Shape Extraction of Volumetric Images of Filamentous Bacteria Using a Topology Preserving Feature Map

U. Bhattacharya
CVPRU, Indian Statistical Institute
Calcutta, India
ujjwal@isical.ac.in

V. Liebscher
IBB, GSF
Neuherberg, Germany
liebsche@gsf.de

A. Datta
CSSC, Indian Statistical Institute
Calcutta, India
amitava@isical.ac.in

S.K. Parui
CVPRU, Indian Statistical Institute
Calcutta, India
swapan@isical.ac.in

K. Rodenacker
IBB, GSF
Neuherberg, Germany
liebsche@gsf.de

B.B. Chaudhuri
CVPRU, Indian Statistical Institute
Calcutta, India
bbc@isical.ac.in

Abstract

The study of the filamentous objects in waste water has recently gained momentum due to its significant effect in environmental pollution. This paper describes a neural network based shape extraction technique for volumetric images of these biofilm objects. These objects are elongated and the extracted shape is in the form of a 3D skeleton. The proposed algorithm is an extension of an earlier neural network based skeletonization technique for 2D objects. Unlike in Kohonen's feature map, the neural network topology of the model here is not fixed or predefined. The topology of the network evolves through learning depending on the input object. In fact, the topology varies from one part of the network to another depending on the object. This adaptive nature of the model makes it quite appropriate for shape extraction for which Kohonen's feature map is unsuitable. Also, the size of the network grows from small to large. The proposed technique implements massively parallel computation since otherwise processing of 3D image data becomes extremely slow. The 3D skeleton in the form of a graph is obtained which may be used for further analysis of these objects.

Keywords :

Topology Preserving Feature Map	Skeleton
3D Shape Extraction	Waste Water
Environmental Pollution Control	Filamentous Bacteria

1. Introduction

The waste water treatment plants in industry usually apply a biological process called activated-sludge process. Activated sludge consists of a mixed community of microorganisms that metabolize and transform organic and inorganic substances into environmentally acceptable forms. The typical microbiology of activated sludge consists of approximately 95 % bacteria. But these bacteria often start to bulk which causes the malfunctioning of the process. Bulking is a condition where the activated-sludge mass begins to grow in volume without a corresponding increase in weight.

During the last few years significant research work has been done in identifying the nature of these microbial species responsible for bulking and in developing an understanding of their population dynamics ([1]). An important aspect of the present research efforts involving such microbials is to develop some efficient tool for quantitative/qualitative analysis of these filamentous organisms. This needs the study of their morphology as well as their spatial distribution in the bio-mass. Microscopic examination of the sludge is necessary for the identification of such bacteria. In fact, the volumetric images of the sludge samples from industrial or municipal waste water can be obtained by Confocal Laser Scanning Microscope (CLSM). Due to enormously complex structures of the filamentous objects and presence of noise in such volumetric images, it is impossible for a micro-biologist to perform necessary analysis manually. The natural solution to this bottleneck is to use the advanced tools of Pattern Recognition and Image Processing.

Recently image processing techniques have been used for the analysis of such filamentous organisms ([2], [3]). Jones ([2]) used a tool, called 2-D Wavelet Packet Analysis, to obtain a unique index of branching morphology of filamentous objects in fungal colonies. Adiga and Chaudhuri ([3]) proposed simple image processing tools based on

heuristics and conventional methods for automatic quantitative evaluation of the filamentous bacteria responsible for industrial sludge bulking. In this paper, we describe a method for a description of shapes of the filamentous bacteria from their volumetric images. This shape description is achieved by transforming the 3D object into a skeleton consisting of line segments. The skeleton contains the essential structure of the bacteria. Skeletons are useful for geometric as well as topological analysis and classification of the shape of these bacteria.

Recent articles ([4]-[7]) are available where ANN model are used for shape extraction of digital objects. In these studies, only 2D objects were considered. In the present paper, we consider a modified version of the Neural-Gas Network model [8] as described in [7] and extend it suitably for application in the 3D images of filamentous objects.

2. Materials and the Model

2.1. Acquisition of 3D images by CLSM

For bulky objects like biofilms, CLSM technique is necessary to capture image data in 3D space ([9]). CLSM offers the third (depth) dimension by a technique which implements optical sectioning of thick samples without cutting physical sections. Using a laser light source, a light spot, with limited diffraction, is produced within the sample and the re-emitted light is imaged by the microscope lens. The effect of optical sectioning is achieved by application of a confocal filter. Only the latest developments of microscope lenses which are especially designed for confocal applications can produce high quality 3D images of depth resolution of the order of 200 μm.

