Quantifying structural heterogeneity in biofilms
by Wei Huang

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Computer Science
Montana State University

Abstract:
In biofilm engineering research, quantifying the biofilm physical structure is important in understanding structure/function relations. To achieve this objective, we use digital image analysis techniques to analyze digital biofilm images collected with confocal scanning laser,microscope (CSLM). A software package on Unix platform with source code written in C++ and Motif was developed. It has image segmentation tools and tools for extracting four quantitative structural parameters from the input biofilm images. The parameters are "2-D porosity," "average cluster size," "fractal dimension," and "texture entropy." The description of a structural feature of the image by each parameter agrees with human visual observation. An image reconstruction program is also written using an annealing algorithm. The program can be used to check whether this set of parameters captures all essential structural information of the original image.

QUANTIFYING STRUCTURAL HETEROGENEITY IN BIOFILMS

by

Wei Huang

A thesis submitted in partial fulfillment
of the requirements for the degree

of

Master of Science

in

Computer Science

MONTANA STATE UNIVERSITY-BOZEMAN
Bozeman, Montana

April 1996

APPROVAL

of a thesis submitted by

Wei Huang

This thesis has been read by each member of the thesis committee and has been found to be satisfactory regarding content, English usage, format, citations, bibliographic style, and consistency, and is ready for submission to the College of Graduate Studies.

Gary Harkin, Chair Gary J Harkin 4/17/96
(Signature) Date

Approved for the Department of Computer Science

Denbigh Starkey, Head J Denbigh Starkey 4/17/96
(Signature) Date

Approved for the College of Graduate Studies

Robert Brown, Dean RL Brown 5/4/96
(Signature) Date

STATEMENT OF PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for a master's degree at Montana State University-Bozeman, I agree that the Library shall make it available to borrowers under rules of the Library.

If I have indicated my intention to copyright this thesis by including a copyright notice page, copying is allowable only for scholarly purposes, consistent with "fair use" as prescribed in the U.S. Copyright Law. Requests for permission for extended quotation from or reproduction of this thesis in whole or in parts may be granted only by the copyright holder.

Signature Wei Huang

Date 4/17/96

ACKNOWLEDGMENTS

I would like to thank the Center for Biofilm Engineering, especially Prof. Zbigniew Lewandowski, and my graduate committee, especially Prof. Gary Harkin, for their support, guidance, and inspiration.

TABLE OF CONTENTS

LIST OF FIGURES

ABSTRACT

In biofilm engineering research, quantifying the biofilm physical structure is important in understanding structure/function relations. To achieve this objective, we use digital image analysis techniques to analyze digital biofilm images collected with confocal scanning laser microscope (CSLM). A software package on Unix platform with source code written in C++ and Motif was developed. It has image segmentation tools and tools for extracting four quantitative structural parameters from the input biofilm images. The parameters are "2-D porosity," "average cluster size," "fractal dimension," and "texture entropy." The description of a structural feature of the image by each parameter agrees with human visual observation. An image reconstruction program is also written using an annealing algorithm. The program can be used to check whether this set of parameters captures all essential structural information of the original image.

CHAPTER 1

INTRODUCTION

In recent studies biofilms have often been shown to be heterogenous, i.e., composed of voids, channels, and irregular clusters [1-7], instead of the simplistic uniform model. This raised a need of more study on biofilm physical structure. Biofilm structure study is also important in understanding structure/function relations. Some structure-related questions are: What are the characteristics of biofilm structures? How does this heterogenous structure evolve with time? How do the biofilm physiological processes affect physical structure? How do the physical structure variables relate to other variables such as the local mass transfer parameter and the fluid velocity? To answer these questions, quantifying biofilm structure is a necessary and essential step.

Various efforts have been directed to biofilm structure study. To visualize biofilm physical structure, traditional light microscopy [1], atomic force microscopy [2], and confocal scanning laser microscope (CSLM) [3,6], etc. have been used. Compared with other microscopies, CSLM has several advantages such as nondestructiveness and three-dimensional capability.

