the facet model, i.e., by a least squares fit of the polynomial to the image data over a window of a certain size [6, 1, 7]. The direction of the line is determined from the Hessian matrix of the Taylor polynomial. Line points are then found by selecting pixels that have a high second directional derivative, i.e., a high curvature, perpendicular to the line direction. The advantage of this approach is that lines can be detected with sub-pixel accuracy without having to construct specialized directional filters. However, because the convolution masks that are used to determine the coefficients of the Taylor polynomial are rather poor estimators for the first and second partial derivatives this approach usually leads to multiple responses to a single line, especially when masks larger than 5×5 are used to suppress noise. Therefore, the approach does not scale well and cannot be used to detect lines that are wider than about 5 pixels.

In this paper an approach to line detection that uses the differential geometric approach of the third category of operators will be presented. In contrast to those, the coefficients of a second order Taylor polynomial are determined by convolving the image with the derivatives of a Gaussian smoothing kernel. Because of this, the algorithm can be scaled to lines of arbitrary width. Furthermore, the behaviour of the algorithm in scale space is investigated for various types of lines. Finally, an algorithm to link the detected line points into a topologically sound data structure of lines and junctions is presented.

2 Detection of Line Points

2.1 Models for Lines in 1D

Many approaches to line detection consider lines in 1D to be bar-shaped, i.e., the ideal line of width $2w$ and height h is assumed to have a profile given by

$$f_b(x) = \begin{cases} h, & |x| \leq w \\ 0, & |x| > w \end{cases} . \tag{1}$$

However, due to sampling effects of the sensor lines usually do not have this profile. Figure 1 shows a profile of a line in an aerial image. As can be seen, no flat bar profile is apparent. Therefore, in this paper lines are assumed to have an approximately parabolic profile. The ideal line of width $2w$ and height h is then given by

$$f_p(x) = \begin{cases} h(1 - (x/w)^2), & |x| \leq w \\ 0, & |x| > w \end{cases} . \tag{2}$$

The line detection algorithm will be developed for this type of profile, but the implications of applying it to bar-shaped lines will be considered later on.

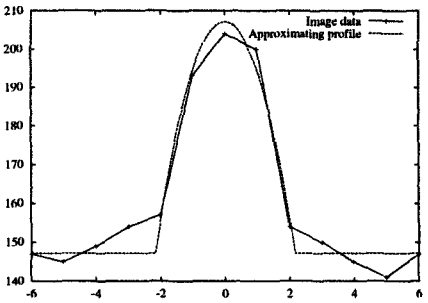


Fig. 1. Profile of a line in an aerial image and approximating ideal line profile

2.2 Detection of Lines in 1D

In order to detect lines with a profile given by (2) in an image $z(x)$ without noise, it is sufficient to determine the points where $z'(x)$ vanishes. However, it is usually convenient to select only salient lines. A useful criterion for salient lines is the magnitude of the second derivative $z''(x)$ in the point where $z'(x) = 0$. Bright lines on a dark background will have $z''(x) \ll 0$ while dark lines on a bright background will have $z''(x) \gg 0$.

Real images will contain a significant amount of noise. Therefore, the scheme described above is not sufficient. In this case, the first and second derivatives of $z(x)$ should be estimated by convolving the image with the derivatives of the Gaussian smoothing kernel

$$g_\sigma(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \quad . \tag{3}$$

The responses, i.e., the estimated derivatives, will be:

$$\begin{aligned} r_p(x,\sigma,w,h) &= g_\sigma(x) * f_p(x) \\ &= \frac{h}{w^2}\big((w^2 - x^2 - \sigma^2)(\phi_\sigma(x+w) - \phi_\sigma(x-w)) - \\ &\qquad 2\sigma^2 x(g_\sigma(x+w) - g_\sigma(x-w)) - \\ &\qquad \sigma^4(g'_\sigma(x+w) - g'_\sigma(x-w))\big) \end{aligned} \tag{4}$$

$$\begin{aligned} r'_p(x,\sigma,w,h) &= g'_\sigma(x) * f_p(x) \\ &= \frac{h}{w^2}\big(-2x(\phi_\sigma(x+w) - \phi_\sigma(x-w)) + \\ &\qquad (w^2 - x^2 - 3\sigma^2)(g_\sigma(x+w) - g_\sigma(x-w)) - \\ &\qquad 2\sigma^2 x(g'_\sigma(x+w) - g'_\sigma(x-w)) - \\ &\qquad \sigma^4(g''_\sigma(x+w) - g''_\sigma(x-w))\big) \end{aligned} \tag{5}$$

$$\begin{aligned} r''_p(x,\sigma,w,h) &= g''_\sigma(x) * f_p(x) \\ &= \frac{h}{w^2}\big(-2(\phi_\sigma(x+w) - \phi_\sigma(x-w)) - \end{aligned}$$