Volumetric image data are obtained by considering a sequence of optical sections as discussed above. The 2D images of these sections are stacked one above the other and thus the 3D information is obtained. Therefore the axial distance of the individual sections is important especially on the background of their further use for quantitative analysis.

2.2. Skeleton - A Shape Descriptor

One of the important morphological descriptors of the shape of a digital object is its skeleton. Skeletonization was introduced to describe the global properties of objects and to reduce the original image into a more compact representation. A basic method of skeletonization is thinning. There are numerous definitions of a skeleton and hence the thinning algorithms differ from each other in implementation and in performance. For a shape, the medial axis is the generally accepted definition of its skeleton and therefore, the output skeleton of an algorithm is often judged on the basis of how close it is to the medial axis of the input object. In recognition of elongated shaped objects, for example, biofilms, character patterns, chromosome patterns etc., such skeletons are often useful.

In the following we describe a method for skeletonization of biofilm images which is an extension of the method described in [7] to 3D objects. Two major problems with the conventional methods for skeletonization are noise sensitivity and rotation dependency. Most of the existing algorithms are sensitive to boundary and interior noise. Also, they are mostly rotation dependent, particularly if the angle of rotation is not a multiple of 90°. On the other hand, the neural network based techniques are highly robust in terms of boundary noise as well as interior noise. Such a method is efficient in terms of invariance under arbitrary rotations and data reduction. Moreover, unlike the conventional algorithms it is grid independent. Also, its output skeleton is a satisfactory approximation of the medial axis ([6]).

2.3. Self-Organizing Feature Map

Kohonen's self-organizing feature map (SOFM) ([10]) uses a network of fixed topology. SOFM is composed of an array A (usually 1- or 2-dimensional) of processors (neurons) receiving input signals from a feature space V to be mapped onto A. Every processor in the net stores a weight vector and the dimension of the weight vectors is the same as that of the space V. These weight vectors are intialized by random values. The initial map is iteratively adapted on the basis of input vectors from the feature space.

Denote the set of processors by $\{\pi_1, \pi_2, \ldots, \pi_n\}$. The *neighbourhood* N_i of a processor π_i is $\{\pi_p | \pi_p$ is connected to $\pi_i\}$ which includes π_i. Let the weight vector for the processor π_i be $W_i(t) = (w_{i1}(t), w_{i2}(t), \ldots, w_{im}(t))$ at time instance t. The starting weight vectors $W_i(0)$ are chosen at random. Suppose the set of input vectors is $S = \{P_1, P_2, \ldots, P_N\}$ where the dimension of each P_j is m. The weight vectors are updated according to the following rule.

Suppose, at iteration or time instance t, P_j is presented to the network. All the processors compete and let $W_k(t)$ be the nearest weight vector to P_j. That is,

$$|W_k(t) - P_j| = \min_i |W_i(t) - P_j| \qquad (1)$$

Then, the weight vectors of the processors within the neighbourhood of π_k are updated as follows ([10]):

$$W_p(t+1) = W_p(t) + \alpha(t)[P_j - W_p(t)] \qquad (2)$$

where $\pi_p \in N_k$ and $\alpha(t)$ $(0 < \alpha(t) < 1)$ is the gain term which decreases with t.

The neighbourhood topology in Kohonen's network is fixed and does not change during learning. Such a network of fixed neighbourhood topology does not work well

in some situations ([11, 12]). This is because during the weight update process, the weight vectors lying in zero-density areas may be affected by input vectors from the surrounding parts of the nonzero distribution. This poses problems in skeletonization tasks because the resulting network does not give the required vector skeleton ([7]).

2.4. Dynamic Network Model

To overcome the limitations of Kohonen's feature map, it is necessary to implement a dynamic change in network topology to fit it into the skeletonization task. Different dynamic versions of the Kohonen's model are reported by several authors in different contexts ([4], [12]-[14]). In the present application we have considered an extension of the Topology-Adaptive and Self-Organizing Neural Network (TASONN) model ([7]) to 3D input space.

