

# Precise Gridding of Microarray Images by Detecting and Correcting Rotations in Subarrays

Yu Wang<sup>1,2</sup> Frank Y. Shih<sup>2</sup> Marc Q. Ma<sup>1,3</sup>

<sup>1</sup>Applied Bioinformatics Laboratory, <sup>2</sup>Computer Vision Laboratory, <sup>3</sup>Center for Applied Mathematics and Statistics  
College of Computer Sciences, New Jersey Institute of Technology, Newark, NJ 07102  
Emails: {yw6, shih, qma}@njit.edu

## Abstract

DNA microarrays are an important tool for massive gene expression profiling and genotyping studies. A microarray consists of thousands of spots that must be scanned and digitized for further analysis. However, the mechanical constraints and other printing issues increase the difficulty in automatic analysis, most of the applications released up to now need different levels of human intervention, which inevitably reduce the efficiency and reproducibility of the entire process. In this paper, we present a precise microarray gridding method, in which potential problems in grid alignment and possible rotation in subarrays are detected and corrected. The gridding process is fully automatic, which eliminates the need for human intervention.

**Keywords:** DNA microarray image, image processing, automatic gridding, subarray rotation

## 1. Introduction

DNA microarrays consist of thousands to hundreds of thousands of spots that must be scanned and digitized for further analysis. By analyzing microarray images, biomedical researchers can perform massive gene expression profiling, genotyping and identify the genes that are involved in particular diseases using data mining and other bioinformatics tools such as gene clustering [1, 2].

The process of extracting features for further analysis from a microarray image can be categorized into three main steps: gridding, segmentation and quantification. Gridding is to assign each spot with individual coordinates. This is the most important and fundamental step in the whole process, and is our focus in this paper. Segmentation is to classify the pixels into foreground, background and others (e.g., noise) [2]. Quantification is to compute unique intensity values for each spot, which are related to the quantity of mRNA present in the solution that hybridize at the particular location of a microarray substrate.

Microarray images contain high-density spots located on regular grids [3]. Ideally, the grids should be highly structured and the analysis should be easy to perform. Nonetheless, in the real world, because of the mechanical constraints in the spotting process, the hybridization inconsistencies, the necessity to print dense arrays in order to increase the throughput, and the environment effect such as contamination [4], there are several issues that have to be addressed: (a) The exact location of each grid may vary from slide to slide. (b) The relative position of subarrays may vary. (c) The array in the image may tilt. (d) The array may have dirt or contamination. (e) The signal received may be weak.

During the last decade, there have been methods (e.g., [5, 6, 7]) and software packages available that deal with one or a few aforementioned problems, such as ScanAlyze (Eisen 1999), Spot (Buckly 2000), and GenePix Pro6 (Axon Instruments Inc. 2004). However, they all require certain level of human intervention for achieving desired accuracy, which imposes big burden on the biologists who use microarrays in their research.

In the microarray image processing, the accuracy and efficiency will inevitably decrease while the level of human intervention increases. Our main goal here is present a precise and automatic microarray gridding method which will eliminate the need for human intervention and correct the potential alignment and rotation problems.

This paper is organized as follows. We describe the procedure of our algorithm and the function in each step in Section 2. Pre-processing, rotation detection, and local gridding methods are presented in Section 3. Experimental results are given in Section 4. Conclusions and future work are provided in Section 5.

## 2. Precise Gridding Process

To precisely address each spot and examine rotations in subarrays in a microarray image, we have developed an iterative refinement method. The flowchart of the precise automatic gridding process shows in Fig. 1.

There are three main components in our method: pre-processing, rotation detection and local gridding refinement. At the pre-processing stage, we estimate the global variables for gridding. At the rotation detection stage, we detect rotated subarrays by computing local gridding variables and comparing them with those global variables obtained in the previous stage. If any rotated region is detected, we successively rotate the area and examine its local features with the global ones until they are matched. After obtaining the rotation angle, we move on to the local gridding refinement stage which is based on the local information of each subarray.

