A Fully-Automatic Method for Gridding Bright Field Images of Bead Based Microarrays

Abhik Datta, *Student Member, IEEE,* Adams Wai-Kin Kong, *Member, IEEE,* and Kin-Choong Yow

Abstract*— In this paper a fully automatic method for gridding bright field images of bead based microarrays is proposed. There have been numerous techniques developed for gridding fluorescence images of traditional spotted microarrays but to our best knowledge, no algorithm has yet been developed for gridding bright field images of bead based microarrays. The proposed gridding method is designed for automatic quality control during fabrication and assembly of bead based microarrays. The method begins by estimating the grid parameters using an evolutionary algorithm. This is followed by a grid fitting step that rigidly aligns an ideal grid with the image. Finally, a grid refinement step deforms the ideal grid to better fit the image. The grid fitting and refinement are performed locally and the final grid is a non-linear (piecewise affine) grid. To deal with extreme corruptions in the image, the initial grid parameter estimation and grid fitting steps employ robust search techniques. The proposed method does not have any free parameters that need tuning. The method is capable of identifying the grid structure even in the presence of extreme amounts of artifacts and distortions. Evaluation results on a variety of images are presented.

***Index Terms*—griddng; bead; microarray; registration;**

I. Introduction

Bead based microarrays have now become ubiquitous in almost all areas of biomedical research. Randomly assembled bead microarrays use micrometer sized spherical beads as the solid support for various kinds of molecular reactions. The bead surface is coupled with capture probes such as proteins, oligonucleotides or peptides which capture the target analytes in a sample. The target analytes are subsequently quantified using fluorophore conjugated detection molecules. The beads are immobilized by trapping them, into wells etched on the surface of a substrate, using fluidic, magnetic or electrostatic forces.

Proper fabrication of bead microarrays requires stable immobilization of beads to sustain fluidic forces exerted during bioassay. Bead deposition however is not a straightforward process, as the number of beads that get deposited cannot be precisely controlled. Fabrication can involve multiple deposition steps in which beads are deposited and washed off alternately. If beads are not washed off properly, clusters of beads will remain on the surface. Similarly, if the washing is too harsh too few beads may get immobilized. Locating and counting immobilized beads, a process known as bead detection, is an important and necessary step to ensure successful bead deposition.

The first step in this bead detection process is gridding, the process of identifying the location of each individual well in the image and assigning an image region to each well. Gridding is, however, complicated by the presence of various artifacts and distortions in the images. The bead deposition process is a wet lab process and involves many fluidics steps. As a result the images can contain a large number of unwanted artifacts such as water spots, dirt particles, pieces of left-over scaffolding etc., (see Fig. 1(d)). Microarray chips may have local non-linear deformations of the grid due to manufacturing imprecisions. The images can also have global non-linear deformations due to spherical aberrations of the camera lens.

Numerous manually assisted, semi-automated and fully-automated techniques have been developed for gridding fluorescence images of traditional spotted microarrays ([1]-[4]). Bright field images of bead based microarrays however, have some characteristics that differentiate them from spotted microarray images. Some typical bright field images are given in the top row of Fig. 1. In bead based microarray images, clusters of beads can be present (see Fig. 1(b)). Spotted microarray images do not have such clusters of spots. Also, bead based microarrays typically use hexagonal grids whereas spotted microarrays use square grids. Moreover, unlike in fluorescence images where only the expressed beads/spots are visible, in the bright field images all the wells and beads are visible. To our best knowledge, no algorithm has yet been developed for gridding bright field images of bead based microarrays.

In this paper, we propose a novel method for gridding bright field images of bead based microarrays acquired during or after bead deposition. The proposed gridding method is fully-automatic and is capable of identifying the correct grid structure in the presence of the following distortions:

1) clusters of beads caused by a subpar deposition process,
2) presence of large number of artifacts and spots in the image,
3) non-linear distortions of the grid, and

This paragraph of the first footnote will contain the date on which you submitted your paper for review.

Abhik Datta is with the School of Computer Engineering, Nanyang Technological University, Singapore 639798 (e-mail: abhik1@e.ntu.edu.sg).

Adams Wai-Kin Kong is with the School of Computer Engineering, Nanyang Technological University, Singapore 639798 (e-mail: adamskong@ntu.edu.sg).

Kin-Choong Yow is with the GIST College, Gwangju Institute of Science and Technology, 123 Cheomdangwagi-ro, Buk-gu, Gwangju 500-712, Republic of Korea (e-mail: kcyow@gist.ac.kr).

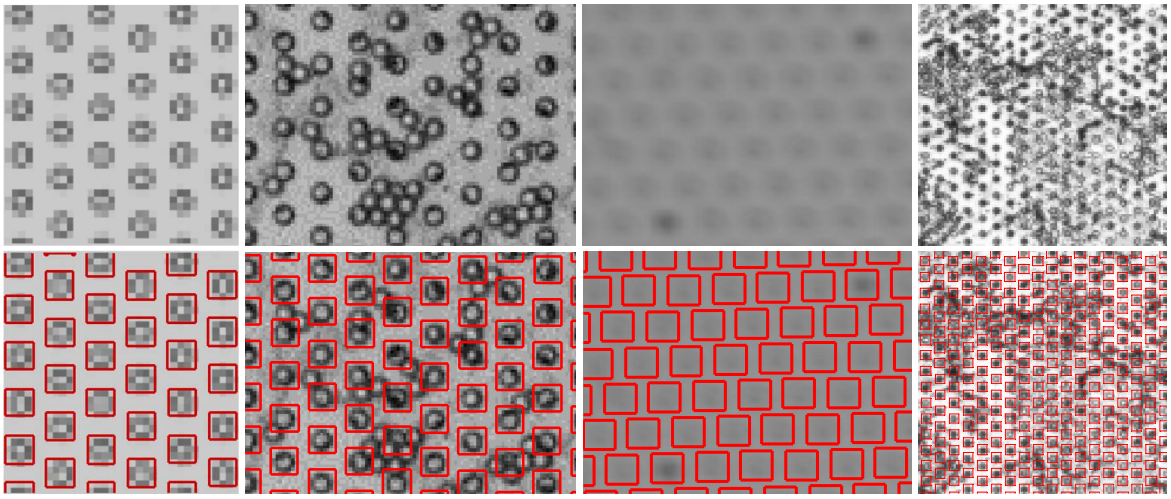


Fig.1. Small regions of original bright field images (top row) and the gridding outputs (bottom row) given by the proposed algorithm. (a) a low resolution image, (b) a high resolution image with bead clusters, (c) a blurry high resolution image and (d) a low resolution image with artifacts.

4) blurring of images due to poorly calibrated image acquisition (see Fig. 1(c)).

The proposed method is examined on five datasets with different image characteristics and varying amounts of distortions.

The rest of the paper is organized as follows. Section 2 discusses previous works on spotted microarray image gridding and points out the difficulties in using them for bead based microarray gridding. Section 3 presents the proposed method in detail and discusses how it overcomes the challenges outlined in this section. Section 4 reports the experimental results, including gridding accuracy as well as visual evaluation. Section 5 concludes the paper and outlines directions for future work.

II. Review of Gridding Algorithms

The use of axis projections ([1], [2], [3], [4]) is the simplest and most common approach for spotted microarray image gridding. It uses two 1-D signals (axis projections) obtained by projecting the integral of pixel intensities along columns and rows onto the horizontal and vertical axes. The crests in the two signals indicate the horizontal and vertical locations of the spots. This approach obviously assumes that the grid is aligned to the horizontal and vertical axes. To overcome this limitation, the Radon transform [5] has been used (for instance in [6], [7]) to correct any arbitrary rotation of the grid.

To better account for noise and artifacts, more sophisticated methods such as, wavelet analysis [8], morphological filtering [9] and optimal multi-level thresholding (OMTG) [10] have also been used to analyze the axis projections. Among these, OMTG is the only parameter-free method. However, OMTG (like the other axis projection based methods) is still prone to failure when a significant portion of the image is corrupted. Moreover, the axis projection based methods cannot deal with non-linear deformations of the grid.

Rueda and Vidyadharan introduced a hill climbing based gridding approach [11] that uses different parametric distributions to perform gradient ascent in the 2-D pixel intensity distribution and detects modes (corresponding to spots) in the image. As with the axis projection based approaches, the hill climbing approach also assumes the grid to be aligned with the horizontal and vertical axes.

Antoniol and Ceccarelli [12] parameterized the grid as a 6-tuple (the 2-D location vector of the top-left spot, the horizontal and vertical gaps and the rotation angles of the two axes). A meta-heuristic search strategy combining an evolutionary algorithm with hill climbing is designed to search (the 6-D parameter space) for the grid that is best aligned to a previously detected set of guide spots.

Zacharia and Maroulis ([13], [14]) formulated the gridding problem as a search for parallel and equidistant lines separating the spots. An evolutionary search strategy is used to separately search for the two sets of lines (one set of horizontal and another set of vertical) separating the spots. Bariamis et al. [15] proposed a max margin gridding approach that uses support vector machines (SVMs) to optimally separate the rows (columns) of spots. Both the max margin approach and the evolutionary search strategy seeking parallel lines do not consider non-linear deformations of the grid.

Hartelius and Carstensen introduced a Bayesian approach [16] to detect both square and hexagonal grids in generic (non-microarray) images. Katzer et al. also developed a Markov Random Field (MRF) based approach [17] for simultaneously gridding and segmenting spots in microarray images. It is a semi-automated method and requires the number of rows and columns as input. Antoniol and Ceccarelli also proposed semi-automatic Bayesian grid refinement schemes ([6], [12]) to handle non-linear deformations of the grid.