In TASONN model, a dynamically defined neighbourhood relationship is used. In this case, it is initially started with a small number of processors without any predefined links between them and the resulting topology is completely data driven. The network grows in size by means of a certain processor insertion mechanism. During the learning process, the processors create/adapt their neighbourhoods dynamically, by means of connection building, on the basis of the input. The model enables the network to learn the weight vectors as well as its topology from the input vectors in an unsupervised manner. The nature of the learnt topology may vary from one part of the network to another depending on the shape of the input making the network a faithful representation of the shape.

In the next section, we will describe the extended version of TASONN model as it is applied to volumetric images of biofilm objects.

3. The Methodology

3.1. Segmentation

Segmentation of the filamentous bacterial structures from the image of the sludge sample is very important for further processing leading to skeletonization of labelled connected components of these microbials. This is necessary because the input to the self-organizing feature map technique is the set of voxel coordinates forming a connected component. The 3D image is segmented by interactively selected (using histogram of the grey levels) threshold. Then it is cleaned using morphological opening. This cleaned image is then used to assign unique label numbers to all the 3D connected components using 6-connected neighbourhood.

3.2. Skeletonization of a Component

Since the filamentous objects, which we consider in this paper, can have widely different 3D shapes, no a priori neighbourhood is assumed for the processors. Instead, we start with a small initial list of n processors $[\pi_1, \pi_2, \ldots, \pi_n]$ where no prior connection between these processors is assumed. The connections or links will be established at a latter stage only. Here, the input feature vectors are the 3-dimensional co-ordinates of the object voxels of the image where $m = 3$. Let N be the number of object voxels and so $S = \{P_1, P_2, \ldots, P_N\}$ is the set of these voxels where $P_j = (x_j, y_j, z_j)$. The initial weight vectors of π_i, say, $(w_{i1}(0), w_{i2}(0), w_{i3}(0))$ are chosen randomly inside the minimal box containing the 3D object. The weight vectors of the processors π_i are updated iteratively on the basis of the voxels in S. Each presentation of an object voxel to the list of processor is called an iteration. Suppose, the voxel P_j is presented at the t-th iteration. Let $dist(P_j, W_k(t)) = \min_i[dist(P_j, W_i(t))]$ and $dist(P_j, W_l(t)) = \min_{i(\neq k)}[dist(P_j, W_i(t))]$. Thus π_k and π_l are the first two nearest processors to the object voxel P_j which modifies the two weight vectors $W_k(t)$ and $W_l(t)$ according to Eq. (2).

If this process of modification of weights is continued, the weights tend to approximate the shape of the object in an orderly fashion. The limiting weight vectors define the ordering. One presentation each of all the voxels in S makes one *sweep* consisting of N iterations. After one sweep is completed, the iterative process for the next sweep starts again from P_1 through P_N. Several sweeps make one *phase*. One phase is completed when the weight vectors of the current set of processors converge, that is, when

$$|W_i(t) - W_i(t')| < \varepsilon \qquad \forall i \tag{3}$$

where t and t' are the iteration numbers at the end of two consecutive sweeps and ε is a predetermined small positive quantity.

Only after a phase T is completed, a list L of pairs of processors is computed. For each object voxel P_j, the two nearest processors are determined. If π_k and π_l are the first two nearest processors of a voxel P_j, then the pair (k, l) is added to this list and we say that a link is established between these two processors. Each entry in this list also contains another number, $N(k, l)$ which is the number of voxels P_j for which π_k and π_l are the first two nearest processors. Thus one sweep through the set of all object voxels is needed to compute this list.

After the list is computed, we search for the maximum value of $N(k, l)$. Let $\mathcal{N} = \max_{(k,l)\in L} N(k, l)$. If

$$\mathcal{N} > \Theta, \tag{4}$$

(where $\Theta > 0$ is a predetermined whole number), then we select the pair of processors (r, s) with $N(r, s) = \mathcal{N}$. If there is more than one such pair with maximum value of $N(r, s)$, then we arbitrarily select one of them. A new processor is inserted between this selected pair of processors π_r and π_s which has the weight vector as $[W_r(T) + W_s(T)]/2$. Thus we have a new set of processors and the object voxels are presented again to this set of processors until the new weights converge (according to condition (3)). The links are computed afresh and a decision is taken whether one more processor has to be added to the present set of processors (according to condition (4)). If a new processor is to be inserted, the process is continued as above. Otherwise, we have the final set of processors and the final links.

