Control and Cybernetics

vol. **31** (2002) No. 2

Probabilistic morphological modeling of hydrographic networks from satellite imagery using Self-Organizing Maps

by

Marek B. Zaremba and Roman M. Palenichka

Départment d'informatique et d'ingénierie, Université du Québec
Hull, Quebec J8Y 3G5, Canada

Abstract: Adequate and concise representation of the shape of irregular objects from satellite imagery is a challenging problem in remote sensing. The conventional methods for cartographic shape representation are usually inaccurate and will provide only a rough shape description if the description process is to be fully automated. The method for automatic cartographic description of water basins presented in this paper is based on Self-Organizing Maps (SOM) - a class of neural networks with unsupervised learning. So-called structured SOM with local shape attributes such as scale and local connections of vertices are proposed for the description of object shape. The location of each vertex of piecewise linear generating curves that represent skeletons of the objects corresponds to the position of a particular SOM unit. The proposed method makes it possible to extract the object skeletons and to reconstruct the planar shapes of sparse objects based on the topological constraints of generating lines and the estimation of local scale. A context-dependent vertex connectivity test is proposed to enhance the skeletonization process. The test is based on the Markov random chain model of vertices belonging to the same generating line and the Bayesian decision-making principle. The experimental test results using Landsat-7 images demonstrate the accuracy of the proposed approach and its potential for fully automated mapping of hydrological objects.

Keywords: self-organizing maps, neural networks, morphological modeling, satellite image processing, skeletonization, Markov chains.

1. Introduction

In several tasks related to the processing of satellite imagery, the main concern is adequate and concise representation of the shape of a particular class of objects. What is of interest in this paper is the task of automated hydrographic mapping hydrological objects, such as rivers, canals, lakes, etc., from the image, and their aggregation in a graphical structure in order to obtain a complete shape description of the objects. In practice, this task is often complicated by the presence of sparseness, discontinuities, and occlusions. A typical example of a hydrographic network is a water basin consisting of many rivers of various widths and lengths. The shape sparseness and discontinuities result here mainly from occasionally narrow segments of rivers, segmentation procedures applied to noisy images, and insufficient resolution of the imaging system. The occlusions are a result of the cloud, bridge or tree coverage. Similar problems are also encountered in tracing road networks from aerial or satellite imagery. What makes automated mapping of hydrological networks more challenging is the need to deal with irregular shapes of widely varying scale.

Most existing algorithms for object detection and tracing are based on structural features capable of providing a description that can be used to extract the objects of interest from a given scene. For example, boundaries of objects, snakes and skeletons (medial axes) are often used as reliable shape descriptors (Haralick and Shapiro, 1992). The approach based on shape skeletons is, in the context of this application, efficient since it can represent in a very concise manner the topology of an object with several connected parts and shape details (Blum and Nagel, 1978). Such a description can be considered as a mixture of region-based and boundary-based shape descriptors.

The classical skeletonization algorithms are based on an iterative thinning of objects in a binary image until predetermined conditions for the resulting skeletal shape are satisfied (Haralick and Shapiro, 1992). Here, they will be not considered for the solution of the above stated problem because they fail to extract skeletons correctly when the shape is sparse. Fortunately, there are several methods for coping with sparse shapes. The most popular, offering a good compromise between efficiency and simplicity, are based on median filtering and morphological set operations: dilation, opening, and closing (Haralick and Shapiro, 1992, Chen and Yu, 1996). Afterwards, a standard thinning algorithm is applied to extract the shape skeleton. However, such methods usually introduce a large number of distortions when the shape sparseness is significant, especially, when discontinuities or occlusions are present.

Recently, a statistical method of principal curves has been developed to extract directly the skeletal description of point sets without using classical thinning procedures (Hastie and Stuetzle, 1989, Kegl et al., 2000). The underlying idea is the generalization of a linear principal component approach to a concise representation of sets of points in the form of principal curves and graphs. The algorithms for drawing principal curves using piecewise linear approximation are, in their initial form, limited to simple curves or manifolds, where, for example, no intersections are allowed. Their application to more complex sets of points requires the use of segmentation and thinning procedures in order to extract a graph that provides a rough approximation of the resulting

Unsupervised neural network methods, such as those based on self-organizing maps (SOM), are quite robust against sparse shapes. This is a novel application of the self-organization principle in image analysis since the SOM approach is applied usually for the clustering image intensity and recognition of objects in images. On the contrary, in application to detection of objects with sparse shape object points on the image plane are self- organized by using a SOM structure. For example, a flow-through SOM is used to obtain the shape skeleton of a connected set of points (Datta, Parui, and Chaudhuri, 1996). The SOM is initialized with a linear topology and evolves to different local patterns based on the angle formed at each map unit by its neighbors and the distance between the two map units. The use of a flow-through SOM in this method is sensitive to the learning schedule and the order in which the data are presented.

Another similar approach, which is based on the batch formulation of the SOM algorithm, has been proposed for the shape skeletonization (Singh, Cherkassky and Papanikopoulos, 2000). The shape skeleton is obtained from a data-driven minimal spanning tree (MST) topology. This method has shown good results, but it requires an initial segmentation of the connected object components. The method deteriorates if the segmented object contains components of various sizes; then the skeleton parts corresponding to larger- scale components become ragged. The idea of adaptive SOM-based skeletonization in an application to shape extraction of simple objects has also been proposed by Der, Balzuweit and Herrmann (1996). An effective method for controlling the skeleton smoothness is the use of an adaptive scale parameter in the neighborhood function of the SOM that performs the skeletonization. Hierarchical shape description using symmetric axes is another useful generalization of the conventional, single-scale, skeletal representation of a planar shape (Pizer et al., 1987, Ogniewicz and Kubler, 1995). It permits extraction of the most relevant shape features at different resolutions by starting from larger scales, and provides a natural association of shape details defined at lower scales with the skeleton lines obtained at larger scales.

A common deficiency of existing approaches to skeletonization is the absence of an underlying model for shape representation. A model-free method, which does not take contextual information into account, is not able of extracting skeletons reliably when the object shape is sparse and includes discontinuities or local occlusions. It makes the theoretical performance analysis of a particular algorithm hard, thus concealing the ways to a further improvement of the algorithm. In particular, the SOM-based algorithms do not take into consideration the size variations of the object elements, and the constraints involved in the analysis of the possible connections between skeleton vertices, which correspond to the SOM units (Singh, Cherkassky and Papanikopoulos, 2000). All this yields poor skeletonization results if the object is composed of parts of various widths.

The proposed SOM-based skeletonization algorithm is based on a morphological image model where the planar shape of the object is represented by lines represent skeletons of the planar object. A piecewise linear representation of curvilinear generating lines has been adopted for the purpose of a uniform model description and the ease of implementation. The concept of so-called *structured SOM* with local shape attributes, such as local scale and connections of vertices, is used to define the object shape. The location of each vertex of the piecewise linear generating curves on the image plane corresponds to the position of a particular SOM unit. The topology of the structured SOM adapts, through the self-organization process, to the skeletal topology of the modeled object, and the estimated attributes of the SOM units describe the skeletal shape attributes. The proposed method permits extraction of skeletons and reconstruction of sparse shapes of planar objects based on the topological constraints of the generating lines and an estimation of the local scale.

In contrast to other methods, the connection of each skeleton vertex with its closest adjacent vertices is tested based on statistical model constraints. In this way, the connectivity problems due to local occlusions, shape discontinuities and sparseness can be to a large extent avoided. Another useful novelty of the proposed algorithm is the incorporation of information on the object local scale into the smoothing kernel function, which controls the self-organization of SOM. It provides a size-invariant self-organization of units along the medial axes for objects consisting of several parts with substantially different sizes. In contrast, the known SOM-based algorithm using MST (Singh, Cherkassky and Papanikopoulos, 2000) gives in practice uneven and ragged skeleton lines for objects, which contain elongated parts of various widths.

Following the introduction in Section 1, the underlying morphological model is presented in Section 2. The formal statement of the problem of shape extraction and tracing is defined in the context of learning the object model. Section 3 presents the proposed method for the extraction of the skeletal shape of objects from multi-spectral images such as satellite images. The section describes the application of structured SOM for the extraction of skeletal shapes in the form of piecewise linear generating lines along with the estimation of scale attributes and connectivity attributes for each SOM unit. Experimental results of object extraction and shape reconstruction, as applied to synthetic and real images from remote sensing, are described in Section 4, and concluding remarks are given in Section 5.