Although much visualization work on biofilm has been carried out, quantification of biofilm structure is in a state of insufficience. The descriptions of biofilm structure in most of the above studies are qualitative. Among a few quantitative studies, Zahid first used fractal dimension to describe biofilms [9], and Zhang studied density, porosity, specific surface area and mean pore radius of biofilm [10]. Both of them used experimental techniques such as microslicing in the quantification of biofilm structure. This has a serious drawback, in that the measurement is not efficient, is destructive, and cannot be combined with other in-situ experiments.

S.W. Hermanowicz's group at Berkeley studied a fractal dimension parameter of CSLM images of biofilms [23]. He computed fractal dimensions of images and investigated its variance with scales, depth, and flow direction. The approach, analyzing CSLM biofilm images with digital image analysis techniques in order to quantify biofilm structure, will also be our approach. Apparently, a single parameter of fractal dimension alone will not suffice for the quantification of biofilm structure. In our work we studied more parameters, each of which captures a unique feature of the biofilm structure, and developed a software package that performs the image analysis.

Our goal for quantifying biofilm structures is to obtain a set of parameters that contain enough structural information such that the reconstructed image, using the set of parameters, is visually similar to the original image.

In the following chapters, after a brief description of image collection and image segmentation steps in Chapter 2 , we describe in Chapter 3 a popular image modeling approach that we attempted. However the result shows that it is not very suitable for biofilm study. Then Chapter 4 presents other image processing approaches, each of which extract a certain structural parameter. The description of an image feature by each parameter agrees with human visual examination. Chapter 5 describes a program that can synthesize images with desired parameter values. We can use this kind of program to check whether a set of parameters can capture *all* essential features of an image. Chapter 6 describes some software design issues. Chapter 7 summarizes the work that has been done and provides a few suggestions for future work.

CHAPTER 2

IMAGE COLLECTION AND IMAGE SEGMENTATION

2.1 Image Collection

A transparent biofilm chemical reactor with biofilm on it is mounted on the CSLM. The CSLM is controlled by a software that can digitize and store images at one or more depths in the sample. The image files can be converted to 768×512 8-bit gray-scale TIFF format files.

2.2 Image Segmentation

To differentiate focused biofilm clusters from surrounding voids in a CSLM biofilm image, image segmentation needs be done. For biofilm pictures, the boundary between cluster and void is indistinct, and the most prominent difference between cluster and void is the difference of brightness. Therefore thresholding seems to be the best technique of segmentation.

Thresholding can be done interactively by instantly viewing thresholded results using various threshold values and choosing the best. The histogram of the image provided in the software can be used to select the best threshold value, if the histogram is bimodal. Usually, for a set of biofilm images in an experiment, CSLM setups are fixed, so threshold value should be the same for all images. In this situation, after interactively choosing the correct threshold, users can create a simple script file to batch-process a set of images. The script repeatedly uses a separate and independent executable file to process image files.

Although the software provides some image segmentation tools, it does not elaborate on this phase. The purpose of segmentation is to get a binary cluster/void image for the next-step's parameter extraction. Therefore image segmentation tools in other existing software can also be used to achieve this result.

CHAPTER 3

MARKOV RANDOM FIELD MODEL

In order to find a set of parameters that contain essential structural information and that can be used to reconstruct similar images, we studied an image modeling technique. In the image processing field, image modeling involves the construction of models or procedures for the specification of images. These models serve a dual role in that they can describe images that are observed and also can serve to generate synthetic images from the model parameters.

Markov random field (MRF) texture model is one of the well known image texture models and is among the most powerful stochastic texture models.

3.1 Basic Concepts

The brightness level at a point in an image is correlated with the brightness levels of neighboring points unless the image is simply random noise. Markov random field is a precise model describing this correlation.

Let $X(i,j)$ denote the brightness level at a point (i,j) on the $N\times N$ lattice L. Sometimes we simplify the labeling of the $X(i,j)$ to be $X(i)$, $i=1,2,\ldots,M$ where $M = N^2$.

Definition 1: Let L be a lattice with G levels. A *coloring of* L denoted $\boldsymbol{X}$ is a function from the points of L to the set $\{0,1,\ldots,G\text{-}1\}$. The notation $\boldsymbol{0}$ denotes the function that assigns each point of the lattice to 0.

