International Journal of Computer Vision 43(2), 99–111, 2001

A Minimum Cost Approach for Segmenting Networks of Lines

JAN-MARK GEUSEBROEK AND ARNOLD W.M. SMEULDERS
Intelligent Sensory Information Systems, Department of Computer Science, Faculty of Science, University of Amsterdam, Kruislaan 403, 1098 SJ Amsterdam, The Netherlands
geusebroek@science.uva.nl

HUGO GEERTS
Neurobiology, CNS Discovery Research, Janssen Research Foundation, Turnhoutseweg 30, B2340 Beerse, Belgium

Abstract. The extraction and interpretation of networks of lines from images yields important organizational information of the network under consideration. In this paper, a one-parameter algorithm for the extraction of line networks from images is presented. The parameter indicates the extracted saliency level from a hierarchical graph. Input for the algorithm is the domain specific knowledge of interconnection points. Graph morphological tools are used to extract the minimum cost graph which best segments the network.

We give an extensive error analysis for the general case of line extraction. Our method is shown to be robust against gaps in lines, and against spurious vertices at lines, which we consider as the most prominent source of error in line detection. The method indicates detection confidence, thereby supporting error proof interpretation of the network functionality. The method is demonstrated to be applicable on a broad variety of line networks, including dashed lines. Hence, the proposed method yields a major step towards general line tracking algorithms.

Keywords: networks, graph morphology, line detection, dashed line detection, minimum cost path, watershed segmentation, differential geometry, mathematical morphology, Gaussian scale-space

The detection of lines in images is an important low-level task in computer vision. Successful techniques are available for the detection of curvilinear structures (Cohen and Kimmel, 1997; Illingworth and Kittler, 1988; Steger, 1998). They are applied in pharmaceutical research, where interesting tissue parameters can be obtained by the extraction of bloodvessels, neurites, or tissue layers. Furthermore, the extraction of roads, railroads, rivers and channels from satellite or aerial images can be used to update geographic information systems.

A higher level of information is obtained by connecting the lines into networks. Applications here can be found in the roads between crossings or highways connecting cities, the railway system in between stations, the neurite system connecting the neurons, all yielding organizational information of the network under consideration. Extraction of line networks rests on the detection of connections, the vertices in the network, as well as their interconnecting curves (Buckley and Talbot, 2000; Vincent, 1998). The linking of line points over the interconnections is an ill-defined problem since the curves are likely to contain gaps and branches. More attractive is to find the minimum cost path between vertices, the path which contains most line evidence (Barzohar and Cooper, 1993; Bellman, 1957; Sha'ashua and Ullman, 1988). The vertices can be used to guide the line tracking. Network extraction is then reduced to tracing lines between vertices.

In this paper, we consider the robust extraction of networks of lines by the application of minimum cost graphs. Design objective is robustness against gaps in lines, which we consider as the most prominent source of error in network extraction. We propose a robust

measure for edge saliency, which indicates confidence for each connection.

1. Network Extraction Algorithm

A network consists of vertices interconnected by lines.

Definition 1 (Network of Lines). A network of lines is defined by a set of vertices indicating line end points, and the corresponding set of lines representing interconnections, where none of the lines do cross.

The definition above implies vertices at crossings. The network can be segmented by tracing the lines between vertices. Therefore, four steps are considered: **a.** the detection of line points, **b.** the detection of vertices, **c.** finding the optimal paths between neighboring vertices yielding the lines, and **d.** the extraction of the network graph from the set of vertices and lines. A flow diagram is given in Fig. 1. Post-processing may include pruning of the graph to remove false branches, and the assignment of confidence levels to the found graph. Graph confidence is given by the saliency of the detected lines, and the basin coverage indicating how much line evidence is covered by the graph. If the network graph covers all line evidence, no lines are missed. However, if not all line evidence is covered by the graph, lines may be missed during extraction. Hence, basin coverage together with edge saliency indicate missed lines and spurious lines in the network graph. Each of these steps are described in further detail below.

1.1. Vertex Detection

For specific applications, the network vertices are geometrical structures which are more obvious to detect than the interconnecting lines. Often, these are salient points in the image. We assume these structures to be detected as landmarks to guide the line tracing algorithm. For a general method one may rely on the detection of saddlepoints, T-junctions, and crossings to obtain vertices (Lindeberg, 1994; ter Haar Romeny, 1994).

1.2. Line Point Detection

Theoretically, in two-dimensions, line points are detected by considering the second order directional derivative in the gradient direction (Steger, 1998). For a line point, the first order directional derivative perpendicular to the line vanishes, where the second order directional derivative exhibits an extremum. Hence, the second order directional derivative perpendicular to the line is a measure of line contrast. The second order directional derivatives are calculated by considering the eigenvalues of the Hessian,

$$H = \begin{pmatrix} f_{xx} & f_{xy} \\ f_{xy} & f_{yy} \end{pmatrix} \tag{1}$$

given by

$$\lambda_{\pm} = f_{xx} + f_{yy} \pm \sqrt{(f_{xx} - f_{yy})^2 + 4f_{xy}^2} \tag{2}$$

where $f(x, y)$ is the grey-value function and indices denote differentiation. After ordering of the eigenvalues by magnitude, $|\lambda_+| > |\lambda_-|$, λ_+ yields the second order directional derivative perpendicular to the line. Bright lines are observed when $\lambda_+ < 0$ and dark lines when $\lambda_+ > 0$ (Lorenz et al., 1998). For both types of lines, the magnitude $|\lambda_+|$ indicates line contrast. Note that this formulation is free of parameters.

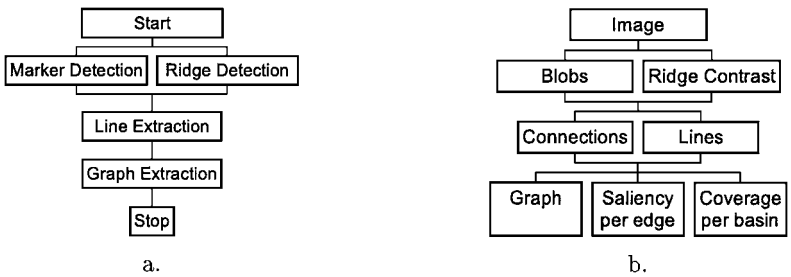


Figure 1. Flow diagram for network extraction. a. Action flow diagram, b. the corresponding data flow. Graph extraction results in the network graph, line saliency indicating confidence for the extracted lines, and basin coverage indicating missed lines.