2. Morphological model

2.1. Representation of multi-spectral images by property maps

In the case of multi-dimensional image data, such as multi-spectral satellite imagery, several images of the same scene can be provided simultaneously. In remote sensing, they represent, for example, a view of the earth surface obtained by using electromagnetic radiation of different wavelengths. A pixel with image

T intensity values corresponding to T wavelength bands. Images obtained from the Landsat-7 satellite using the Enhanced Thematic Mapper Plus (ETM+) scanning radiometer instrument are produced with seven bands of reflected energy and one band of emitted energy. Since one of the eight ETM+ channels is a panchromatic channel, in Landsat-7 imagery $T = 7$. In many application areas, multi-dimensional image data are also obtained from a single image by extracting local intensity properties with respect to each image point (i, j). This approach is frequently applied to images with textured objects and backgrounds. In the case of Landsat-7, such extraction could be performed on the panchromatic channel data.

There are usually correlations—especially in the thermal bands—between the intensity components of the same pixel. The correlation between adjacent pixel intensities for each image component $g(i, j)$ of a multi-spectral image can be described using a dynamic polynomial regression model (Palenichka and Ivasenko, 1999). The model is valid in every image point but may have different parameters in different points in order to represent the non-homogeneity of image intensity. This image model states that the image intensity function $g(i, j)$ can be represented by a polynomial function of order q within a neighborhood region (window) around the current point and an independent white noise term.

In order to consider both the intensity and the shape description of given objects in multi-dimensional images, an intermediate image representation, called *property map*, will be used in the modeling. The property map is obtained from the initial multi-dimensional image by extracting one relevant image property per point. The map generates a piecewise constant model of objects to be segmented in the initial multi-dimensional image by defining a function $f_{i,j}(b_1, b_2, \ldots, b_T)$, which is an implicit function of components of the multi-band intensity vector $[b_1, b_2, \ldots, b_T]$ in point (i, j). It is also assumed that a random zero-mean perturbation term $y(i, j)$ with a unit variance is present in point (i, j). Thus, the property map can be defined as:

$$f_{i,j}(b_1, b_2, \ldots, b_T) = \lambda \cdot y(i,j) + \sum_{l} \tau_l(b_1, b_2, \ldots, b_T) \cdot \varphi_l(i,j) \tag{2.1}$$

where $\tau_l(b_1, b_2, \ldots, b_T)$ is a vector of constant intensity values of image plane segments corresponding to objects belonging to class l (e.g., a water basin), the indicator function $\varphi_l(i, j)$ is a binary map of objects of the lth class, and λ is the perturbation level, i.e., the standard deviation of the white noise term $\lambda \cdot y(i, j)$. The indicator function $\varphi_l(i, j)$ is equal to zero in the whole image plane except for the points belonging to objects of interest in the lth class. In the modeling by Eq. (2.1), it is supposed that a majority of elements (points) of the perturbation term in (2.1) is composed of independent identical normally distributed random variables with variance λ^2 except for a relatively small number of outliers. An uncorrelated outlier occurs with a certain probability and has a different distribution of its intensity. The outlier points are responsible for by comparing $f_{i,j}(b_1, b_2, \ldots, b_T)$ with two constant thresholds: lower threshold δ_l and upper threshold Δ_l. If $f_{i,j}(b_1, b_2, \ldots, b_T)$ is within the thresholds, then the point (i, j) is considered as a point of a class l object; otherwise it belongs to objects of other classes. The outlier probability is equivalent, in this case, to the percentage of shape sparseness π. Other types of shape distortions, such as discontinuities and occlusions, can also be modeled by outliers, but with a higher degree of correlation and under the assumption of a particular distribution law.

Since the explicit form of function $f_{i,j}(b_1, b_2, \ldots, b_T)$, which extracts the intensity property, is not known, the self-organization principle used in SOM can be exploited to obtain this functional transformation implicitly. It applies the well-known capability of the SOM-type neural networks (Kohonen et al.,1996) to perform clustering of input vectors, in this case vectors with coordinates $(b_1, b_2, \ldots, b_T)$. Such a clustering provides an effective segmentation of the input multi-band image in the form of a binary image $\eta(i, j)$. Each image point (i, j) whose property is located within a given cluster (clusters), i.e. belongs to an object of class l is labeled in $\eta(i, j)$ by "1", the other points of the image $\eta(i, j)$ are labeled by "0". The actual shape sparseness π for objects of class l can be measured from the binary image $\eta(i, j)$ and binary map $\varphi_l(i, j)$ by the percentage of object points with the value "1" with respect to all the object points. The total number of object points of lth class is equal to all non-zero points in $\varphi_l(i, j)$.

2.2. Morphological representation of planar shapes

The objective of planar shape modeling is to obtain a concise representation of an object whose property map satisfies model (2.1). An efficient approach to defining the planar shape is to use the multi-scale morphological image model that describes the object by using structuring elements, axes of symmetry, and skeletons as well as a contour description (Haralick and Shapiro, 1992). Some interesting extensions of the skeletal shape description to the multi-scale representation of planar shape were discussed, for instance, in Pizer et al. (1987), Ogniewicz and Kubler (1995). In our approach, a simple multi-scale morphological model is assumed when developing a skeletonization algorithm. In this model, one initial structuring element S_0 of a minimal size is selected that determines the size and resolution of the imaged objects. It has a symmetric disk shape to insure the rotation invariance as well as a reasonable approximation of different possible shapes. Two types of scales can be distinguished, *uniform scales* and *logarithmic scales.* The structuring element with diameter d_m at scale m in the uniform scales system is formed, as shown in Fig. 1, by a consecutive binary dilation (denoted by $\oplus$) by S_0, $S_m = S_{m-1} \oplus S_0$, $m = 1, 2, \ldots, M - 1$,

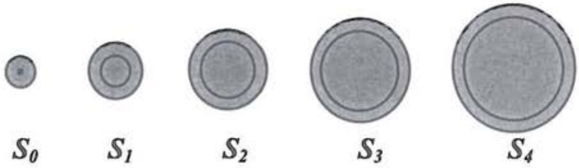

Figure 1. An example of formation of 5 uniform scales by consecutive scale dilation of S_0.

Generation of a planar object shape can be modeled, in the continuous and discrete cases, by a growth process (Maragos, 1989). Formation of an isolated object starts from a single point called *seed*. The seed begins to grow along a *generating line*, straight or curvilinear, forming a growth path. In the discrete case, the growth path is represented by a generating set. The generating set is a one-pixel wide connected set of points, each of them—except for the end points—having two adjacent points. A piecewise linear approximation in the form of straight-line segments is used here to represent curvilinear generating lines as well as direct lines. A constraint of constant curvature, meant as the same angle between two adjacent straight-line segments, is imposed on the generating line. A scale value is assigned to each vertex point and a generating line is represented as a concatenation of straight-line segments. Given two vertices, a blob-like object is formed by two structuring elements, S_k and S_l, with diameters d_k and d_l, associated with the end vertices u_k and u_l of a given straight-line segment G (see example in Fig. 2a).

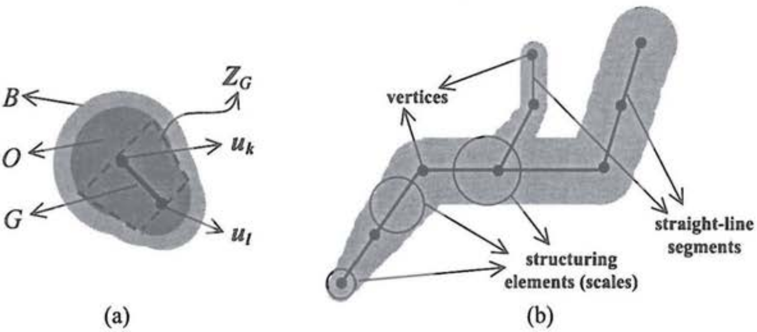

Figure 2. An example of a blob-like object with a segment trapezoid $Z_G(u_k, u_l)$ shown in dashed lines in (a) and multi-scale formation of a simple object (b) by the concatenation of blob-like objects. The generating lines composed of straight-line segments

The domain region U of the blob-like object is formed by using the operation of dilation of a generating straight-line segment (set) G with a structuring element, $S(G)$, of variable size (scale):

$$U = G \oplus S(G) = \bigcup_{(i,j)\in G} S_m(i,j), \tag{2.2}$$

where $S_m(i,j)$ is the structuring element with size d as the following function of point (i,j) of the generating line G:

$$d(i,j) = \alpha_k(i,j)d_k + \alpha_l(i,j)d_l, \tag{2.3}$$

where $\alpha_k(i,j)$ and $\alpha_l(i,j)$ are the ratios of distances between the current point (i,j) and the end points of the generating line divided the length of the generating line segment. The object points that are projected on the straight-line segment G (excluding the vertices) are within the segment trapezoid $Z_G(u_k,u_l)$ (see Fig. 2a). A multi-scale object of interest is formed from the blob-like objects by concatenating their vertices, as shown in Fig. 2b. The result of concatenation at a single scale produces a *primitive object*. Thus, an object of interest composed of parts with different scales is a concatenation, according to Eq. (2.3), of primitive objects of different scales.