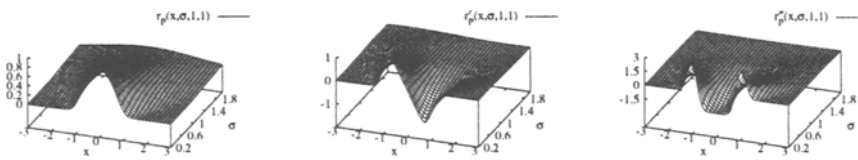


Fig. 2. Scale-space behaviour of the ideal line f_p when convolved with the derivatives of Gaussian kernels for $x \in [-3, 3]$ and $\sigma \in [0.2, 2]$

$$\begin{aligned} &4x(g_\sigma(x+w) - g_\sigma(x-w)) + \\ &(w^2 - x^2 - 5\sigma^2)(g'_\sigma(x+w) - g'_\sigma(x-w)) - \\ &2\sigma^2 x(g''_\sigma(x+w) - g''_\sigma(x-w)) - \\ &\sigma^4(g'''_\sigma(x+w) - g'''_\sigma(x-w))) \end{aligned} \tag{6}$$

where

$$\phi_\sigma(x) = \int_{-\infty}^{x} e^{-\frac{t^2}{2\sigma^2}} dt \ . \tag{7}$$

Equations (4)–(6) give a complete scale-space description of how the ideal line profile f_p will look like when it is convolved with the derivatives of Gaussian kernels. Figure 2 shows the responses for an ideal line with $w = 1$ and $h = 1$ (i.e., a bright line on a dark background) for $x \in [-3, 3]$ and $\sigma \in [0.2, 2]$. As can be seen from this figure, $r'_p(x, \sigma, w, h) = 0 \Leftrightarrow x = 0$ for all σ. Furthermore, $r''_p(x, \sigma, w, h)$ takes on its maximum negative value at $x = 0$ for all σ. Hence it is possible to determine the precise location of the line for all σ. Furthermore, it can be seen that because of the smoothing the ideal line will be flattened out as σ increases. This means that if large values for σ are used, the threshold to select salient lines will have to be set to an accordingly smaller value. Section 4 will give an example of how this can be used in practice to select appropriate thresholds.

For a bar profile without noise no simple criterion that depends only on $z'(x)$ and $z''(x)$ can be given since $z'(x)$ and $z''(x)$ vanish in the interval $[-w, w]$. However, if the bar profile is convolved with the derivatives of the Gaussian kernel, a smooth function is obtained in each case. The responses will be:

$$r_b(x, \sigma, w, h) = h\big(\phi_\sigma(x+w) - \phi_\sigma(x-w)\big) \tag{8}$$

$$r'_b(x, \sigma, w, h) = h\big(g_\sigma(x+w) - g_\sigma(x-w)\big) \tag{9}$$

$$r''_b(x, \sigma, w, h) = h\big(g'_\sigma(x+w) - g'_\sigma(x-w)\big) \ . \tag{10}$$

Figure 3 shows the scale-space behaviour of a bar profile with $w = 1$ and $h = 1$ when it is convolved with the derivatives of a Gaussian. It can be seen that the bar profile gradually becomes "round" at its corners. The first derivative will vanish only at $x = 0$

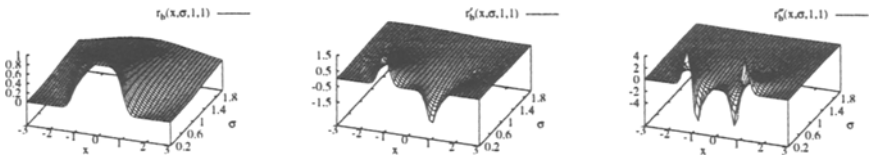


Fig. 3. Scale-space behaviour of the bar-shaped line f_b when convolved with the derivatives of Gaussian kernels for $x \in [-3, 3]$ and $\sigma \in [0.2, 2]$