The final list of links is used to construct the skeleton of the object. A pair of processors in the list means that they are neighbours of each other. In fact, this list of links can be looked upon as a graph with the weight vectors of the corresponding processors as the vertices and the links or neighbourhoods as the edges. Thus, the skeleton of the object is obtained in the form of a spatial graph providing a line-segment approximation to the medial axis of the input object. A 3D raster image of the skeleton can be generated in the following way. For each line segment describing the skeleton, those voxels (object or otherwise) are considered whose positions (the 3D lattice points) are within a distance of 0.5 from the line segment (a voxel is a unit cube). The set of all such voxels gives the raster skeleton. The proposed algorithm can now be stated briefly as follows:

The Algorithm

Step 1: [Initialization]
Initialize sweep no. t = 0, phase no. T=0;
Select the initial number (n) of processors;
Obtain the list of coordinates of the object voxels;
Initialize the weight vectors $W_i(t)$, $(i = 1, 2, \ldots, n)$ with random values.

Step 2: [Sweep]
Set sweep number $\tau = \tau + 1$.
For all object voxels $P_j, j = 1, 2, \ldots, N$, modify the weight vectors according to rule (2).

Step 3: If condition (3) is not satisfied, go to Step 2.

Step 4: [Phase]
Set phase number $T = T + 1$ and reset sweep number $\tau = 0$;
Compute the list of links among the present set of processors;
If condition (4) is satisfied insert a new processor and go to Step 2.

Step 5: Stop.

3.3. Initial list of processors

The size of a biofilm can be very large and also its shape may be very complicated with a large number of branches in different orientations. Such features of input objects slow down the convergence of the above algorithm greatly. To overcome this problem of slow convergence, we propose to divide the whole input object into several parts and select at least two or more initial processors from each of the smaller parts of the object. This approach to selecting an initial list of weight vectors makes the process several times faster.

3.4. Choice of parameter values

The speed of convergence of each phase during learning is largely controlled by the choice of the two parameters α and ε. Though a constant value of α $(0 < \alpha < 1)$ may be used throughout the learning process, it is better to make α decreasing with time for faster convergence of each phase. For convergence of the weight modification rule, the decreasing sequence of α should satisfy a set of conditions which have been discussed in [15] in relation to a similar learning phenomenon. In this present paper, we will consider $\alpha(t) = \frac{\alpha(0)}{1+(t/20)}$, where t denotes the sweep number in a phase and $\alpha(0)$ is the value of α at the beginning of a phase. In the simulation runs described in the next section we have considered $\alpha(0)$ lying between 10^{-2} and 10^{-3}.

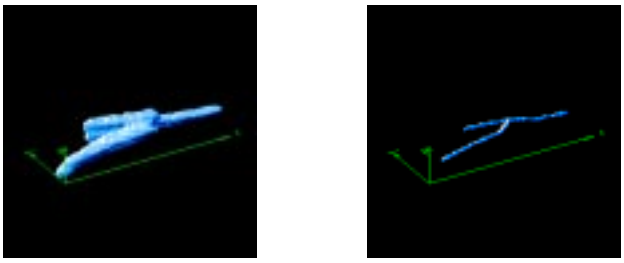
The value of ε determines the termination point of a phase during learning. A smaller value of ε means more accurate skeleton. It also means a phase will have a larger number of sweeps. In fact, the optimal method of selection of the value of ε is to use larger values during the earlier phases (particularly during skeletonization of the smaller parts of the object) to get a crude shape and smaller values during the final phases for a more accurate shape. In the simulation results, we have used $\varepsilon = 0.01$ during processing smaller parts of the object and $\varepsilon(T) = \max\left(0.0005, \frac{0.005}{1+(T/10)}\right)$, where T denotes the phase number during processing the whole object together.

In fact, the stopping criterion of the algorithm, as mentioned earlier in the description of the method, is based on the value of the parameter Θ. It also, in effect, determines the proximity of neighbouring/adjacent processors representing the skeleton of the object. The framing of a rule for the selection of a suitable value of Θ is not very straightforward. Instead, one may use trial and error method for the purpose.