The only gridding techniques for hexagonal grids of bead based microarrays that we are aware of were developed by Giannakeas et al. [18] and Galinsky [19]. However, both these

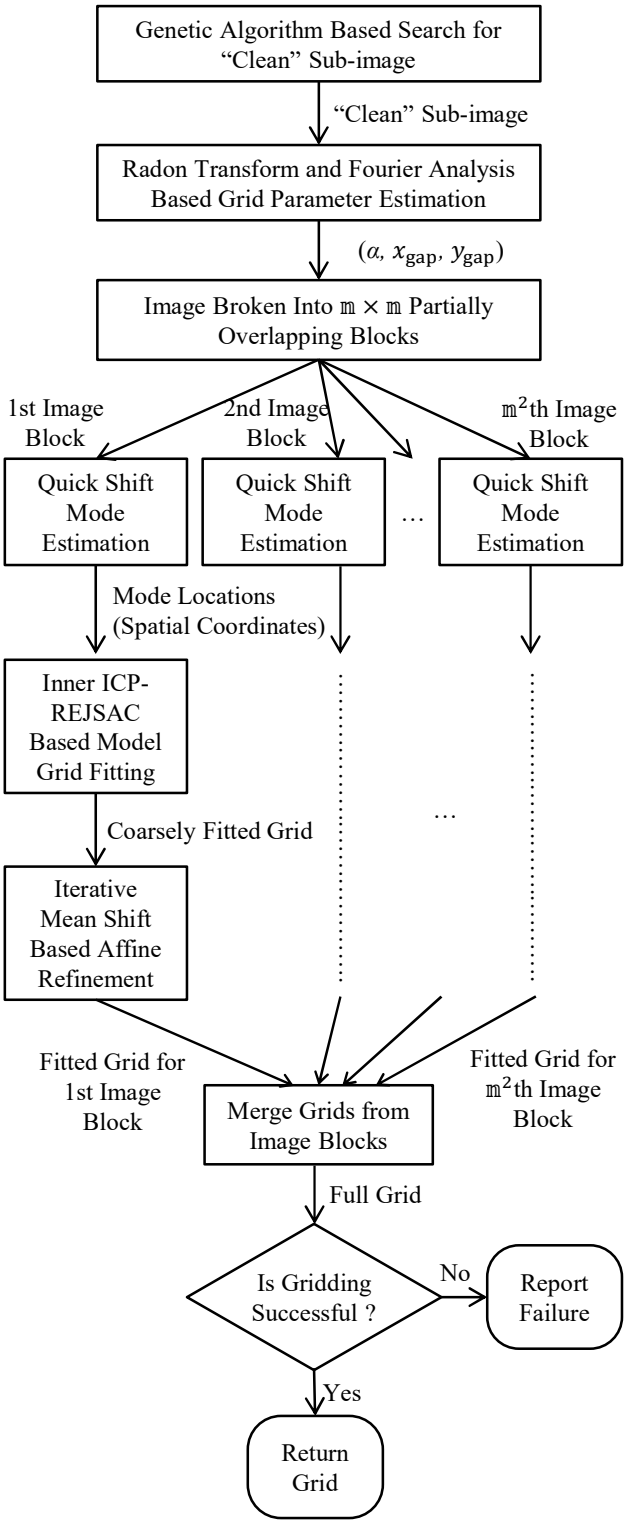


Fig.2. Overview of the gridding method.

techniques were designed for gridding relatively clean fluorescence images and are not well-suited for dirty bright field images.

III. The Proposed Method

The proposed gridding method consists of three key segments: 1) grid parameter estimation, 2) grid fitting, and 3) grid refinement. A schematic diagram of the proposed method is presented in Fig. 2.

In the grid parameter estimation, the Radon transform and the 1-D axis projections are used to generate the initial estimate of grid parameters. However, instead of using the entire image, first an evolutionary algorithm is used to search for a small patch in the image that is free from distortions and artifacts. Axis projection is then applied only to the small clean patch to estimate the initial grid parameters. The proposed grid parameter estimation is elaborated in section III A.

To perform grid fitting the image is broken into smaller blocks and an ideal hexagonal grid is fitted to each block. Quick shift mode estimation [20] is used to approximately locate the wells in the blocks. A variant of the Iterative Closest Point [21] pattern matching algorithm is then used within an Inner-RANSAC [22] like robust parameter estimation scheme to rigidly fit the ideal hexagonal grid to the approximately located wells. The proposed grid fitting method is referred to as the *Inner ICP – REJSAC (Inner Iterative Closest Point-Rejection Sampling and Consensus)* based grid fitting. The method is detailed in section III B.

The fitted grid is then refined using an iterative approach referred to as the *Iterative Mean Shift based Affine Refinement*. The iterative refinement alternates between estimating local grid deformations at selected well locations and estimating a smooth global deformation of the grid based on the local deformations. The mean shift algorithm [23] is used to estimate the local deformations at well locations and an affine global deformation of the grid is estimated. The refinement scheme is discussed in section III C.

After the grids have been refined for all the blocks they are joined to form the non-linear (piecewise affine) grid. The joining involves removing duplicate wells in the overlapping block regions and performing a check to ensure that the grids of all the blocks agree. The grid merging and checking steps are discussed in section III D.

A. Grid Parameter Estimation

Let $\mathbb{B} = \{b_i\}$ be a bright field image of width z_W and height z_H where $b_i \in \mathbb{Z}^+$ and $\mathbf{x}_i = [x_i, y_i]^T$ is the spatial coordinates of the i^{th} pixel in $\mathbb{B}$. The objective of gridding is to locate the spatial coordinates of the well centers $\mathbf{g}_i = [x_i, y_i]^T$ that form the grid. An ideal hexagonal grid can be defined using five parameters; a rotation angle α, two distances x_{gap} and y_{gap} and the 2-D spatial coordinate of the top left well. The first three parameters are shown in Figs. 3(b) and (c). In this section, we discuss the proposed method for estimating these three parameters: the rotation angle α and the distances x_{gap} and y_{gap}. First the Radon transform is used to estimate the angle α, and after rotating the image according to α, x_{gap} and y_{gap} are computed by analyzing the Fourier spectrum [24] of the axis projections. Since noise and artifacts can significantly affect the axis projections, a Genetic Algorithm (GA) is used to search for a patch of the image that is likely to be noise and

artifact free and the parameters are estimated using only the patch instead of the entire image.

A patch (or sub-image) $\mathcal{B}$ of image $\mathbb{B}$ is defined using a 4-tuple $[x_t, y_t, \mathbb{s}_w, \mathbb{s}_h]$, where (x_t, y_t) is the spatial coordinate of the top-left corner pixel of $\mathcal{B}$ (in the coordinate system of $\mathbb{B}$), and $\mathbb{s}_w$ and $\mathbb{s}_h$ are respectively the width and height of $\mathcal{B}$. In the proposed GA, the chromosome is represented as a sequence of the four integer values stored in these four variables. The effects of non-linear or affine distortions are more pronounced in large sub-images than in small sub-images. Therefore, to avoid selecting sub-images that are too large, the permissible values of $\mathbb{s}_w$ and $\mathbb{s}_h$ are restricted to the ranges $[100, (z_W/10)]$ and $[100, (z_H/10)]$ respectively.

To compute the fitness score for a given sub-image, the wells in the sub-image are extracted using a connected component analysis. Otsu's thresholding [25] is used to generate a binary image from which connected components are extracted. In a noise-free (good) sub-image, the connected components (CCs) are likely to correspond to the wells. Otsu's thresholding will produce a good foreground and background segmentation only when their intensity distributions are well separated; a property that is desirable in a good sub-image. A good sub-image is also characterized by equally sized wells placed with uniform spacing throughout the sub-image. Low variance in the distance between neighboring CCs (σ_n) and low variance in size of the CCs (σ_a) are both indicators of a good sub-image. Also, the index of dispersion (I_{disp}) [26] of the CCs in a good sub-image is expected to be close to one. The index of dispersion is calculated by breaking the sub-image into a grid of 3x3 blocks. Note that a larger sub-image will make the parameter estimation more robust and therefore, the number of CCs in the sub-image (n_b) should also be maximized. Based on these, a quality (fitness) score of a sub-image $\mathcal{B}$ is defined as:

$$\psi_{\text{fitness}} = \frac{n_b}{\sigma_a \sigma_n |1 - I_{\text{disp}}| + 1}. \quad (1)$$

The GA is initialized with a population of 50 sub-images that are uniformly distributed throughout the image and have random widths and heights (within their respective permissible ranges). A deterministic tournament selection strategy is used to select reproducing parents along with 10% elitist selection. A single-point crossover is applied to all reproducing parents and mutations are introduced with a 50% probability. The algorithm is allowed to evolve for 10 generations.