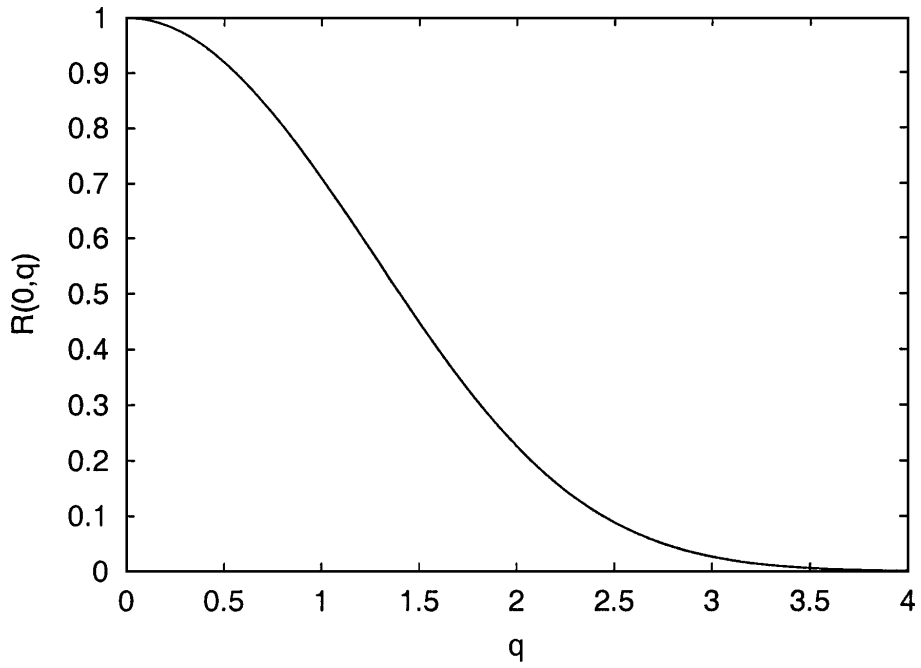


Figure 2. Response of $R(.)$ Eq. (3) at the centerline as function of relative line width $q = w/\sigma$.

In practice, one can only measure differential expressions at a certain observation scale (Florack et al., 1992; Koenderink, 1984). By considering Gaussian weighted differential quotients, ($f_x^\sigma = G(\sigma)_x * f$, a measure of line contrast is given by

$$R(x, y, \sigma) = \sigma^2 \left|\lambda_+^\sigma\right| \frac{1}{b^\sigma} \tag{3}$$

where σ, the Gaussian standard deviation, denotes the scale for observing the eigenvalues, and where line brightness b is given by

$$b^\sigma = \begin{cases} f^\sigma & \text{if } \lambda_+^\sigma \leq 0, \\ W - f^\sigma & \text{otherwise} \end{cases} \tag{4}$$

Line brightness is measured relative to black for bright lines, and relative to white level W (255 for an 8-bit camera) for dark lines. The original expression Eq. (2) is of dimension [intensity/pixel^2]. Multiplication by σ^2, which is of dimension [pixel^2], normalizes line contrast Eq. (3) for the differential scale. Normalization by line brightness b results in a dimensionless quantity. As a consequence, the value of $R(.)$ is within $[0 \ldots 1]$.

The response of the second order directional derivate $|\lambda_+|$ does not only depend on the image data, but it is also affected by the Gaussian smoothing scale σ. By analysis of the response to a given line profile as function of scale, one can determine the optimal scale for line detection. For a bar-shaped line profile of width w, the response of $R(.)$ Eq. (3) as function of the quotient $q = w/\sigma$ is plotted in Fig. 2. The response of $R(.)$ is biased towards thin lines, and gradually degrades for larger w. For a thin line $w \to 0$ the response equals line contrast, whereas for a large value of w relative to σ the response vanishes. Hence, the value of σ should be large enough to capture the line width. For optimal detection of lines, the value of σ should at least equal the width of the thickest line in the image,

$$\sigma \geq w. \tag{5}$$

When line thickness varies, one can set the value of σ to the size of the thickest line $\hat{w}$ to expect,

$$\sigma = \hat{w}. \tag{6}$$

In this case, response is slightly biased to thin lines.

The differential equation Eq. (3) is a point measure, indicating if a given pixel belongs to a line structure or not. The result is not the line structure itself, but a

set of points accumulating evidence for a line. In the sequel we will discuss how to integrate line evidence to extract line structures.

1.3. Line Tracing

Consider a line and its two endpoints S_1 and S_2. For all possible paths Ξ between S_1 and S_2, the path which integrates most line evidence is considered the best connection between the vertices. Therefore, we reformulate the line tracing to a minimum cost optimization problem. First, let $r(x, y, \sigma)$ be a cost function depending on $R(.)$ Eq. (3),

$$r(x, y, \sigma) = \frac{\epsilon}{\epsilon + R(x, y, \sigma)} \tag{7}$$

and let us define the path integral, taking σ for granted, to be

$$c(S_1, S_2) = \min_{\Xi} \int_{S_1}^{S_2} r(x(p), y(p))\, dp. \tag{8}$$

Here, $(x(p), y(p))$ is the Cartesian coordinate of the path parameterized by the arclength p. The path integral $c(S_1, S_2)$ now yields the integrated cost Eq. (7) over the best defined path in terms of line contrast $R(.)$. For high line contrast, the line is well-defined, the cost $r(.)$ should be determined by $1/R(.) \approx 0$. For a low value of $R(.)$, the cost should approximate 1, such that the Euclidian shortest path is traced. Hence, the constant term ϵ in Eq. (7) determines the trade-off between either following the maximum line contrast or taking the shortest route. The value of ϵ is typically very small, e.g. $\epsilon = 0.001$, compared to the line contrast, henceforth assures that plateaus are crossed. Note that line extraction does not introduce additional parameters.