Finally, this morphological model of planar shape is combined with the image property model (2.1) in such a way that every binary map $\varphi_l(i,j)$ for objects belonging to class l is represented by the morphological model, using Eq. (2.2) and (2.3).

2.3. Markov chain model for generating lines

The correlation of a current vertex position with the positions of the previous two vertices connected in a straight-line segment offers additional information for checking the vertex connectivity, provided they belong to the same generating line. A Markov random chain model is suitable to describe in a natural and very simple manner the position of the next vertex of a generating line as dependent on the direction of the previous straight-line segment (Kemeny and Snell, 1976). It can be considered as a consistent model for the growth process of a generating line. Namely, it describes a macro-growth process in the sense that the growth is observed along the entire generating line, with a subsequent scale interpolation by Eq. (2.2) to form the object planar shape. The model consistency is valid for hydrographic networks whose planar shape can be modeled using the proposed morphological model with relatively long and smoothly varying generating lines. The constraint that the direction of such generating line cannot vary abruptly is quite natural in this case.

The Markov chain model is used here to provide the probability of the position of the next vertex of a generating line with respect to the positions of the line macro-growth process can be defined by the conditional probabilities of the new vertex position with respect to the positions of the previous two adjacent vertices on the same generating line. Let u_s, u_l and u_k be three consecutive vertices of two straight-line segments $[u_s, u_l]$ and $[u_l, u_k]$ of the same generating line. The Markov chain model of a generating line G can be defined in terms of conditional probability of the slope $\theta_{l,k}$ of line segment $[u_l, u_k]$ with respect to the slope $\theta_{s,l}$ of its preceding straight-line segment $[u_s, u_l]$:

$$\{P(\theta_{l,k}, [u_l, u_k] \subset G) = P(\theta_{l,k}/\theta_{s,l}, \forall\{[u_s, u_l], [u_l, u_k]\} \subset G\}, \tag{2.4}$$

where $P(\theta_{l,k}/[u_l, u_k] \in G)$ is the unconditional probability of the slope of the straight-line segment between vertices k and l of the same line G, and $P(\theta_{l,k}/\theta_{s,l}, \forall\{[u_s, u_l], [u_l, u_k]\} \subset G\}$ is the conditional probability of the slope of the straight-line segment between vertices k and l of the same generating line G provided that the slope of straight-line segment $[u_s, u_l] \subset G$ equals $\theta_{s,l}$.

In practice, two approaches can be used in order to define explicitly the probability (2.4): the empirical distribution function and the probability density function with estimated parameters. The distribution parameters are estimated during the model learning stage (see Section 2.4). The parametric approach is quite suitable in this case since, for example, a uniform distribution of the slope difference $\theta = \theta_{k,l} - \theta_{s,l}$ with zero mean describes quite well the real situations. A Gaussian density function $\chi(\theta)$ is also a good approximation of the real situations since the angle difference θ will oscillate around zero for a smoothly varying generating line:

$$\chi(\theta) = 1/\sqrt{2\pi\xi^2} \cdot \exp(-\theta^2/2\xi^2), \tag{2.5}$$

where ξ^2 is the variance of the random variable θ. It is assumed that the variance ξ^2 is constant for all generating lines, whereas the mean value of the slope difference is equal to zero.

2.4. SOM learning of morphological models

In application of our model, the problem of shape extraction for objects of class l consists in solving two separate problems: *segmentation of objects* and *morphological shape extraction*. The solution of the segmentation problem corresponds to finding a clustering mapping of the primary feature space (dimension T) to the property map (dimension 1) for class l of the objects having their cluster center at τ_l (Eq. (2.1)). The solution of the shape extraction problem in the case of piecewise linear skeletons consists in determining a multi-scale structural graph of the entire planar shape. As discussed earlier, the generation of the morphological model can be reduced to the definition of the vertices of line segments and their local connectivity as well as the vertex scales. The approach based on self-organizing maps has been applied to solve both problems. However, the of the first problem involves multi-dimensional sets, i.e., vectors of primary features with T band components, for training a conventional (unstructured) SOM (Zaremba et al., 2000). In contrast, the point sets defined in two respective coordinates of segmented images are the input data for the model-based shape extraction that uses a structured SOM.

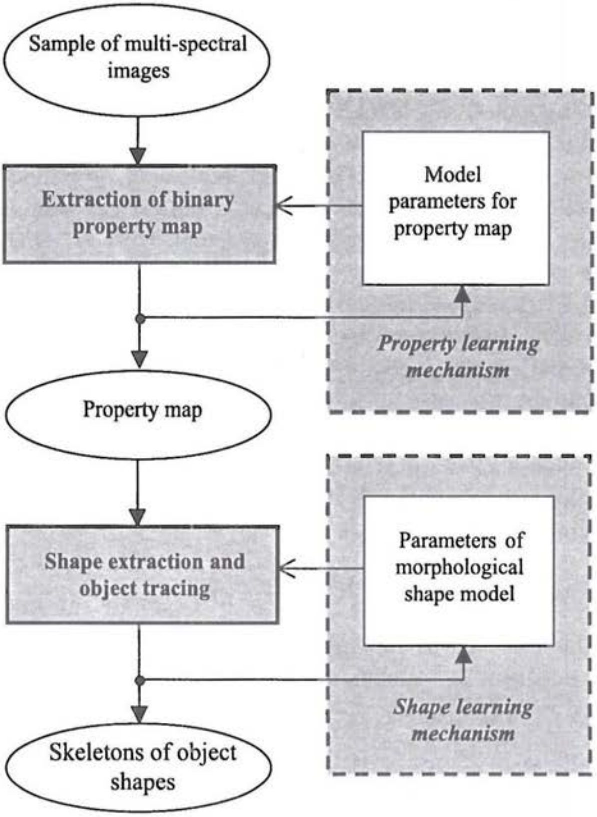
Sample of multi-spectral images
Extraction of binary property map
Model parameters for property map
Property learning mechanism
Property map
Shape extraction and object tracing
Parameters of morphological shape model
Shape learning mechanism
Skeletons of object shapes

Figure 3. Shape learning and extraction for object tracing in multi-spectral images.

An enhanced fuzzy SOM (FSOM) learning algorithm was used to train the unstructured SOM. In this algorithm, the learning rate is replaced by a membership value, which is related to the distance between the centres of clusters and samples. The winning cluster takes the largest membership value. The fuzzy SOM algorithm can be presented as the following sequence of steps:

STEP 1. Randomly initialize the weights w_{ij}, select the number of clusters c, neighborhood size, neighborhood decrease rate, m_0 $(1 < m_0 < \infty)$, STEP 2. Update all the winning neurons, for which the Euclidean distance is smallest, and their neighborhood:

$$v_j(k+1) = v_j(k) + \frac{\sum_{i=1}^{n} \alpha_{ij}(k)^{m(k)}(x_i(k) - v_j(k))}{\sum_{i=1}^{n} \alpha_{ij}(k)^{m(k)}} \tag{2.6}$$

where $\alpha(k)$ is calculated by using equations:

$$\alpha_{ij}(k) = \left[\sum_{k=1}^{c} \left(\frac{|x^i - v_j|}{|x^i - v_k|} \right)^{\frac{2}{m-1}} \right]^{-1}, \tag{2.7}$$

$$v_i = \frac{\sum_{i=1}^{n} x_i(\alpha_{ij})^m}{\sum_{i=1}^{n}(\alpha_{ij})^m}. \tag{2.8}$$

The parameter m in (2.7) is equal to $m(k)$ in (2.6) and changes with time according to:

$$m(k) = m_0(1 - e^{-E(k)}) + m_{\min} \tag{2.9}$$

where m_0 and $m_{\min}$ are some positive constant greater than one.

STEP 3. Calculate

$$E(k) = \max\{\|v_j(k+1) - v_j(k)\|^2\}. \tag{2.10}$$

If $E(k) < \varepsilon$, stop. Else, if the neighborhood update time is reached and the neighborhood > 0, reduce the neighborhood by 1, and then go to Step 2.

The degree of fuzziness is related to m. As $m \to \infty$, $\alpha_{ij} \to 1/c$ and when $m \to {}^{+}1$, $\alpha_{ij} \to 1$ or 0. At the beginning, $m(k)$ is almost equal to $m_0 + m_{\min}$, as $E(k)$ is large. The winner and its neighborhood are updated at almost the same learning rate. During learning, $E(k)$ decreases, and the learning rate of the winner will be larger than of its neighborhood. Finally, $m(k)$ will be closer to $m_{\min}$, and $m \to +1$.