Once a good sub-image $\mathcal{B}_g$ has been found the rotation of the grid is detected using the Radon transform. Fig. 3(c) shows an image after rotation correction. The Radon transform gives the 1-D projection of the image along a projection line as:

$$R(\theta, d) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} b(x, y)\delta(d - x\cos\theta - y\sin\theta)dxdy, \quad (2)$$

where $b(x, y)$ is the intensity of the pixel at location (x, y), θ is the angle between the projection line and the x-axis, d is the shortest displacement from origin to the projection line and $\delta(\cdot)$ is the Dirac delta function. $R(\theta, d)$ is the integral of pixel intensities along the straight line defined by θ and d. The

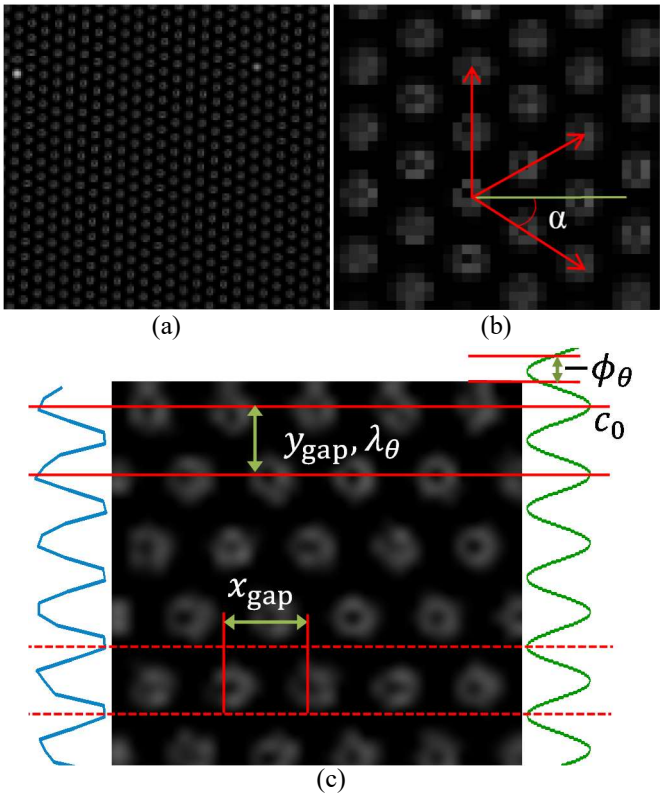


Fig.3. (a) Small portion of a bright field image, (b) closer look at the bright field image and the rotation angle α, (c) a region of the bright field image shown in (a) after counter clockwise rotation by α degrees.

$R(90°, d)$ for the image in Fig. 3(c) is shown on the left of the image (blue curve).

In a hexagonal grid, there are six possible values of α that can be used to rotate the image such that the rows of wells are aligned horizontally (the three directional vectors shown using red arrows in Fig. 3(b) and their opposite vectors). In an ideal grid, the angle between these vectors is 60°. However, to account for small affine distortions, the range of the search for α is set to $[0°, 70°]$. The step size of the search is set to 1°. Let $R_\theta(d) = R(\theta, d)$, where $\theta = 0°, 1°, \ldots, 70°$ be the Radon Transform of image $\mathcal{B}_g$ for a certain θ. For each θ, the Fourier spectrum of $R_\theta(d)$ is analyzed. For a discrete $R_\theta(d)$ with $d = 0, \ldots, N-1$, the Fourier spectrum is computed as:

$$F_\theta(K) = \sum_{d=0}^{N-1} R_\theta(d) e^{-i2\pi K \frac{d}{N}}, \qquad K = 0, \ldots, N-1. \quad (3)$$

The wavelength λ_θ and the phase-shift ϕ_θ (both in pixels) corresponding to the dominant frequency in $R_\theta(d)$ are then estimated from F_θ. When the rows of wells are horizontally aligned, λ_θ gives the distance between the rows, as shown in Fig. 3(c). The location of the crest c_i in $R_\theta(d)$ is determined by $c_i = -\phi_\theta + (i + 0.75)\lambda_\theta$. $-\phi_\theta$ corrects for the phase shift and the first crest of the phase corrected wave (a sine wave) is found at $0.75\lambda_\theta$. A score ψ_θ is then computed for each θ as:

$$\psi_\theta = \frac{\left((\boldsymbol{LR})^T \mathbf{1}_p\right) * \left(\left((\mathbf{1}_{pN} - \boldsymbol{L})\mathbf{1}_N\right)^T \mathbf{1}_p\right)}{\left((\boldsymbol{L}\mathbf{1}_N)^T \mathbf{1}_p\right) * \left(\left((\mathbf{1}_{pN} - \boldsymbol{L})\boldsymbol{R}\right)^T \mathbf{1}_p\right)}. \quad (4)$$

where p is the number of crests detected in R_θ, $\boldsymbol{R} = [R_\theta(0), \dots, R_\theta(N-1)]^{\boldsymbol{T}}$, $\boldsymbol{L}(i,j) = \begin{cases} 1, & |c_i - j| \le 0.25\lambda_\theta \\ 0, & |c_i - j| > 0.25\lambda_\theta \end{cases}$ is an $p \times N$ logical matrix, $\mathbf{1}_{pN}$ is an $p \times N$ matrix of ones, and $\mathbf{1}_p$ and $\mathbf{1}_N$ are $p \times 1$ and $N \times 1$ vectors of ones. The rotation angle α is determined by:

$$\alpha = \underset{\theta \in [0°, 70°]}{\arg\max}\ \psi_\theta. \quad (5)$$

The λ_θ corresponding to F_α is considered as y_{gap}. Image $\mathcal{B}_g$ is then rotated by α degrees to get the rotated image $\mathcal{B}'_g$, similar to the image shown in Fig. 3(c).

Using the detected crests c_i and well height λ_θ, n_r (row) sub-images are extracted from $\mathcal{B}'_g$, such that each (row) sub-image contains a single row of wells. One (row) sub-image is shown in Fig. 3(c) using dashed red lines. The Fourier spectrum of the vertical intensity projections of a (row) sub-image is used to produce one estimate of the horizontal gap (x_{gap}) between the wells. The median of the n_r estimates obtained from the n_r (row) sub-images is considered as x_{gap}.

To facilitate the quick shift mode estimation using a symmetric kernel in the next stage, image $\mathbb{B}$ is rescaled along the horizontal axis using a scaling factor ($x_{\text{gap}}/y_{\text{gap}}$) so that the horizontal and vertical gaps between wells in the rescaled image $\mathbb{B}'$ are equal.

B. Grid Fitting

To make the proposed gridding method robust against non-linear distortions, $\mathbb{B}'$ is broken down into $\mathbb{m}^2$ overlapping blocks ($\mathcal{B}'$) in a $\mathbb{m} \times \mathbb{m}$ grid pattern and the grid fitting and refinement processes are performed locally and separately for each image block.

This section discusses the steps involved in fitting an ideal grid $\boldsymbol{M}_{n\times 2} = [\mathbf{m}_1, \mathbf{m}_2, \dots \mathbf{m}_n]^T$, where $\mathbf{m}_i = [x_i, y_i]^T$, to a given image block $\mathcal{B}'$. In an ideal hexagonal grid all the points are at exactly the same distance from their 6 neighboring points. A pre-computed ideal grid can be used as long as the grid is large enough to cover the entire image block. The ideal grid is rotated and scaled to fit the image block.

The grid fitting method involves two sub-steps, 1) detecting a set of likely well locations (guide points) in the image block and 2) estimating the similarity transform that aligns the ideal grid M with the set of detected guide points.

Throughout the rest of the paper, the following notations are used to describe neighborhoods: a rectangular region of size $n \times m$ centered at $\mathbf{x}$ is denoted as $H(\mathbf{x}, n, m)$ and a circular region of radius r centered at $\mathbf{x}$ is denoted as $H(\mathbf{x}, r)$.

1) Guide Point Detection

When considering a bright field image as a 2-D density function, the wells in the image can be viewed as the modes in the density. Therefore, mode seeking algorithms such as quick shift and mean shift ([20], [23]) can be used to locate the wells. In the proposed method, quick shift is used to detect guide points and mean shift is used in the grid refinement step. Both quick shift and mean shift rely on the same principle of gradient ascent. Starting from an arbitrary location $\mathbf{x}$ in the feature space, $\mathbf{x}$ is repeatedly shifted in the opposite direction of the density gradient to reach the peak of the hill that $\mathbf{x}$ is on.

When discrete density estimates are available, the underlying density function P at a point $\mathbf{x}$ can be estimated as the convolution between a suitable kernel K and the empirical density estimates:

$$P(\mathbf{x}) = \sum_{\mathbf{x_i} \in \boldsymbol{H}(\mathbf{x}, r, r)} K(\mathbf{x_i} - \mathbf{x}) \mathbb{b}_i. \quad (6)$$

where $\mathbb{b}_i$ is the empirical density estimate at $\mathbf{x_i}$ and r is the kernel bandwidth. For symmetric kernel functions such as a Gaussian or a uniform kernel, the mean shift vector at $\mathbf{x}$,

$$\mathbf{x}_{\text{shift}} = \frac{\sum_{\mathbf{x_i} \in \boldsymbol{H}(\mathbf{x}, r, r)} K(\mathbf{x_i} - \mathbf{x}) \mathbb{b}_i (\mathbf{x_i} - \mathbf{x})}{\sum_{\mathbf{x_i} \in \boldsymbol{H}(\mathbf{x}, r, r)} K(\mathbf{x_i} - \mathbf{x}) \mathbb{b}_i}, \quad (7)$$

points to the opposite direction of the gradient of the convolution surface P [27].

