Automatic gridding of microarray images based on spatial constrained *K*-means and Voronoi diagrams

Mónica G. Larese [†] [*]
Intelligent Systems Group, Instituto de Física Rosario
Rosario, Santa Fe, Argentina
mlarese@ifir.edu.ar

Ariel A. Bayá [†]
Intelligent Systems Group, Instituto de Física Rosario
Rosario, Santa Fe, Argentina
abaya@ifir.edu.ar

Juan C. Gómez [†]
Intelligent Systems Group, Instituto de Física Rosario and
Laboratory for System Dynamics and Signal Processing, FCEIA, UNR
Rosario, Santa Fe, Argentina
jcgomez@fceia.unr.edu.ar

Abstract

Images from complementary DNA (cDNA) microarrays need to be processed automatically due to the huge amount of information that they provide. In addition, automatic processing is also required to implement batch processes able to manage large image databases. Most of existing softwares for microarray image processing are semiautomatic, and they usually need user intervention to select several parameters such as positional marks on the grids, or to correct the results of different stages of the automatic processing. On the other hand, many of the available automatic algorithms fail when dealing with rotated images or misaligned grids. In this work, a novel automatic algorithm for cDNA image gridding based on spatial constrained K-means and Voronoi diagrams is presented. The proposed algorithm consists of several steps, *viz*., image denoising by means of median fi ltering, spot segmentation using Canny edge detector and morphological reconstruction, and gridding based on spatial constrained K-means and Voronoi diagrams computation. The performance of the algorithm was evaluated on microarray images from public databases yielding promising results. The algorithm was compared with other existing methods and it shows to be more robust to rotations and misalignments of the grids.

Keywords: Bioinformatics, cDNA microarrays, image analysis, K-means, automatic gridding.

[*] Author to whom all correspondence should be addressed.
[†] The group is supported by CONICET, ANPCyT under grant PICT 11-15132 and Universidad Nacional de Rosario.

1 INTRODUCTION

cDNA microarrays are a new technology that allows to investigate, monitor and analyze the activity and interaction of thousands of genes simultaneously [3]. They can be used, for example, to identify genes that are involved in certain diseases (such as cancer) and organism's malfunctions by comparing the genetic expression of normal and abnormal cells. They help to understand gene expression, that is the biochemical process by which proteins are synthesized.

A cDNA microarray experiment can be described briefly as follows (for a detailed description of the process the reader is referred to [3, 5, 19, 29]). Two single-stranded DNA sequences are considered, each one made of multiple genes and coming from two different sources. They are obtained through messenger RNA (mRNA) reverse transcription. One of the samples is called the *target sample* and the other the *reference sample*. Both of them are labelled using two different colour fluorescences: red Cy5 and green Cy3 dyes, respectively. They are mixed and printed simultaneously on a glass slide provided with a control sample bound to its surface. cDNA printing is carried out by a robot head with arrayed tips that plunge into the mix and place it onto the slide using an array layout of printed spots. Usually the spots are grouped into grids within the slide.

After the cDNA labelled samples are bound and hybridized to the control sample, the slide is washed to remove unhybridized strands. Then the slide is scanned using two different lasers, one sensible to Cy5 and the other to Cy3 fluorescence, resulting in two 16-bit different images, usually called the red and green channel. Each image shows the mRNA abundance for each sample, and the combination of both of them allows to measure the relative mRNA abundance.

Digital image processing and analysis are then required to extract important features from the images. This process is generally described in the literature to be composed by three major steps: gridding of spot arrays, segmentation of spots and feature extraction. Pre-processing is also a very important task, including denoising, background correction, contrast enhancement, among others.

Due to the large number of images in existing databases, it is desirable to develop efficient automatic algorithms not requiring user intervention during the process. Although many researchers are working on microarray image processing, there are few works focused on automatic gridding. Most of the existing techniques grid the images through a manual or semiautomatic process that need user intervention, for instance, to set the values of different parameters, or to position marks on the grids, or to correct the obtained results in the intermediate stages. For example, in the case of the software MAGIC [16] the user must specify not only the number of existing grids, but also point with the mouse the top left spot, top right spot and bottom row. The user can apply the same configuration or specify a particular layout for each grid. This can lead to a time-consuming process when handling a large image database.

Yang *et al*. [29] proposed a semiautomatic method which requires the user to specify the top left spot in each grid and the bottom right spot in the bottom right grid of a *template*. The template is then used to process the whole batch of images.

Several automatic techniques that calculate vertical and horizontal intensity profiles from the original image have been proposed. They extract the grids by finding the minima in those profiles [23, 6]. Angulo and Serra [2] also apply morphological filtering to the profiles in order to extract the grids and spots. The drawback of these methods is that they fail when dealing with rotated images or misaligned grids or spots. Li *et al*. [20] made an improvement of this technique computing local profiles inside a moving window of approximately the size of a normal spot. The user needs to specify the number of rows and columns of spots in the image as well as the size of the moving window.