1.4. Graph Extraction

Now consider an image containing vertices $S = \{S_1, S_2, \ldots, S_n\}$. For our case, lines are connecting neighboring vertices. The aim is to extract the network graph $G = (S, E)$ with vertices S, and edges E, the interconnecting lines given by the minimum cost paths. As there will be no crossing paths (see Section 1.1), the graph G may be found by a local solution. Hence, we concentrate on connecting neighboring vertices.

Neighbors are defined by assigning a zone of influence to each vertex, where each region $Z(S_i)$ defines the area for which all points p are closer to S_i than to any other vertex (Vincent, 1989),

$$Z(S_i) = \{p \in \mathbb{R}^2, \forall A \in S \backslash \{S_i\}, c(p, S_i) < c(p, A)\}. \tag{9}$$

The partitioning is known as the weighted Voronoï graph. Here, distance is measured with respect to cost $c(p, S_i)$ Eq. (8). The regions of influence correspond to the catchment basins of the topographical watershed algorithm (Meyer, 1994). Neighboring vertices to S_i are defined by the set of all vertices for which the zone of influence touches that of S_i. Hence, neighboring vertices share an edge in the topographical watershed segmentation. The minimum cost path Ψ_{ij} between S_i and S_j runs over the edge shared by S_i and S_j.

Note that we consider two types of graphs here. Viewing the image as a mountain landscape, the valleys being marker points, the borders of the weighted Voronoï graph run over the crest points of the mountains. The second graph is given by the steepest descent from saddle points in the mountain crest to the valleys, the graph being the solution G to the shortest path problem. The graph is dual to the weighted Voronoï partitioning.

The graph G is computed by applying the topographical watershed transform. First, the grey-weighted distance transform is applied on the cost image given by r Eq. (7), with the vertices S as mask. The grey-weighted distance transform propagates the costs from the masks over their influence area, resulting in a wavefront collision at places where two zones of influence meet. The collision events result in the edges between neighboring vertices, yielding the watershed by topographic distance. The minimum cost path between two neighboring vertices runs over the minimum in their common edge. Therefore, any edge between two neighboring vertices is traced for its minimum. Steepest descent on each side of the saddlepoint results in the minimum cost path between the vertices. Tracing the steepest descents for all different borders between the zones of influence results in the network graph G.

The described algorithm requires one distance transform to find the zones of influence. Hence, the order of the algorithm is determined by the grey-weighted distance transform. An efficient grey-distance transform algorithm based on Fast Marching is given by Sethian (1996). Note that the graph algorithm is free of parameters.

1.5. Edge Saliency and Basin Coverage

A natural measure of edge saliency is the integrated line contrast Eq. (3) over the edge,

$$s(S_1, S_2) = \int_{S_1}^{S_2} R(x(p), y(p))\, dp \qquad (10)$$

where S_1, S_2 are start and end node, and where $(x(p), y(p))$ is the path parameterized by arclength p. Note that $s(.)$, as R, is a parameter free quantity. A confidence measure indicating how well the edge is supported by the image data is given by the average saliency over the line,

$$\bar{s}(S_1, S_2) = \frac{1}{l} s(S_1, S_2) \qquad (11)$$

where l is the path length. Again, $\bar{s}$ is a dimensionless quantity, with range $[0 \ldots 1]$, a high value indicating a well-defined line.

Each basin in the minimum cost graph is surrounded by a number of connected paths forming the basin perimeter. An indication of segmentation confidence for a basin B may be obtained by considering the average saliency over the surrounding lines, compared to the average line contrast inside the graph basins. The average saliency over the basin boundary is given by

$$\bar{s}_B(B) = \frac{1}{l} \oint_B R(x(p), y(p))\, dp \qquad (12)$$

l being the basin perimeter, and p representing the arclength. A high value, in the range $[0 \ldots 1]$, indicates a well-outlined basin.

The average line contrast within basin B is measured by

$$\bar{c}_B(B) = \frac{1}{A(B^{\ominus})} \iint_{B^{\ominus}} R(x(p), y(p))\, dx\, dy. \qquad (13)$$

$B^{\ominus}$ is the basin eroded by a band of thickness given by σ. Erosion is applied to prevent the detected line points, smoothed by the Gaussian at scale σ, to influence the basin contrast. In Eq. (13), $A(.)$ is the area of the eroded basin. The value of $\bar{c}_B$ increases when line structures are present inside the basin, maybe due to a missed line in the graph. Coverage of the graph G is defined by the ratio of the line contrast remaining inside the basins relative to the line contrast covered by the graph edges,

$$\bar{c}(B) = 1 - \frac{\bar{c}_B(B)}{\bar{s}_B(B)}. \qquad (14)$$

When all line points are covering the basin boundary, $\bar{c}$ will be close to one. For a basin containing a missed line, the average line contrast over the basin will be high. When a spurious edge outlines the basin, summed contrast over the edges will be low, yielding a lower coverage value.

1.6. Thresholding the Saliency Hierarchy

The graph G is constructed such that neighboring vertices are connected, regardless the absence of interconnecting lines. For a spurious connection saliency will be low, since evidence of a connecting line is lacking. Pruning of the graph for spurious lines may be achieved by thresholding on saliency. Pruning by saliency of G imposes a hierarchy on the graph, ranging from graph G with all edges included, up to the graph consisting of the one best defined edge in terms of contrast. The threshold parameter indicates the saliency level of the hierarchy. Note the introduction of a parameter, indicating the application dependent hierarchy level of the graph. We propose two methods to prune edges by saliency.

First, global pruning may proceed by removing all ill-defined lines for which

$$\bar{s}(S_1, S_2) < \alpha. \qquad (15)$$

The resulting graph consists of the most contrasting lines, removing lines for which contrast is below the threshold. The method is applicable when a clear distinction between lines and background is present.

For the case of textured background, a local pruning method based on local comparison of edge saliency may be applied. Pruning of low confidence edges is installed by removing all edges for which an alternative path can be found in graph G, via other vertices, with higher confidence. Path confidence between S_1 and S_n via vertices S_i is defined by the average saliency over the n edges,

$$\bar{s}(S_1, S_2, \ldots, S_n) = \frac{1}{l} \left(\sum_{i=1}^{n-1} s(S_i, S_{i+1}) \right). \qquad (16)$$

Here, l is the total path length. The direct path between S_1 and S_n is pruned when

$$\min \bar{s}(S_1, \ldots, S_n) < \alpha \bar{s}(S_1, S_n) \qquad (17)$$

where the minimum is taken over all alternative paths in graph G between S_1 and S_n. Locally ill-defined lines are removed from the graph, the degree of removal given by α. For $\alpha = 0$, no lines are removed, whereas for $\alpha = 1$ all lines are removed for which a detour via another vertex yields higher saliency. Hence, short ill-defined paths are pruned when longer well-defined paths exist between the vertices. The method is applicable for a textured background, and when enough connections are present to determine alternative routes between neighboring vertices. The minimum of Eq. (17) may be computed by using a weighted graph distance transform (Vincent, 1989), thereby disregarding the direct connection between S_1 and S_n.