Definition 2: The point j is said to be a neighbor of the point i if:

$$p(X(i)\,|\,X(1),X(2),\ldots,X(i\text{-}1),X(i+2),\ldots,X(m))$$

depends on $X(j)$. (In most applications, we assume the neighbors of a point are its physically close points.)

Definition 3: A *Markov random field* is a joint probability density on the set of all possible colorings $\boldsymbol{X}$ of the lattice L subject to the following conditions:

1) Positivity: $p(\boldsymbol{X}) > 0$ for all $\boldsymbol{X}$.

2) Markovianity: $p(X(i)\,|$ all points in the lattice but i)

= $p(X(i)\,|$ neighbors of i)

3) Homogeneity: $p(X(i)\,|$ neighbors of i) depends only on the configuration of neighbors and is translation invariant with respect to translation with the same neighborhood configuration).

The Hammersley-Clifford theorem and Besag's work [13] provide a formulation relating the probability of a point $X(i,j)$ having gray level k with the parameter determined by its neighbors, as the following shows.

The probability $p(X(i,j)=k|$neighbors of $(i,j))$ is binomial with parameter $\theta(T)$ and G-1, where G is the number of gray levels. The value of T is given in (1)-(4) for MRF models of various orders. The a and $b(m,n)$ are the parameters of the model and are 0 for all i larger than the order.

$$\theta = \frac{e^{T}}{1+e^{T}}$$

where a first-order model has the form for *T*:

(1) $$T = a + b(1,1)[X(i-1,j)+X(i+1,j)] + b(1,2)[X(i,j-1)+X(i,j+1)].$$

A second-order model has a *T* of the form:

(2) $$\begin{aligned} T = {} & a + b(1,1)[X(i-1,j)+X(i+1,j)] + b(1,2)[X(i,j-1)+X(i,j+1)] \\ & + b(2,1)[X(i-1,j-1)+X(i+1,j+1)] + b(2,2)\,[X(i-1,j+1)+X(i+1,j-1)]. \end{aligned}$$

A third-order model takes the form:

(3) $$\begin{aligned} T = {} & a + b(1,1)[X(i-1,j)+X(i+1,j)] + b(1,2)[X(i,j-1)+X(i,j+1)] \\ & + b(2,1)[X(i-1,j-1)+X(i+1,j+1)] + b(2,2)\,[X(i-1,j+1)+X(i+1,j-1)] \\ & + b(3,1)[X(i-2,j)+X(i+2,j)] + b(3,2)\,[X(i,j-2)+X(i,j+2)]. \end{aligned}$$

A fourth-order model is obtained by adding an additional term of the form

(4) $$\begin{aligned} & b(4,1)[X(i-1,j-2)+X(i-2,j-1)+X(i+1,j+2)+X(i+2,j+1)] \\ & + b(4,2)[X(i+1,j-2)+X(i+2,j-1)+X(i-1,j+2)+X(i-2,j+1)] \end{aligned}$$

to the form for the third-order *T*. Additional high-order terms can be obtained by extending the orders in a similar way.

For binary image case that we are interested in, we obtain the conditional probability:

$$p(X{=}x\,|\,neighbors) = \frac{e^{xT}}{1+e^{T}}$$

where x is one of the {0,1}.

3.2 Meanings of Model Parameters

Each parameter of the set $\{a, b(m,n)\}$ does not independently correspond to a certain image feature. In fact, the set of several parameters together determines an infinite number of images with some common features. The relationships between those common features and parameter values are indirect and so are usually not easy to decide.

We can specify some of these relationships as follows. If $b(1,1)$ and $b(1,2)$ are positive and a is negative, then the bigger $b(1,1)$ and $b(1,2)$, the bigger the clusters in an image. If $b(1,1)$ is not equal to $b(1,2)$, the image is anisotropic.