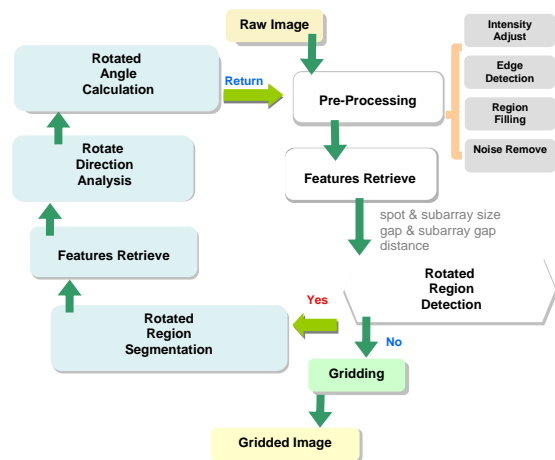


Fig. 1: Flowchart of the precise automatic gridding process.

### 3. Pre-Processing, Rotation Detection and Local Gridding

To avoid altering the original intensities of the microarray images, we use the duplicate image from the input to compute the information we need and keep the input image intact for further analysis. In the following, we will give details for each step.

#### 3.1. Pre-Processing

The pre-processing is to obtain the global values of the structure components of microarray image for further examination. We apply the state-of-the-art methods to extract the important features of the microarray image, including intensity adjustment, Canny edge detection, region filling, noise removal by morphological opening, and projection profile calculation.

The intensity adjustment maps the pixel values in the input image to new values such that the values below the “low\_in” threshold will be changed to a single “low\_out” value and those values above the “high\_in” threshold will be changed to the same

“high\_out” value. This method helps us to obtain the maximal information from any spots.

In order to distinguish spot regions from non-spot regions in the image, we apply Canny edge detection method [8] on the intensity adjusted image to segment the objects into edge and non-edge groups. By using two different thresholds to detect strong and weak edges, and including the weak edges in the output only if they are connected to the strong edges, the Canny edge detection is less likely to be fooled by noises and more likely to detect true weak edges than other methods, *e.g.*, Sobel, Prewitt, and zero-crossing [9].

The image after applying thresholding and Canny edge detection turns into a black-and-white binary image with only edges being presented. Here we implement the region filling method with 4-connected background neighbors to fill the holes (Note: 8-connected background neighbors are too restricted in this situation). After the filling process, the applicable spot regions are filled with the intensity of 1.

The noises that have been emphasized by intensity adjustment step need to be removed. Mathematical morphology [6] has been widely used in image processing especially when dealing with the geometry shape change. Here we use the morphological opening (succession of erosion then dilation) to eliminate the noise from the local region.

At the end of this stage, we calculate the intensity projection profile of the microarray image by summing up the intensity of each pixel along the  $x$  and  $y$  axis. By calculating and analyzing the values on the projection profiles, we also obtain global values such as the average size of spot, the distances between rows and columns, the distances between subarrays and the width and height of subarrays.

#### 3.2. Rotation Detection

There are two steps in rotation detection stage: subarray rotation identification and angle calculation. Because the true values of the original images will affect by any rotation involving interpolation, therefore, we use the black-and-white binary image from the previous steps to examine the possible rotation problems.

Subarray rotation identification is achieved by examining the intensity projection profile along  $x$ - and  $y$ -axes. If there is any abnormal block whose size is greater than the average subarray’s width and height, then that region is identified as the rotated region. The graphical representation is shown in Fig. 2.

By comparing the intensity sum of the top one-third region with the bottom one-third region along the horizontal and vertical axis in the rotated region, we can detect both clockwise and counter-clockwise

rotation directions. The graphical representation is shown in Fig. 3.

For calculating the rotation angle, we iteratively rotate the region by a quarter degree until its projection profile is closely matched to the normal one. Although the linear iteration method is always computationally expensive, we use it here instead of the binary algorithm for three reasons. First, the rotation angle in a subarray can only be within a small value due to the high-density feature in the microarray image. Second, applying the binary algorithm will need an extra process to examine the rotation direction to check whether it is over-rotated. Third, the image that we manipulate here is only a small portion of the whole image, so that we can complete the process in a very short time. Fig. 4 illustrates the method.

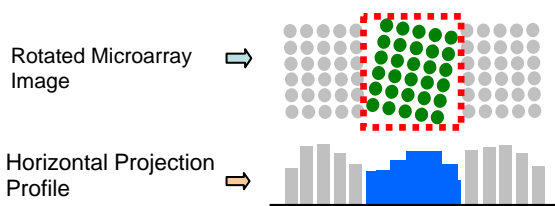


Fig. 2: Intensity projection profile of the rotated region. By comparing the projection profile of the rotated region with the normal one, it can be easily identified.