While in mean shift the point $\mathbf{x}$ is shifted smoothly in the feature space, in quick shift, $\mathbf{x}$ shifts to the nearest neighbor data point $\mathbf{x}'$ which has a higher density. Shifting the data points repeatedly in this manner connects all the data points into a tree. Any point $\mathbf{x}$ that shifts to another point $\mathbf{x}'$**,** is a child of $\mathbf{x}'$ in the tree**.** Local modes are extracted from the tree by breaking of branches that represent shifts greater than a user specified distance d in the feature space. It naturally follows, that if at a given data point the density is higher than the density at any other data point that lies within a distance d of it, then the given data point must be a local mode. The proposed guide point detection works in a similar way: if at a given pixel $\mathbf{x}$**,** the density $P(\mathbf{x})$ is greater than the density at any other pixel that lies within the $H(\mathbf{x}, x_{\text{gap}}, x_{\text{gap}})$ region, then pixel $\mathbf{x}$ must contain a well center. Formally, a pixel $\mathbf{x}$ is denoted as a well center if

$$\underset{\mathbf{x}_i \in H(\mathbf{x}, x_{\text{gap}}, x_{\text{gap}})}{\text{argmax}} \quad P(\mathbf{x}_i) \quad (8)$$

is $\mathbf{x}$. The underlying density $P(\mathbf{x}_i)$ at the pixel center $\mathbf{x}_i = [x_i, y_i]^T$ is estimated using equation (6). The pixel intensity b_i is used as the empirical density estimate at $\mathbf{x}_i$ and a symmetric Gaussian kernel K of radius $r = x_{\text{gap}}$ and standard deviation ($x_{\text{gap}}/4$):

$$K(z) = \frac{1}{{x_{\text{gap}}}^2 \sqrt{2\pi \left(\frac{x_{\text{gap}}}{4}\right)^2}} e^{-\frac{||\mathbf{z}||^2}{2\left(\frac{x_{\text{gap}}}{4}\right)^2}} \quad (9)$$

is used as the kernel function.

2) Transformation Estimation

To estimate an affine transformation between the model grid $\boldsymbol{M}$ and the guide points $\boldsymbol{S}_{\ell\times 2} = [\mathbf{s}_1, \dots \mathbf{s}_\ell]^{\text{T}}$, where $\mathbf{s}_i = [x_i, y_i]^T$ a variant of the Inner-RANSAC algorithm called *Inner*

ICP-REJSAC (Inner Iterative Closest Point-Rejection Sampling and Consensus) is proposed. As the name suggests, the *Inner ICP-REJSAC* method uses a rejection sampling technique rather than random sampling, to select the samples from which the parameters of the similarity transform are estimated.

The rejection sampling is performed by selecting a triplet of points from the set of guide points S, such that the points are nearest neighbors of each other. Let the guide points selected in the k^{th} iteration be $\boldsymbol{S}_{3\times2}^{k} = [\mathbf{s}_1^k, \mathbf{s}_2^k, \mathbf{s}_3^k]^T$. The selected sample $\boldsymbol{S}^k$ is only considered further if the three pairwise Euclidean distances d_{12}^k, d_{13}^k and d_{23}^k, defined as $d_{ij}^k = [\mathbf{s}_i^k - \mathbf{s}_j^k]^T[\mathbf{s}_i^k - \mathbf{s}_j^k]$, are all greater than $0.75x_{\text{gap}}$ and less than $1.25x_{\text{gap}}$; otherwise the sample is rejected. The range $(0.75x_{\text{gap}}, 1.25x_{\text{gap}})$ was selected empirically. Three points $\boldsymbol{M}_{3\times2}^{'} = [\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3]^T$, which are nearest neighbors of each other are selected from the model grid M. Note that in the model grid all triplets of points which are nearest neighbor of each other are equivalent and therefore only one $\boldsymbol{M}'$ needs to be chosen at the start.

If a sample $\boldsymbol{S}^k$ is not rejected the similarity transformation between $\boldsymbol{M}'$ and $\boldsymbol{S}^k$ is computed by solving the system of linear equations:

$$\boldsymbol{S}^k = \boldsymbol{A}^k\boldsymbol{M}' + \boldsymbol{L}^k\,, \tag{10}$$

where $\boldsymbol{A}^k = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$ encodes the scaling and rotation parameters and $\boldsymbol{L}^k = \begin{bmatrix} t_x \\ t_y \end{bmatrix}$ is the translation vector. The model grid points $\boldsymbol{M}$ are then transformed using $\boldsymbol{A}^k$ and $\boldsymbol{L}^k$ to generate the transformed model grid points $\boldsymbol{T}^k = \boldsymbol{A}^k\boldsymbol{M} + \boldsymbol{L}^k$.

RANSAC assumes that the parameters estimated from outlier-free samples will be consistent with all inliers. However this is rarely the case. For instance, the transformation estimated from three pairs of true correspondences may not be the optimal alignment between the two point sets, if the points are perturbed by noise. To overcome this problem Inner-RANSAC was developed by Chum et al. [22]. In Inner-RANSAC a local optimization is performed to refine the estimated model parameters after the initial estimate has been generated from the samples. In the proposed *Inner ICP-REJSAC* method the local optimization is performed using a variant of the Iterative Closest Point (ICP) algorithm that uses bootstrapping and least trimmed squares.

In the Dual-Bootstrap ICP [28], bootstrapping regions are defined for both the point sets (the grid points and the guide points) and the transformation parameters are estimated using only those correspondences that lie within the bootstrap regions. The idea being that, if a good alignment can be established between the two bootstrapping regions, then by bootstrapping the registration in these regions and then growing them in subsequent iterations, a better alignment can be established. In the proposed method, before the local optimization (the inner ICP) is performed, an initial transformation that registers the points in $\boldsymbol{M}'$ and $\boldsymbol{S}^k$ was already established. The neighborhoods of $\boldsymbol{M}'$ and $\boldsymbol{S}^k$ are also likely to be well registered under this initial transformation. Therefore these neighborhood regions are used as the bootstrap regions. Note that here the initial alignment is generated from all-inlier samples and therefore is close to optimal. The circular bootstrap region for the n^{th} iteration is defined as $H(\mathbf{s}_c, \rho)$ where $\rho = 1.25x_{gap} * n$ and $\mathbf{s}_c$ is the mean of the three points in $\boldsymbol{S}^k$. Correspondences between $\mathbf{s}_i$ and $\mathbf{t}_i \in \boldsymbol{T}^k$ that lie within $H(\mathbf{s}_c, \rho)$ are used to bootstrap the local optimization (Inner-ICP). The correspondences are established using a nearest-neighbor approach.

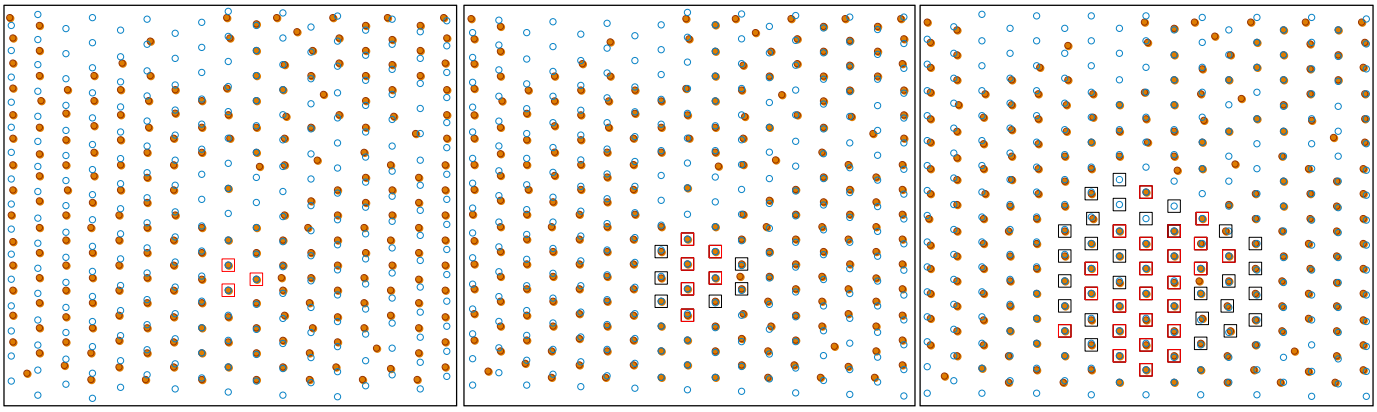
Fig. 4 illustrates the bootstrapping method. The two point sets are shown as blue empty circles and orange filled circles. Fig. 4(a) shows the initial alignment obtained using the three corresponding pairs (points inside the red boxes). In Figs. 4(b) and (c), the bootstrap region (illustrated as points inside boxes) is extended and the alignment quality is also improved.

The Trimmed ICP [29] uses only a subset of the correspondences that are likely to be good correspondences, rather than using all possible correspondences to calculate the least squares transformation parameters. This makes the alignment process robust to outliers. A similar approach is employed in the proposed method. Only those correspondences that lie within the bootstrap region, and for which the alignment error is less than a threshold, are used to estimate the transformation parameters. In Figs. 4(b) and (c), the correspondences that are used to estimate the alignment parameters are marked with red boxes and the correspondences that are marked with black boxes are trimmed.

Let $\{\mathrm{s_i}, \mathrm{t_i}\}$ be a correspondence between a guide point $\mathrm{s_i}$ and a transformed model point $\mathrm{t_i} \in \mathrm{T_h^k}$ where $\mathrm{T_h^k}$ is the transformed model grid point set estimated in the h^{th} iteration of the local optimization (inner iteration). For each correspondence inside the bootstrap region, the squared Euclidean distance $\mathrm{d_i} = (\mathrm{s_i} - \mathrm{t_i})^{\mathrm{T}}(\mathrm{s_i} - \mathrm{t_i})$ is calculated. The correspondence pairs, for which $\mathrm{d_i}$ is less than the median of the $\mathrm{d_i}$s, are used to estimate the new similarity transformation matrices $\mathrm{A^{hk}}$ and $\mathrm{L^{hk}}$ and the new transformed model grid point set $\mathrm{T_{h+1}^k}$, for the $(\mathrm{h}+1)^{\text{th}}$ inner iteration.

