

Microarray core detection by geometric restoration

Jimmy C. Azar^{a,*}, Christer Busch^b and Ingrid B. Carlbom^a

^a*Centre for Image Analysis, Uppsala University, Uppsala, Sweden*

^b*Rudbeck Laboratory, Department of Genetics and Pathology, Uppsala University, Uppsala, Sweden*

Abstract. Whole-slide imaging of tissue microarrays (TMAs) holds the promise of automated image analysis of a large number of histopathological samples from a single slide. This demands high-throughput image processing to enable analysis of these tissue samples for diagnosis of cancer and other conditions. In this paper, we present a completely automated method for the accurate detection and localization of tissue cores that is based on geometric restoration of the core shapes without placing any assumptions on grid geometry. The method relies on hierarchical clustering in conjunction with the Davies-Bouldin index for cluster validation in order to estimate the number of cores in the image wherefrom we estimate the core radius and refine this estimate using morphological granulometry. The final stage of the algorithm reconstructs circular discs from core sections such that these discs cover the entire region of each core regardless of the precise shape of the core. The results show that the proposed method is able to reconstruct core locations without any evidence of localization. Furthermore, the algorithm is more efficient than existing methods based on the Hough transform for circle detection. The algorithm's simplicity, accuracy, and computational efficiency allow for automated high-throughput analysis of microarray images.

Keywords: Tissue microarrays (TMAs), whole-slide scanning, core detection, hierarchical clustering, Davies-Bouldin index, morphological granulometry, morphological opening/closing, convex hull

1. Introduction

Tissue microarrays [13] (TMAs) allow pathologists to study hundreds of tissue samples on a single slide. When imaged as one slide, these tissue arrays hold the promise of high-throughput, automated analysis and staging of cancer and other illnesses.

The preparation of tissue microarrays is carried out manually. Due to mechanical strain, human error, and acquisition conditions, microarray layouts often contain a number of irregularities in the 2D grid geometry [6] such as rotations, geometric distortions, and variations in inter-core spacing. In addition non-uniformities may arise in the foreground and background intensity distributions due to the slide

preparation procedure or the acquisition process. Tissue microarrays thus present a challenging task for automated image analysis. In this paper we focus on what is referred to as gridding, namely to localize the center and extent of each tissue core in a slide.

Several methods in the literature describe gridding of TMA images. Commercial gridding systems (such as [1, 2, 21]) are not well documented in regard to performance and errors. Some methods [9, 16] are not fully automated while others [12] are computationally expensive and tend to fail in detecting weak signal cores. In [15] the proposed gridding method sums up intensities across the rows and columns in the microarray image in order to obtain 1D profiles of peak/valley patterns and to identify the location of cores. But the method assumes that the cores are distributed horizontally and vertically in the image and therefore does not take into account rotations and geometric transformations that frequently occur (see Cases

*Corresponding author: Jimmy C. Azar, Centre for Image Analysis, Uppsala University, Uppsala, Sweden. Tel.: +46 18 471 3474; Fax: +46 18 55 34 47; E-mail: jimmy.azar@cb.uu.se.

8–12 in the *Results* Section). Moreover the method in [15] is not fully automated since the user must pre-define the number of cores in each row and column as well as the length of a sliding window equivalent to the width of the cores in order to detect local minima in the 1D profiles. A support vector machine based gridding method following rotation-correction is proposed in [5] for DNA microarrays; however the support vector classifier utilized was linear and may therefore work well only with DNA microarrays which often have cores arranged regularly in rows and columns.

Our solution does not place any assumptions on grid geometry or inter-core spacing, thus giving our proposed method a great advantage over existing techniques. Since we consider that the cores may be randomly positioned in the image without any assumed regularity or uniformity in distribution, our method is not affected by missing or misaligned cores. Furthermore, having no reliance on 1D image projections as in [15], our method is not negatively affected by image intensity variations.

The purpose of our method is to detect and localize all the cores in microarray images regardless of grid geometry, including those that are partial or missing considerable portions. We leave it to the pathologist to accept or reject any of the detected cores depending on whether the underlying core regions are representative or not. In what follows, we present an accurate, robust, and computationally efficient method for automating microarray core detection and localization.

2. Materials and methods

The extraction of cores from microarray images proceeds in five stages as illustrated in Fig. 1. In Stage 1 preprocessing produces gray scale images from the original microarray images followed by global thresholding using Otsu's method [19]. Stage 2 derives a rough estimate of the microarray cores radii by employing hierarchical clustering in conjunction with

the Davies-Bouldin index [8] as a cluster validation measure. Note that only in Stage 2 the image is first downsampled in order to boost computational performance since this stage does not require high image resolution. Stage 3 employs morphological granulometry [17, 18] to obtain a refined estimate of the core radius that is used to define a morphological structuring element. Stage 4 consists of a series of sequential morphological operations using a disk structuring element with its radius defined in the previous stage. Finally, Stage 5 uses classical geometry to restore ideal circles from the partial/defective cores at the output of Stage 4. In what follows we discuss each of these stages in greater detail.

2.1. Stages 1-2: Clustering and validation

In order to extract tissue cores, we apply a series of morphological operations to identify the cores or portions of these and localize their position. Morphological operations [20] such as closing and opening depend on a structuring element of a predefined shape and size. Microarray cores are roughly circular, and therefore a suitable structuring element is a disc. To estimate the core radii for a microarray image of arbitrary size without manual guidance, we transform the problem into a simpler one by instead estimating the number of cores, K , present in the image. The total area of the circular cores (i.e. the area of the foreground after applying global thresholding) and their number, K , give an estimate of the core radius.