1.7. Overview

The algorithm is illustrated in Fig. 3. The figure shows the extraction of cell borders from heart tissue (Fig. 3(a), extracted vertices are indicated). Line contrast is calculated according to Eq. (3), shown in Fig. 3(b). The tracing of minimum cost paths is shown in Fig. 3(c). Most of the lines are correctly detected, together with some spurious lines. Local pruning the graph results in Fig. 3(d). Here, all edges which are not supported by the image data are removed. Figure 3(f) shows the area coverage, where black indicates $\bar{c} = 1$, and white indicates $\bar{c} = 0$.

In summary, we have proposed a one-parameter algorithm for the extraction of line networks. The parameter indicates the saliency level in a hierarchical graph. The graph tessellates the image into regions, where each edge travels over the minimum cost path between vertices. The resulting graph is labeled by edge saliency and area coverage, both derived from line contrast.

1.8. Error Analysis

The robustness of the proposed algorithm can best be evaluated when considering the different types of errors that may occur in forming the network graph. Table 1 gives an overview of possible errors and their consequences on the network graph G. The columns give a complete representation of the consequences an error may have on the network graph G. The rows overview the errors which may result from the vertex and line detection. In the sequel we discuss the sensitivity of the proposed algorithm to these types of errors.

When the image contains textured regions, the texture may cause a high response for the line point detection. Hence, the algorithm will falsely respond to the texture as being an underbroken line and find an optimal path, illustrated in Fig. 4(a). Further, when

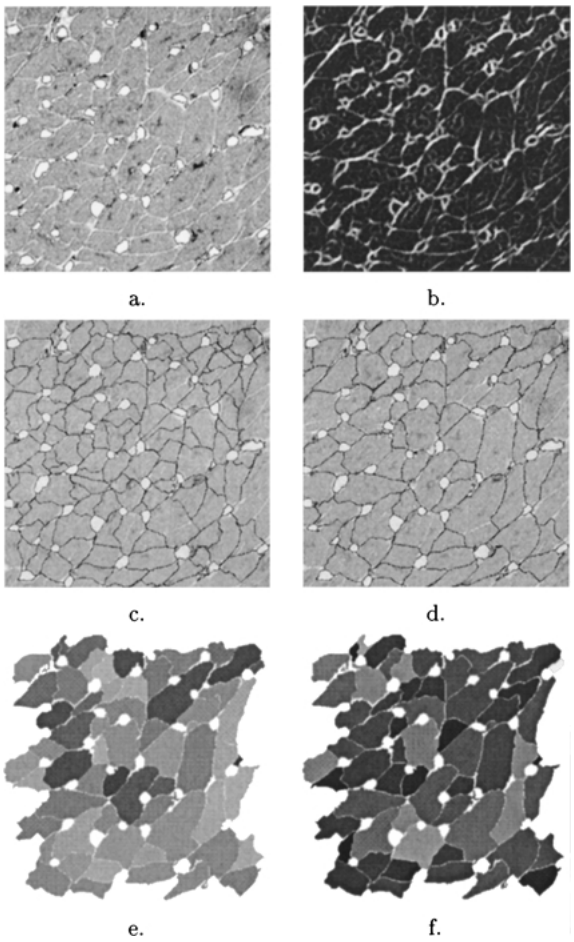


Figure 3. Example of line detection on heart tissue (**a**), observed by transmission light microscopy. The dark contours show the segmented blood vessels, superimposed on the original image. Line contrast $R(.)$ is shown in (**b**), the minimum cost graph in (**c**). The final segmentation (**d**) after local pruning of spurious edges for $\alpha = 0.9$ Eq. (17). The estimated saliency (**e**) Eq. (11) and area coverage (**f**) Eq. (14), dark representing high confidence in the result.

Table 1. Types of errors, in general for extraction of networks of lines and their consequences. Columns denote events in graph construction, whereas rows represent detection errors. Wanted sensitivity of the proposed algorithm is indicated by "+", whereas unwanted sensitivity to errors is indicated by "−". Robustness of the proposed method to errors is indicated by □.

Error type	Vertex insert	Vertex delete	Edge insert	Edge delete	Edge deviation	Saliency Eq. (10)	Coverage Eq. (14)
Spurious line	□	□	□	□	−	□	−
Gap in line	□	□	□	□	□	+	□
Line off vertex	□	□	□	□	□	+	□
Missed line	□	□	□	−	□	□	+
Spurious vertex							
off line	−	□	−	□	□	+	□
at line	−	□	□	□	□	□	□
Missed vertex							
at line	□	−	□	□	□	□	□
at fork	□	−	□	−	□	□	+
at line end	□	−	□	−	−	□	+

spurious line structures are present in the image data, without being part of the network, distortions may occur when the line is near other interconnections. In that case, the best path between vertices may be via the spurious line. An example is shown in Fig. 4(b), where text interferes with dashed line structures. For missed lines, basin coverage degrades. As the line structure is not part of the network, such sensitivity is unwanted.

Gaps in lines, or lines slightly off the vertex, illustrated by Fig. 4(c), will have no consequences except that saliency degrades.

When a line structure is of too low contrast to contribute enough to form a line, the line maybe pruned after confidence thresholding. An example of a missed line is shown in Fig. 4(d). As a consequence, coverage degrades, thereby indicating the event of a missed line.

For the case of a falsely detected vertex off a line (no example available), the vertex will be connected to the network. Saliency of the spurious lines will be low as line evidence is missing from the image. Hence, pruning of the network by saliency is likely to solve such errors.