4. Simulation Results

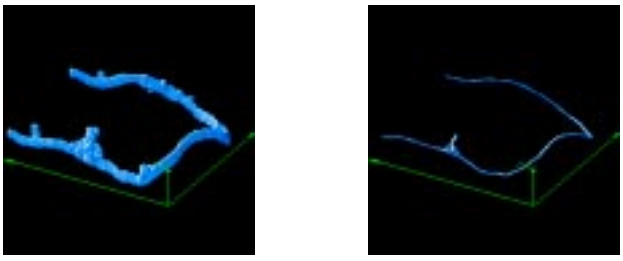
A number of 3D images of biofilm objects of different sizes and shapes have been used to test the proposed skeletonization method. The pseudo surface display of two such

filament objects are shown in Figs.$1(a)$ and $2(a)$. Similar displays of the raster images of their output skeletons are shown in Figs.$1(b)$ and $2(b)$ respectively.



(a) (b)

Figure 1. Pseudo surface of 3D images of one filamentous bacteria and its skeleton

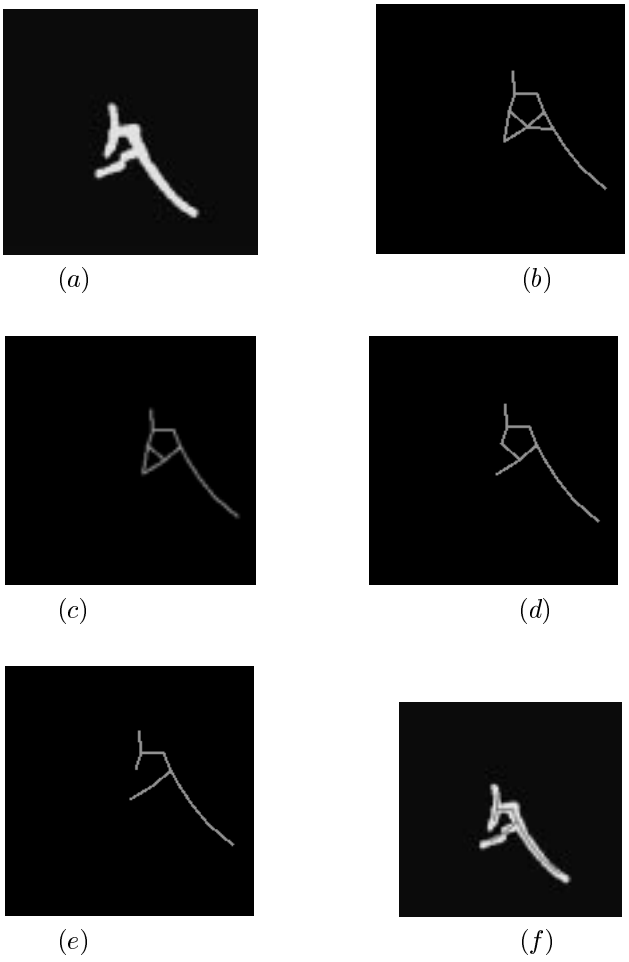


(a) (b)

Figure 2. Pseudo surface of 3D images of another filamentous bacteria and its skeleton

The present algorithm generates a list of links which is in turn used to get the skeleton of the digital object. Usually, this list contains a few spurious links along with the genuine ones. After studying several lists of links corresponding to different objects, we reached at the conclusion that for such spurious links the values of $N(r,s)$ are always significantly smaller compared to those of the genuine links. Thus such spurious links may be easily avoided in the final skeleton by setting some threshold value ($\mathcal{L}$) of $N(r,s)$ on the basis of which a pair of processors π_r and π_s in the list of links may be joined to obtain the final skeleton.

We select one biofilm object (its projection on the horizontal plane is shown in Figure $3(a)$) to demonstrate how this approach for the removal of spurious links works. In Figures $3(b) - 3(e)$, similar projections of the skeletons of this object is shown when the values of $\mathcal{L}$ are 1, 50, 70 and 90 respectively. The values of $N(r,s)$ for all the genuine links corresponding to this skeleton are more than 200. In Figure $3(f)$ the projection of the object is superimposed with the projection of the skeleton formed by the genuine links only.



(a) (b) (c) (d) (e) (f)

Figure 3. Proper selection of the value of $\mathcal{L}$ helps to obtain the required skeleton. In (b) $\mathcal{L} = 1$, (c) $\mathcal{L} = 50$, (d) $\mathcal{L} = 70$ and (e) $\mathcal{L} = 90$. In (f) the skeleton (when $\mathcal{L} = 90$) and the object are superimposed.