Several other methods perform a pre-processing for correcting the rotation angle of the image, some of them through a human guided process [17, 28, 10] and others automatically [27, 30, 25]. Misalign-

ments of the grids are not always considered.
Finally, an automatic technique which does not require rotation of the image is proposed by Katzer *et al.* in [19]. The method is based on a Markov Random Field (MRF) approach.
A new automatic algorithm for cDNA image gridding based on spatial constrained K-means and Voronoi diagrams, not requiring user intervention, is proposed in this work. The basic steps of this algorithm are:

- Pre-processing: noise removal using median filtering, segmentation of the spots by Canny edge detection and morphological reconstruction.
- Automatic gridding: connected components labelling and clustering based on spatial constrained K-means and Voronoi diagrams.

The algorithm was tested on public image databases and proved to perform accurately. The performance was compared with that of other existing methods, showing more robustness against rotation of the images and misalignments of the grids.

The rest of the paper is organized as follows. In section 2 the proposed automatic gridding algorithm is described. Experimental results showing the performance of the algorithm on public image databases are presented in section 3. Finally some concluding remarks and perspectives for future work are discussed in section 4.

2 AUTOMATIC GRIDDING ALGORITHM

The algorithm presented in this paper processes 16-bit/8-bit raw microarray images corresponding to the red Cy5 and green Cy3 channels in TIFF and GIF format. The images are first processed by applying a square root transformation in order to perform contrast enhancement and to reduce their bit depth.
The next step consists of applying the pre-processing techniques described in subsection 2.1, involving noise removal by median filtering, and image segmentation. After these procedures are applied, the spot centres are localized and spatial constrained K-means is used to extract the grids in combination with Voronoi diagrams, as it is described in subsection 2.2. The complete algorithm is schematically depicted in Figure 1.

2.1 Microarray Image Pre-processing

The pre-processing step consisting on noise removal and image segmentation is described in this section.

2.1.1 Noise Removal

Median filtering [15] is a well-known image processing technique which is very effective to remove impulse noise. The filter works by replacing the value of the pixel by the median of the intensity values in the neighbourhood (usually a 3x3 mask centred on the pixel). One important advantage of this type of filter is that it results in less blurring than a smoothing filter.
As shown in Figure 1, the median filter is used to remove the noise present in the Cy5 and Cy3 8-bit images (hereafter denoted as $R(i,j)$ and $G(i,j)$, respectively) using a 3x3 mask.

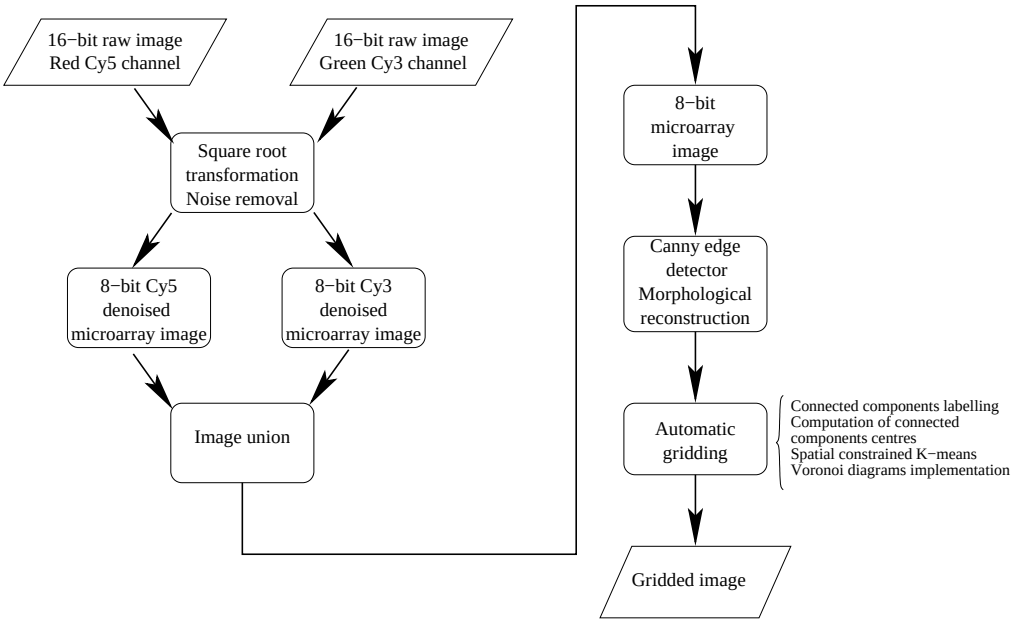


Figure 1: Outline of the proposed algorithm for automatic gridding.

After the noise is removed from each image, an 8-bit combined grayscale image $RG(i, j)$ is obtained by computing the image union of both channels as

$$RG(i, j) = max(R(i, j), G(i, j)) \tag{1}$$

This procedure was already successfully applied by Wu [28], resulting in a unique 8-bit grayscale image containing the information from both the Cy5 and Cy3 channels. An example of such images, corresponding to the whole yeast genome taken from [18], is presented in Figure 2, where the red (a) and green (b) channels are shown in conjunction with the resulting union (only the four top-left grids are included due to space limitations).