3.3 Compute Model Parameters from an Image

The method used to estimate the parameters $\{b(i,j)\}$ is the maximum likelihood estimation. Let $p(x|.)$ denote the conditional probability $p(X(i,j)=k|$neighbors of $(i,j))$. The usual log likelihood is given by:

$$A = \sum_{X} ln(p(X|.))$$

where the summation extends over all points of the lattice.

To maximize A, we must find the optimal solution of a multi-variable function. We used the “downhill simplex method” algorithm from the book “Numerical Recipes in C” [17]. For all of the sample images in Cross’s paper [12], our program gives correct parameter values.

3.4 Construct Images from Model Parameters

Theorem: Let $\boldsymbol{X}$ and $\boldsymbol{Y}$ be two colorings of the Markov random field lattice L. Then

$$\frac{p(\boldsymbol{Y})}{p(\boldsymbol{X})} = \prod_{i=1}^{M} \frac{p(X(i)=y(i)|X(1),X(2),...,X(i-1),Y(i+1),...,Y(N))}{p(X(i)=x(i)|X(1),X(2),...,X(i-1),Y(i+1),...,Y(N))}$$

Other mathematical theorems [12] along with the above theorem guarantee that the application of the algorithm in Fig. 1 will eventually result in a lattice with desired Markov field.

```
while not STABLE do
     begin
          choose sites X(1),X(2) with X(1) <> X(2);
          r:= P(Y)/P(X);
          If r >= 1
               then switch X(1), X(2)
               else
                    begin
                         u := uniform random on [0,1];
                         If r > u
                              then switch X(1), X(2)
                              else retain X
                    end
     end
```

Figure 1. Algorithm for image annealing with a set of Markov parameters.

The algorithm takes an arbitrary image that has the desired histogram and the set of desired parameters as input and then begins the annealing process. Initial configurations affect the rate at which equilibrium is reached.

We use an image with random pixel values as input. Experiments by other authors and our own showed that the annealing should take at least 10M attempted switches between pixels with different colors, where M is the size of image. Usually we use 100M attempted switches. The equilibrium is manifested when more attempts change the appearance of the image little. The measured parameter values of the output image are very close to the desired parameters, with difference usually less than 10%.

3.5 Results and Discussion

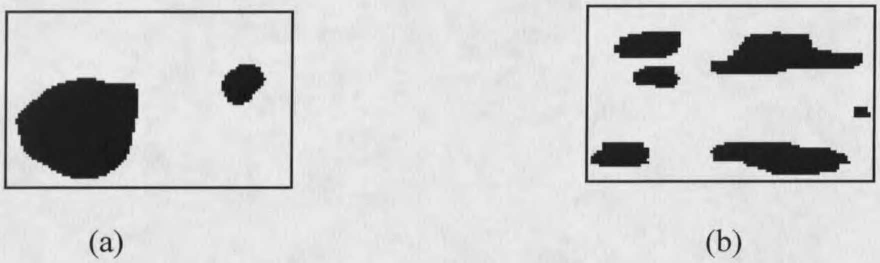


Figure 2. Markov image reconstruction. (a): an original image, (b): the generated image.

Fig. 2 shows original and generated pictures, using the first-order Markov method. The measured parameters for the original image are: $a = -22.269$, $b(1,1) = 18.809$, and $b(1,2) = 3.167$. The measured parameters for the generated image are: a=-24.433, $b(1,1) = 20.199$, and $b(1,2) =$ 3.597. We can see that the generated image grasps these properties of the original image: general sizes of clusters and directionality. (But note that horizontal directionalities of the two images have different causes: one is due to the arrangement of clusters and the other is due to the shapes of clusters.)

From the above figure we may speculate that if we use higher order parameters, we may achieve satisfactory results. However, from our tests, we found that higher order parameters behave quite unreliably and unpredictably. For example, given a set of parameters, we generate an image; measuring the image shows that the first order parameters are close to the original parameters, but the higher order ones differ from the original ones completely.

Besides the above drawback, from our tests and from the nature of the Markov method, we can see the following limitations of the technique for use in the quantification of biofilm structure. (1) Due to the homogeneous nature of the model, if, in the original image, there are a few very small clusters clustering in a subregion, in the generated image those very small clusters will be spread out through the whole image. (2) This method is standalone, that is, we cannot incorporate other parameters into this method. (3) There is no direct physical meaning for any one of the set of parameters. The parameters are dependent with each other.

