Detection of curvilinear structures and reconstruction of their regions in gray-scale images

Jeong-Hun Jang, Ki-Sang Hong*

Image Information Processing Laboratory, Department of Electrical Engineering, San 31, Hyojadong, Namku, Pohang, Kyungbuk, 790-784, South Korea

Received 5 May 2000; accepted 19 March 2001

Abstract

In this paper, we present a new method for detecting curvilinear structures and reconstructing their regions in gray-scale images. The concept of skeleton extraction is introduced to detect more general structures such as tapering structures. A candidate skeleton is extracted from the Euclidean distance map that is constructed based on the edge map of an input image. The extracted skeleton is usually noisy due to small protrusions and gaps existing on edge contours. Unnecessary skeletal points are effectively removed with a method combining previously proposed and our own methods. Then, each skeletal point is classified as one of three types (RIDGE, RAVINE, or STAIR), and connected points of the same type are grouped to form a skeletal segment. Finally, the reconstruction of curvilinear structure regions is performed based on the skeletal segment classification result. Experimental results show that our detector contains many of the desirable properties required of a curvilinear structure detector. Furthermore, since the range of widths that our detector can detect at one time is wide, it is very useful, for example, when an input image includes curvilinear structures of various widths or tapering structures whose width varies greatly. Our algorithm for reconstructing curvilinear structure regions enables us to decompose an image into several types of regions. The reconstruction result, together with the skeleton extraction result, is expected to be useful to make a simplified scene description of an image. © 2002 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

Keywords: Feature detection; Curvilinear structure detection; Skeleton extraction; Euclidean distance map; Reconstruction of curvilinear structure regions

1. Introduction

Feature detection in a gray-scale image is one of the major research interests in computer vision and pattern recognition communities, and has a long history. Good features describe information contained in an image compactly and effectively, and facilitate higher-level processes such as matching or recognition. Frequently used features are edges, lines, corners, etc.

In this paper, we deal with the problem of detecting another useful feature, called a curvilinear structure, in a gray-scale image. A curvilinear structure represents a line or a curve with some width, and it differs from conventional line or curve features, which are usually extracted based on edges. Curvilinear structures are more structured features and contain more information than edges. They can be found in most natural images, and their detection is particularly useful, for example, when trying to find roads or rivers in aerial images, blood vessels or bones in medical images, and characters in text images.
Desirable properties required of a curvilinear structure detector are listed below:

- It should be able to detect structures that are bent severely.
- It should be able to detect structures without user interaction.
- It should be robust to noises.
- Detected positions should be close to the center lines of structures.
- Selection of structure’s width should be allowed.
- The detectable range of structure’s width should be wide.

There are many publications addressing the problem of curvilinear structure detection. Most of the recently proposed methods are based on one of following three approaches:

1. Locally parallel edge based approach [1–3].
2. Ridge based differential geometric approach [4–9].
3. Active contour model based approach [10,11].

Modifications or new ideas have been added to these approaches to overcome their inherent limitations. For example, Koller et al. [2] proposed an algorithm based on Approach 1, where two (left and right) Gaussian derivative filters are applied perpendicular to a line, and the responses of both filters are combined in a nonlinear way to produce the final response. Steger [9] suggested a method based on Approach 2, where the width and the center line of curvilinear structures are estimated after their positions are roughly found by detection of ridges. Zlottnick and Carine [11] proposed a method for automatic seed point detection to overcome the major weakness of Approach 3: the necessity of user interaction.

Since each approach has its own strong and weak points, and no method includes all the desirable properties previously listed, it is the user’s responsibility to choose a method appropriate to the given situation. However, the above approaches have a fundamental limitation. Since they are designed to detect only elongated structures with small variations of width along their center lines, they are not adequate for the detection and description of more general structures such as tapering structures. Let us examine why such a limitation exists in conventional curvilinear detectors. Most curvilinear structure detectors take a width value of a structure to be detected as a parameter explicitly or implicitly. For example, in Approach 2, since the centers of a bar-shaped structure are flat, a Gaussian filter is applied to make them convex. The size of the filter kernel, $\sigma$, should be large enough to detect wide curvilinear structures, but then thin lines or curves are blurred out. Therefore, the detectable range of structure’s width is confined by the given value of $\sigma$. We could find structures of different width by applying a detector repeatedly with different scales, but it is very costly, and it is not easy to integrate the detection results.

To solve the above problem, we bring the concept of skeleton extraction, which is famous in the binary image domain, to the gray-scale image domain. With this approach, as will be shown later, a curvilinear structure detection problem becomes a sub-problem of selecting skeletal segments according to their properties. The proposed detector includes many of the desirable properties required of a curvilinear structure detector, and an extracted skeleton is adequate for describing more complex structures.

A skeleton is a well-defined concept in the binary image domain. The skeleton of a binary image object can be thought of as a collection of centers of inner disks that touch the boundary of the object at least twice. An inner disk satisfying the condition is called a maximal disk. In a gray-scale image, no clear boundary of an object or region is defined. Whether a pixel becomes a boundary point or not depends on a given situation. This means that some decision rule needs to be involved to determine boundaries of regions. It is obvious that at boundaries, some kind of discontinuity of an image surface will be observed, and in fact, it is a so-called edge.