In order to reduce the computational time in equation (2.7), the neighborhood size also shrinks with time. When the neighborhood size is becoming smaller, because only the membership values of winner and its neighborhood will be calculated, the computational time is reduced, especially for a large number of clusters. In the fuzzy SOM, updates are non- sequential, thus the updated weights are independent of the data labels. Rather than terminating at an assigned number of iterations, fuzzy SOM algorithm terminates when it converges.

Labeling of object classes can be performed manually, for example, by the user pointing on pixels belonging to objects of a river basin. In contrast, the learning phase on the stage of shape extraction using a structured SOM consists simply of the determination of basic parameters in the underlying morphological model. It includes the computation of the range of object sizes (local scales) as lines (Section 2.3). Additionally, some parameters of the property map model are also computed during the shape learning. For example, the shape sparseness π (Section 2.1) is estimated during the model learning in order to be used later in the skeletonization algorithm while connecting skeleton vertices.

3. Tracing of skeletal shapes

3.1. Segmentation of objects of interest and extraction of skeletal shapes

As discussed above, the proposed method for model-based tracing of object shapes consists of three main stages: segmentation of objects belonging to a given class, determination of object multi-scale skeletal shape by piecewise linear skeletonization, and morphological reconstruction of the planar shape. The principle of self-organization is used for both image segmentation and planar shape extraction. The SOM-based segmentation is applied prior to starting the skeletal shape extraction. A conventional self-organizing network performs clustering of multi-spectral images and formation of the property map followed by a binarization procedure (e.g., Ritter, Martinez and Schulten, 1992). As a result of the training process the SOM is tuned to classify objects of interest belonging to the same class. A one-dimensional linear or two- dimensional eight-neighbor SOM topology can be successfully used for this purpose. If the linear topology is selected, then the cluster border for the lth object class is defined by two threshold values thresholds δl and Δ_l in the SOM topological space. Such a linear topology establishes a correspondence between the position of the SOM unit in the linear self- organizing map and the continuous values of $f_{i,j}(b_1, b_2, \ldots, b_T)$ in the model (2.1) of the property map. If a pixel magnitude of the image property map falls within the two then it belongs to class l, and the corresponding pixel of the resulting binary image $\eta(i,j)$ will be labeled by "1". At a proper selection of the total number of used units, a single neural unit can be selected for labeling object points of lth class. This means that the two thresholds δ_l and Δ_l will coincide. The obtained image $\eta(i,j)$ is an estimate of the model for the indicator function $\varphi_l(i,j)$ in Eq. (2.1) and represents the objects of that class in the form of a binary image. After this, the shape extraction algorithm (see Section 3.2) based on the notion of structured SOM takes the object points labeled by "1" of image $\eta(i,j)$ as input data for the skeletonization. The SOM units, which correspond to vertices of the shape skeleton, are connected in a multi-scale structural graph (MSG) based on a connectivity test. The connectivity attributes describe the skeleton topology on the image plane and the scale attribute associated with each SOM unit provides the local size of the objects of interest.

The skeletonization approach based on the structured SOM with shape attributes allows us to reconstruct the planar shape of objects of lth class, i.e.

tons of sparse objects in $\varphi_l(i,j)$ are extracted correctly, then the reconstructed shape will be not sparse and will not contain discontinuities or occlusions if such were initially present.

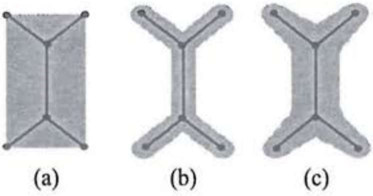

Figure 4. Different shape reconstruction results from the same piecewise-linear skeleton.

Scale plays an important role in adequate reconstruction of the object shape from its skeletal representation, since skeletons alone give ambiguity in shape representation. This is illustrated in Fig. 4 for the case of a bar-like object. In Fig. 4a there are 6 vertices connected by applying the minimal spanning tree (MST) algorithm without relying on the object scales. As a result, the object is reconstructed as a rectangle. In Fig. 4b we have the same 6 vertices connected by the MST algorithm with the scale information; the same scale is assigned to all the vertices. Similarly, the two central vertices in Fig. 4c have the same scale factor, whereas the four peripheral vertices have a smaller scale. Let us note that in this example, scale-based linear interpolation (2.2) has been used to reconstruct morphologically the objects of interest.

3.2. Determination of skeleton vertices using structured SOM

The procedure of finding the vertices of piecewise linear skeletons is based on a batch-mode SOM training algorithm, since in this case the final positions of map units are not sensitive to the order of presentation of image pixels (Singh, Cherkassky and Papanikopoulos, 2000). The algorithm using the MST topology has shown a good performance but is not well suited for a multi-scale shape. In the proposed approach a structured SOM is used, where the units contain—besides the conventional coordinates in the data space - such attributes as local scale and local connectivity. Local connectivity attributes are in the form of a list of adjacent units connected to the current unit. The scale attribute is important for correct skeletonization of multi-scale objects, and it is also necessary to determine the connectivity between the vertices, as well as to reconstruct the object shape. Correct determination of the local connectivity between vertices is of particular importance, since there is no known general approach to evaluating the connectivity in the case of sparse shapes.

The proposed procedure for shape learning and extraction is illustrated in Fig. 5. Vertices of the piecewise linear skeletons are determined by iterating four are shown within oval blocks. The image data after segmentation of objects of a given class are represented as binary shape map, i.e. a binary image $\eta(m,n)$. Additionally, the learned or statistically estimated parameters of the underlying morphological model (see Section 2.2) are available at all steps of the algorithm for the purpose of utilization of object shape constraints when dealing with sparse shape data.

The SOM-based method of skeletonization using MST (Singh, Cherkassky and Papanikopoulos, 2000) does not check for sparse connections between the tree vertices. Therefore, it is not suitable for describing circular and other connected contours without performing an additional connectivity check. These shortcomings of the MST approach were a motivation for using, instead, a multi-scale structural graph (MSG) for skeletonization. The proposed SOM-based algorithm using MSG consists in iterative updating of unit weights while progressively decreasing the span of a SOM kernel function. The kernel function of the SOM, which takes into account the scale attributes of SOM units, controls the updating of unit coordinates in the data space. Eventually, the object skeleton in the form of a MSG is determined by iterating Steps 1–4 in the following algorithm (see Fig. 5):

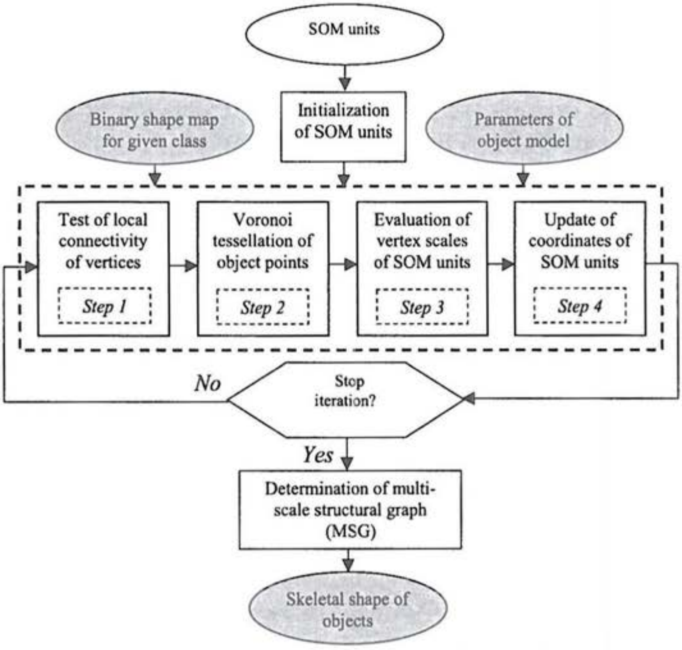
SOM units
Initialization of SOM units
Binary shape map for given class
Parameters of object model
Test of local connectivity of vertices
Step 1
Voronoi tessellation of object points
Step 2
Evaluation of vertex scales of SOM units
Step 3
Update of coordinates of SOM units
Step 4
No
Stop iteration?
Yes
Determination of multi-scale structural graph (MSG)
Skeletal shape of objects

Figure 5. General flowchart for the SOM-based extraction of skeletal shapes using the