Spurious or missed vertices at lines has, except for the insertion or deletion of a vertex, respectively, no consequence for the extracted network. An examples of spurious vertices is given in Fig. 4(e). The measure of saliency is invariant for insertion and deletion of vertices. This is proven by considering the path integral Eq. (10). Insertion of a vertex S_x at the path S_1, S_2 results in

$$\begin{aligned} s(S_1, S_x) + s(S_x, S_2) &= \int_{S_1}^{S_x} R(x(p), y(p))\, dp \\ &\quad + \int_{S_x}^{S_2} R(x(p), y(p))\, dp \\ &= \int_{S_1}^{S_2} R(x(p), y(p))\, dp \\ &= s(S_1, S_2) \end{aligned}$$

which is of course similar to the original saliency. Invariance for vertex deletion follows from the reverse argumentation. For the average line contrast within the graph basins is not affected by insertion or deletion of vertices at edges, coverage Eq. (14) is invariant for vertex insertion or deletion at lines.

More critical is overlooking a vertex at a fork or line-end. An example of a missing vertex is shown in Fig. 4(f). In both cases, an edge is missed in the resulting graph, and coverage degrades as not all line points are covered by the graph edges. For the missing of a vertex at a line end, the line is maybe connected to a different vertex, causing the crossing of the background by the minimum cost path. Pruning of the network by saliency is likely to solve the error.

Except for errors general for the extraction of networks of lines, the proposed algorithm generates errors specific for minimum cost path based methods. In general, only one path between two vertices is of

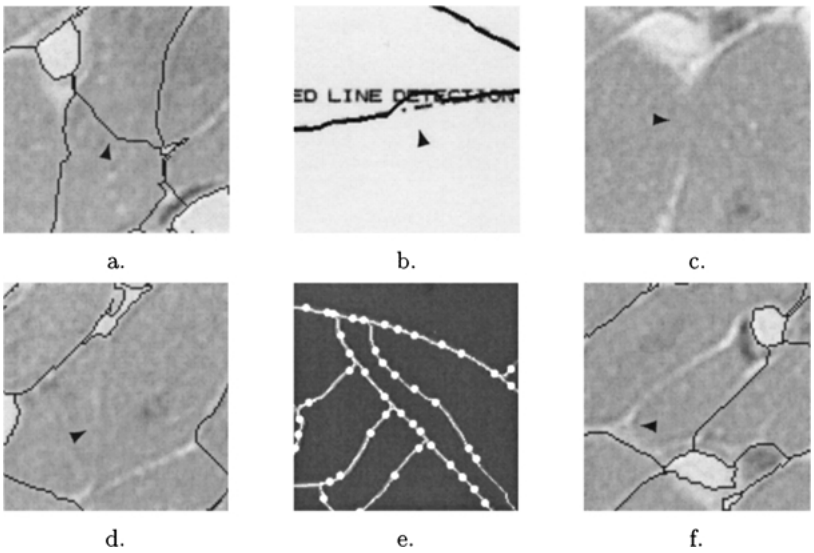


Figure 4. Example of failures in the line detection. **a.** The detection of a spurious line due to a textured region. **b.** The deviation of a line due to spurious line structures, the text, in the image. **c.** A gap in a line; the line is without errors correctly detected by the algorithm (result not shown). **d.** A missing connections due to lack of line evidence. **e.** Extra vertices on the line does not influence the algorithm performance. **f.** Missing of a vertex at a fork, resulting in a missed line in the network graph.

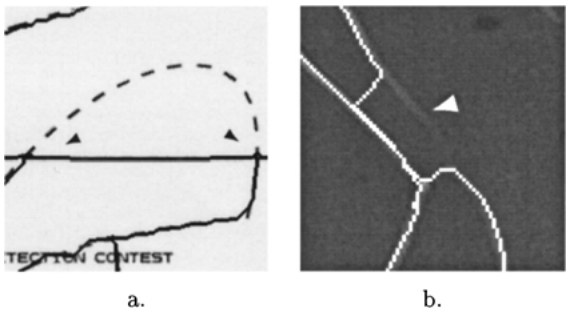


Figure 5. Example of failures specific for algorithms based on minimum cost paths. **a.** The missing of a line due to the double linking of vertices, for which the best connection is preserved. **b.** A shortcut along a better defined lines to optimally connect two vertices.

minimum cost. Any other path connecting the same vertices will be removed, as illustrated in Fig. 5(a). As a consequence, an edge is missed in the graph, and basin coverage degrades indicating the event of a missed line.

Further, when a better defined path exists in the neighborhood of the traced path, the algorithm tends to take a shortcut via the better defined path, as shown in Fig. 5(b). In that case, coverage degrades to indicate the missed line, whereas saliency increases due to the better defined route.

In conclusion, the proposed method is robust against: **a.** gaps in lines, **b.** lines slightly off their vertex, **c.** spurious lines, and **d.** spurious vertices at lines. The algorithm is sensitive to: **a.** missed lines, **b.** spurious vertices off lines, and **c.** missed vertices at forks. For missed vertices, the resulting graph is degraded. For missed lines, the graph may be degraded, and confidence of the area in which the missed line is situated may be too high. Specific for the algorithm is the sensitivity to shortcuts, and the inability to trace more than one line between connections.

2. Illustrations

2.1. Heart Tissue Segmentation

Figure 3 illustrates the application of the proposed algorithm on the extraction of cells from heart tissue. The tissue consists of cardiac muscle cells, the dark textured areas, and bloodvessels, the white discs. Cell borders are transparent lines surrounding all cardiac muscle cells. Due to the dense packing of cells, bloodvessels are squeezed between the cells. The cell borders appear as bright lines connecting the bloodvessels. Further, the dense packing causes gaps in the lines at places were light microscopic resolving power is too low to examine the cell border.

In the cardiac muscle cell application, the bloodvessels are considered as initial vertices. The vessels

are detected by dome extraction (Beucher and Meyer, 1993) (Fig. 3(a)). The extracted network graph, together with basin saliency and coverage, is shown in Fig. 3(d), (e) and (f). The heart tissue segmentation is a successful application in that a large number of cells is correctly segmented by the proposed algorithm. Individual cell parameters may be estimated after selecting those cells with high saliency and coverage. The amount of cells extracted from the tissue is in the same range as for qualitative studies based on interactively outlining the cells by experts (Ausma et al., 1997; Beltrami et al., 1994; Vliegen et al., 1987). Hence, the algorithm enables the quantitative assessment of morphological changes in heart tissue at a cellular level.