for all $\sigma > 0$ because of the infinite support of $g_\sigma(x)$. However, the second derivative $r''_b(x, \sigma, w, h)$ will not take on its maximum negative value for small σ. In fact, for $\sigma \leq 0.2$ it will be approximately zero. Furthermore, there will be two distinct minima in the interval $[-w, w]$. It is, however, desirable for $r''_b(x, \sigma, w, h)$ to exhibit a clearly defined minimum at $x = 0$. After some lengthy calculations it can be shown that

$$\sigma \geq w/\sqrt{3} \tag{11}$$

has to hold for this. Furthermore, it can be shown that $r''_b(x, \sigma, w, h)$ will have its maximum negative response in scale-space for $\sigma = w/\sqrt{3}$. This means that the same scheme as described above can be used to detect bar-shaped lines as well. However, the restriction on σ must be observed. The same analysis could be carried out for other types of lines as well, e.g., roof-shaped lines. However, it is expected that no fundamentally different results will be obtained. For all σ above a certain value that depends on the line type the responses will show the desired behaviour of $z'(0) = 0$ and $z''(0) \ll 0$ with $z''(x)$ having a distinct minimum.

The discussion so far has assumed that lines have the same contrast on both sides of the line. This is rarely true in real images, however. For simplicity, only asymetrical bar-shaped lines

$$f_a(x) = \begin{cases} 0, & x < -w \\ 1, & |x| \leq w \\ h, & x > w \end{cases} \tag{12}$$

are considered ($h \in [0, 1]$). The corresponding responses will be:

$$r_a(x, \sigma, w, h) = \phi_\sigma(x + w) - (h - 1)\phi_\sigma(x - w) \tag{13}$$

$$r'_a(x, \sigma, w, h) = g_\sigma(x + w) - (h - 1)g_\sigma(x - w) \tag{14}$$

$$r''_a(x, \sigma, w, h) = g'_\sigma(x + w) - (h - 1)g'_\sigma(x - w) \ . \tag{15}$$

The location where $r'_a(x, \sigma, w, h) = 0$, i.e., the position of the line, is given by

$$x = -\frac{\sigma^2}{2} \ln(1 - h) \ . \tag{16}$$

This means that the line will be estimated in a wrong position when the contrast is significantly different on both sides of the line. The estimated position of the line will be within the actual boundaries of the line as long as

$$h \leq 1 - e^{-\frac{2w}{\sigma^2}} \quad . \tag{17}$$

If $\sigma = 1$ and $w = 1$, for example, the estimated location of the line will be within the actual line if $h \leq 0.86466$. This means that relatively large contrast differences can be handled. Note, however, that as $h \to 1$, i.e., as the bar line profile is gradually transformed into a step edge profile, the location of the line $x \to \infty$. Fortunately, $r''_a(x, \sigma, w, h)$ will have a small value as $h \to 1$, so by simple thresholding these erroneously located lines can be eliminated.

2.3 Lines in 1D, Discrete Case

The analysis so far has been carried out for analytical functions $z(x)$. For discrete signals only two modifications have to be made. The first one is the choice of how to implement the convolution in discrete space. Integrated Gaussian kernels were chosen as convolutions masks, mainly because they give automatic normalization of the masks and a direct criterion on how many coefficients are needed for a given approximation error. The integrated Gaussian is obtained if one regards the discrete image z_n as a piecewise constant function $z(x) = z_n$ for $x \in (n - \frac{1}{2}, n + \frac{1}{2}]$ and integrating the continuous Gaussian kernel over this area. The convolution masks will be given by:

$$g_{n,\sigma} = \phi_\sigma(n + \tfrac{1}{2}) - \phi_\sigma(n - \tfrac{1}{2}) \tag{18}$$

$$g'_{n,\sigma} = g_\sigma(n + \tfrac{1}{2}) - g_\sigma(n - \tfrac{1}{2}) \tag{19}$$

$$g''_{n,\sigma} = g'_\sigma(n + \tfrac{1}{2}) - g'_\sigma(n - \tfrac{1}{2}) \quad . \tag{20}$$

The approximation error is set to 10^{-4} in each case. Of course, other schemes, like Lindeberg's discrete Gaussian derivative approximations [10] or a recursive computation [3], are suitable for the implementation as well.