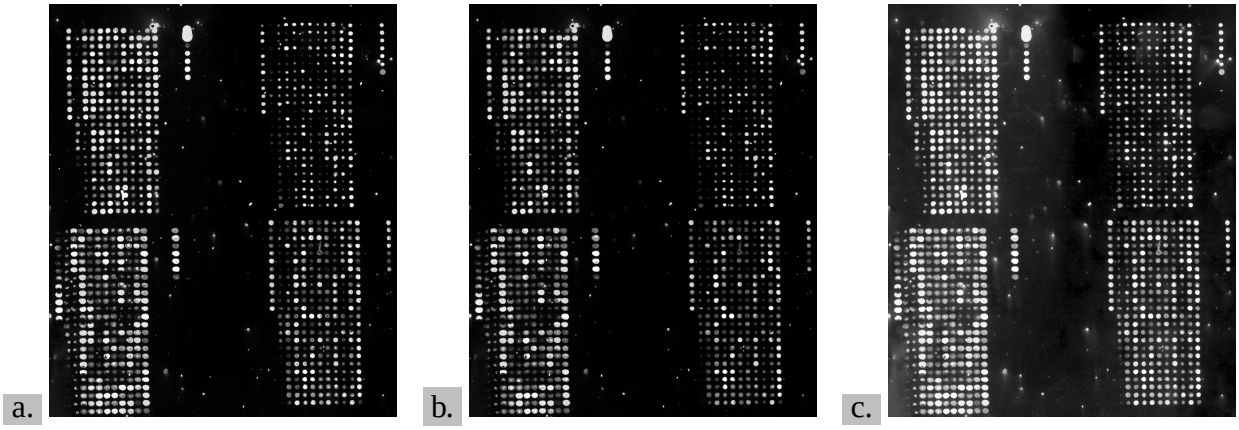


Figure 2: Yeast genome microarray image. a. Red Cy5 channel. b. Green Cy3 channel. c. Image union of both channels.

2.1.2 Image Segmentation

Image segmentation is implemented by first detecting the edge borders of the spots using Canny edge detector on the 8-bit grayscale image RG and then filling the resulting hollow regions. These procedures were also applied to cDNA microarray images within the pre-processing stage by Wang *et al*. [27].

Canny edge detector [9] is a powerful method to detect the edges of the objects present in the image. It first smoothes the image using a Gaussian filter and then detects the local gradient and the edge direction. The locally maximum points in the gradient direction are considered to be edge points. These maxima are then tracked to identify and preserve ridge edge points, setting to zero the rest of them, and thus producing a thin line in the output. Then the ridge pixels are thresholded using two thresholds, T_1 and T_2, being $T_1 < T_2$. Those ridge pixels between T_1 and T_2 are called "weak" while those greater than T_2 are called "strong". Finally, weak pixels are linked to strong pixels to produce continuous lines using 8-neighbours connections. After edges are detected, hollow regions in the binary edges image RG_{edge} are filled using morphological binary reconstruction.

Morphological reconstruction is an operator derived from mathematical morphology. The reader is referred to [26] for a more extensive explanation about binary and grayscale morphological reconstruction and practical algorithms to compute them. According to the scope of this work only binary morphological reconstruction is considered.

Let I, J be two binary images, with $J \subseteq I$. J is called the *marker* and I the *mask*. The marker determines the starting stage for the reconstruction while the mask acts as a constrain. These binary images can be seen as sets in $\mathbb{Z}^2$ with 1-valued pixels as elements. Then, morphological binary reconstruction can be defined in terms of the notions of *geodesic distance* and n-iterative *geodesic dilations*.

The geodesic distance d_I between two points $p, q \in I$ is defined as the length of the shortest path included in I that joins them. This distance depends on the kind of pixel connectivity used, i.e., 4 or 8-connectivity. Thus, geodesic dilation of size $n \geq 0$ of J within I, denoted as $\delta_I^{(n)}(J)$, can be defined as the set of pixels of I whose geodesic distance to J is smaller than or equal to n, *i.e.*

$$\delta_I^{(n)}(J) = \{p \in I | d_I(p, J) \leq n\} \tag{2}$$

Geodesic dilations are extensive transformations, *i.e.*, $J \subseteq \delta_I^{(n)}(J)$. Iteratively combining elementary geodesic dilations of J, morphological reconstruction becomes

$$\rho_I(J) = \bigcup_{n \geq 1} \delta_I^{(n)}(J) \tag{3}$$

Morphological reconstruction has very useful practical applications. The resulting effect depends on the images that are used as marker and mask. One of those applications consists on filling the holes inside objects in an image. In this case, the mask is set to be the complement of the image whose holes are to be filled, I^C. The marker J is chosen to be all 0s except on those pixels at coordinates (x, y) on the image border, where it has values of $1 - I$, that is

$$J_m(x, y) = \begin{cases} 1 - J(x, y), & \text{if } (x, y) \text{ is on the border of } J; \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

The resulting image RG_{bw} having all of its holes filled is obtained by computing the complement of the morphological reconstruction using the marker and mask images mentioned above. An algorithm for filling hollow regions using morphological reconstruction can be found in [24].