2.2. *Neurite Tracing*

A second example Fig. 6 shows interactive segmentation of neurites. The neurite starting points at the cell bodies are interactively indicated, and used as initial markers for the network segmentation algorithm. The resulting network is shown in Fig. 6(b). In this case, pruning of lines is not possible since no alternative routes between the markers are present. Paths between cells which are not connected are removed by thresholding the saliency Eq. (15). Note that no errors are caused from lack of line structure, indicated in Fig. 6(a). The overall saliency of the result is $\bar{s} = 0.44$, indicating that the line contrast spans almost half the dynamic range of the camera. Coverage $\bar{c} = 0.95$, indicating that 95% of the line structures present in the image is covered by the network graph. Hence, the result is considered highly accurate.

2.3. *Crack Detection*

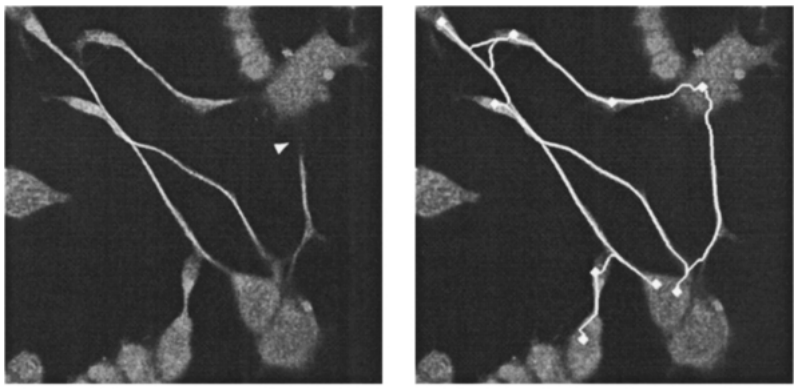
An example of general line detection is shown in Fig. 7, where cracks in ink at high magnification are traced. The image shows an ink line, at such a magnification that the ink is completely covering the image. Cracks in the ink form white lines, due to the transmission of light, against a background of black ink. Note that no natural markers are present.

For the general case of line detection, saddlepoint detection may be used to extract markers. The saddlepoints on bright lines are detected by

$$f_x^\sigma = 0, \quad f_y^\sigma = 0, \quad \lambda_+^\sigma < 0, \quad f_{xx}^\sigma f_{yy}^\sigma - f_{xy}^{\sigma^2} < -\alpha. \quad (18)$$

Here, α indicates salient saddlepoints, and is typically small to suppress spurious saddlepoints due to noise. The saddlepoints are used as markers for the network extraction algorithm.

The detected saddlepoint are highlighted in Fig. 7(b). The result of the proposed algorithm, the saddlepoints as vertices, is shown in Fig. 7(c). Average saliency is thresholded Eq. (15) to remove paths which cross the background. Overall saliency of the graph is 0.313, and coverage 0.962. The cracks are successfully extracted by the proposed algorithm, except that the crack ends are missing when end markers are absent. In that case, the detected cracks are too short.



a. b.

Figure 6. Extraction of a neurite network (**a**); note the gaps present in the neurites. The traced network is shown in (**b**). The dots represent the interactively indicated neurite start points at the cell bodies.

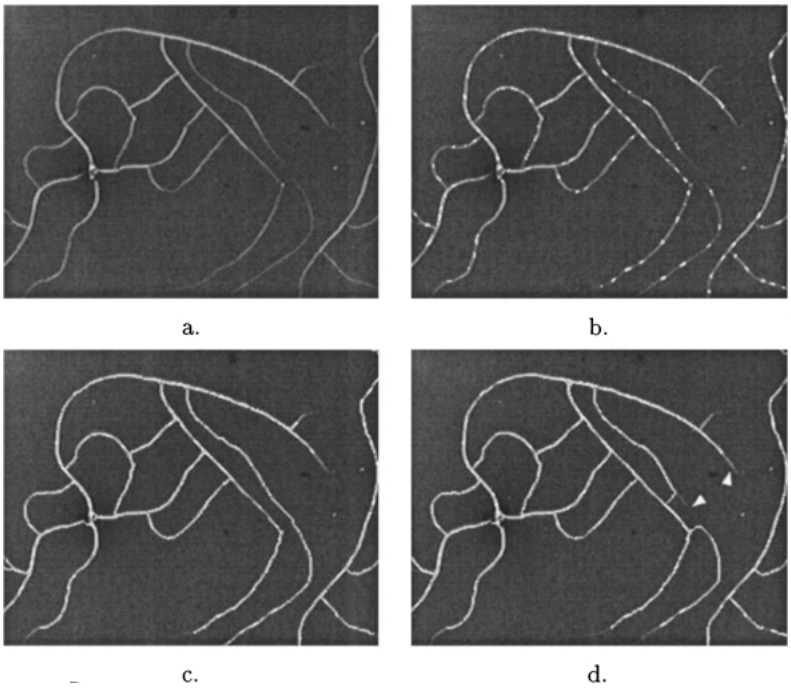


Figure 7. Extraction of a general line network. **a.** shows a high magnification image of ink, completely covering the image, distorted by white cracks through which light is transmitted. No natural markers are present. **b.** The saddlepoints at the bright lines Eq. (18). **c.** The detected lines, overall saliency $\bar{s} = 0.313$, coverage $\bar{c} = 0.962$. **d.** The result for half the number of markers, overall saliency $\bar{s} = 0.316$, coverage $\bar{c} = 0.960$. Note the shortcut and the removal of line ends.

Since no natural markers are present, the algorithm should be robust against marker insertion or deletion at lines. Figure 7(d) shows the result after random removal of half the markers in Fig. 7(c). Errors in the result include a shortcut via a more contrasting line. Further, line ends are pruned due to the absence of markers. Note that saliency is only marginally affected by the new situation, 0.316 instead of 0.313, whereas coverage likewise is reduced marginally, from 0.962 to 0.960. Hence, the algorithm is robust for variations in the threshold value α for saddle point detection Eq. (18).