The local optimization (inner iterations) continues till the bootstrap region covers the entire image block $\mathcal{B}'$. The final transformed model grid $\boldsymbol{T}_f^k$ is used to calculate the mean squared error (e_{mse}^k) for the sample $\boldsymbol{S}^k$ as:

$$e_{mse}^k = \frac{tr\left(\left(\boldsymbol{L}\boldsymbol{T}_f^k - \boldsymbol{S}\right)\left(\boldsymbol{L}\boldsymbol{T}_f^k - \boldsymbol{S}\right)^{\boldsymbol{T}}\right)}{\mathbf{1}_n{}^{\boldsymbol{T}}\boldsymbol{L}^T\mathbf{1}_\ell}, \tag{11}$$



(a) (b) (c)

Fig.4. The bootstrapped and least trimmed squares ICP algorithm. The two point sets are shown using orange dots and blue circles. (a), (b) and (c) the first, second and third iterations of the algorithm. The correspondences marked with squares form the bootstrap region. Only the correspondences marked with red boxes are used to estimate the transformation in each iteration.

where $\boldsymbol{L}(i,j) = \begin{cases} 1, & if\ \{\mathbf{s}_i, \mathbf{t}_j\}\ is\ a\ correspondence \\ 0, & otherwise \end{cases}$ is an $\ell \times n$ logical matrix, $tr(\cdot)$ returns the trace of a matrix, and $\mathbf{1}_n$ and $\mathbf{1}_\ell$ are $n \times 1$ and $\ell \times 1$ vectors of ones.

In RANSAC [30] the number of iterations (k) required to achieve a probability of success η is:

$$k = \frac{\log(1-\eta)}{\log(1-\epsilon^3)} \tag{12}$$

where ϵ is the fraction of inliers among the guide points. Since in the proposed *Inner ICP-REJSAC* method rejection sampling is used, ϵ is set to a relatively high value of 0.75. Using $\eta = 0.9999$ the value of k is 17 (rounded off from 16.808) and therefore 17 samples are drawn through rejection sampling.

The transformed model point set $\boldsymbol{T}_f^k$ with the minimum mean squared error e_{mse}^k is selected as the rigidly aligned grid $\boldsymbol{\mathcal{G}}$ for the image block $\mathcal{B}'$.

C. Grid Refinement

In this section, the *Iterative Mean Shift based Affine Refinement* is discussed. This step uses a similar iterative bootstrapping and trimming approach as the previous step to prevent the refinement from deforming the grid wildly from the initial rigid alignment. The bootstrap region is initialized and grown in the same way as in the grid fitting step. Let $\mathbf{g} \in \boldsymbol{\mathcal{G}}$ be one of n grid points inside the bootstrap region in the k^{th} iteration. These grid points are used to initialize n mean shift mode estimations.

The mean shift vector $\mathbf{g}_{\text{shift}}$ for each $\mathbf{g}$ is computed using equation 7. The pixel intensities are used as the empirical density estimates at pixel centers and an $x_{\text{gap}} \times x_{\text{gap}}$ uniform kernel is used as the kernel function.

Starting at $\mathbf{g}$ the grid point is repeatedly shifted until a local maximum is reached. Although the mean shift algorithm is guaranteed to converge to some local maxima indicated by a zero gradient, in practice the rate of convergence can become very slow when the window gets close to the maxima. Therefore, the mean shift process is stopped when the shift falls below 0.1 pixels i.e., when $||\mathbf{x}_{\text{shift}}|| < 0.1$.

Fig. 5 shows the mean shift mode detection process. The green squares indicate the initial location of the kernel centers, the red dots show the positions of the shifted kernel centers and the blue dots represent the located modes.

In noisy and dirty images the mean shift can sometimes shift the grid point farther away from the grid center. To negate this error, the mean shifts for the grid points are smoothed using a circular median filter with a radius of $2x_{\text{gap}}$ i.e., a filter centered at a grid point and covering its 12 neighboring grid points. For each $\mathbf{g}$ its smoothly shifted version $\hat{\mathbf{g}}$ is computed as:

$$\hat{\mathbf{g}} = \mathbf{g} + \text{median}\ \left([(\mathbf{x}_1 - \mathbf{g}_1), \ldots, (\mathbf{x}_j - \mathbf{g}_j)]\right), \tag{13}$$

where $\mathbf{g}_i \in H(\mathbf{g}, 2x_{\text{gap}})$, $\mathbf{x}_i = [x_i, y_i]^T$ is the mode detected by mean shifting from $\mathbf{g}_i$ and $\text{median}(\boldsymbol{X})$ returns the $n \times 1$ median vector of the $n \times m$ matrix $\boldsymbol{X}$. A smoothness score ψ_{smooth} is then computed for each smoothed $\hat{\mathbf{g}}$ as:

$$\psi_{\text{smooth}} = \frac{1}{tr\left(\text{cov}([(\hat{\mathbf{g}}_1 - \mathbf{g}_1), \ldots, (\hat{\mathbf{g}}_j - \mathbf{g}_j)])\right)}, \tag{14}$$

where $\mathbf{g}_i \in H(\mathbf{g}, 2x_{\text{gap}})$, $\hat{\mathbf{g}}_i$ is the smoothed version of $\mathbf{g}_i$, $tr(\cdot)$ returns the trace of a matrix and $\text{cov}(\boldsymbol{X})$ returns the $n \times n$ covariance matrix of a $n \times m$ matrix $\boldsymbol{X}$. The points in $\boldsymbol{\mathcal{G}}$ whose corresponding ψ_{smooth} scores are lower than the median of the ψ_{smooth} scores are selected as the control points ($\boldsymbol{\mathcal{G}}'$) for estimating an affine deformation. A lower ψ_{smooth} score for a point indicates that the differences in shifts between that point and its neighboring points are small, meaning that the shifts are in good agreement.

Using the selected (untrimmed) control points $\boldsymbol{\mathcal{G}}' = [\mathbf{g}_1', \ldots, \mathbf{g}_q']^T$, and their corresponding smoothed versions $\hat{\boldsymbol{\mathcal{G}}}' = [\hat{\mathbf{g}}_1', \ldots, \hat{\mathbf{g}}_q']^T$, an affine transformation matrix $\boldsymbol{A}$ is computed by solving the system of linear equations:

$$\hat{\boldsymbol{\mathcal{G}}}_1' = \boldsymbol{A}\boldsymbol{\mathcal{G}}_1', \tag{15}$$

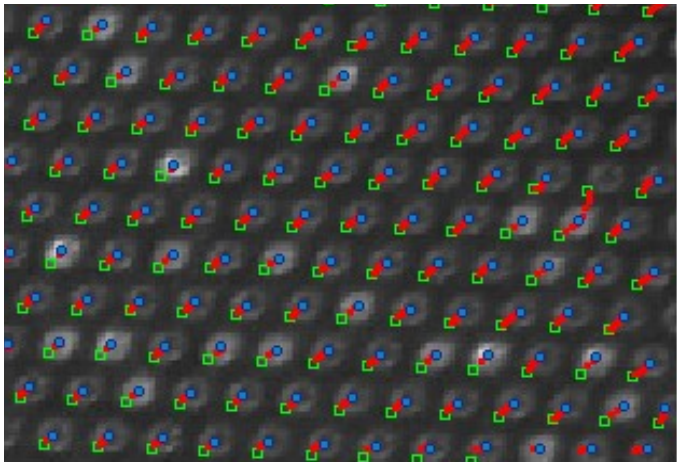


Fig.5. Mean shift mode estimation. An inverted bright field image with the mean shift initialization locations marked with green squares, the location of the center of the kernel windows during the mean shift iterations marked with red dots and the detected modes marked with blue circles.

where $= \begin{bmatrix} a_{11} & a_{12} & t_1 \\ a_{21} & a_{22} & t_2 \\ 0 & 0 & 1 \end{bmatrix}$, $\boldsymbol{\mathcal{G}_1}' = \begin{bmatrix} \mathbf{g}_1{}' & \mathbf{g}_2{}' & \mathbf{g}_3{}' \\ 1 & 1 & 1 \end{bmatrix}$ and $\widehat{\boldsymbol{\mathcal{G}}_1}' = \begin{bmatrix} \hat{\mathbf{g}}_1{}' & \hat{\mathbf{g}}_2{}' & \hat{\mathbf{g}}_3{}' \\ 1 & 1 & 1 \end{bmatrix}$. $\boldsymbol{\mathcal{G}}$ is then transformed using $\boldsymbol{A}$ to generate $\widehat{\boldsymbol{\mathcal{G}}} = \boldsymbol{A}\boldsymbol{\mathcal{G}}$. $\widehat{\boldsymbol{\mathcal{G}}}$ is then set as the new $\boldsymbol{\mathcal{G}}$. The bootstrap region is grown and the process is repeated until the entire block $\mathcal{B}'$ has been covered by the bootstrap region.