2.2 Spatial-Constrained *K*-means Clustering Overview

K-means is a fast partitional clustering algorithm that can be used to perform image segmentation. Given a number K of clusters, the algorithm divides the dataset D in K disjoints subsets of points and assigns them to a center according to some metric, usually the Euclidean metric. The updating criterion for the centers is performed by minimizing an error or cost function. In Table 1, the outline of the algorithm is presented. Given the number K of centroids [1], the algorithm initializes them at random, assigning each of the n points of the set to a center μ_i according to a distance function, for instance, the Mahalanobis distance, the Manhattan distance, or the Euclidean distance [13]. After that each center μ_i is updated and the process is iterated until there are no further changes in M.

Table 1: K-means algorithm

Given: n: number of points, K: number of clusters and M: a set of K vectors
1. Initialize $M = \{\mu_1, \mu_2, ..., \mu_k\}$
2. **do**
2.1. classify each point of the dataset D according to the nearest μ_i
2.2. recalculate each μ_i
until no change in μ_i
3. return M

The algorithm described in Table 1 can be implemented as a gradient descent or an Expectation Maximization (EM) algorithm [11]. These algorithms minimize a cost or an error function [2], which for the case of the EM approach is given by

$$J = \sum_{j=1}^{k} \sum_{x_i \in C_j} \parallel x_i - \mu_j \parallel \tag{5}$$

where $\parallel . \parallel$ represents the Euclidean norm, K is the number of clusters, C_j represents cluster j, and x_i is a pattern from D. For the EM case the cost function gives a measure of how good is the hypothesis for a given centroid. The mean vectors are then reestimated as

$$\mu_j = \frac{1}{N_j} \sum_{i=1}^{n} x_i \tag{6}$$

where N_j represents the number of points in cluster j, to obtain a new hyphotesis that minimizes the cost function in (5).
Equations (5) and (6) describe mathematically the steps of estimation and maximization respectively, used in the EM algorithm [22, 7]. Bottou and Bengio [7] present the corresponding equations for the gradient descent and they also provide a discussion about convergence of K-means methods.
For image segmentation tasks, conventional K-means will cluster the observations in feature space, *i.e.*, will cluster the pixel intensities. Luo *et al*. [21] pointed out the necessity of also considering spatial information of the pixels. They propose an improvement based on the use of spatial constrained region growing to obtain better general purpose segmentation, referring to it as *spatial-constrained K-means*. They develop a multi-level K-means clustering, and on each level they implement region

[1] A centroid is the center of a cluster.
[2] In the clustering literature these two functions are also been referred to as distortion.

growing of the pixels that correspond to the same feature cluster using connected components to label the image regions. On the next level K-means clustering is used on these regions separately.
In this work a variation of this technique is proposed. K-means clustering is carried out on the spot centres that are identified from the segmented image RG_{bw}, instead of using all the pixels in the image. The spot centres are detected through connected components analysis [15]. Let S be a subset of pixels in an image. A connected component of S is the set of all the pixels that are connected to any pixel p in S. Two pixels p and q are said to be connected if there exists a path between them that is composed entirely of pixels of S.
After connected components are detected, the mass centres of those components whose area is higher than the average area of the spots are computed. In this way the spot centres are clustered by grouping those which belong to the same grid, and avoiding parts of the same spot to be assigned to different grids, as it could happen when considering all the pixels in the image. The algorithm starts with two grid centres located at random, and clusters all the spot centres in these two classes. Then, each grid centre is split into two other centres and the algorithm clusters all the spot centres belonging to that class into the two new classes. This recursive procedure is iterated until the total number of split grid centres reaches the total number of grids, which are specified by the user at the beginning of the algorithm. Once spot centres (and consequently the spots) are clustered in their corresponding grids, the separation lines between the centres of adjacent grids are computed via Voronoi diagrams, and the spots belonging to the same Voronoi cell are considered to constitute a grid. Note the reader that not always the clusters found by spatial constrained K-means are definitive, due to the corrective effect of Voronoi diagrams.
Voronoi diagrams were firstly formulated by Descartes (1644) and Dirichlet (1850). They can be defined briefly as follows [14]. Let consider two points p and q belonging to a plane, and let $d(p, q)$ be the Euclidean distance between them. The Voronoi diagram $V(S)$ for a set S of n distinct points of interest in the plane is the subdivision of this plane in n Voronoi cells, one for each point in S, such that q lies in the cell of p_i iff $d(q, p_i) < d(q, p_j)$, for each $p_j \in S$, with $i \neq j$.
Each edge of $V(S)$ is a bisector of a pair of points in S, and each vertex is the intersection of two or more Voronoi edges. A typical Voronoi diagram is depicted in Figure 3. Mathematical definitions and properties about Voronoi diagrams can be found in [4], while their connection with convex hull and Delaunay tesselation is discussed in [14]. A practical algorithm (Qhull) for performing Voronoi diagrams via Delaunay triangulation is proposed by Barber *et al.* [8].

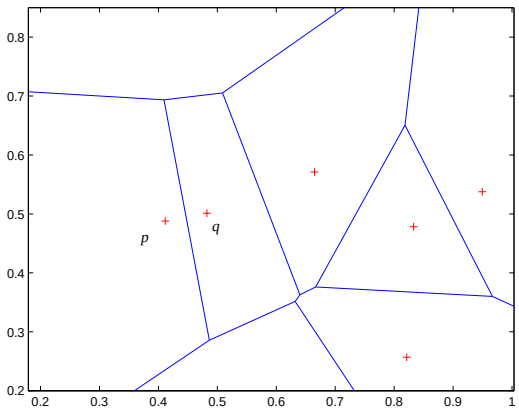


Figure 3: A typical Voronoi diagram.