We make use of detected edgels (edge pixels) as the alternative to boundary points, though they are imperfect. Once the boundary points are determined, we can utilize powerful tools that have been used for skeleton extraction in the binary image domain for many years. For various reasons, which will be explained later, we take an approach in which the Euclidean distance transform is performed on an edge map, and ridge points, which are regarded as candidate skeletal points, are extracted from the constructed Euclidean distance map. Many of the extracted ridge points are usually false, which are caused by noisy boundary shapes, incomplete edges, etc. In order to remove them effectively, we combine several previously proposed and our own methods. The remaining ridge points constitute a skeleton. Since the detected edges have no direct relationship with the width of the structures to be detected and a skeleton is extracted based on these edges, structures with various widths or even widths that vary greatly can be detected at once.

One of three types (RIDGE, RAVINE, or STAIR) is assigned to each skeletal point by observing the cross-sectional gray-level values of an image in the neighborhood of the point. Then, connected skeletal points of the same type are grouped to form a skeletal segment. Unstable skeletal segments, for example, very short ones embedded between two long segments, are reclassified depending on their neighboring segments’ types. This classification procedure, together with the Euclidean distance map, enables us to find curvilinear structures of selected width and type.
Optionally, the regions of the curvilinear structures that have been detected in the form of labeled skeletal segments can be reconstructed. Points on a skeletal segment, which constitute a center line of the corresponding region, act like seed points of the region. Starting from those points, the region grows up to its boundary, which is roughly defined by the distance values of the seed points on the distance map. To reconstruct the original shape of the region more accurately, new seed points are selected based on the previous result, and region growing is performed iteratively. With this technique, we can decompose an image into different kinds of structures.

This paper is organized as follows. In Section 2, the skeleton extraction procedure is explained step by step. In Section 3, a description of how to classify skeletal segments is given in detail. Reconstruction of curvilinear structure regions is considered in Section 4. Section 5 shows various experimental results. Conclusions are given in Section 6.

2. Skeleton extraction

In this section, skeleton extraction in a gray-scale image is considered. The skeleton extraction procedure is composed of several steps, and its overall flow is shown in Fig. 1. In following subsections, algorithms used in each step are described in detail.

2.1. Detecting edgels

The skeleton extraction procedure begins with the detection of edgels in an input gray-scale image. Detected edgels serve as boundary points of meaningful structures. In our work, the Canny edge detector is used because of its many desirable properties [12]. Three parameters are involved in the algorithm: the size of a Gaussian kernel, $\sigma$ in Gaussian smoothing, and two thresholds, $T_1$ and $T_2$ in hysteresis thresholding. After edgels are detected, isolated edgels are removed. An input gray-scale image is given in Fig. 2(a), which will be used as an example throughout this paper, and its edge image is shown in Fig. 2(b).

2.2. Constructing a Euclidean distance map

Once boundary points are determined by edge detection, we can utilize powerful tools that have been used for skeleton extraction in the binary image domain for many years. We adopted the "medial axis extraction from a distance map" approach among several existing skeletonization approaches, because it was considered to be most adequate for our situation. For example, it does not require closed boundary contours, and it provides the width of detected structures.

We slightly modified the region-growing Euclidean distance transform algorithm proposed by Cuisenaire [13]. It yields as a result, a Euclidean distance map where each pixel site has a value of distance to the nearest edgel, and the position of that edgel. The latter information is also needed since it plays an important role when removing unnecessary ridge points in Subsection 2.4, and classifying skeletal segments in Section 3. The distance map obtained by applying the algorithm to edgels of Fig. 2(b) is shown in Fig. 3(a).

2.3. Detecting ridge points

Ridge points (i.e., local maxima) of a distance map constitute candidate skeletal points, since theoretically, a ridge point corresponds to the center of a maximal disk. We implemented the algorithm proposed by Arcelli and Baja which is specially designed to detect ridges in a distance map [14]. One of advantages of this method is to guarantee connectedness of ridges. Extracted ridges are thinned to be one pixel wide. The resulting image of ridges detected from the distance map of Fig. 3(a) is shown in Fig. 3(b).

2.4. Removing unnecessary ridge points

The step of Subsection 2.3 results in a number of superfluous ridge points due to noisy boundaries. Therefore, it needs to remove ridge points that make little contribution to the description of meaningful structures.

Malandain and Fernández-Vidal introduced two parameters $\phi$ and $d$ for the local characterization of skeletal points [15]. After removing unwanted skeletal points by thresholding the parameters, they applied a method called topological reconstruction to get a reliable skeleton. This method is preferred because it is simple, easy to implement, and the Euclidean distance map and its related information obtained in Subsection 2.2 can be fully utilized.