D. Grid Merging and Checking

Once the grids have been established for all the image blocks, duplicate grid points (points representing the same well) are identified, as grid points from different image blocks that lie in an overlapping region and are within a distance of $0.25x_{\text{gap}}$ of each other. Typically duplicate points are very close to each other, in most cases less than a pixel apart. For each set of duplicate points, the mean intensity of pixels within a distance $0.375x_{\text{gap}}$ of the points are computed, and only the point with the highest mean intensity is kept.

If any two or more grid points of an image block in the overlapping region are not within a distance of $0.25x_{\text{gap}}$ of any grid points in another image block, the gridding process is assumed to have failed. This is a situation where the grid of one image block is not aligned with the grid of a neighboring image block. In such a situation the gridding process can be restarted.

IV. Experiments and Results

The proposed bead microarray bright field image gridding method was tested on 189 images grouped into 5 datasets. The number of images and the properties of the images in each dataset are listed in Table I. The *High Resolution Dense* (HRD) dataset contained relatively clean high (well) resolution images with tightly packed wells, where the distance between the wells is much less than the size of the wells. The *High Resolution Blurry* (HRB) dataset contained high resolution blurry images with moderate amounts of artifacts. The *Clustered* (CLUST) dataset contained images with medium well resolution and large amounts of bead clustering. The *Low Resolution Clean* (LRC) and *Low Resolution Dirty* (LRD) datasets contained images acquired at 5x optical zoom and had low resolution of the wells. The *Low Resolution Dirty* dataset images additionally had extreme amounts of noise and artifacts, in some cases spanning over 70% of the image area. All the images suffered from varying degrees of spherical aberration. Fig. 6 shows the gridding achieved on an image from the LRD dataset. Notice that the image also has blurring in addition to dust and noise. The insets shown in Figs. 6(c) and 6(e) are less sharp than the insets shown in Figs. 6(b) and 6(d).

The images were obtained using a DSLR camera (EOS 5D Mark II, Canon, Japan) mounted on top of an infinity corrected objective lens (UIS2, Olympus, Singapore). The Ayoxxa biochip (Ayoxxa Living Health Technologies, Singapore) and polystyrene beads from different providers were used for the experiments. The proposed method was implemented in Java and all the experiments were performed on a standard desktop computer with an Intel Xeon Hexa Core 3.2 GHz processor and 16GB RAM. Multithreading was used with up to 12 threads.

The proposed method was run five times for each image. The detected well regions as well as the well centers were plotted on the image with super-pixel accuracy using a MATLAB script and the output was visually analyzed. Successful gridding was achieved on all five occasions for 187 of the images. The proposed method failed to grid one of the images in the HRB dataset with extreme amounts of blur in two out of the five runs. Another image in the LRD dataset failed in one of the five attempts because of extreme amounts of dirt. However, even in all the three failure cases, the gridding did not fail completely. It only failed for a few image blocks. One of the advantages of performing the gridding locally in image blocks is that failure can be detected automatically (while joining the grids) and the gridding process can be automatically restarted.

To objectively evaluate the gridding accuracy, ground truth data was generated by manual labelling. 10 images were randomly selected from each of the five datasets giving a total of 50 images. Each image was broken into 25 blocks in a 5×5 grid pattern. Two individuals who were familiar with the data and were involved in performing the biological experiments were asked to manually draw tight rectangular boxes around one randomly selected well in each of the 25 blocks. The centers of the manually labelled rectangles were used as reference well centers. The accuracy of gridding was calculated using the distance $\mathrm{d_{err}}$ between the manually labelled well center and the corresponding well center detected using the proposed method.

Table II shows the mean, median and standard deviation of the 500 d_{err} values (25×2 labels for each image ×10 images per dataset) obtained for each of the 5 datasets. Both the actual pixel values of d_{err} and the d_{err} as a percentage of the well width, defined as $d_{err\%} = \frac{d_{err}}{d_w} \times 100\%$, where d_w is the width of the well, are presented. Note that $d_{err\%}$ is scale (resolution) invariant. Table II also shows the statistics from the two sets of manual labels separately. The gridding results give good agreement on the two sets of labels for all the datasets. The mean and median of d_{err} for all cases are less than 1 pixel. Notice that, while the well centers estimated using the proposed method are continuous values, the manually marked

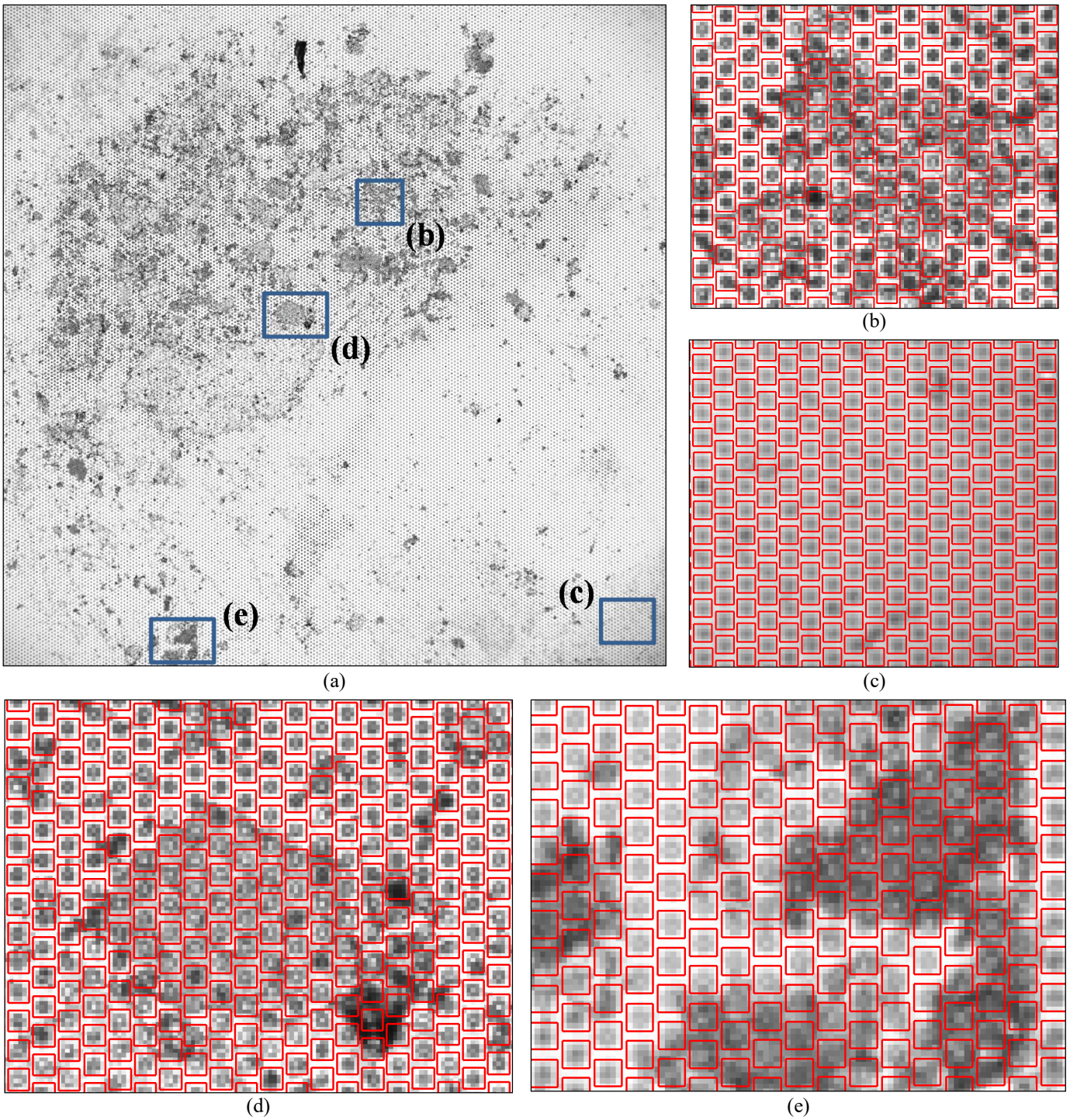


Fig. 6. The gridding achieved on a dirty image from the LRD dataset using the proposed method. (a) the original bright field image. The approximate regions of the image from which the insets shown in figures (b), (c),(d) and (e) are extracted are marked and labelled. (b) the gridding achieved in a region of the original image with large amount of dirt. (c) the gridding achieved in a blurry region of the image. (d) the gridding result on another dirty region of the image. (e) the gridding result on a blurry region with large amounts of dust particles.

well centers have discrete values. During the manual labelling process, the marked rectangles did not have sub-pixel accuracy, and therefore the corresponding well centers could only lie either at the boundary or at the center of a pixel. Even so, a mean d_{err} of 0.31 pixels was achieved for the LRC dataset. Because the wells in the LRD and LRC datasets are only a few (~5 or 6) pixels wide, even small variations between two markings of a well region result in a significant difference in terms of the percent error $d_{err\%}$. Nevertheless, the mean and median of d_{err} is less than 0.85 pixels for all the cases. The slightly higher d_{err} values for the HRB dataset can be attributed to the extreme amount of blur which makes manually identifying the true well boundaries difficult in some cases.