The second problem that has to be solved is how to determine the location of a line in the discrete case. In principle, one could use a zero crossing detector for this task. However, this would yield the position of the line only with pixel accuracy. In order to overcome this, the second order Taylor polynomial of z_n is examined. Let r, r', and r'' be the locally estimated derivatives at point n of the image that are obtained by convolving the image with g_n, g'_n, and g''_n. Then the Taylor polynomial is given by

$$p(x) = r + r'x + \frac{1}{2}r''x^2 \quad . \tag{21}$$

The position of the line, i.e., the point where $p'(x) = 0$ is

$$x = -\frac{r'}{r''} \quad . \tag{22}$$

The point n is declared a line point if this position falls within the pixel's boundaries, i.e., if $x \in [-\frac{1}{2}, \frac{1}{2}]$ and the second derivative r'' is larger than a user-specified threshold. Please note that in order to extract lines, the response r, which is the smoothed local image intensity, is unnecessary and therefore does not need to be computed.

2.4 Detection of Lines in 2D

Curvilinear structures in 2D can be modeled as curves $s(t)$ that exhibit a characteristic 1D line profile (e.g., f_p or f_b) in the direction perpendicular to the line, i.e., perpendicular to $s'(t)$. Let this direction be $n(t)$. This means that the first directional derivative in the direction $n(t)$ should vanish and the second directional derivative should be of large absolute value. No assumption can be made about the derivatives in the direction of $s'(t)$. For example, let $z(x,y)$ be an image that results from sweeping the profile f_p along a circle $s(t)$ of radius r. The second directional derivative perpendicular to $s'(t)$ will have a large negative value, as desired. However, the second directional derivative along $s'(t)$ will also be non-zero.

The only problem that remains is to compute the direction of the line locally for each image point. In order to do this, the partial derivatives r_x, r_y, r_{xx}, r_{xy}, and r_{yy} of the image will have to be estimated. This can be done by convolving the image with the appropriate 2D Gaussian kernels. The direction in which the second directional derivative of $z(x,y)$ takes on its maximum absolute value will be used as the direction $n(t)$. This direction can be determined by calculating the eigenvalues and eigenvectors of the Hessian matrix

$$H(x,y) = \begin{pmatrix} r_{xx} & r_{xy} \\ r_{xy} & r_{yy} \end{pmatrix} . \tag{23}$$

The calculation can be done in a numerically stable and efficient way by using one Jacobi rotation to annihilate the r_{xy} term. Let the eigenvector corresponding to the eigenvalue of maximum absolute value, i.e., the direction perpendicular to the line, be given by (n_x, n_y) with $\|(n_x, n_y)\|_2 = 1$. As in the 1D case, a quadratic polynomial will be used to determine whether the first directional derivative along (n_x, n_y) vanishes within the current pixel. This point will be given by

$$(p_x, p_y) = (tn_x, tn_y) , \tag{24}$$

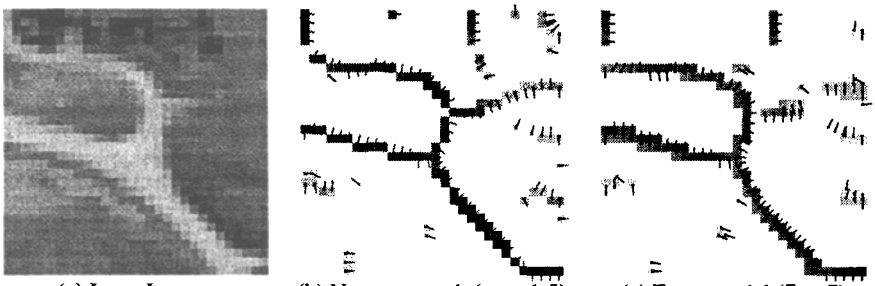
where

$$t = -\frac{r_x n_x + r_y n_y}{r_{xx} n_x^2 + 2 r_{xy} n_x n_y + r_{yy} n_y^2} . \tag{25}$$

Again, $(p_x, p_y) \in [-\frac{1}{2}, \frac{1}{2}] \times [-\frac{1}{2}, \frac{1}{2}]$ is required in order for a point to be declared a line point. As in the 1D case, the second directional derivative along (n_x, n_y), i.e., the maximum eigenvalue, can be used to select salient lines.