The meaning of $\phi$ and $d$ can be understood easily from Fig. 4, where $p$ and $n$ represent a skeletal point (i.e., ridge point) and its neighbor, respectively, and $e_p$ and $e_n$ represent the boundary points (i.e., edgels) closest to $p$ and $n$, respectively. Parameters $\phi$ and $d$ of a skeletal point $p$ are given by

$$\phi(p) = \max_{n \in N_p} \frac{180}{\pi} \arccos \frac{\overrightarrow{pe_p} \cdot \overrightarrow{pe_n}}{|| \overrightarrow{pe_p} || \cdot || \overrightarrow{pe_n} ||}$$

and

$$d(p) = \max_{n \in N_p} || \overrightarrow{e_pe_n} ||,$$

where $N_p$ denotes a neighborhood of $p$. These parameters are closely related with a maximal disk centered on $p$. Points $e_p$ and $e_n$ in Fig. 4 are, in fact, only approximations of two touching points of the maximal disk. In this paper we use only one parameter $\phi$, whose property is shown well in Fig. 5, where $\phi_1$ and $\phi_2$ of two points $p_1$ and $p_2$
Fig. 1. Skeleton extraction procedure.

Fig. 2. (a) Input gray-scale image. (b) Edge image produced by the Canny edge detector. Parameters: $\sigma = 1.3$, $T_l = 1.0$, and $T_h = 8.0$.

lying on skeletal branches generated by small protrusions of boundaries (i.e., noises), have small values compared to $\phi_3$ of $p_3$ on the main skeleton.

Simple thresholding of the parameter $\phi$ is insufficient to remove unnecessary ridge points effectively, since it does not preserve the important topological structure of the original shape. To solve the problem, the two thresholds scheme is adopted where two thresholds, $\phi_h$ and $\phi_l$ ($\phi_h > \phi_l$), are used to produce two skeletons $S_h$ and $S_l$ ($S_h \subset S_l$), respectively. $S_h$ reflects the detail level of the skeleton, i.e., the important parts to be preserved, while $S_l$ gives the topology to be preserved. Based on $S_h$ and $S_l$, a method called *topological reconstruction* is performed to get a robust skeleton. Let $S_{l/h}$ be a set of points belonging to $S_l$ but not to $S_h$. The topological reconstruction of $S_h$ with respect to $S_l$ is to add to $S_h$ some
Fig. 3. (a) Distance map produced by the region-growing Euclidean distance transform. Distance values greater than 10.0 are thresholded to 10.0 for the purpose of display. (b) Ridges extracted from the distance map of (a).

Fig. 4. Definition of the two parameters $\phi$ and $d$.

Fig. 5. Property of the parameter $\phi$.

Fig. 6. Labeling of $S_{1/h}$.

3. Remove skeletal segments from $S_{1/h}$ if they have any end point labeled END. Then, go to Step 1. Repeat the loop until no segments are removed.

4. Merge $S_{1/h}$ with $S_h$ to yield a final skeleton.

In Fig. 7, the result is shown when two thresholds, $\phi_h = 120^\circ$ and $\phi_l = 60^\circ$, are applied to the ridge points of Fig. 3(b). Fig. 8 shows a skeleton produced by topological reconstruction based on the result of Fig. 7. Note that many of the noisy skeletal points are removed while most of the important connections between junctions are maintained.

The above method was originally designed for extracting skeletons from binary images, where boundary contours are always closed. When this method is applied to edge maps, the relationship between the thresholds ($\phi_l$ and $\phi_h$) of the method and existing edge gaps is rather unclear. To show the relationship explicitly, consider the case of locally parallel edge lines, where one of them contains a gap, as shown in Fig. 9. In this figure, the edge lines are $w_g$ pixels apart from each other and the width of the gap is given by $w_g$. The edge gap produces a ridge segment penetrating itself, which we want to remove. The ridge points have smaller values of $\phi$ as they become farther away from the gap. Let $\phi_m$...
represent the smallest value of $\phi$ that can be produced by the gap. According to the topological reconstruction method, the ridge points with $\phi \geq \phi_h$ always remain alive, while those points with $\phi < \phi_l$ are always removed. The ridge points with $\phi_l \leq \phi < \phi_h$ survive only when they are enclosed by the ones with $\phi \geq \phi_h$. Therefore, among the ridge points produced by the edge gap, only those points with $\phi \geq \phi_h$ are left after the topological reconstruction if $\phi_l$ is set to be greater than $\phi_m$. Let $r_m$ represent a radius of a maximal disk of the ridge point with the smallest value of $\phi$. The maximal disk should be tangent to the edge line that is on the opposite side of the gap as well as pass through two end points of the gap to give the smallest value of $\phi$. From Fig. 9, it is easy to show that $r_m$ is given by

$$r_m = \frac{w_x}{2} + \frac{w_y^2}{8w_x}$$

and $\phi_m$, the smallest value of $\phi$, is given by

$$\phi_m = 2 \tan^{-1}\left( \frac{4w_x w_y}{4w_y^2 - w_x^2} \right).$$

From Eq. (4), the ratio $w_x/w_y$ is given by

$$\frac{w_x}{w_y} = 1 + \sqrt{1 + \tan^2(\phi_m/2)}.$$  

It should be noted that since $w_x/w_y$ is a monotonically decreasing function of $\phi_m$, for the given threshold $\phi_l$, the condition