3 EXPERIMENTAL RESULTS

In this section the different stages of the algorithm will be illustrated through an example using the microarray image in Figure 2, corresponding to the whole yeast genome. Further, the performance of the proposed algorithm will be compared with that of other existing methods, in particular with the work by Angulo and Serra in [2]. Robustness of the proposed method against grid rotations and misalignments will also be demonstrated on real microarray images.
Figure 4a. shows the results corresponding to Canny edge detection applied on the 8-bit combined image in Figure 2c. Figure 4b. shows the result of applying morphological reconstruction on the previous image. Finally, Figure 4c. displays the resulting image after performing connected components labelling and removing those elements whose area is smaller than a given threshold (usually the average area of the spots).

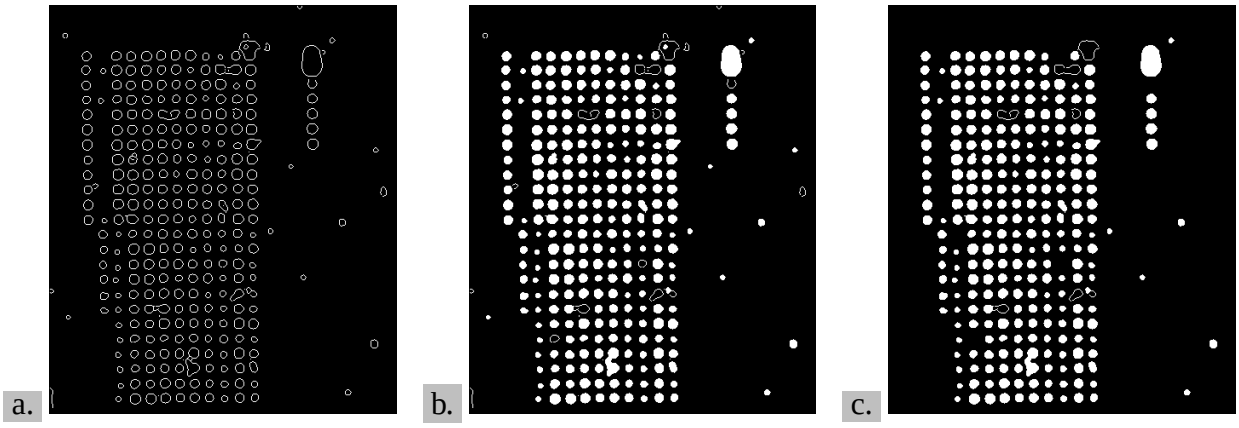


Figure 4: a. 8-bit image union after Canny edge detection (RG_{edge}). b. RG_{edge} image after morphological reconstruction (RG_{bw}). c. RG_{bw} image after connected components labelling. (Only the top-left grid is included due to space limitations.)

Figure 5a. corresponds to the resulting image after computation of the mass centres of the connected components. The detected spot centres are marked with red crosses. These centres are clustered using spatial constrained K-means and then, on these results, Voronoi diagrams are implemented. The resulting grids are shown in Figure 5b. Spots belonging to the different grids are colour labelled. The grid separating lines generated by Voronoi diagrams (in red) are also included in this figure.
In Figure 6a. the complete image corresponding to the whole yeast genome taken from [18] and consisting of 32 grids is presented. Figure 6b. shows the results obtained by the method proposed in [2], while the resulting gridded image based on the method proposed in this paper is presented in Figure 6c. As it is noticeable from Figure 6b. the grids are correctly extracted and all of them are found, but also some isolated groups of spots are extracted as if they were independent grids in themselves. Appart from this comment, the method based on morphological filtering of the intensity profiles and the one proposed in this work seem to perform in a similar way.
In order to evaluate the robustness of the proposed algorithm against rotation and misalignments of the grids the complete microarray image corresponding to the gene expression of the diffuse large B-cell lymphoma (coded as 'lc7b017rex2.gif') is included in Figure 7a. This image was taken from a public database of the Lymphoma/Leukemia Molecular Profiling Project[3] described in [1]. Note the reader the high degree of rotation and misalignment of the different grids. Figure 7b. shows the results of the algorithm introduced in [2]. It is apparent that the algorithm fails in separating the grids.

[3]The microarray images are downloadable from: http://llmpp.nih.gov/lymphoma/data/rawdata/.