In conclusion, we think the Markov model is not a good way for quantifying biofilm structures, although it does capture certain features of an image and it has the reconstruction ability.

CHAPTER 4

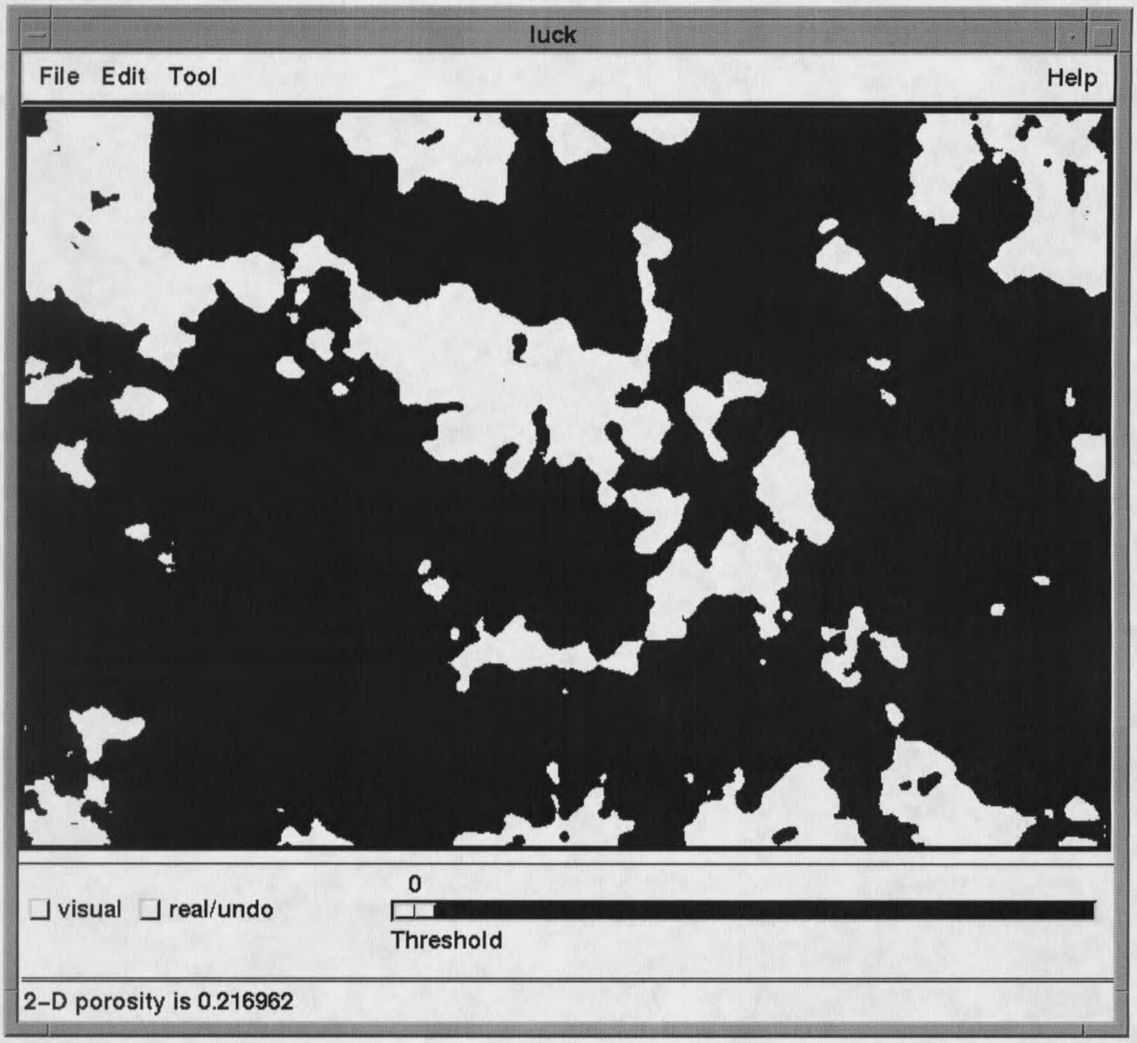
FOUR STRUCTURAL PARAMETERS

We developed four other parameters to quantify biofilm images. Two of these parameters, porosity and average size, have straightforward morphological meanings. They directly describe certain geometric features. The other two parameters, texture entropy and fractal dimension, are more abstract and describe image features at higher levels.