$$\frac{w_x}{w_y} > 1 + \sqrt{1 + \tan^2(\phi_l/2)}$$

must be satisfied to prevent the skeletal segment produced by the edge gap from being connected to a main skeleton. If, for example, $\phi_l$ is set to $60^\circ$, $w_x/w_y$ must be greater than 1.866. The length $l_k$ of a ridge segment remaining
after the topological reconstruction is simply given by

\[ l_h = \frac{w_g}{\tan(\phi_h/2)}. \]  

(7)

Since the gap width \( w_g \) is usually small, the length \( l_h \) is also small, and the remaining short ridge segment will be removed in Subsection 2.6.

2.5. Disconnecting skeletal points

The problem solved in Subsection 2.4 arises in both the cases of binary and edge images. However, there are some problems that can occur only in the case of edge images, and we are to solve them in this subsection.

It is easy to understand that points in a skeleton can be categorized into three kinds of points (end, link, and junction points) according to their role in a set of connected points, and it can be shown that they can be identified by observing their eight neighboring points. In our case, however, identifying the points without taking edgels into account causes problems depicted in Fig. 10. Two skeletal points \( p_1 \) and \( p_2 \) in Fig. 10(a) are connected to each other, considering an eight-connected neighborhood. However, they should be regarded as disconnected, since they are crossing an edge which serves as a boundary. Let \( S \) and \( E \) denote skeletal and edge point sets, respectively, and let \( P = S \cup E \). Points to be disconnected can be found in \( P \) using the two operators in Fig. 11, where \( p \) represents a skeletal point under consideration, and \( n \) and \( e \) designate neighboring skeletal and edge points, respectively. \( P \) is scanned sequentially, and if at least one operator in Fig. 11 applies successfully at a point \( p, p \) and \( n \) are marked in \( S \). After scanning of \( P \) has been completed, points in \( S \) are identified as end, link, and junction points. If a point \( p \) in \( S \), which is to be identified, is a marked point, other marked neighbors of \( p \) are removed temporarily, and \( p \) is identified. Then, the removed neighboring points are restored. Let \( Q \) denote the resultant set of identified skeletal points.

Fig. 9. Scene near the edge gap.

The problem depicted in Fig. 10(b) happens due to imperfect detection of edges. In the case of a Canny edge detector, detection of edgels sometimes fails particularly near junctions, which results in unwanted skeletal points like \( p_3 \) in Fig. 10(b). Such points cannot be removed by the method introduced in Step 4, since most of them have the value of \( \phi \) close to 180°. Therefore, they should be treated separately. Fig. 12 illustrates a scene near the incomplete junction of edges, where \( p \) is a skeletal point to be removed, and our aim is to find \( p' \). What to do first is to find a region containing \( p' \), which is represented by a square \( R \) in Fig. 12. The center of \( R \) is located on the end point of an edge \( (e_c \) in the figure), and its width \( w \) is given as twice the maximum of expected gap width between edges. A point \( p' \), which is a candidate for \( p' \), is searched for in \( R \) using the assumption that \( p' \) is the point closest to \( e_c \) among skeletal points on a line extended from \( e_c \) in the direction of \( e_c \), where \( e_l \) is a neighboring link edgel of \( e_c \). Actually, \( p' \) is given by

\[ p' = \arg \min_{p \in T} \| p - e_c \|, \]  

(8)

\[ T = \left\{ p \mid \frac{\| e_c - p \|}{\| e_c \|} > \cos\left(\arctan\left(\frac{2}{w}\right)\right), \right. \]  

(9)

where \( Q_{\text{link}} \) represents a set of link points belonging to \( Q \). It is an important condition in Eq. (9) that the angle \( \theta \) between \( e_c \) and \( e_l \) should be small (i.e., \( \theta < \arctan(2/w) \), since if we choose the skeletal point closest to \( e_c \) and \( p' \), not taking the angle into consideration, it will produce an undesirable result such as the point \( p_4 \) in Fig. 13(a).

All \( p' \)'s satisfying Eq. (8) are not accepted as \( p' \) due to the problem illustrated in Fig. 13(b). In this figure, the point \( p_2 \) also satisfies Eq. (8), but removing \( p_2 \) seems to cause an unnatural disconnected skeleton. It is because the gap width at \( e_c \) is almost the same as the width of the structure represented by the skeleton. Thus, it is important to measure how rapidly the width of the structure varies in the vicinity of \( p' \). Let \( S_r \) denote a skeletal segment containing \( p' \). We can consider two points \( p_o \) and \( p_b \) (\( p_o, p_b \in S_r \)) farthest from \( p' \) within the circle \( C \) centered on \( p' \) with the radius \( r \) (see Fig. 12). The points \( p_o \) and \( p_b \) can be obtained by tracking points belonging to \( S_r \), starting from \( p' \) in opposite directions. Tracking stops
when it reaches the end points of $S_r$ or the boundary of the circle $C$. The last points tracked become $p_a$ and $p_b$. We use the distance values on the Euclidean distance map $M_d$ to measure the deviation of the structure’s width within $C$. The radius $r$ of $C$ is set to $4M_d(p'_{e})$. The factor of 4 was chosen empirically. The point $p_{r}^{'}$ is accepted as $p_r$ if the condition