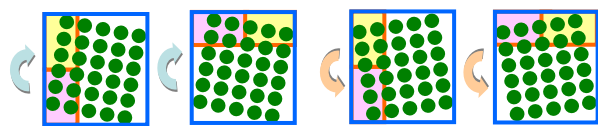


Fig. 3: Rotation direction analysis. The left two figures represent the clockwise rotation image. The right two figures represent the counter-clockwise rotation image. By summing up the intensities of the spots within the two enclosed areas (pink and yellow) respectively, we can identify rotation direction using comparison. Computing both horizontal and vertical enclosed areas makes our method more reliable.

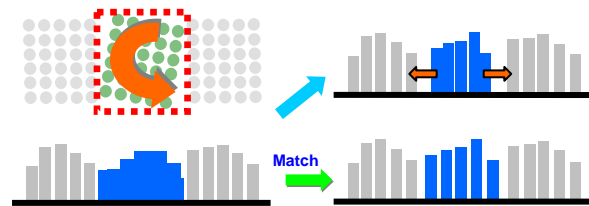


Fig. 4: Rotation correction by successively rotating the target region and comparing its projection profile with the normal region until both patterns are closely matched, the rotation angle can be calculated.

### 3.3. Local Gridding

The global values of the microarray image obtained in the pre-processing stage can not be simply applied to each individual subarray directly because there are variations among the subarrays. Since we have some idea about the location of each subarray, we apply the pre-processing method again to each subarray to obtain the specific features for the local subarray. Fig. 5 shows the difference between the global and local gridding.

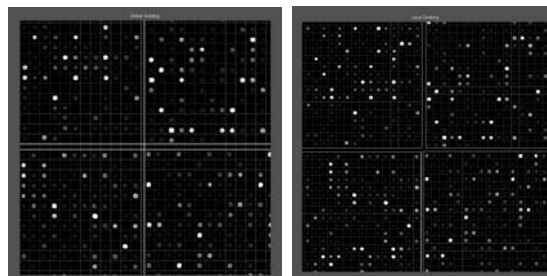


Fig. 5: Global gridding (left) and local gridding (right).

## 4. Experimental Results

Fig. 6 shows a partial original image with  $3 \times 2$  subarrays. Fig. 7 shows the projection profile of the original input image along  $x$ - and  $y$ -axes. Fig. 8 shows the images after successively implementing the “Intensity Adjustment,” “Canny Edge Detection,” “Region Filling,” and “Morphological Opening.” Fig. 9 shows the projection profile after our pre-processing methods. From the projection profile, each subarray, each spot, and the abnormal regions can be identified.

After the pre-processing stage, the Rotation Detection algorithm is applied. After the rotated area that is identified by our algorithm, Fig. 10 shows the successive rotation angle process.

Finally, our program completes the gridding process based on each subarray’s local features. Fig. 11 shows part of the result of the gridding process, from which we observe that the gridding applied to each subarray can truly reflect the local features.

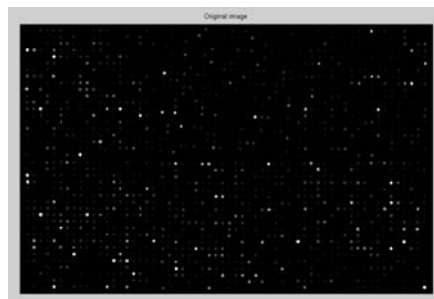


Fig. 6: A partial original image with  $3 \times 2$  subarrays.

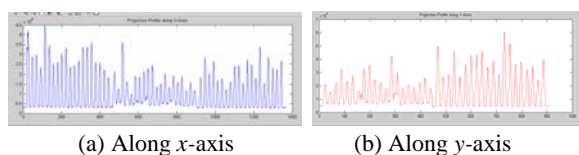


Fig. 7: Projection profile of the partial original input image.