Input and output data of the algorithm. The structured SOM is given by its neural units $u_1, \ldots, u_K$ and unit weight vectors $\mathbf{w}_1, \ldots, \mathbf{w}_K$. Two kinds of shape attributes are associated with each unit: one scale attribute ρ_r, where $r = 1, \ldots, K$, and K binary connectivity attributes. The range of a scale attribute is determined by the used scale system with M structuring elements $\{Sm\}$, $m = 1, 2, \ldots, M-1$. For a current unit, the connectivity attributes given in an incidence matrix indicate to which units the current one is connected. The incidence matrix is a diagonal symmetric matrix of size $K \times K$ with elements $\{a_{k,l}\}$ equal to one or zero depending on the connectivity of units u_k and u_l. Input data vectors $\mathbf{v}_1, \ldots, \mathbf{v}_N$ represent the image plane coordinates of object points given in binary image $\eta(i, j)$, which is obtained after the image intensity clustering (segmentation) using a conventional SOM learning procedure. N is the total number of non-zero points in the binary image $\eta(i, j)$. The *output data* of the algorithm is a MSG, a graph with vertex attributes that describes the skeletal shape of objects in the binary image $\eta(i, j)$. In fact, the MSG is the structured SOM obtained after the self-organization process based on the data of image $\eta(i, j)$. Each vertex of the MSG corresponds to a particular SOM unit u_r, $r = 1, \ldots, K$, although some SOM units may be void (eliminated) after the self-organization process. Eventually, the MSG can be a weighted graph with probabilities of connectivity attached to the graph edges. This is the case when a measure of connectivity strength (e.g., a probability of connectivity) is evaluated during the formation of the MSG.

STEP 0. *Initialization.* SOM units $u_1, \ldots, u_K$ are initialized in the data space. For example, a linear plane initialization can be made with respect to weight vectors $\mathbf{w}_1, \ldots, \mathbf{w}_K$, which represent a two-dimensional grid topology. The initial total number of units K (a certain number of units may later be deleted) should be comparable with the maximal number of vertices in the object skeletons according to the model data. This parameter can be determined based on the maximal rank of vertices and the length of minimal straight-line segment. Here, the rank of a current vertex indicates how many vertices are connected to this vertex. The scale attributes of all units, $\rho_1, \ldots, \rho_K$, are set to a maximal scale value, which is comparable with the image size. This means that each unit is initially connected to its q nearest neighboring SOM units, where q is the maximal allowed rank of vertices.

STEP 1. *Determination of vertex neighborhood connectivity.* At the beginning, SOM units are locally connected into local structures by checking the connectivity between units in a given region (see Section 3.4 for details). During the first iteration, the full connectivity will be established since the unit scales are all equal and maximal in size. The connectivity test is based on the analysis of the input data vectors that are located between two units having the same range of scales. Connectivity attributes are assigned to all the units.

STEP 2. *Matching of point coordinates to vertices.* This step corresponds to a similarity measure. The number z_p of a SOM unit in the topological space, which is closest to pth data point, is determined as

$$z_p = \arg\min_r \|\mathbf{v}_p - \mathbf{w}_r\|, \quad p = 1, \dots, N. \tag{3.1}$$

As a result, the region of object points of the input binary image $\pi(i,j)$ will be partitioned into N Voronoi regions, $\{V_p\}$. The new weights of rth unit, u_r, will be determined based on pixels belonging to the rth Voronoi region and to the regions, which are adjacent to V_r.

STEP 3. *Evaluation of vertex scale.* The scale attribute of rth SOM unit, ρ_r, is estimated by considering the Voronoi region of the SOM unit , and is proportional to the number of points in rth Voronoi region, V_r. However, more accurate scale estimation is made by finding the maximal disk inscribed into the Voronoi region V_r as follows:

$$\begin{aligned}\rho_r = \arg\max_k \Bigg\{ \max_{(i,j)\in V_k} \Bigg\{ \Bigg| \frac{1}{|S_k|} \sum_{(m,n)\in S_k(i,j)} \eta(m,n) \\ - \frac{1}{|R_k|} \sum_{(m,n)\in R_k(i,j)} \eta(m,n) \Bigg| \Bigg\} \Bigg\},\end{aligned} \tag{3.2}$$

where $k = 0, \dots, M-1$, $\eta(m,n)$ is the pixel value of the binary image at point (m,n) obtained after the object segmentation, $S_k(i,j)$ is the kth structuring element centered in point (i,j), $R_k(i,j)$ is a ring of points around it, $|S_k|$ denotes the total number of points inside the set S_k, and M is the total number of scales. The structuring element $R_k(i,j)$ is a disk with diameter d_k centered at point (i,j) (see Fig. 1). The width of the ring $R_k = S_{k+1} \setminus S_k$ is selected as being equal to the minimum distance between two separated objects in order to extract isolated objects. Such a scale estimate is used due to the assumption of shape sparseness: the direct method of the maximal inscribed disk gave poor results. Point $\mathbf{t}_r = (i_r, j_r) \in V_r$, which corresponds to the maximum value of the scale calculated according to (3.2) is called the *attraction point* of the region V_r. The use of the scale estimate and the attraction point is needed when calculating conditional expectation of unit coordinates in the next step of this algorithm.

STEP 4. *Updating of coordinates using conditional expectation.* This is the stage of updating the weights of SOM units, i.e., the vertex coordinates. We use here a modified approach to conditional expectation, so that it takes into account the scale attribute associated with a current unit (Singh, Cherkassky and Papanikopoulos, 2000):

$$w_r = \frac{\sum_{p=1}^{K} v_p \Phi(\mathbf{w}_r, \mathbf{w}_p, \mathbf{t}_r)}{\sum_{p=1}^{K} \Phi(\mathbf{w}_r, \mathbf{w}_p, \mathbf{t}_r)}, \quad r = 1, \dots, K \tag{3.3}$$

where $\Phi(\mathbf{w}_r, \mathbf{w}_p, \mathbf{t}_r)$ is a monotonically decreasing kernel function defined indi-

rth and pth SOM units, and distance $\Delta_2(\mathbf{w}_r, \mathbf{t}_r)$ between the position of unit r and the position of attraction point $\mathbf{t}_r \in V_r$. A Gaussian kernel function has been used as the smoothing kernel function $\Phi(\Delta_1 + \Delta_2)$ of the distance in (3.3):

$$\Phi(\Delta_1 + \Delta_2) = \exp\left(-\frac{(\Delta_1 + \Delta_2)^2}{2\sigma^2}\right), \tag{3.4}$$

where parameter σ determines the span of the kernel function. The distance function $\Delta_1(\mathbf{w}_r, \mathbf{w}_p)$ is defined within the SOM units connected to u_p, i.e., within a primitive object. The simultaneous use of two variables in the kernel function makes it possible to adapt to scale differences in object segments, and to extract correctly the skeleton points of object parts of different size. The use of the smoothing function in (3.4) is justified by the fact that it represents the conditional probability that point $\mathbf{v}_p$ belongs to unit u_r in the current iteration step (Mulier and Cherkassky, 1995; Villmann et al., 1997). This probability is determined by two factors: the distance between points $\mathbf{v}_p$ and $\mathbf{w}_r$, provided $\mathbf{v}_p$ is connected with $\mathbf{w}_r$, and the distance between point $\mathbf{v}_i$ and the attraction point $\mathbf{t}_r$, provided it belongs to V_r.

STEP 5. *Checking the iteration condition.* The span of the kernel function in (3.3) is linearly decreased in this step by decreasing the values of σ in (3.4). The termination condition for iterative computation of MSG vertices in Steps 1–4 of the algorithm is the condition of unchanged position of vertices $w_1, \dots, w_K$ with respect to the previous iteration.

STEP 6. *Determination of multi-scale structural graph.* During this step, the connectivity between local structures determined in Step 1 is checked. The verification consists in determining the connectivity of all non-connected vertices between the local structures. During this step, the connectivity between vertices and elementary straight-line segments is also determined. If a vertex is connected to a line segment, then this vertex is added to the skeleton and a new SOM unit with its local connectivity attributes is inserted into the list of units. Such a vertex insertion provides more exact approximation of skeletons with crossing lines and other higher-order connections.

Further details of this algorithm, concerning mainly determination of connectivity between vertices, are given in the next sections.