$$|M_d(p_{a}) - M_d(p_{r}^{'})| > d_f M_d(p_{r}^{'})$$

or

$$|M_d(p_{b}) - M_d(p_{r}^{'})| > d_f M_d(p_{r}^{'})$$

is satisfied, where $d_f$ is a user-specified deviation factor. Then $p_r$ is removed from $Q$.

Skeletal points before and after applying the above algorithms to the previous result in Fig. 8 are shown, respectively, in Figs. 14(a) and (b). One can see that unwanted bridges between main skeletal segments are broken, and points are identified correctly in the neighborhood of edgels.

### 2.6. Pruning skeleton branches and beautifying a skeleton

In the previous subsection, some skeletal segments are split up intentionally, which may result in some noisy skeletal branches that are not useful to shape analysis. Thus, we need to prune such branches. We use Arcelli and Baja’s pruning algorithm because of its simplicity and efficiency [16]. Pruning starts from an end point of each branch. For each skeletal point $p$ in the branch that ends with $p_e$, the quantity

$$r(p, p_e) = M_d(p_e) - M_d(p) + \| p - p_e \|$$

is computed, where $M_d$ represents the distance map obtained in Step 2. If $r$ is less than a given threshold $T_r$, $p$ is removed from the branch, and pruning goes on. It is stopped when either $r$ becomes greater than or equal to $T_r$, or the other end of the branch is reached. The quantity $r(p, p_e)$ in Eq. (11) can be interpreted as the loss of information we get, in terms of reconstruction of an initial shape, if the branch from $p$ to $p_e$ is pruned away.
Fig. 14. Identified skeletal points (a) before and (b) after applying the two algorithms in Subsection 2.5 to the result in Fig. 8. Parameters: \( w = 5 \), \( d_f = 0.5 \).

Fig. 15. Final skeleton image where edgels and skeletal points are represented by black and gray pixels, respectively. Parameter: \( T_r = 1 \).

After pruning has been finished, very short skeletal segments are deleted, and the final step of the skeleton extraction procedure, beautification, is performed. Beautifying a skeleton means straightening its zigzags that are mostly caused by the unit-width thinning operation. Refer to [16] for the detailed algorithm. In Fig. 15, a final skeleton image is shown.

3. Skeletal segment classification

Since we are dealing with a gray-scale image, not a binary image, it would be meaningful to provide the skeleton obtained in Section 2 with the ability to represent the gray-scale intensity information of underlying structures as well as their shapes. Each skeletal point can be classified as one of three types (RIDGE, RAVINE, or STAIR) according to the cross-sectional shape of the image in its neighborhood. A typical example of a cross-sectional shape for each type is shown in the first column of Fig. 16. In our method, the direction and the size of the cross-section at the position of a skeletal point \( p \) are decided by \( p \) and its two nearest edgels \( e_p \) and \( e_n \) (see Fig. 4). In more specific terms, the intensity values of pixels lying on two line segments \( e_n p \) and \( p e_p \) (from \( e_n \) to \( e_p \) through \( p \) ) constitute the profile of the cross-section.

Let \( L_p \) denote a set of pixels on \( e_n p \) and \( p e_p \) except \( e_n \) and \( e_p \), and let \( I(p) \) denote the gray-scale intensity value at \( p \). It should be noted that it is reasonable to use the Gaussian smoothed version of an input image whose value of \( \sigma \) is equal to that used in the Canny edge detector, rather than an input gray-scale image directly, since the locations of edges, which serve as boundaries of structures to be detected, are affected by the value of \( \sigma \). To determine
what class \( p \) belongs to, the following rules are applied:

\[
\text{let } i_{\text{max}}(p) = \max_{q \in L_p} I(q), \\
i_{\text{min}}(p) = \min_{q \in L_p} I(q); \\
\text{let } e_{\text{max}}(p) = \max(I(e_p), I(e_n)), \\
e_{\text{min}}(p) = \min(I(e_p), I(e_n)); \\
\text{if } (i_{\text{max}}(p) > e_{\text{max}}(p) \text{ and } i_{\text{min}}(p) > e_{\text{min}}(p)) \\
\quad p \text{ is labeled RIDGE}; \quad \text{// see Fig. 16(a) and (b).} \\
\text{else if } (i_{\text{max}}(p) < e_{\text{max}}(p) \text{ and } i_{\text{min}}(p) < e_{\text{min}}(p)) \\
\quad p \text{ is labeled RAVINE}; \quad \text{// see Fig. 16(c) and (d).} \\
\text{else if } (i_{\text{max}}(p) < e_{\text{max}}(p) \text{ and } i_{\text{min}}(p) > e_{\text{min}}(p)) \\
\quad p \text{ is labeled STAIR}; \quad \text{// see Fig. 16(e) and (f).} \\
\text{else} \\
\text{if } ((e_{\text{min}}(p) - i_{\text{min}}(p)) < (i_{\text{max}}(p) - e_{\text{min}}(p))) \\
\quad p \text{ is labeled RIDGE;} \\
\text{else } p \text{ is labeled RAVINE; } \quad \text{// see Fig. 16(g).} \\
\]