2.4. *Directional Line Detection*

Characteristic for the proposed algorithm is that line evidence is accumulated over the line. When line evidence is absent, the algorithm optimizes the shortest path to the neighboring line parts to continue integration. As a result, when large gaps are present, the algorithm may find an alternative route by crossing the background to a neighboring line, tracking that line, and jumping back to the original line after the gaps. The problem may be solved by including line orientation information into the algorithm.

To proceed, we consider directional filtering for detection of line contrast. Consider Eq. (3), which was measured by isotropic Gaussian filters of scale σ. For the directional filtering, we consider anisotropic Gaussian filters of scale σ_l and σ_s, for longest and shortest axis, respectively, and of orientation θ. Hence, line contrast is given by

$$R'(x, y, \sigma_l, \sigma_s, \theta) = \sigma_s^2 \left| f_{ss}^{\sigma_l, \sigma_s, \theta} \right| \frac{1}{b^{\sigma_l, \sigma_s}}, \qquad (19)$$

where $f_{ss}^{\sigma_l, \sigma_s, \theta}$ is the Gaussian weighted derivative in the shortest axis direction, and b^{σ_l, σ_s} is given by Eq. (4). The scale σ_s depends on line width as given by Eq. (6), whereas σ_l is tuned to adequately capture line direction. Hence, σ_l should be large enough to bridge small gaps, but should be not too large to prevent errors when line curvature is high. In practice, an aspect ratio of $\sigma_s = \hat{w}$ and $\sigma_l = 3\sigma_s$ is often sufficient.

Now we have established how to filter in a particular direction, the filter need to be tuned to the

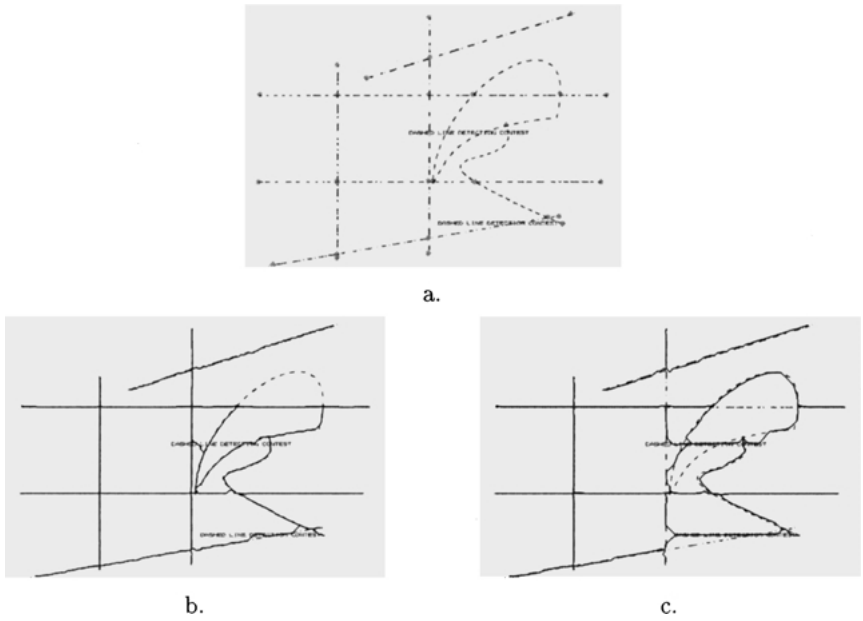


Figure 8. Extraction of a dashed line network (**a**), taken from (Kong et al., 1996); markers are interactively selected at line crossings and line end points, indicated by grey dots. The extracted network is shown in (**b**) Errors made include some shortcuts, and the missing of the bend line part due to the presence of a second, shorter connection between the markers. Note the difference for the isotropic result for scale σ_l (**c**) The scale σ_l was taken to integrate line evidence over the gaps in the dashed lines. Deviation from centerline in the isotropic case is the result of the large integration scale compared to line width.

line direction. Therefore, two options are considered. First, eigenvector analysis of the Hessian results in the principal line direction. One could apply a first undirectional pass to obtain line direction, as described by Steger (1998). A second pass yields the tuning of the filter at each pixel in the line direction to obtain line contrast. Note that the filter orientation may be different for each position in the image plane. Instead of tuning the filter, sampling the image at different orientations may be applied. One applies Eq. (19) for different orientations. When the filter is correctly aligned with the line, filter response is maximal, whereas the filter being perpendicular to the line results in low response. Hence, the per pixel maximum line contrast over the orientations yields directional filtering.

The proposed method is applied to an example of a dashed line pattern as given in Fig. 8(a), taken from (Kong et al., 1996). The example is taken from the hardest class, the "complex" patterns. The grey dots represent interactively selected markers, indicating crossings and line end points. Orientation filtering is applied at 0°, 30°, 60°, 90°, 120°, and 150°, for which the maximum line contrast per pixel is taken over the sampled orientations. The result after graph extraction and saliency thresholding is shown in Fig. 8(b). The crude sampling of orientation space causes some of the lines to be noisy. A better sampling enhances the result. Further, one line part is missed, due to a shorter line connecting the same markers. The text present in the example causes the algorithm to follow parts of the text instead of the original line. Not that isotropic line detection does not adequately extract the graph (Fig. 8(c)).

3. Conclusion

The extraction and interpretation of networks of lines from images yields important organizational information of the network under consideration. We present a one-parameter algorithm for the extraction of line networks from images. The parameter indicates the extracted saliency level from a hierarchical graph. Input for the algorithm is the domain specific knowledge of interconnection points. The algorithm results in the network graph, together with edge saliency, and catchment basin coverage.

The proposed method assigns a robust measure of saliency to each minimum cost path, based on the average path cost. Edges with a low saliency compared to alternative routes are removed from the graph, leading to an improved segmentation result. The correctness of the network extraction is indicated by the edge saliency and area coverage. Hence, confidence in the final result can be based on the overall network saliency.

Design issues are robustness against general errors summarized in Table 1. The proposed method is robust against: **a.** gaps in lines, **b.** lines slightly off their vertex, **c.** spurious lines, and **d.** spurious vertices at lines. The algorithm is sensitive to: **a.** missed lines, **b.** spurious vertices off lines, and **c.** missed vertices at forks. Thresholding on saliency reduces the errors caused by spurious vertices. Missed lines are signaled by a measure of coverage Eq. (14), indicating how much of the line evidence is covered by the network graph. Specific for the algorithm is the sensitivity to shortcuts, and the inability to trace more than one line between connections. Any algorithm based on minimum cost paths is sensitive to these types of errors.