2.5 Examples

Figure 4(b) gives an example of the results obtainable with the presented approach. Here, bright line points were extracted from the input image given in Fig. 4(a). This image is part of an aerial image with a ground resolution of 2 m. Figure 4(c) shows the results that were obtained using the facet model. In both cases the sub-pixel location (p_x, p_y) of the line points and the direction (n_x, n_y) perpendicular to the line are symbolized by vectors. The strength of the line, i.e., the absolute value of the second directional derivative along (n_x, n_y) is symbolized by gray values. Line points with high saliency have dark gray values.



(a) Input Image (b) New approach ($\sigma = 1.5$) (c) Facet model (7×7)

Fig. 4. Line points detected in image (a) using the new approach (b) and using the facet model (c)

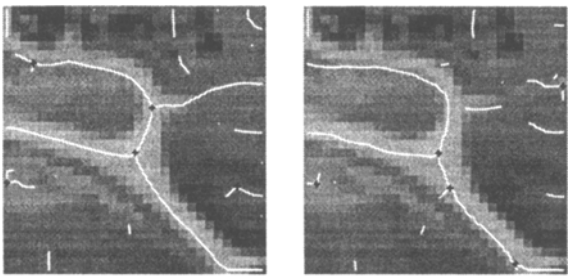
From Fig. 4 it can be seen that in the approach presented here there will always be a single response to a given line. When the facet model is used, multiple responses are quite common. Note, for example, the line that enters the image in the middle of the left hand side. This makes linking the individual line points into lines rather complicated. In [1] the response of the operator is thinned before linking to get around this problem. However, this operation throws away useful information since diagonal lines will be thinned unnecessarily. In the new approach the linking will be considerably easier and no thinning operation is needed.

3 Linking Line Points into Lines

After individual line pixels have been extracted, they must be linked into lines. In order to facilitate later mid-level vision processes, e.g., perceptual grouping, the resulting data structure should contain explicit information about the lines as well as the junctions between them. This data structure should be topologically sound in the sense that junctions are represented by points and not by extended areas as in [1]. Furthermore, since the presented approach yields only single responses to each line, no thinning operation needs to be performed prior to linking. This assures that the maximum information about the line points will be present in the data structure.

Since there is no suitable criterion to classify the line points into junctions and normal line points in advance without having to resort to extended junction areas, another approach has been adopted. From the algorithm in Sect. 2 the following data are obtained for each pixel: the orientation of the line $(n_x, n_y) = (\cos\alpha, \sin\alpha)$, a measure of strength of the line (the second directional derivative in the direction of α), and the sub-pixel location of the line (p_x, p_y).

Starting from the pixel with maximum second derivative, lines will be constructed by adding the appropriate neighbour to the current line. Since it can be assumed that the line point detection algorithm will yield a fairly accurate estimate for the local direction of the line, only three neighbouring pixels that are compatible with this direction are examined. For example, if the current pixel is (c_x, c_y) and the current orientation of the line is in the interval $[-22.5°, 22.5°]$, these points will be $(c_x + 1, c_y - 1)$,



(a) New approach ($\sigma = 1.5$) (b) Facet model (7×7)

Fig. 5. Linked lines detected using the new approach (a) and using the facet model (b). Lines are drawn in white while junctions are displayed as black crosses.

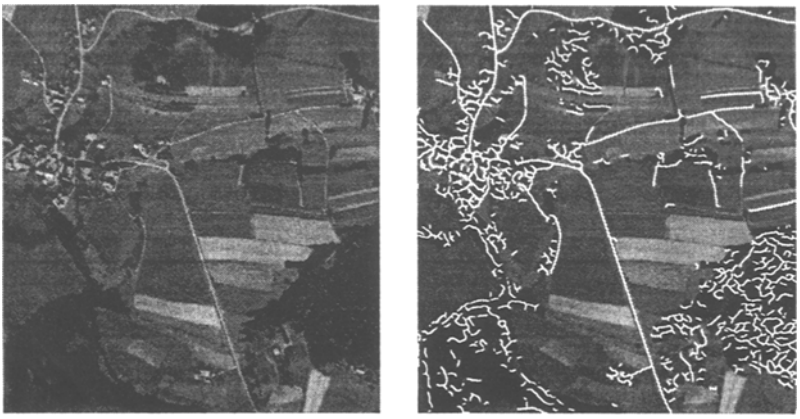
$(c_x + 1, c_y)$, and $(c_x + 1, c_y + 1)$. The choice about the appropriate neighbour to add to the line is based on the distance between the respective sub-pixel locations and the angle difference of the two points. Let $d = \|p_2 - p_1\|_2$ be the distance between the two points and $\beta = |\alpha_2 - \alpha_1|$, $\beta \in [0, \pi/2]$, be the angle difference between those points. The neighbour that is added to the line is the one that minimizes $d + w\beta$. In the current implementation, $w = 1$ is used. This algorithm will select each line point in the correct order. At junction points, it will select one branch to follow without detecting the junction. This will be detected later on. The algorithm of adding line points is continued until no more line points are found in the current neighbourhood or until the best matching candidate is a point that has already been added to another line. If this happens, the point is marked as a junction, and the line that contains the point is split into two lines at the junction point.