The above rules can be understood intuitively by looking at Fig. 16. Attention should be paid particularly to the second column of the figure, where each profile represents a special case of the corresponding type. For example, Fig. 16(b) looks different from the typical cross-sectional shape of a ridge such as (a). In Fig. 16(b), the peak of the profile is positioned far from the center of the structure, and moreover, the intensity value at the center \( p \) is less than the value at the boundary point \( e_p \). In fact, such deviations from a standard ridge shape are what distinguishes a curvilinear structure detector from a simple ridge detector, as pointed out in [9].

After all skeletal points have been classified and labeled, points that are connected and have the same label are grouped to form a skeletal segment. Note that merging cannot continue beyond junction points. Skeletal segments with a length less than or equal to a threshold \( T_s \) (\( T_s \) is usually small) are relabeled UNDETERMINED because they seem to be unstable and noisy. If they are adjacent to stable segments whose length is greater than \( T_s \), the points belonging to the unstable segments...
4. Reconstruction of curvilinear structure regions

In this section, we describe a method for reconstructing the regions of the curvilinear structures that have been found in the form of labeled skeletal segments. Each skeletal segment has been assigned one of three labels (RIDGE, RAVINE, or STAIR) with the method introduced in Section 3. The reconstruction of the region of some curvilinear structure can be said to find a region that gives rise to a corresponding skeletal segment. Since the skeletal segment is obtained based on edge information, the

[Fig. 17. Relabeling of unstable skeletal points ($T_s = 3$). (a) Construction of initial skeletal segments. (b) Segments with length less than $T_s$ are relabeled UNDETERMINED. (c) Merging after the first iteration. (d) Merging after the second iteration. Merging has been finished. (e) Isolated short segment. (f) Relabeling result for the segment in (e).]

[Fig. 18. Skeletal segment classification result. Parameter: $T_s = 5$.]
Fig. 19. Pseudo-code for the rough reconstruction algorithm of curvilinear structure regions.

Procedure: RoughReconstruction

Variables:
- $T_{sl}$: seed point location table (input).
- $M_d$: distance map obtained in Subsection 2.2 (input).
- $L$: label array (input, output).
- $A_{nsl}$: nearest seed point index array (output).
- $M_s$: distance map constructed inside this procedure.
- $Q_{1,2}$: queues in which pixel locations can be stored.

Initialization:
- $L$: initialized before this procedure is called.
- $M_s$: filled with a very large value.
- $A_{nsl}$, $Q_{1,2}$: empty.

Main:
- for $(i = 1; i \leq \#$ of entries in $T_{sl}; i++)$
  - $Q_1(i) = T_{sl}(i); \ M_s(T_{sl}(i)) = 0, \ A_{nsl}(T_{sl}(i)) = i$;
- while ($Q_1$ is not empty) {
  - for $(k = 0; \ i$th pixel $p_i$ in $Q_1$) {
    - $s_n = T_{sl}(A_{nsl}(p_i))$
    - for $(j$th neighboring pixel $n_j$ of $p_i$) {
      - $d = \|n_j - s_n\|$;
      - if ($d < M_s(n_j)$ and $d \leq M_d(s_n)$)
        - $L(n_j) = L(p_i), \ M_s(n_j) = d, \ A_{nsl}(n_j) = A_{nsl}(p_i), \ Q_2(k++) = n_j$;
    }
  - $Q_1 = Q_2$;
  }

Fig. 19. Pseudo-code for the rough reconstruction algorithm of curvilinear structure regions.

region is expected to be enclosed by edges. However, the edge information cannot be used directly to determine the region due to existing edge gaps. In our method, distance map information is used to reconstruct curvilinear structure regions instead of the edge information. The reconstructed regions are represented by labeled sets of connected pixels, where the label of a set is equal to that of a corresponding skeletal segment. Our pixel labeling scheme is as follows.

From now on, we use the term seed point instead of "skeletal point" since the role of labeled skeletal points is similar to that of seed points in conventional "region segmentation by region growing" algorithms. Let $L(p)$ denote a label assigned to a pixel $p$ and let $S$ represent the set of labeled seed points obtained in Section 3. Then, $p$ is labeled by the following rule:

$$L(p) = L(s), \ s = \arg \min_{q \in S_p} \| p - q \|,$$

$$S_p = \{ q \mid \| q - p \| \leq M_d(q), \ q \in S \},$$  \hspace{1cm} (12)$$

where $M_d(q)$ is equal to the radius of a maximal disk at a seed point $q$. $S_p$ represents a set of seed points whose maximal disks include $p$. The pixel $p$ is assigned the same label as that of the seed point that is closest to $p$ among ones belonging to $S_p$.