TABLE I.
SUMMARY OF THE FIVE EVALUATION DATASETS

Dataset	Image Properties				
	No. of Images	*Image Resolution (pixels)*	*Approx. Well Resolution (pixels)*	*Approx. Gap Between Wells (pixels)*	*Approx. No. of Wells per Image*
HRD	38	5616 × 3744	14 × 14	8	50000
HRB	13	3416 × 2744	14 × 14	13	20000
CLUST	10	1364 × 1007	8 × 8	6	10000
LRC	64	1300 × 1300	5 × 5	3	40000
LRD	64	1300 × 1300	5 × 5	3	40000

TABLE II
WELL DETECTION ACURACY ON THE FIVE DATASETS

Dataset	Statistic	d_{err} (pixels) [$d_{err\%}$]		
		Overall	*Manual Label 1*	*Manual Label 2*
HRD	Mean	**0.560 [3.73]**	0.556 [3.71]	0.564 [3.76]
	Median	0.522 [3.48]	0.523 [3.48]	0.521 [3.47]
	Std. Dev.	0.281 [1.87]	0.288 [1.92]	0.279 [1.86]
HRB	Mean	**0.764 [5.09]**	0.808 [5.38]	0.719 [4.80]
	Median	0.795 [5.30]	0.841 [5.60]	0.708 [4.72]
	Std. Dev.	0.371 [2.48]	0.364 [2.43]	0.374 [2.49]
CLUST	Mean	**0.328 [3.64]**	0.369 [4.10]	0.286 [3.18]
	Median	0.299 [3.33]	0.330 [3.67]	0.254 [2.82]
	Std. Dev.	0.216 [2.40]	0.214 [2.37]	0.214 [2.38]
LRC	Mean	**0.316 [5.26]**	0.321 [5.35]	0.311 [5.18]
	Median	0.304 [5.06]	0.298 [4.97]	0.308 [5.14]
	Std. Dev.	0.123 [2.06]	0.130 [2.17]	0.116 [1.93]
LRD	Mean	**0.389 [6.48]**	0.370 [6.16]	0.408 [6.80]
	Median	0.360 [6.01]	0.348 [5.79]	0.374 [6.24]
	Std. Dev.	0.192 [3.20]	0.185 [3.08]	0.198 [3.29]

TABLE III
RUNNING TIMES OF THE PROPOSED ALGORITHM

	Database				
Running Times	***HRD***	***HRB***	***CLUST***	***LRC***	***LRD***
Avg. (secs)	74.18	23.06	6.58	4.81	4.95
Std. Dev. (secs)	5.98	1.75	0.21	0.33	0.80

The receiver operator characteristic (ROC) curves of the proposed method are given in Fig. 7. The x-axes of the plots are $d_{err\%}$ and the y-axes of the plots are $1 - err(\tau)$, where $err(\tau)$ is defined as the ratio of manually labelled well centers that do not have their corresponding estimated well centers within a given distance $\tau = d_{err\%}$. The ROC curves of the proposed method before the grid refinement are also shown (blue lines) to demonstrate the improvement achieved through the refinement process. In the cases of the HRD, HRB, and CLUST datasets the d_{err} are less than 2 pixels for all the manually labelled wells and for the LRC and LRD datasets the d_{err} are less than 1 pixel for almost all the manually labelled wells.

An example of a grid before and after refinement is shown in Fig. 8. The image shows a region of the grid where four blocks overlap. Fig. 8(a) shows the grids detected before the refinement. For each well, only one of the block's grid is shown. Fig. 8(b) shows the grid in the same region after refinement. The grid in Fig. 8(b) is smooth whereas the grid points in Fig. 8(a) belonging to the four blocks are distinguishable. In particular, the grid points in Fig. 8(a) belonging to the top-left corner block can be easily identified, as they are heavily skewed towards the top right of the wells.

The run-times of the proposed method are presented in Table III. They are primarily dependent on the well resolution and well density and are only marginally affected by the level of distortion and noise in the image. The average running time for the dirty LRD images, for instance, is only slightly higher than that of the clean LRC images.

V. DISCUSSION AND CONCLUSIONS

Gridding is often the first step in many microarray image analysis tasks and has continued to attract considerable interest from the computer vision community over the past decade. In this paper, a novel method is presented to grid bead microarray bright field images. The proposed method uses a robust Genetic algorithm to search for clean regions in the image and then learns the parameters of the grid from them. It then fits a model grid to a set of detected guide points and then progressively refines the grid through a number of controlled fitting steps.

Microarray experiments produce large amounts of image data and manual intervention can be a major bottleneck in the analysis pipeline. Automatic gridding has therefore been the holy grail of gridding algorithms. While the proposed method does have some parameters, for most cases these parameters will not need tuning.

Our experiments show that with the configuration outlined above, the GA is able to find very high quality sub-images consistently in a varied set of images. Small variations in the GA parameters have little impact on the outcome. In fact, for moderately distorted images a primitive random search evaluating a sufficiently large number of image patches is often able to find patches of acceptable quality. The GA also performs a random search but it further improves the outcome by searching the neighborhoods of good solutions. In extreme cases, where large amounts of noise and distortions extend throughout the image, it may be useful to compute the grid parameters multiple times from the n best patches found through the GA, provided that the differences of the scores between the best patch and the $n - 1$ other selected patches are small.

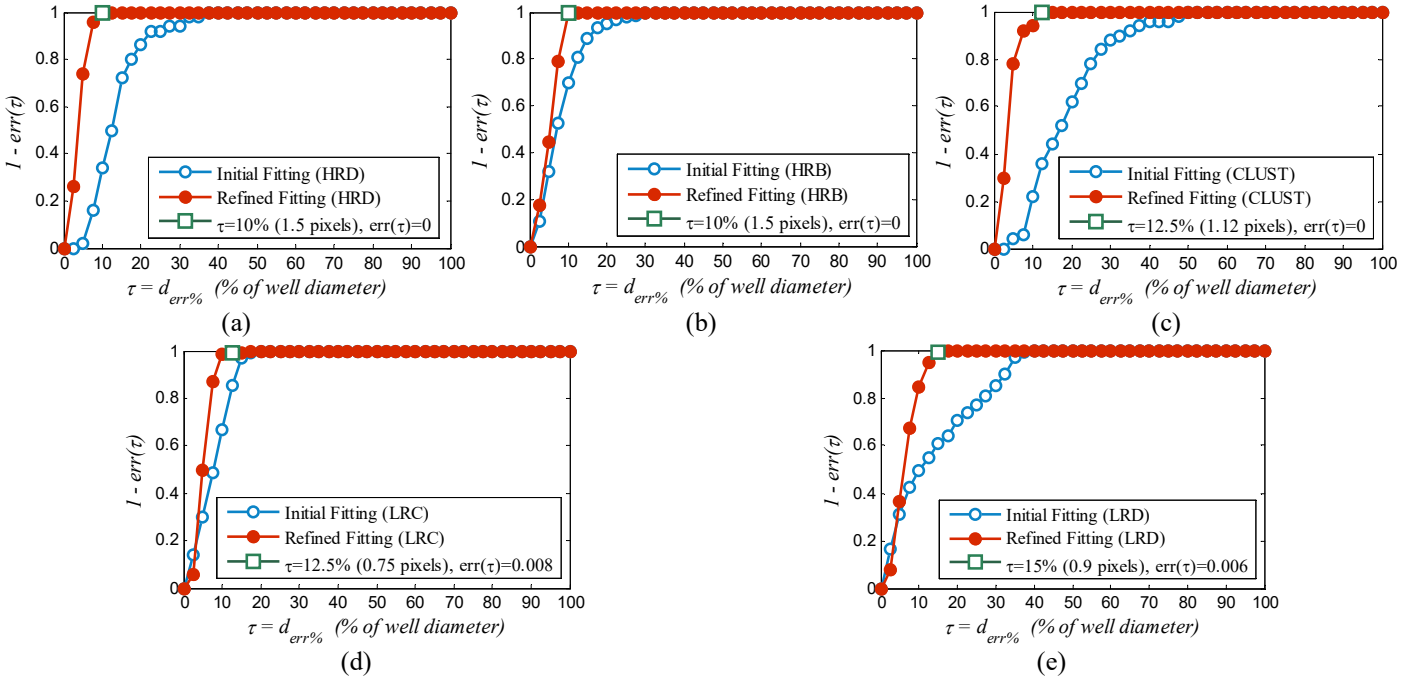


Fig.7. The distributions of d_{err} achieved using the proposed method on the 5 datasets. (a), (b), (c), (d) and (e) the distribution of d_{err} for the HRD, HRB, CLUST, LRC and LRD datasets respectively. The blue and the red curves shows the distribution of d_{err} before and after the affine refinement respectively. The green square marks the distance threshold τ at which $err(\tau)$ falls below 0.01.