For each of these parameters, we tested with a few real biofilm images. The description of image features by each parameter agrees with human visual judgment. The software will be used by other engineering researchers in the CBE, and at that time, with extensive experimental data, those parameters will relate to the concrete research problems in biofilms.

4.1 Two-dimensional Porosity

Since biofilms often consist of voids and channels, there should be a parameter describing such porous effect, which porosity does. It is the ratio of void volume to the total volume for an object. Straightforwardly, we define 2-D porosity as the ratio of void area to total area for a cross section plane and use this parameter as an indicator of porousness. The 2-D porosity is closely related to 3-D porosity. If we have a set of images scanned at different depths above the same spot, the average of 2-D porosities of these planes differs from the 3-D porosity only by a constant factor. Fig. 3 shows a biofilm image with the computed 2-D porosity displayed as 0.217.



2-D porosity is 0.216962

Figure 3. Compute 2-D porosity of a thresholded biofilm image

4.2 Average Cluster Size

As we can see in Fig. 3, biofilms are usually very irregularly shaped. They are often interconnected and have filament-like components. Usual descriptions of sizes such as diameter are based on circle-like shapes and are not appropriate for many biofilms. To obtain a value that describes the average size of biofilm clusters, we defined a unique parameter. The definition was inspired by chemical diffusion processes. For each cluster pixel we compute its shortest distance to cluster/void borders and then average over all pixels to get an average distance. The average distance thus obtained is not only directly related to cluster size, but also suggests in average how much depth the exterior chemicals need to penetrate to reach the interior microorganisms. Therefore, the parameter is ready to be combined with other studies on biofilm such as the mass transfer coefficient study.

To get a quantitative sense of the average cluster size parameter, let us calculate it for an input of a circle with radius R in Fig. 4:

$$average\,size = \frac{1}{\pi R^2}\int_0^R 2\pi r(R-r)dr = \frac{R}{3}$$

It is not surprising that the average size is proportional to the radius. As a digression, the above calculation was also used in the verification of our software.

It seems computationally very complex to compute the average cluster size. The Euclidean Distance Mapping algorithm [15] greatly reduces computational burden of our problem.

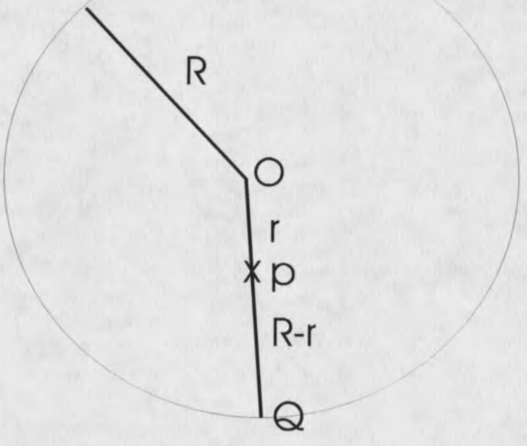


Figure 4. Compute average cluster size for a circular cluster. R is the radius. R-r is the shortest distance to the border from the point P.

4.2.1 Euclidean Distance Mapping

In two dimensional rectangular space, several distance metrics are defined, in which, metric

$$d_e((i,j),(h,k)) = \sqrt{(j-i)^2 + (k-h)^2}$$

is called Euclidean metric d_e . Obviously, we assume that (i,j) and (h,k) are two points on the two dimensional space.