New lines will be created as long as the starting point has a second directional derivative that lies above a certain, user-selectable upper threshold. Points are added to the current line as long as their second directional derivative is greater than another user-selectable lower threshold. This is similar to a hysteresis threshold operation [2].

The contour linking approach presented here is similar to that given in [6]. However, there the best neighbour is determined from a neighbourhood that does not depend on the current direction of the line. Furthermore, the author does not mention whether explicit junction information is generated by the algorithm.

With a slight modification the algorithm is able to deal with multiple responses if it is assumed that with the facet model approach no more than three parallel responses are generated. No such case has been encountered for mask sizes of up to 13×13. Under this assumption, the algorithm can proceed as above. Additionally, if there are multiple responses to the line in the direction perpendicular to the line (e.g., the pixels $(c_x, c_y - 1)$ and $(c_x, c_y + 1)$ in the example above), they are marked as processed if they have roughly the same orientation as (c_x, c_y). The termination criterion for lines has to be modified to stop at processed line points instead of line points that are contained in another line.

Figure 5 shows the result of linking the line points in Fig. 4 into lines. The results



(a) Input image (b) Detected lines

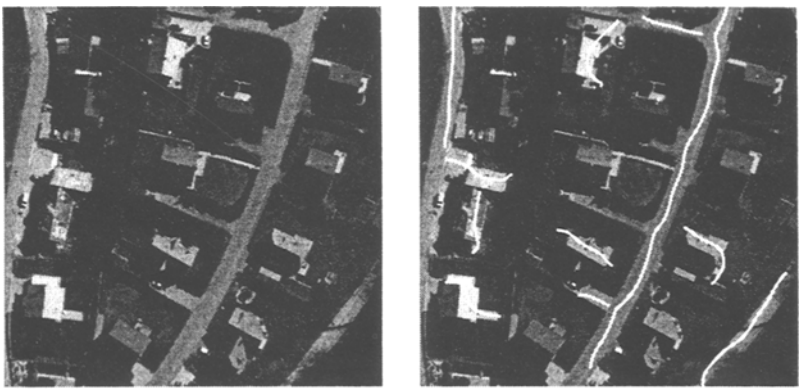
Fig. 6. Salient lines detected in the complete aerial image

are overlaid onto the original image. In this case, the upper threshold was set to zero, i.e., all lines, no matter how faint, were selected. If an upper threshold of 5 were used only the salient lines would be selected. It is apparent that the lines obtained with the new approach are much smoother than the lines obtained with the facet model. Furthermore, the geometric accuracy in case of unequal contrast is better with the new approach. Note, for example, the line that enters the image at the bottom right corner. This line has quite a different contrast on both sides. With the new approach the line is within half a pixel of the true location of the line while with the facet model it lies more than one pixel from the true line.

4 Further Examples

In this section some more examples of the versatility of the proposed approach will be given. Figure 6(a) shows the complete aerial image from which the image in Fig. 4 was taken. In this example, $\sigma = 1.5$ and only bright lines that had a second derivative with an absolute value larger than 8 were selected. The lower threshold for the hysteresis was set to 3. It can be seen from Fig. 6(b) that the algorithm is able to extract most of the salient lines from the image.

Figure 7 shows that the presented approach scales very well. In Fig. 7(a) an aerial image with a ground resolution of $\approx 25\,\text{cm}$ is displayed. The lines in this image are approximately bar-shaped. If 50 pixel wide lines are to be detected, i.e., if $w = 25$, according to (11), a $\sigma \geq 14.4337$ should be selected. In fact, $\sigma = 15$ was used for this image. If lines with a contrast of $h \geq 70$ are to be selected, (10) shows that these lines will have a second derivative of ≈ -0.10316. Therefore, the threshold for the absolute value of the second derivative was set to 0.1. The lower threshold was set to 0.025. Figure 7(b) displays the lines that were detected with these parameters. As can be seen,



(a) Input image (b) Detected lines

Fig. 7. Salient lines detected in a high resolution aerial image

all of the roads were detected. Most of the lines in this image have different contrasts on both sides of the line. Therefore it is not surprising that the detected lines deviate slightly from the true centers of the lines. This is especially true for the line in the bottom right corner of the image. However, even this line is detected within the boundaries of the actual line.