To realize the above rule, we made some modifications to the region-growing Euclidean distance transform algorithm used in Subsection 2.2. The resulting code is shown in Fig. 19. We applied this algorithm to the
Fig. 20. Reconstruction results of curvilinear structures. (a) Rough reconstruction. (b) Accurate reconstruction.

Procedure: AccurateReconstruction

Variables:
- $I_s$: labeled skeleton image obtained in Section 3 (input).
- $M_d$: distance map obtained in Subsection 2.2 (input).
- $L$: label array (output).
- $T_{1,2}$: seed point location tables.
- $A_{nisi}$: nearest seed point index array.

Initialization:
- $L$: filled with UNDETERMINED.
- $T_{1,2}, A_{nisi}$: empty.

Main:
- for (ith seed point $s_i$ in $I_s$) $L(s_i) = I_s(s_i), T_1(i) = s_i$;
- while ($T_1$ is not empty) {
  RoughReconstruction ($T_1, M_d, L, A_{nisi}$);
  for ($k = 1$; ith region boundary point $p_i$ in $L$) {
    $s_n = T_1(A_{nisi}(p_i))$;
    if ($A_{nisi}(p_i) \neq 0$ and $M_d(p_i) \neq 0$ and $M_d(p_i) < M_d(s_n)$) $T_2(k++) = p_i$;
  }
  $T_1 = T_2$;
}

skeletal segment classification result of Fig. 18, and obtained the result shown in Fig. 20(a). Note that the resulting image is not composed of three kinds of regions, but four. Regions labeled UNDETERMINED are necessary because there may exist a case where $S_p$ is a null set in Eq. (12). Such a case frequently happens around region boundaries, and occurs due to the pruning operation performed on the skeleton.

We extended the algorithm of Fig. 19 to get more accurate region boundaries. Let us consider region boundary
points. A region boundary point $p_b$ is a point satisfying the condition that $p_b$ is not of the UNDETERMINED type and at least one of its four neighboring points (upper, lower, left, and right) is of the UNDETERMINED type. It should be noted that in our case, true region boundaries are determined by detected edges, and that edgels have zero on the distance map. Therefore, $M_d(p_b) \neq 0$ implies that the current region boundary point $p_b$ may not have reached a true region boundary point, and we need to consider expanding the current region to some degree in the neighborhood of $p_b$.

In our extended algorithm, the selection of new seed points and the expansion of regions are carried out iteratively until no new seed points are found. A point $p$ is selected as a new seed point if $p$ is a region boundary point, $M_d(p) \neq 0$, and $M_d(p) < M_d(s)$, where $s$ is a seed point from which $p$ originated. Once new seed points are chosen, regions are expanded using the procedure of Fig. 19. The condition $M_d(p) < M_d(s)$ is needed to prevent the regions from growing outward through gaps in edge contours. The pseudo-code of the extended algorithm is given in Fig. 21. Finally, very small holes remaining in the label array $L$ are filled with the values of neighboring pixels. The final result of region segmentation is shown in Fig. 20(b). Note that pixels labeled UNDETERMINED around the region boundaries have been completely removed.

5. Experimental results

In this section, various examples of results obtainable with our method are shown. The first example is the synthetic image shown in Fig. 22(a), which is a simple multi-level image created with a paint program. Its edge image, which was obtained using the Canny edge detector, is shown in (b), where one can see that there are some gaps in the edge contours. The small gaps in the middle of the image were caused by the Canny edge detector, which are typical phenomena that happen occasionally near junctions where different intensity levels of regions meet. However, somewhat large holes in the right part of the image were made intentionally to show that our method is not very sensitive to such gaps. In most cases, skeletal segments passing through the gaps are removed by pruning operations. Extracted skeletal segments and their classification results are shown in Figs. 22(c) and (d), respectively. The reconstructed curvilinear structure regions are also shown in Figs. 22(e) and (f). Note that the boundaries of the reconstructed regions were not affected by the gaps in the edge contours.

In Fig. 23, four ridge-like structure detection results are shown for the purpose of comparison. The results in (a) and (c) were obtained using Steger’s method [9], which is one of differential geometric ridge detection methods.
Fig. 23. Ridge-like structure detection results. (a), (c) Steger’s method. (b), (d) Our method. Parameters: (a) $\sigma = 1.5$, $T_l = 1.5$, $T_h = 5.0$. (b) $\sigma = 1.3$, $T_l = 1.0$, $T_h = 8.0$, $\phi_l = 60^\circ$, $\phi_h = 120^\circ$, $T_r = 1$. (c) $\sigma = 4.0$, $T_l = 0.2$, $T_h = 0.5$. (d) $\sigma = 2.5$, $T_l = 2.0$, $T_h = 6.0$, $\phi_l = 30^\circ$, $\phi_h = 120^\circ$, $T_r = 2$.