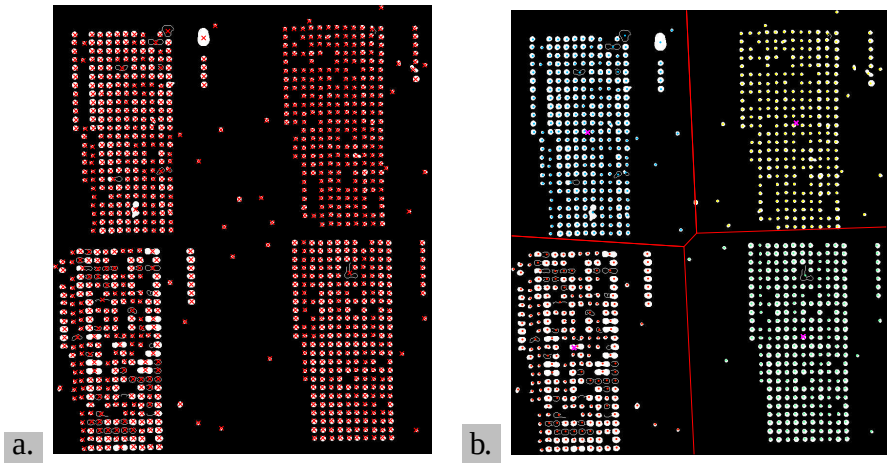


Figure 5: a. Spot centres detected using connected components labelling. b. Gridded image computed using the combination of spatial-constrained *K*-means and Voronoi diagrams. Spots for each grid are marked with different colours, and the center of each grid is marked in pink.

This is due to the fact that the algorithm assumes that the original images are not rotated and the grids are aligned. In contrast, the method proposed in this work does not make these assumptions on the original images, and therefore it works properly even for the cases where noticeable rotations and misalignments of the grids are present. This can be observed from Figure 7c.

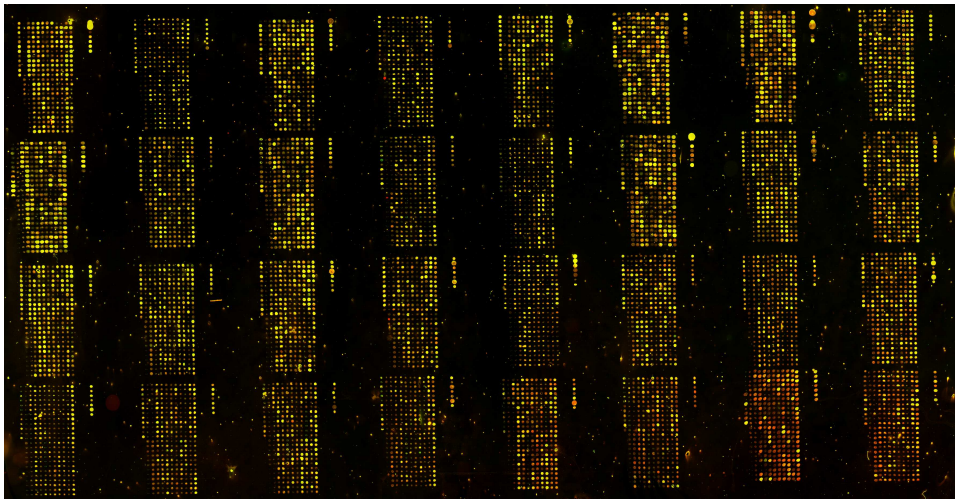
4 CONCLUSIONS AND FUTURE WORK

In this work a new automatic technique for automatic gridding of cDNA microarray images has been introduced. In a pre-processing stage noise is removed by means of median filtering, then image segmentation is performed through Canny edge detector and morphological reconstruction. The automatic gridding is then performed by first labelling connected components and computing the mass centres of the segmented spots. Spatial constrained K-means is in turn used to cluster the spot centres in order to assign them to their corresponding grid. The proposed technique avoids pixels from the same spot to be assigned to different grids. Once this clustering has been performed, Voronoi diagrams are computed on the resulting grid centres to delimit the grid areas. In this way, the previous clustering is corrected by assigning to the same grid all those spot centres that are contained in the same Voronoi cell. Separating lines between grids are also obtained.

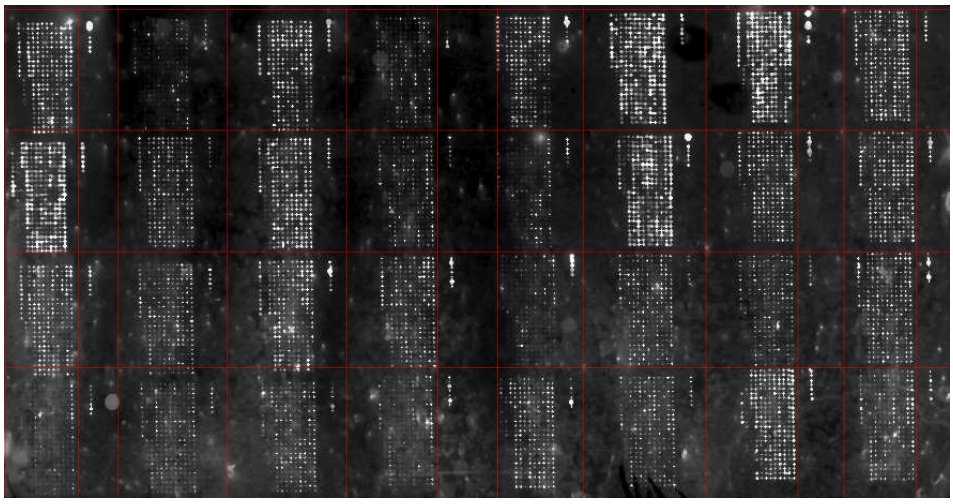
The proposed algorithm was applied on public domain images and was compared with the method based on morphological filtering of horizontal and vertical intensity profiles described in [2]. In contrast to the method in [2], the algorithm described in this work showed to be robust against rotations and misalignments of the grids.