3.3. Optimal aggregation of vertices in a MSG

Let us suppose that all vertices of MSG have been determined by the above-described method using a structured SOM. Now the problem consists in connecting the graph vertices in an optimal way and according to the actual image data. Let us consider a MSG graph that is a weighted graph without vertex attributes. The edge weights of the graph can represent the probabilities of the represent the skeleton segments of objects. It is natural to demand that the total probability of vertex connectivity be maximal. Let us assume that the MSG is composed of a spanning forest with respect to the vertices. The total probability of vertex connectivity, $P_c(A)$, in a spanning forest A is:

$$P_c(A) = \prod_{\substack{u_l, u_k \in A \\ \forall k \neq l}} P_c(u_k \wedge u_l), \tag{3.5}$$

where $P_c(u_k \wedge u_l)$ is the probability that the skeleton vertices u_l and u_k are connected in the MSG. Since the probability measure is non-negative, the maximization of the total probability in (3.5) can be made according to a logarithm function. The solution to this problem is equivalent to the combinatorial solution of the maximum weight forest problem in the graph theory (Korte and Vygen, 2000). On the other hand, it can be reduced to the solution of the minimum spanning tree (MST) problem by modifying the graph weights. Two basic combinatorial algorithms solving the MST problem are known: Kruskal's algorithm and Prim's algorithm (Korte and Vygen, 2000). Prim's algorithm in its most general form generates the MST by adding each time a new vertex with the maximal probability of connectivity over all the vertices already contained in the MST graph. The initialization of the MST is made by finding a pair of vertices with the maximal probability of connection between them. It is proven that Prim's algorithm works correctly and its computational complexity is $O(K^2)$, where K is the total number of vertices.

At this point, the problem consists in determining the connectivity probability, based on the image data and the underlying model. The Bayesian paradigm is a suitable approach in this case. It involves two components: prior probability of connection between vertices u_k and u_l, $P(u_k \wedge u_l)$, and the probability of image data in the neighborhood of vertices provided the vertices are connected. Instead of considering a joint conditional distribution for all data points, a random variable, called the *connectivity feature* x, is defined. This feature relates to the connectivity between any two vertices u_k and u_l, and must have identical conditional distributions $P(x/u_k \wedge u_l)$ for any pair of vertices. The resulting logarithm of the posterior probability $P(u_k \wedge u_l/x)$ of vertex connectivity, i.e. after applying the Bayes rule, becomes:

$$\ln P(u_k \wedge u_l/x) = \ln(P(u_k \wedge u_l) \cdot P(x/u_k \wedge u_l)) - \ln(P(x)), \tag{3.6}$$

where $P(x)$ is the marginal probability of x having a uniform distribution. This is quite natural assumption, which matches well the empirical marginal distribution of x. Since a uniform distribution is assumed for $P(x)$, the value of $P(x)$ will be identical for all vertex pairs (u_l, u_k), and the term $\ln(P(x))$ can be ignored. The prior probability $P(u_k \wedge u_l)$ can be determined based on the actual relative positions of vertices without taking into account the image data. For in determining prior probabilities provided the two vertices, u_l and u_k, are adjacent vertices. In this case, the exponential probability density function is a fairly good approximation for the distribution of $d(u_l, u_k)$ taken as the distribution law for the prior probability, $P(u_k \wedge u_l)$, in (3.6). With regard to the connectivity feature x, the averaged deviation of intensity within the segment trapezoid $Z(u_k, u_l)$ from its mean value (Fig. 2a) will satisfy the relevance condition. In the context of the model for property map from Section 2.1, it follows a normal distribution $\mathrm{N}(0, \lambda^2/M)$ provided the vertices are connected, where M is the total number of points involved in the averaging within trapezoid $Z_G(u_k, u_l)$. Parameter λ is the same as in (2.1). The mean value intensity of the property map is evaluated as the average intensity value within structuring elements with vertex scales ρ_k and ρ_l for u_k and u_l.

Taking the above into account, the basic step of Prim's algorithm will involve optimal joining of the current vertex u_k to that vertex, $u_l \in V(A^s)$ from the intermediate spanning tree A^s at step s, which minimizes the following value:

$$\Delta(u_k, u_l) = (d(u_k, u_l) - a) + \alpha \cdot \delta^2(u_k, u_l), \tag{3.7}$$

where a is the closest possible distance between the adjacent vertices, $\delta^2(u_k, u_l)$ is the squared deviation of intensity within the segment trapezoid $Z(u_k, u_l)$ (see Fig. 2a), and α is a constant coefficient calculated based on the parameters of two distribution laws: exponential for prior probability and normal for conditional probability. Factor α determines the relative influence of the second term corresponding to the posterior probability with respect to the first term, which characterizes the prior probability. The value of $\Delta(u_k, u_l)$ is non-negative since $d(u_l, u_k)$ is always greater than a.

3.4. Determination of vertex connectivity

Application of Prim's algorithm for the aggregation of vertices in a MSG provides only a method for combining extracted vertices in a graph structure without final and explicit determination of their connectivity. If two vertices are connected, the graph of the MSG contains an edge between these vertices. One possible solution to the connectivity problem would be the statistical hypothesis testing based on the value of connectivity feature x (see Section 3.3) with respect to current vertices u_k and u_l. If x is less than a threshold, then the vertices are connected. However, such a simple test of connectivity between any two vertices may not work properly because of the shape sparseness and present local discontinuities in object shapes.

In order to circumvent this limitation of the classical statistical hypothesis testing, a decision has to be made by using a context-dependent connectivity test. The context-dependent test is based on the Markov random chain model of vertices belonging to the same generating line and the Bayesian principle of a decision-making process. Such a test is effective only when the connectivity for be done at the stage of vertex aggregation in a MSG by comparing the current value of x with a threshold. It is supposed that the prior probability $P(u_k \wedge u_l)$ of connectivity between vertices u_k and u_l is given or can be easily determined by the model data and the actual positions of vertices. The posterior probabilities are determined based on the object points actually between the vertices u_k and u_l.

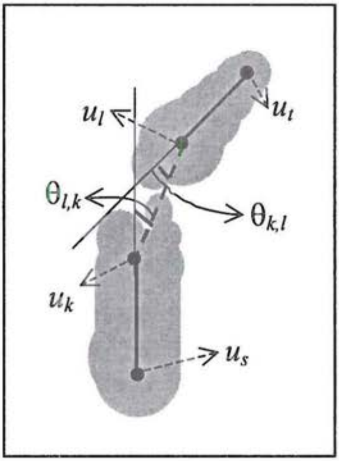

Figure 6. Example of an object part, which contains 4 skeleton vertices, u_k, u_l, u_s, and u_t, with a discontinuity at the middle. The connectivity between vertex pairs (u_k, u_s) and (u_l, u_t) is established through a context-free test of local connectivity, whereas the connectivity between vertices u_k, and u_l can be established based on the Bayesian decision-making process.

The adopted method for the determination of prior probabilities using the Markov chain model (see Section 2.4) is illustrated on the example of Fig. 6. This figure shows the case when a discontinuity occurs between two straight-line segments that connect vertices of the same skeleton line. The prior probability of connectivity $P(u_k \wedge u_l)$ can be evaluated based on relative angular positions of vertices u_k and u_l with respect to straight-line segments $[u_k, u_s]$ and $[u_l, u_t]$. The connectivity between vertices u_k and u_s or vertices u_l and u_t has been established during aggregation of vertices in a MSG. The Markov chain model provides the two probabilities for relative positions of vertices u_k and u_l as functions of the slope of segment $[u_k, u_l]$ relative to segments $[u_k, u_s]$ and $[u_l, u_t]$. Since the relative positions of both vertices contribute to the connectivity of u_k and u_l, the prior probability of their connectivity will be:

where $P(\theta_{k,l})$ and $P(\theta_{l,k})$ are the probabilities of slope values $\theta_{k,l}$ and $\theta_{l,k}$ of the segment $[u_k, u_l]$ with respect to the preceding straight-line segments $[u_k, u_s]$ and $[u_l, u_t]$. The probability $P(\theta_{k,l})$ and $P(\theta_{l,k})$ are given in the Markov chain model of generating lines. Here, it was assumed that the maximal rank of vertices $q = 2$, i.e. vertices u_t, u_s, u_l and u_k belong to one generating line without bifurcations.

The connectivity feature x defined as a function of data in trapezoid $Z(u_k, u_l)$ is used again (see Section 3.3) to determine the posterior probability of x, regardless of whether the vertices are connected or not. The posterior probability of connected vertices u_k and u_l is:

$$P(u_k \wedge u_l/x) = \frac{P(u_k \wedge u_l) \cdot P(x/u_k \wedge u_l)}{P(x)}, \tag{3.8}$$

where $P(u_k \wedge u_l)$ is the prior probability of connectivity, $P(x/u_k \wedge u_l)$ is the conditional probability of the random connectivity feature x, provided the vertices u_k and u_l are connected. After adopting a limit probability, p_{lim}, in the connectivity test by a probability thresholding, i.e., $P(u_k \wedge u_l/x) \geq p_{\text{lim}}$, the connectivity test can be written in an explicit form (Palenichka et al., 1998):

$$\frac{P(x/u_k \wedge u_l)}{P(x/u_k \,\|\, u_l)} \geq \frac{p_{\text{lim}}}{(1 - p_{\text{lim}})} \cdot \frac{P(u_k \,\|\, u_l)}{P(u_k \wedge u_l)}, \tag{3.9}$$

where $P(u_k \,\|\, u_l) = 1 - P(u_k \wedge u_l)$ is the prior probability of no connection between vertices u_k and u_l, and $P(x/u_k \,\|\, u_l)$ is the conditional probability of the connectivity feature x provided vertices u_k and u_l are not connected. The connectivity test by (3.9) guarantees that the error due to a wrong connection of vertices will be less than $(1 - p_{\text{lim}})$.