We restricted ourselves to locally defined line networks, where lines are connecting neighboring vertices. For globally defined networks, like electronic circuits, the algorithm can be adapted to yield a regional or global solution. Therefore, several distance transforms have to be applied, at the cost of a higher computational complexity. The pruning of the network, and the measure of saliency is again applicable for the global case.

Incorporation of line directional information into the algorithm results in better estimation of line contrast, hence improves graph extraction. The eigenvector analysis of the directional derivatives yields an estimate of the local direction of the line. The directional information may be included by considering an anisotropic metric for the line contrast filtering. Experiments showed a better detection of the network graph for dashed line detection. The example given is considered as a complex configuration, according to (Kong et al., 1996). Disadvantage is a longer computation time, due to the anisotropic filtering pass.

The proposed method results in the extraction of networks from connection to connection point. The routing from a starting connection to its final destination depends on the functionality of the network, and is not considered in this paper. Correct interpretation of the network in the presence of distortion obviously requires information on the function of the network.

For the extraction of line networks the proposed method has proven to be a useful tool. The method is robust against gaps in lines, and against spurious vertices at lines, which we consider as the most prominent source of error in line detection. Hence, the proposed method enables reliable extraction of line networks. Furthermore, the method indicates detection confidence, thereby supporting error proof interpretation of the network functionality. The proposed method is applicable on a broad variety of line networks, including dashed lines, as demonstrated by the illustrations. Hence, the proposed method yields a major step towards general line tracking algorithms.

Acknowledgments

The authors are grateful to Geert Streekstra and Rein van den Boomgaard for valuable suggestions. The example figures Fig. 3(a) and Fig. 6(a) are by courtesy of Gerrit Dispersyn and Rony Nydens.

References

Ausma, J., Wijfels, M., Thoné, F., Wouters, L., Allessie, M., and Borgers, M. 1997. Structural changes of atrial myocardium due to sustained atrial fibrillation in the goat. *Circulation*, 96:3157–3163.

Barzohar, M. and Cooper, D. 1993. Automatic finding of main roads in aerial images by using geometric-stochastic models and estimation. In *IEEE Conference on Computer Vision and Pattern Recognition*, New York, NY, June 15–17, 1993. Computer Society Press: Washington, DC, pp. 459–464.

Bellman, R.E. 1957. *Dynamic Programming*. Princeton University Press: Princeton, NJ.

Beltrami, C.A., Finato, N., Rocco, M., Feruglio, G.A., Puricelli, C., Cigola, E., Quaini, F., Sonnenblick, E.H., Olivetti, G., and Anversa, P. 1994. Structural basis of end-stage failure in ischemic cardiomyopathy in humans. *Circulation*, 89:151–163.

Beucher, S. and Meyer, F. 1993. The morphological approach to segmentation: The watershed transformation. In *Mathematical Morphology in Image Processing*, E.R. Dougherty (Ed.). Marcel Dekker: New York, Ch. 12, pp. 433–481.

Buckley, M. and Talbot, H. 2000. Flexible linear openings and closings. In *Mathematical Morphology and its Applications to Image and Signal Processing*, J. Goutsias, L. Vincent, and D.S. Bloomberg (Eds.). Kluwer Academic Publishers: Dordrecht, The Netherlands.

Cohen, L.D. and Kimmel, R. 1997. Global minimum for active contour models: A minimal path approach. *Int. J. Computer Vision*, 24:57–78.

Florack, L.M.J., Romeny, B.M.t.H., Koenderink, J.J., and Viergever, M.A. 1992. Scale and the differential structure of images. *Image and Vision Computing*, 10(6):376–388.

Illingworth, J. and Kittler, J. 1988. A Survey of the Hough transform. *Computer Vision, Graphics, Image Process.*, 44:87–116.

Koenderink, J.J. 1984. The Structure of images. *Biol. Cybern.*, 50:363–370.

Kong, B., Phillips, I.T., Haralick, R.M., Prasad, A., and Kasturi, R. 1996. A bench mark: Performance evaluation of dashed-line detection algorithms. In *Graphics Recognition Methods and Applications*, R. Kasturi and K. Tombre (Eds.). Springer-Verlag: Berlin, pp. 270–285.

Lindeberg, T. 1994. *Scale-Space Theory in Computer Vision*. Kluwer Academic Publishers: Boston.

Lorenz, C., Carlsen, I.C., Buzug, T.M., Fassnacht, C., and Weese, J. 1998. *Scale Space Theories in Computer Vision*. Springer-Verlag: Berlin, pp. 152–163.

Meyer, F. 1994. Topographic distance and watershed lines. *Signal Processing*, 38:113–125.

Sethian, J.A. 1996. A fast marching level set method for monotonically advancing fronts. *Proc. Natl. Acad. Sci. USA*, 93:1591–1595.

Sha'ashua, A. and Ullman, S. 1988. Structural saliency: The detection of globally salient structures using a locally connected network. In *Second International Conference on Computer Vision* Tampa, FL, December 5–8, 1988. Computer Society Press: Washington, DC, pp. 321–327.

Steger, C. 1998. An unbiased detector of curvillinear structures. *IEEE Trans. Pattern Anal. Machine Intell.*, 20:113–125.

ter Haar Romeny, B.M. (Ed.). 1994. *Geometry-Driven Diffusion in Computer Vision*. Academic Publishers: Boston.

Vincent, L. 1989. Graphs and mathematical morphology. *Signal Processing*, 16:365–388.

Vincent, L. 1998. Minimal path algorithms for the robust detection of linear features in gray images. In *Mathematical Morphology and its Applications to Image and Signal Processing*, H. Heijmans and J. Roerdink (Eds.). Kluwer Academic Publishers: Dordrecht, The Netherlands, pp. 331–338.

Vliegen, H.W., van der Laarse, A., Huysman, J.A.N., Wijnvoord, E.C., Mentar, M., Cornelisse, C.J., Eulderink, F. 1987. Morphometric quantification of myocyte dimensions validated in normal growing rat hearts and applied to hypertrophic human hearts. *Cardiovasc. Res.*, 21:352–357.