Improvements need to be made when dealing with images where small separation between adjacent grids is present. In adition, other methods need to be explored in order to obtain the spot centres, to discard noisy and control spots that can not be completely removed from the images in the pre-processing stage. A work is under preparation where the denoising pre-processing stage of the algorithm is been implemented based on Discrete Wavelet Transform (DWT) [12].

a.



b.



c.

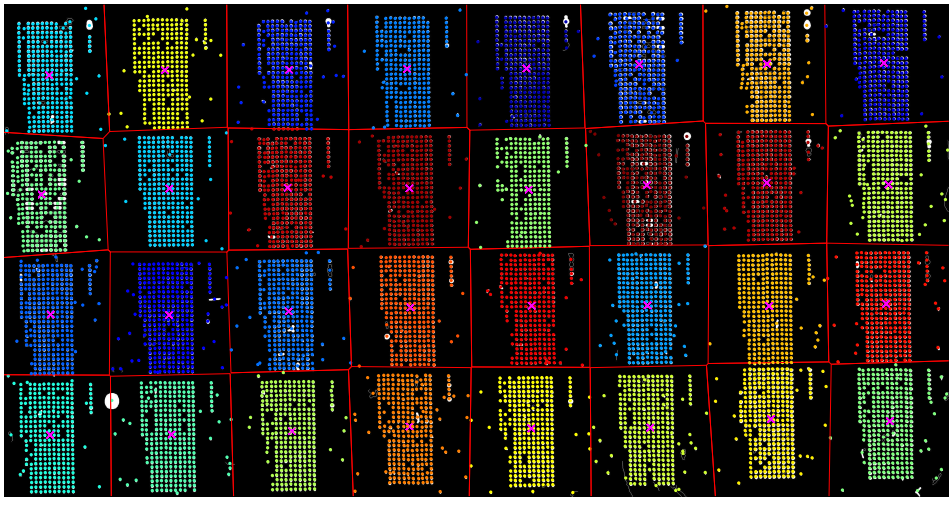


Figure 6: a. Original microarray image taken from [18]. b. Gridded image using morphological filtering. c. Gridded image using spatial-constrained K-means in combination with Voronoi diagrams.

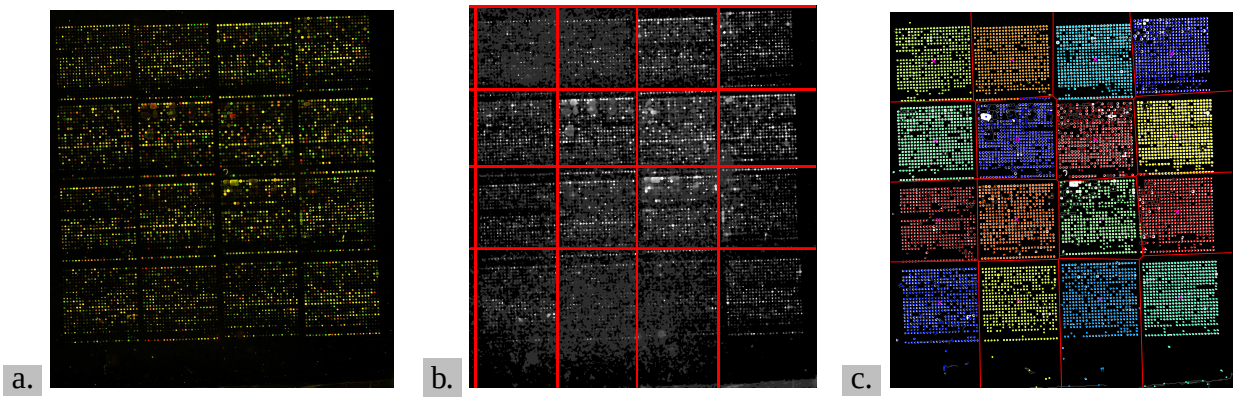


Figure 7: a. Original microarray image taken from [1]. b. Gridded image using morphological filtering. c. Gridded image using spatial-constrained K-means in combination with Voronoi diagrams.

REFERENCES

[1] Alizadeh A., Eisen M., *et al*. Distinct types of diffuse large B-cell lymphoma identifi ed by gene expression profi ling. *Nature*, 403(6769):503–511, February 2000.

[2] Angulo J. and Serra J. Automatic analysis of DNA microarray images using mathematical morphology. *Bioinformatics*, 19(5):553–562, 2003.

[3] Attwood T. and Parry-Smith D. *Introduction to bioinformatics*. Addison Wesley Longman Limited, Harlow, England, 1999.

[4] Aurenhammer F. and Klein R. *Handbook of Computational Geometry*, chapter V, pages 201–290. Elsevier Science Publishing, 2000.

[5] Berrar D., Dubitzky W., and Granzow M., editors. *A practical approach to microarray data analysis*. Kluwer Academic Publishers, Dordrecht, 2003.