It is supposed that the variable x follows a Gaussian distribution law $\mathrm{N}(0, \lambda^2/M)$ provided the vertices are connected, where M is the total number of points within trapezoid $Z(u_k, u_l)$, λ^2/M is the variance of x. Similarly, a Gaussian probability distribution density $\mathrm{N}(h, \lambda^2/M)$ will be valid if the vertices are disconnected. Here, it is assumed that the background intensity (intensity of objects belonging to other classes) has the same variance λ^2 as in the model of property map in Section 2.1. Parameter h, the so-called object-to-background contrast, is estimated as the mean value of the intensity difference between object points and background points. In the case of a binary image $\eta(i, j)$ obtained after the object segmentation it represents an estimate for the shape sparseness π in image modeling by (2.1). Then, the decision about the vertex connectivity is accepted if the following inequality is valid:

$$x \leq \frac{h}{2} - \frac{\lambda^2}{h \cdot M} \cdot \ln(\Psi(u_k, u_l)), \tag{3.10}$$

where $\Psi(u_k, u_l)$ is the right-hand side of (3.9). The decision according to (3.10) takes into account the ratio of prior probabilities in such a way that the threshconnection is greater than 0.5. The undefined parameter, the variance of object points λ^2 in the property map, and the contrast h are supposed to be constant over the image plane and have to be estimated during the learning phase (see the right side of Fig. 3) by pointing on the object and background sub-regions and computing the local density variances and local contrasts.

In the absence of prior probabilities, equal values of probabilities, $P(u_k \parallel u_l) = P(u_k \wedge u_l)$, have to be used in (3.9). This assumption does not require a correlation model for the vertices of generating lines, as is the case with a Markov random chain. Nevertheless, it tolerates a certain amount of shape sparseness between the vertices due to the low threshold value in the right-hand side of (3.10), since the contrast h is proportional to the shape sparseness.

4. Experimental results

The algorithm of SOM-based skeletonization was tested on both satellite images of water basins and synthetic images in order to investigate its ability to extract skeletons of different types of objects. The main goal of testing the algorithm on synthetic images was the numerical evaluation of its performance with respect to known skeletal shapes. For this purpose, a multi-scale morphological shape model (see Section 2) was used to generate objects of interest based on object skeletons in the form of connected straight-line segments of generating lines. The approach of model skeletons offers reference sets of points, which are later compared with the skeletons obtained using the proposed skeletal shape extraction algorithms. Noise has been added to the objects obtained from the reference skeletons.

A convenient measure of the skeletonization error is the symmetric Hausdorff distance between two skeleton sets G_1 and G_2 (Huttenlocher et al., 1993):

$$H(G_1, G_2) = \max\{h(G_1, G_2), h(G_2, G_1)\},$$

where $h(G_1, G_2)$ is the directed Hausdorff distance from a set of points G_1 to a set of points G_2 defined as follows:

$$h(G_1, G_2) = \max_{a \in G_1} \min_{b \in G_2} \{d(a, b)\},$$

where $d(a, b)$ is the Euclidean distance between points a and b. This skeleton distance measure is stricter than the normalized Euclidean distortion used for skeleton comparison in the SOM-based algorithm using MST (Singh, Cherkassky and Papanikopoulos, 2000).

The Hausdorff distance was calculated as a function of noise level for two methods: the proposed scale-based algorithm and the SOM-based algorithm using MST (Singh, Cherkassky and Papanikopoulos, 2000). One such synthetic image used in the experiments is shown in Fig. 7 and a plot of the evaluated skeletonization error computed as Hausdorff distance for this image is given in

The MST-based algorithm gave poor results for images with multi-scale objects, such as the one in Fig. 7a. The smooth evolution of skeletons based on the distance between SOM units fails to provide good results when the two specified thresholds (for merging or insertion of vertices) do not depend on the local scale of objects (Singh, Cherkassky and Papanikopoulos, 2000). Both algorithms showed weak dependence on shape sparseness when the distribution of noisy pixels was uniform.

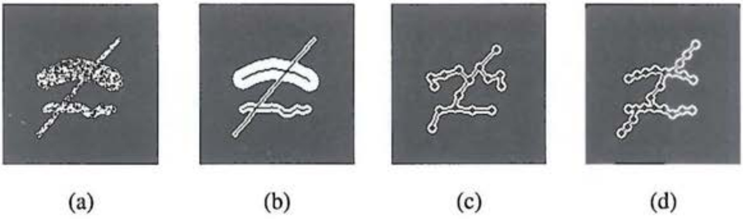

Figure 7. Example of results of two skeletonization algorithms: (a) – initial noisy image with 40% of noisy pixels; (b) – model skeletons of objects in the initial image; (c) – result of skeletonization with SOM-based algorithm using MST; (d) – result of skeletonization by the proposed algorithm.

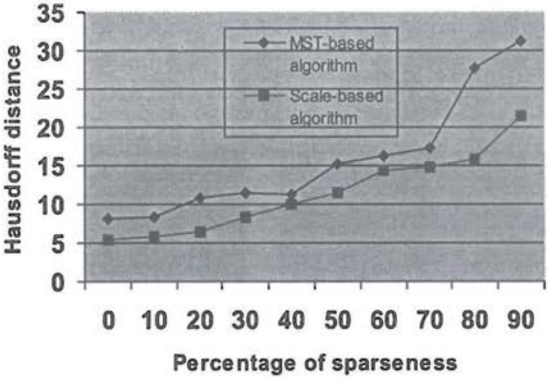

Figure 8. Hausdorff distance for the image shown in Fig. 7.

Another method for evaluating the numerical performance of a skeletonization algorithm is to measure skeleton distortion with respect to the skeleton obtained from an initial image without added noise (Singh, Cherkassky and Papanikopoulos, 2000). However, this is a relative measure of performance, which characterizes only the robustness of a skeletonization algorithm against

An example of an image fragment with a sparse shape of a river basin obtained from Landsat-7 imagery after segmentation is shown in Fig. 9a. The segmentation of the river image was made in two stages: property map extraction using conventional SOM and binarization by comparing the property value in each point with two constant thresholds. The result of SOM-based skeletonization of this image is given in Fig. 9b. Other tests of the developed algorithm were performed with a view to assessing the accuracy of shape reconstruction from MSG (see Fig. 10).

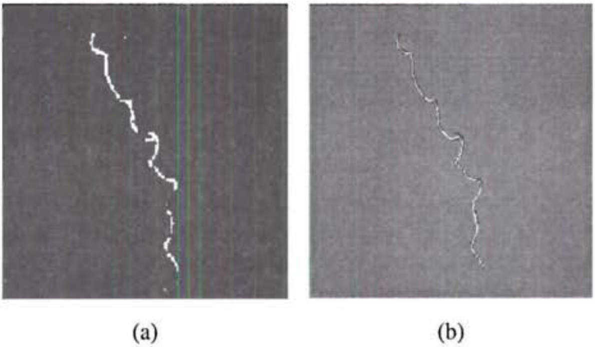

Figure 9. Image segment with sparse shape after the property map extraction and binarization (a) and example of skeletal shape extraction using the structured SOM approach (b). The skeleton pixels are shown in white superimposed on object points in black.

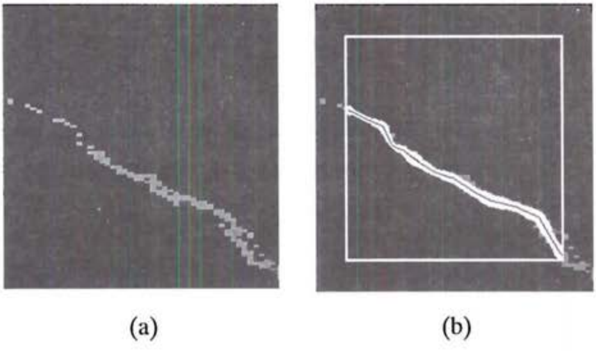

Figure 10. A river basin image (a) and sparse shape reconstruction using the scale-based algorithm of skeletonization (b).