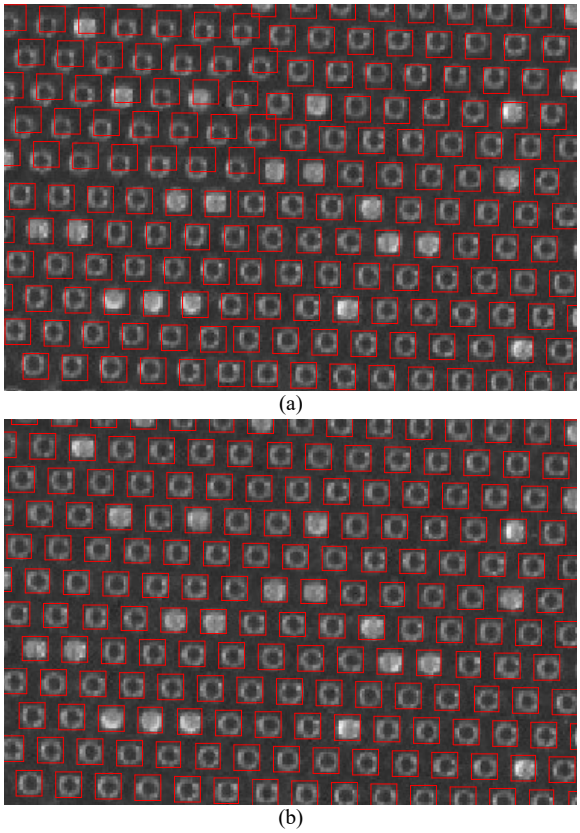
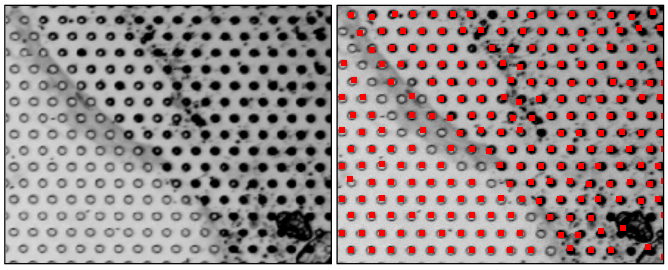


Fig.8. The grid before and after affine refinement. (a) the initial grid fitting output on a portion of an inverted bright field image where four image blocks meet, (b) the gridding achieved after refinement on the same image region. In Fig. (a) the top left block's grid points can be clearly seen as being skewed towards the top-right of the beads.

The other empirically determined parameter, the range $(0.75x_{\text{gap}}, 1.25x_{\text{gap}})$ used in the rejection sampling scheme, is also robust against images with diverse quality. Typically, each sub-image contains hundreds or even thousands of wells. Even in highly distorted sub-images, a fairly large number of wells are detected by the quick shift mode detection. Fig. 9 shows the quick shift mode detection output on an image region with irregular illumination and dirt. The selected range allows for significant shifts of the wells from their ideal grid locations. If no good samples are found in a sub-image for this range, it is likely that the image or the grid is extremely corrupted and the gridding will (and should) fail in such a case.

For fluorescence images it can be beneficial to exclude regions of the image that are affected by noise and dirt as they can adversely affect the subsequent quantitation [31]. However, for bright field images it is often useful to perform gridding even in the noisy and dirty regions. In the sequential bead deposition scheme of [32] for instance, the initial deposition images may contain a lot of dirt and scaffolding which can be washed away during the subsequent deposition steps. In such a case, it is necessary to grid the entire dirty bright field images obtained after the first few deposition steps.

The algorithms used in the proposed method to achieve robustness against noise and distortion require a considerable amount of computation. To keep the execution times reasonable an emphasis was made on parallelizability throughout the design of the method. The genetic algorithm, mean shift algorithm, Radon transform and Inner ICP-REJSAC are inherently parallelizable. Additionally breaking the image into blocks and independently gridding the blocks allow for even more parallelism. The proposed method also has the



(a) (b)

Fig.9. (a) A portion of a brightfield image with irregular illumination and dirt, (b) modes detected in the brigh field image using quick shift. Note that the quick shift mode detection identifies pixels as modes. Here, larger squares are drawn centered over the mode pixels for better visualization.

desirable property of being able to detect failures. This is important in building user confidence in an automated system.

In this paper, the proposed method is presented in the context of hexagonal grids, as they are most common in bead based microarrays. However, it can also be easily extended to detect square grids.

ACKNOWLEDGMENT

The authors would like to thank Ayoxxa, Singapore for their kind cooperation in providing the test data and for their feedback.

REFERENCES

[1] A. Jain, T. Tokuyasu, A. Snijders, R. Segraves, D. Albertson and D. Pinkel, "Fully automatic quantification of microarray image data," *Genome Research,* vol. 12, no. 2, pp. 325-332, 2002.

[2] P. Bajcsy, "Gridline: automatic grid alignment in DNA microarray scans," *IEEE Transactions on Image Processing,* vol. 13, no. 1, pp. 15-25, 2004.

[3] K. Blekas, N. Galatsanos, A. Likas and I. Lagaris, "Mixture model analysis of DNA microarray images," *IEEE Transactions on Medical Imaging,* vol. 24, no. 7, pp. 901-909, 2005.

[4] M. Katzer, F. Kummert and G. Sagerer, "Methods for automatic microarray image segmentation," *IEEE Transactions on Nanobioscience,* vol. 2, no. 4, pp. 202-214, 2003.

[5] A. G. Ramm and A. I. Katsevich, The radon transform and local tomography, CRC Press, 1996.

[6] M. Ceccarelli and G. Antoniol, "A deformable grid matching approach for microarray images," *IEEE Transactions on Image Processing,* vol. 15, no. 10, pp. 3178-3188, 2006.

[7] C. C. Charalambous and G. K. Matsopoulos, "A new method for gridding DNA microarrays," *Computers in Biology and Medicine,* vol. 43, no. 10, pp. 1303-1312, 2013.

[8] G. Bidaut, F. Manion, C. Garcia and M. Ochs, "WaveRead: automatic measurement of relative gene expression levels from microarrays using wavelet analysis," *Journal of Biomedical Informatics,* vol. 39, no. 4, pp. 379-388, 2006.

[9] J. Angulo and J. Serra, "Automatic analysis of DNA microarray images using mathematical morphology," *Bioinformatics,* vol. 19, no. 5, pp. 553-562, 2003.

[10] L. Rueda and I. Rezaeian, "A fully automatic gridding method for cDNA microarray images," *BMC Bioinformatics,* vol. 12, no. 113, 2011.

[11] L. Rueda and V. Vidyadharan, "A hill-climbing approach for automatic gridding of cDNA microarray images," *IEEE/ACM Transactions on Computational Biology and Bioinformatics,* vol. 3, no. 1, pp. 72-83, 2006.

[12] G. Antoniol and M. Ceccarelli, "Microarray image gridding with stochastic search based approaches," *Image and Vision Computing,* vol. 25, no. 2, pp. 155-163, 2007.

[13] E. Zacharia and D. Maroulis, "An original genetic approach to the fully automatic gridding of microarray images," *IEEE Transactions on Medical Imaging,* vol. 27, no. 6, pp. 805-813, 2008.

[14] E. Zacharia and D. Maroulis, "Microarray image gridding via an evolutionary algorithm," in *Proc. 15th IEEE International Conference on Image Processing*, San Diego, 2008.

[15] D. Bariamis, D. K. Iakovidis and D. Maroulis, "M3G: Maximum margin microarray gridding," *BMC Bioinformatics,* vol. 11, no. 49, pp. 1471-2105, 2010.

[16] K. Hartelius and J. Carstensen, "Bayesian grid matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 25, no. 2, pp. 162 - 173, 2003.

[17] M. Katzer, F. Kummert and G. Sagerer, "A markov random field model of microarray gridding," in *Proc. 18th ACM Symposium on Applied Computing*, Melbourne, Florida, pp. 72-77, 2003.

[18] N. Giannakeas, F. Kalatzis, M. G. Tsipouras and D. I. Fotiadis, "Spot addressing for microarray images structured in hexagonal grids," *Computer Methods and Programs in Biomedicine*, vol. 106 (1), pp. 1-13, 2012.

[19] V. Galinsky, "Automatic registration of microarray images. II. Hexagonal grid," *Bioinformatics,* vol. 19, no. 14, pp. 1832-1836, 2003.

[20] A. Vedaldi and S. Soatto, "Quick shift and kernel methods for mode seeking," in *Proc. European Conference on Computer Vision*, Marseille, pp. 705-718, 2008.

[21] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 14, no. 2, p. 239–256, 1992.

[22] O. Chum, J. Matas and J. Kittler, "Locally optimized RANSAC," *Pattern Recognition,* vol. 2781, pp. 236-243, 2003.

[23] K. Fukunaga and L. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," *IEEE Transactions on Information Theory,* vol. 21, no. 1, pp. 32-40, 1975.

[24] C. V. Loan, Computational frameworks for the fast Fourier transform, Philadelphia: Society for Industrial and Applied Mathematics, 1992.

[25] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man and Cybernetics,* vol. 9, no. 1, p. 62–66, 1979.

[26] D. R. Cox and P. A. W. Lewis, "The statistical analysis of series of events," in *The Statistical Analysis of Series of Events*, London, Chapman and Hall, 1966, p. 72.

[27] R. T. Collins, "Mean-shift blob tracking through scale space," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Madison, Wisconsin, pp. 234-240, 2003.

[28] C. V. Stewart, C.-L. Tsai and B. Roysam, "The dual-bootstrap iterative closest point algorithm with application to retinal image registration," *IEEE Transactions on Medical Imaging,* vol. 22, no. 11, pp. 1379-1394, 2003.

[29] D. Chetverikov, D. Svirko, D. Stepanov and P. Krsek, "The trimmed iterative closest point algorithm," in *Proc. 16th International Conference on Pattern Recognition*, Quebec, pp. 545-548, 2002.

[30] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM,* vol. 24, no. 6, p. 381–395, June 1991.

[31] M. L. Smith, M. J. Dunning, S. Tavare and A. G. Lynch, "Identification and correction of previously unreported spatial phenomena using raw Illumina BeadArray data," BMC Bioinformatics, vol. 11, no. 208, 2010.

[32] A. Datta, A. Kong, S. Ghosh and D. Trau, "A fast point pattern matching algorithm for robust spatially addressable bead encoding," in *Proc. 13th IEEE International Conference on Bioinformatics and Bioengineering*, Chania, Greece, pp. 1-7, 2013.