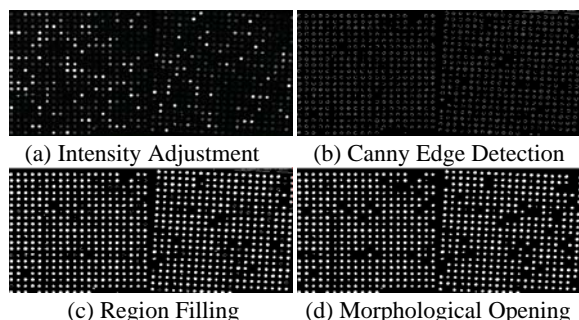


Fig. 8: Images after “Intensity Adjustment,” “Canny Edge Detection,” “Region Filling,” and “Morphology Opening.”

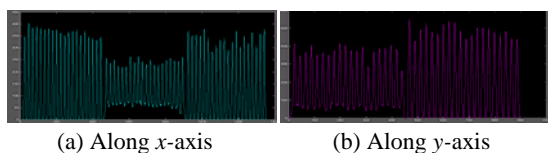


Fig. 9: The projection profile after pre-processing.

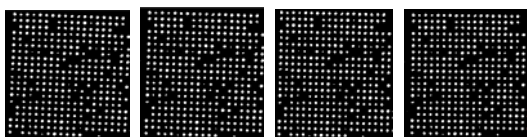


Fig. 10: The successive rotation angle correction process.

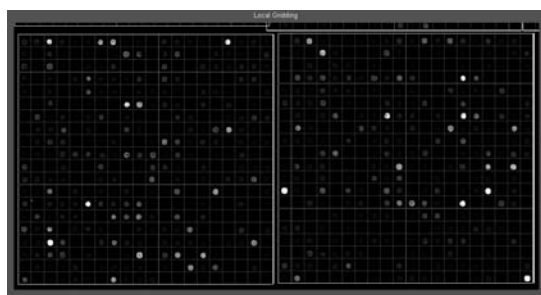


Fig. 11: The final result of our gridding algorithm. Each subarray is addressed individually base on its local features.

## 5. Conclusions and Future Work

In this paper, we have described a novel gridding algorithm by applying the state-of-the-art image processing methods along with our iterative feature extraction algorithm. Our method not only can precisely address each subarray’s location for correcting the potential alignment and rotation problems but also eliminate the human intervention to

achieve the high-throughput requirement. If big chunk of contaminations or noises appear to be the similar size of the subarray, misclassification of rotated/non-rotated subarrays will happen in our algorithm.

Our future work will focus on adding the state-of-the-art classifier such as LIBSVM [10] in our method to increase the robustness and flexibility. Preciser addressing program to the single spot level could be developed as well by examining each spot’s foreground and background intensity individually.

## 6. References

- [1] R.S.H. Istepanian, “Microarray image processing: current status and future directions,” *IEEE Transactions on Nanobioscience*, vol. 2, no. 4, pp. 173-174, Dec. 2003.
- [2] S. Wu and H. Yan, “Microarray image processing based on clustering and morphological analysis,” *Proceedings of the First Asia-Pacific Bioinformatics Conference on Bioinformatics 2003*, vol. 19. pp. 111-118, 2003.
- [3] L. Stefano and Y. Luo, “Gridding and compression of microarray images,” *Proceedings of the 2004 IEEE Computational Systems Bioinformatics Conference*, 2004.
- [4] S. Draghici, *Data Analysis Tools For DNA Microarrays*, CRC Press, Boca Raton, FL, 2003.
- [5] Y.H. Yang, M.J. Buckley, and T.P. Speed, “Comparison of methods for image analysis on cDNA microarray data,” *Journal of Computational and Graphical Statistics*, vol. 11, no. 1, pp. 108-136, 2002.
- [6] F.Y. Shih and O.R. Mitchell, “Threshold decomposition of grayscale morphology into binary morphology,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 11, no. 1, pp. 31-42, Jan. 1989.
- [7] M. Katzer, F. Kummert, and G. Sagerer, “A markov random field model of microarray gridding,” *Symposium on Applied Computing*, pp.72-77, March 9-12, 2003.
- [8] J.F. Canny, “A computational approach to edge detection,” *IEEE Trans Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679-698, Nov. 1986.
- [9] S. Selvarajan and C.W. Tat, “Extraction of man-made features from remote sensing imageries by data fusion techniques,” *22<sup>nd</sup> Asia Conference on Remote Sensing*, 2001.
- [10] C.C. Chang and C.J. Lin, “LIBSVM: a library for support vector machines,” *Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>*, 2004.