Given a binary image with two set of pixels,

S = set of 1's, the objects

$\underline{S}$ = set of 0's, the background

a distance map L(S) is an image such that for each pixel $(i,j) \in S$, there is a corresponding pixel in L(S) where

$$L(i,j) = min(d[(i,j),\underline{S}])$$

i.e., each pixel in S has been assigned a label in L(S) that amounts to the distance to the nearest background $\underline{S}$. Obviously we can define a similar map $L(\underline{S})$ for the background. The computational procedure L: $S \rightarrow L(S)$ is called distance mapping.

There have been efficient algorithms for distance maps based on the metrics d_4 and other quasi-Euclidean metrics. The efficient algorithm for true Euclidean distance mapping did not occur until 1980 [15]. With this algorithm the seemingly computationally complex problem can be solved more easily.

The algorithm operates on the picture L, which is a two dimensional array with the elements

$$L(i,j) \quad 0 \leq i \leq M-1,\ 0 \leq j \leq N-1.$$

Each element is a two-element vector

$\boldsymbol{L} = (L_i , L_j)$, L_i , L_j being positive integers,

$\boldsymbol{L}(i,j) = (L_i(i,j), L_j(i,j)).$

The size of a vector $\boldsymbol{L}(i,j)$ is defined by

$$|\boldsymbol{L}(i,j)| = \sqrt{(L_i^2 + L_j^2)}.$$

The four-point sequential Euclidean distance mapping algorithm (4SED) is defined as follows.

Initially:

```
        $\boldsymbol{L}(i,j) = (0,0)$   if $(i,j) \in S$,
        $\boldsymbol{L}(i,j) = (Z,Z)$   if $(i,j) \in \underline{S}$,
```

where Z is the largest integer that can be stored without inconvenience and where S is the object and $\underline{S}$ is the background in the original $M \times N$ binary picture.

First picture scan:

```
    For $j = 1,2,...,N-1$,
        for $i = 0,1,2,...,M-1$,
            $\boldsymbol{L}(i,j) = \min(\boldsymbol{L}(i,j), \boldsymbol{L}(i,j-1) + (0,1)$;
        for $i = 1,2,...,M-1$,
            $\boldsymbol{L}(i,j) = \min(\boldsymbol{L}(i,j), \boldsymbol{L}(i-1,j) + (1,0)$;
        for $i = M-2,M-3,...,1,0$,
            $\boldsymbol{L}(i,j) = \min(\boldsymbol{L}(i,j), \boldsymbol{L}(i+1,j) + (1,0)$.
```

Second picture scan:

```
    For $j = N-2,N-3,...,1,0$,
        for $i = 0,1,2,...,M-1$,
            $\boldsymbol{L}(i,j) = \min(\boldsymbol{L}(i,j), \boldsymbol{L}(i,j+1) + (0,1)$;
        for $i = 1,2,...,M-1$,
            $\boldsymbol{L}(i,j) = \min(\boldsymbol{L}(i,j), \boldsymbol{L}(i-1,j) + (1,0)$;
        for $i = M-2,M-3,...,1,0$,
            $\boldsymbol{L}(i,j) = \min(\boldsymbol{L}(i,j), \boldsymbol{L}(i+1,j) + (1,0)$.
```

The algorithm first assigns a maximum label to all background pixels. The real assignments take place in two scans of the complete picture where each picture scan involves for each line *j*, first a one-step propagation in the *j*-direction, then a sequential propagation in both *i*-directions.. Each label ***L****(i,j)* is involved in six comparisons.

Detailed analysis of the algorithm shows that it produces a distance map that is error-free except for very sparsely scattered pixels that may be assigned a distance label with an absolute error less than 0.29 pixel units.

4.2.2 Compute Average Cluster Size

In the first step, we need to threshold the input biofilm image into a binary image. In the second step, we apply the above Euclidean distance mapping algorithm and get a distance map for the cluster regions. In the third step, for those regions, we sum up distance values over all pixels and divide it by the total pixel number.

Fig. 5 shows a thresholded biofilm image with computed average cluster size value displayed at the bottom of the window. Biofilm image in Fig. 5 is thresholded and then filtered from the original biofilm image in Fig. 6. (The black parts are clusters.) The unit of the size is one pixel. By knowing the magnification setup of CSLM, we can convert this value to the actual length.