5 Conclusions

In this paper a low-level approach to the extraction of curvilinear structures from images was presented. An analysis of the scale-space behaviour of two distinct line types was carried out. The results of this analysis help tremendously in the selection of the appropriate parameters for the algorithm. The advantages of this approach are that line extraction is done using only the first and second directional derivatives of the image. No specialized directional filters are needed. This makes the approach computationally efficient. For instance, the 520 × 560 image in Fig. 6 was processed in 8 seconds on a HP 735 workstation. Furthermore, since the derivatives are estimated by convolving the image with the derivatives of a Gaussian smoothing kernel, only a single response is generated for each line. The algorithm has no problems extracting line points where three or more lines meet.

An algorithm has been presented that links the extracted line points into a data structure containing lines and junctions. Although the algorithm itself does not attempt any perceptual grouping, the data structure that is generated will facilitate this in a higher-level step.

The presented approach shows two fundamental limitations. Firstly, if a line has highly different contrasts on each side oft the line, the position of the line will be estimated in a different position than the actual center of the line. This is a fundamental limitation of other approaches as well [9, 1]. In this paper, an analysis was carried out that

shows how the position will vary with differing contrasts. Secondly, only a combined estimate of the width and height of the line is returned. This means, that narrow lines with high contrast will result in similar responses as broad lines with low contrast. This contrasts with the approach given in [9] that returns an estimate of the width of the line as well as the height of the line at the expense of computational complexity. However, if only lines of a certain range of widths are present in an image, the combined estimate presents no fundamental limitation since it will then depend only on the contrast of the lines.

References

1. Andreas Busch. Fast recognition of lines in digital images without user-supplied parameters. In *International Archives of Photogrammetry and Remote Sensing*, volume 30, part 3/1, pages 91–97, 1994.
2. John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.
3. Rachid Deriche. Recursively implementing the gaussian and its derivatives. Rapport de Recherche 1893, INRIA, Sophia Antipolis, France, April 1993.
4. M. A. Fischler, J. M. Tenenbaum, and H. C. Wolf. Detection of roads and linear structures in low-resolution aerial imagery using a multisource knowledge integration technique. *Computer Graphics and Image Processing*, 15:201–223, 1981.
5. Martin A. Fischler. The perception of linear structure: A generic linker. In *Image Understanding Workshop*, pages 1565–1579, San Francisco, CA, USA, 1994. Morgan Kaufmann Publishers.
6. Frank Glazer. Curve finding by ridge detection and grouping. In W. Kropatsch and H. Bischof, editors, *Mustererkennung*, Informatik Xpress 5, pages 109–116. Deutsche Arbeitsgemeinschaft für Mustererkennung, 1994.
7. Robert M. Haralick, Layne T. Watson, and Thomas J. Laffey. The topographic primal sketch. *International Journal of Robotics Research*, 2(1):50–72, 1983.
8. Bruno Jedynak and Jean-Philippe Rozé. Tracking roads in satellite images by playing twenty questions. In A. Gruen, O. Kuebler, and P. Agouris, editors, *Automatic Extraction of Man-Made Objects from Aerial and Space Images*, pages 243–253, Basel, Switzerland, 1995. Birkhäuser Verlag.
9. T. M. Koller, G. Gerig, G. Székely, and D. Dettwiler. Multiscale detection of curvilinear structures in 2-D and 3-D image data. In *Fifth International Conference on Computer Vision*, pages 864–869. IEEE Computer Society Press, 1995.
10. Tony Lindeberg. Discrete derivative approximations with scale-space properties: A basis for low-level feature extraction. *Journal of Mathematical Imaging and Vision*, 3(4):349–376, 1993.
11. J. Brian Subirana-Vilanova and Kah Kay Sung. Multi-scale vector-ridge-detection for perceptual organization without edges. A.I. Memo 1318, MIT Artificial Intelligence Laboratory, Cambridge, MA, USA, December 1992.