5. Conclusions

We have proposed an extension of the skeletonization technique using TASONN model to 3D digital objects in the context of shape analysis of volumetric images of biofilm objects. There exist several conventional techniques which can as well be used to find the skeleton of 3D digital objects. But the proposed neural network based technique of skeletonization has certain advantages over the conventional ones. One of them is noise immunity of the technique for which the reason is as follows. The resulting vector skeleton here consists of the final weight vectors and the links between them. The weight vectors converge to the

centre of gravity of the 3D Voronoi regions generated by the weight vectors and these Voronoi regions are not greatly affected by noise. On the other hand, most of the existing conventional methods for skeletonization use a rigid definition of connectedness of the object which causes noise sensitivity.

We have discussed the present technique of skeletonization as a tool for the analysis of 3D images of Biofilm objects. Recently the study of these objects has gained importance due to environmental reasons. This technique based on a neural network model generates a graph which is used to form the raster skeleton of the image. This skeletonal image can be used for better visualization and for further analysis of the 3D object.

Acknowledgements: The authors like to thank Prof. (Dr.) Winkler, IBB, GSF, Munich for his encouraging interests in the work and Dr. Hutzler, Inst. of Path., GSF, Munich for the data used in this paper. This work has been supported partly by International Buro, DLR, Germany, (INR 027) and ICMR, India under the Indo-German collaboration for Biomedical Image Processing.

References

[1] D. Jenkins, M.G. Richard and G.T. Daigger, *Manual on the causes and control of activated-sludge bulking and foaming*. Lewis Publishers, Ann Arbor, 1993.

[2] C.L. Jones, 2-D wavelet packet analysis of structural self-organization and morphogenic regulation in filamentous fungal colonies. In *Complex Systems Conference - "From local interaction to global phenomena"*, July 14-17, 1996, Albury - Wodonga, N.S.W., Australia.

[3] P.S.U. Adiga and B.B. Chaudhuri, Analysis of volumetric images of filamentous bacteria in industrial sludge. In *Proc. 14th Int. Conf. on Pattern Recognition*, Australia, 1998, pp.1735-1737.

[4] A. Datta, T. Pal and S.K. Parui, A modified self-organizing neural net for shape extraction. *Neurocomputing*. 1997: 14: 3-14.

[5] A. Datta and S.K. Parui, Skeletons from dot patterns : a neural network approach. *Pattern Recognition Letters* 1997: 18: 335-342.

[6] A. Datta and S.K. Parui, Shape extraction : a comparative study between neural network based and conventional techniques. *Neural Computation and Applications* 1998: 7: 343-355.

[7] A. Datta and S.K. Parui, A skeleton generating neural network : a prerequisite for character recognition. In *Proc. Int. Conf. on Computational Linguistics, Speech and Document Processing*, 1998, Indian Statistical Institute, Calcutta.

[8] T. Martinetz and K. Schulten, A Neural-Gas network that learns topologies. In *"Artificial Neural Networks"*. Eds. T. Kohonen, O. Simula and J. Kangas, Elsevier Science Publishers B.V. (North-Holland), 1991, pp. 397-402.

[9] M. Wagner, P. Hutzler and R. Amann, Three dimensional analysis of complex microbial communities by combining confocal laser scanning microscopy and fluroscence in situ hybridization. In *Digital Image Analysis of Microbes*, Eds. M.H.F. Wilkinson and F. Schut, John Wiley & Sons, Chichester, 1998.

[10] T. Kohonen, *Self-Organization and Associative Memory*, Springer-Verlag, 1989.

[11] J.A. Kangas, T. Kohonen and J. Laaksonen, Variants of self-organizing maps. *IEEE Trans. Neural Networks* 1990: 1: 93-99.

[12] D. Choi and S. Park, Self-creating and organizing neural networks. *IEEE Neural Networks* 1994: 5: 561-575.

[13] B. Fritzke, Let it grow - self-organizing feature maps with problem dependent cell structure. In Kohonen T et al. (eds.) *Artificial Neural Networks, vol.* 1, North-Holland, 1991.

[14] M. Sabourin and A. Mitiche, Modeling and classification of shape using a Kohonen associative memory with selective multiresolution. *Neural Networks* 1993: 6: 275-183.

[15] U. Bhattacharya and S.K. Parui, On the rate of convergence of perceptron learning. *Pattern Recognition Letters* 1995: 16: 491-497.