Results of shape extraction from a river basin image by the structured SOM, and structured SOM for shape description, are depicted in Fig. 11. It can be seen that the proposed SOM-based algorithm copes well with significant shape discontinuities (see upper left fragment in Fig. 11a) by applying Bayesian decision-making process. Fig. 12 illustrates a water basin segment of a river with discontinuities and occlusions, and shows the results of skeletal shape extraction.

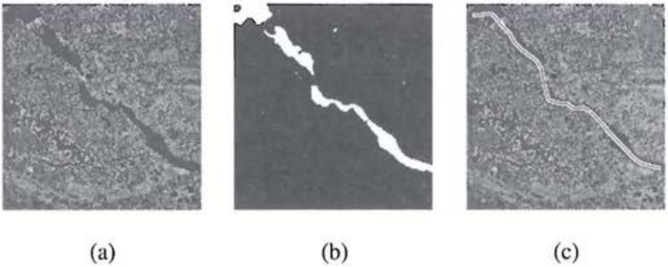

Figure 11. Skeletal shape extraction for a water basin image: (a) – property map extracted by using SOM-based clustering; (b) – binary shape map as the result of binarization of the property map; (c) – extracted skeletal shape of the river.

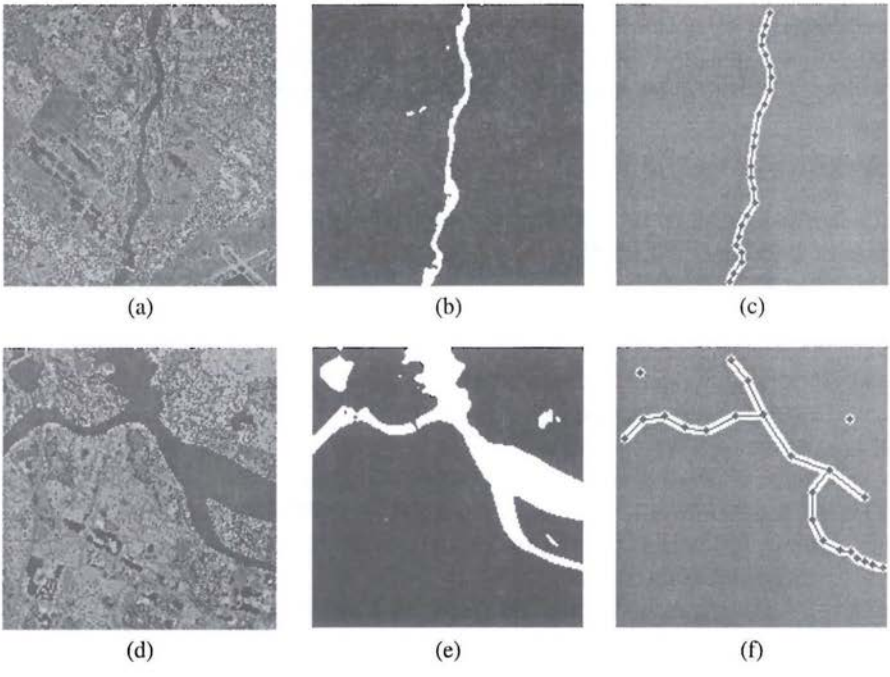

Figure 12. Property maps (a, d), binary maps (b, e), and model-based skeletonization

5. Conclusions

A method for multi-scale structural description and model-based extraction of the skeletal shape of objects by using structured self-organizing maps was developed and tested on synthetic images and Landsat-7 satellite images. Comparative testing results has confirmed the advantages of the structured SOM approach to modeling and skeletonization of objects with sparse shape over the classical thinning algorithms. A distinctive feature of the proposed structured SOM approach is the introduction of such attributes as local scale and connectivity of SOM units, which are used besides the unit weights. Their determination is based on the underlying image model of shape representation of objects in the form of piecewise-linear generating (skeleton) lines and a system of vertex scales. Such a hierarchical multi-scale representation makes the extraction of skeletons more robust for objects with sparse shapes, discontinuities or occlusions, as well as provides the possibility of shape reconstruction from MSG.

The two main differences of the presented algorithm with respect to the SOM-based skeletonization using MST (Singh, Cherkassky and Papanikopoulos, 2000) consist in the utilization of the scale-based updating phase of SOM units and the realization of a probabilistic model-based connectivity test while connecting adjacent SOM units, i.e., skeleton vertices. In contrast, the known algorithm gave ragged skeleton lines for elongated thick objects and false connections between skeleton vertices of disconnected object parts. These two advantages are implemented according to the proposed morphological model, which provided shape constraints helping to cope with the shape sparseness.

Acknowledgements

The authors wish to acknowledge the support of the Canadian Network of Centres of Excellence GEOIDE.

References

BLUM, N. and NAGEL, R.N. (1978) Shape description using weighted symmetric axis features. *Pattern Recognition*, **10**, 167–180.

CHEN, Y.S. and YU, Y.T. (1996) Thinning approaches for noisy digital patterns. *Pattern Recognition*, **29** (11), 1847–1862.

DATTA, A., PARUI, S.K. and CHAUDHURI, B.B. (1996) Skeletal shape extraction from dot patterns by shape-organization. *Proc. 13th Int. Conf. Pattern Recognition*, **4**, 80–84.

DER, R., BALZUWEIT, G. and HERRMANN, M. (1996) Building nonlinear data models with self-organizing maps. *Lecture Notes in Computer Science*, **1112**, Springer, 821–826.

HARALICK, R.M. and SHAPIRO, L.G. (1992) *Computer and Robot Vision*. Read-

HASTIE, T. and STUETZLE W. (1989) Principal curves. *Journal of the American Statistical Association*, **84** (406), 502–516.

HUTTENLOCHER, D.P., KLANDERMAN, G.A. and RUCKLIDGE, W.J. (1993) Comparing images using the Hausdorff distance. *IEEE Trans. Pattern Analysis and Machine Intelligence*, **15** (9), 850–863.

KEGL, B. *et al.* (2000) Learning and design of principal curves. *IEEE Trans. Pattern Analysis and Machine Intelligence*, **22** (3), 281–297.

KEMENY, J.G. and SNELL, J.L. (1976) *Finite Markov Chains*, Springer-Verlag, N.J.-Heidelberg-Berlin.

KOHONEN, T. *et al.* (1996) Engineering applications of self-organizing maps. *Proc. IEEE*, **84**, 1358–1384.

KORTE, B. and VYGEN, J. (2000) *Combinatorial Optimization: Theory and Algorithms.* Springer, Berlin.

MARAGOS, P. (1989) Pattern spectrum and multi-scale shape representation. *IEEE Trans. Pattern Anal. Mach. Intelligence*, **11** (7), 701–717.

MULIER, F. and CHERKASSKY, V. (1995) Statistical analysis of self-organization. *Neural Networks*, **8**, 717–727.

MULIER, F. and CHERKASSKY, V. (1995) Self-organization as an iterative kernel smoothing process. *Neural Computation*, **7** (6), 1165–1177.

OGNIEWICZ, R.L. and KUBLER, O. (1995) Hierarchical Voronoi skeletons. *Pattern Recognition*, **28** (3), 343–359.

PALENICHKA, R.M. *et al.* (1998) Model–based generation of structural statistical hypotheses for flaw detection in radiographic images. *Proc. Int. IEEE Workshop IMDSP'98*, Alpbach, Austria, 51–55.

PALENICHKA, R.M. and IVASENKO, I.B. (1999) Fast and robust parameter estimation in the polynomial regression model of images. *Proc. SPIE Nonlinear Image Processing X*, **3646**, 28–37.

PIZER, S.M., OLIVER, W.R. and BLOOMBERG, S.H. (1987) Hierarchical shape description via the multi-resolution symmetric axis transform. *IEEE Trans. Pattern Analysis and Machine Intelligence*, **9** (4), 505–511.

RITTER, H., MARTINEZ, T.M. and SCHULTEN, K. (1992) *Neural Computation and Self Organizing Maps.* Reading, MA, Addison-Wesley.

SINGH, R, CHERKASSKY, V. and PAPANIKOPOULOS, N. (2000) Self-organizing maps for the skeletonization of sparse shapes. *IEEE Trans. on Neural Networks*, **11** (1), 241–248.

VILLMANN, T., DER, R., HERRMANN, M. and MARTINEZ, T.M. (1997) Topology preservation in self-organizing feature maps: Exact definition and measurement. *IEEE Trans. on Neural Networks*, **8** (2), 256–266.

ZAREMBA, M.B. *et al.* (2000) Integration of self-organizing maps with spatial indexing for efficient processing of multi-dimensional data. *Proc. 8th ACM Int. Symposium GIS2000*, Washington, DC, 77–82.