[6] Blekas K., Galatsanos N., and Georgiou I. An unsupervised artifact correction approach for the analysis of DNA microarray images. IEEE Int. Conf. on Image Processing (ICIP 2003), Barcelona, September 2003.

[7] Bottou L. and Bengio Y. Convergence properties of the K-means algorithms. In G. Tesauro, D. Touretzky, and T. Leen, editors, *Advances in Neural Information Processing Systems*, volume 7, pages 585–592. The MIT Press, 1995.

[8] Bradford Barber C., Dobkin D., and Huhdanpaa H. The Quickhull algorithm for convex hulls. *ACM Trans. on Mathematical Software*, 22(4):469–483, 1996.

[9] Canny J. A computational approach for edge detection. *IEEE Trans. on Pattern Anal. and Machine Intell.*, 8(6):679–698, 1986.

[10] Demirkaya O., Asyali M., Shoukri M., and Abu-Khabar K. Segmentation of microarray cDNA spots using MRF-based method. 25th Annual Conf. of the IEEE EMBS, Cancun, Mexico, 2003.

[11] Dempster A., Laird N., and Rubin D. Maximum likelihood from incomplete data via the EM algorithm. *J. of the Royal Statistical Society B.*, 39:1–38, 1977.

[12] Donoho D. Denoising by soft-thresholding. *IEEE Trans. on Information Theory*, 41(3):613–627, 1995.

[13] Duda R. and Hart P. and Stork G. *Pattern classification*. J. Wiley & Sons, New York, 2nd. edition, 2001.

[14] Fisher J. Visualizing the connection among convex hull, Voronoi diagram and Delaunay triangulation. 37th Annual Midwest Instruction and Computing Symp., MN, 2004. URL: http://www.cs.mtu.edu/∼shene/PUBLICATIONS/2004/Hull2VD.pdf.

[15] Gonzalez R. and Woods R. *Digital image processing*. Prentice Hall, 2nd. edition, 2002.

[16] Heyer L., Moskowitz D., Abele J., Karnik P., Choi D., Malcolm Campbell A., Oldham E., and Akin B. MAGIC tool: integrated microarray data analysis. *Bioinformatics*, 21(9):2114–2115, 2005.

[17] Hirata Jr. R., Barrera J., Hashimoto R., Dantas D., and Esteves G. Segmentation of microarray images by mathematical morphology. *Real-Time Imaging*, 8:491–505, 2002.

[18] Hoopes L. and Brown J. Whole yeast genome (Y01 Version). Department of Biology, Davidson College, Davidson, NC, http://www.bio.davidson.edu/projects/magic/magic.html, 2003.

[19] Katzer M., Kummert F., and Sagerer G. Methods for automatic microarray image segmentation. *IEEE Trans. on Nano-Bioscience*, 2(4):202–214, 2003.

[20] Li Q., Fraley C., Eugene Bumgarner R., Yeung K., and Raftery A. Donuts, scratches and blanks: robust model-based segmentation of microarray images. *Bioinformatics*, 21(12):2875–2882, 2005.

[21] Luo M., Ma Y.-F., and Zhang H.-J. A Spatial Constrained K-Means Approach to Image Segmentation. *Fourth IEEE Pacific-Rim Conference On Multimedia*, pages 738 – 742, December 2003.

[22] Mitchell T. *Machine learning*. Mac Graw Hill, Boston, 1997.

[23] Rueda L. and Vidyadharan V. A hill-climbing approach for automatic gridding of cDNA microarray images. *IEEE/ACM Trans. on Computational Biology and Bioinformatics*, 3(1):72–83, January 2006.

[24] Soille P. *Morphological image analysis: principles and applications*. Springer-Verlag, 1999.

[25] Uehara C. and Kakadiaris I. Towards automatic analysis of DNA microarrays. In *WACV '02: Proc. of the 6th. IEEE Workshop on Applications of Computer Vision*, page 57, Washington DC, 2002. IEEE Computer Society.

[26] Vincent L. Morphological grayscale reconstruction in image analysis: applications and efficient algorithms. *IEEE Trans. on Image Processing*, 2(2):176–201, 1993.

[27] Wang Y., Shih F., and Ma M. Precise gridding of microarray images by detecting and correcting rotations in subarrays. JCIS/CBGI, Salt Lake City, Utah, 2005.

[28] Wu S. and Yan H. Microarray image processing based on clustering and morphological analysis. In Yi Ping Phoebe Chen, editor, *Conferences in Research and Practice in Information Technology*, volume 19. First Asia-Pacific Bioinformatics Conference, Adelaide, Australia, 2003.

[29] Yang Y., Buckley M., Dudoit S., and Speed T. Comparison of methods for image analysis on cDNA microarray data. Technical Report #584, Department of Statistics, University of California, Berkeley, November 2000. URL: http://www.stat.berkeley.edu/users/terry/zarray/TechReport/584.pdf.

[30] Zhang K., Ma M., Wang H.-Y-and Wang Y., and Shih F. Integrated data analysis for genotyping microarrays. Technical Report #0405-33, Center for Applied Mathematics and Statistics, New Jersey Institute of Technology, University Heights, Newark, 2005. URL: http://web.njit.edu/∼qma/pub/report33.pdf.