The results of our method for the same input images are shown in (b) and (d). Fig. 23(a) and (c) illustrate well the common weakness of conventional curvilinear structure detection methods. In these methods, as mentioned in Section 1, the detectable range of a curvilinear structure’s width is confined by the given value of $\sigma$. Therefore, if structures are distributed over a wide range of width values in an input image, they often yield an unsatisfactory result. For example, in the case of Steger’s method, it could not properly detect the tapering structure (1) and the road (2) in Figs. 23(a) and (c). It also failed to detect junctions (e.g., (3) in Fig. 23(c)), where more than two curvilinear structures meet when the differences among the widths of the structures are large. As can be seen in Figs. 23(b) and (d), our method is successful in detecting such structures.

Another problem of conventional ridge detection methods is shown by (4) in Fig. 23(c), where the detected positions are deviating from the center line of the road. It can be shown that in the case of an asymmetrical bar-shaped line, the position of the line estimated by conventional ridge detectors moves farther away from the true center position as $\sigma$ becomes larger [9]. Let us suppose that our goal is to find roads in the aerial image of Fig. 23(c). Since it contains various roads of different widths, $\sigma$ should be set to a reasonably large value to cover all the roads, but that would result in biased position estimation for the structures of small width.
like (4) in Fig. 23(c). Steger focused on this matter, and proposed a method for the unbiased estimation of a line position [9]. Note that when obtaining the results in Fig. 23(a) and (c), his unbiased estimation technique was not used intentionally to show the problem of conventional ridge detectors. When detecting edges using Canny’s method, the value of $\sigma$ also affects the positions of detected edges. Discussion on the relation between the value of $\sigma$ and edge positions can be found in [17]. In our method, however, this effect on the positions of detected structures is very small compared to that of conventional ridge detectors, since the value of $\sigma$ does not need to be larger to detect structures of large width. One can see that in (4) of Fig. 23(d), the position of the road detected by our method is correct.

In some cases our method might produce results worse than Steger’s method. A typical case is that a skeleton extracted with our method is sometimes disconnected where the local surface of a gray-scale image has a saddle-like shape. That happens because in such a place an edge tends to be detected whose direction is perpendicular to the direction of the underlying structure. We can find examples of the case in Fig. 23(b). (See (5) in the figure).

Another case is that Steger’s method outperforms our method when brightness at the boundary of a curvilinear structure to be detected varies smoothly. In this case, most edges representing the structure’s boundary are likely to be undetected, which results in failure of our method (See (6) in Fig. 23). The above two cases are common weaknesses of edge-based methods.

Ridge-like structures and their corresponding regions obtained by applying our method to an aerial image of the Nile river are shown in Fig. 24. The width of detected structures is in the range of 4 to 9 pixels and their length is greater than 10 pixels. The width at a skeletal point $p$ is defined as $2(M_d(p) + 1)$. Fig. 24 shows the ability of our method to select the width and the length of curvilinear structures to be detected.

The reconstruction results of curvilinear structure regions of other example images are shown in Fig. 25. Comparing each input with the corresponding region segmentation result would be helpful to figure out characteristics of our method. Notice that complex scenes in images can be decomposed into several types of structures by our method. Thus, a region segmentation result, together with the skeleton extraction result, is expected...
to be useful to make a simplified scene description of an image.

6. Conclusions

In this paper, we proposed a new method for detecting and describing meaningful structures in gray-scale images. Our work is closely related with previous research on curvilinear structure detection. We introduced the concept of skeleton extraction to the gray-scale image domain to overcome the limitations of conventional approaches. Our detector is capable of detecting and describing more general and complex structures. The proposed method for the reconstruction of curvilinear structure regions based on skeleton extraction and skeletal segment classification enables us to decompose an image into several types of regions. Experimental results show that our method works well on both synthetic and natural images. We are now investigating the possibility of a hierarchical description of an image using our method, and looking for new, feasible applications.

References


About the Author—JEEONG-HUN JANG was born in Seoul, Korea, in 1970. He received the B.S. degree in Electrical Engineering
   from Hanyang University, Korea, in 1994 and the M.S. degree in Electrical & Electronic Engineering from POSTECH, Korea, in
   1996. He is now in the Ph.D. program at POSTECH. His research interests include feature extraction, character recognition, and
   face recognition.

About the Author—KI-SANG HONG received the B.S. degree in Electronic Engineering from Seoul National University, Korea,
   in 1977, and the M.S. degree in Electrical & Electronic Engineering from KAIST, Korea, in 1979. He also received the Ph.D. degree
   from KAIST in 1984. During 1984–1986, he was a researcher in the Korea Atomic Energy Research Institute and in 1986, he
   joined POSTECH, Korea, where he is currently an associate professor of Electrical & Electronic Engineering. During 1988–1989,
   he worked in the Robotics Institute at Carnegie Mellon University, Pittsburgh, PA, as a visiting professor. His current research
   interests include computer vision and pattern recognition.