Starting from a microarray color image, global thresholding using Otsu's method (see Fig. 2b) transforms the image into grayscale (Stage 1). The binary image is then downsampled by a factor determined automatically such that the output image is less than 200×200 pixels. The reason for downsampling is to gain speed; it is used only at this stage since to obtain a rough estimate for the number of cores, high resolution proves unnecessary. The resulting image is then segmented into K cores by means of agglomerative hierarchical clustering [10] using complete linkage

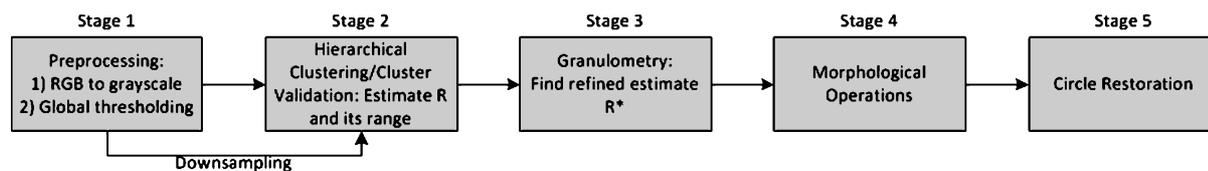


Fig. 1. Flowchart of our method. R represents the approximate radius of the cores.

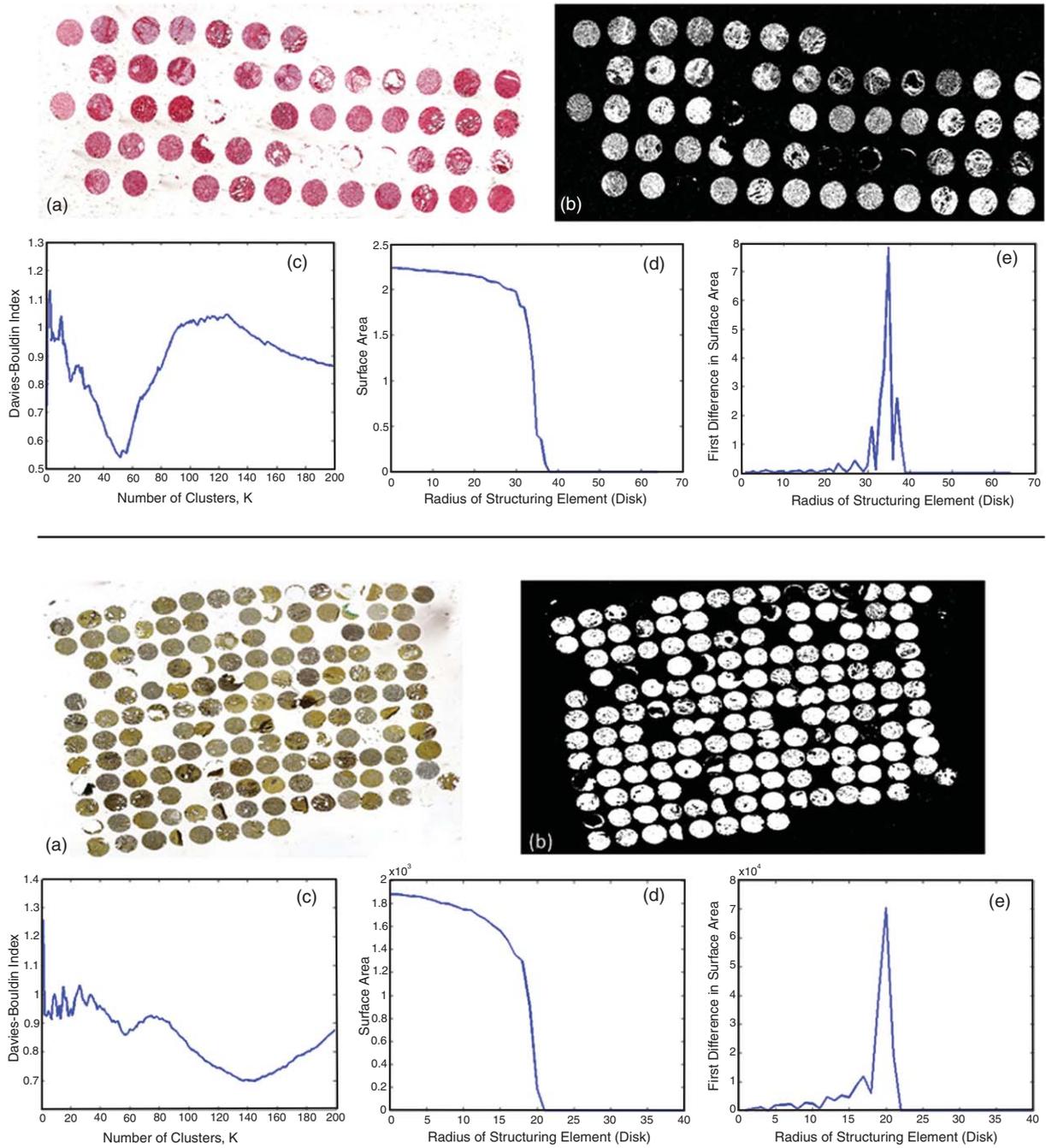


Fig. 2. An illustration of morphological granulometry. (a) Original TMA image. (b) Global thresholding by Otsu's method. (c) Davies-Bouldin index versus number of clusters. (d) Granulometry function. (e) Object size distribution.

with Euclidean distance metric wherein the distance between two clusters C^i and C^j is given by

$$d(C^i, C^j) = \max\{d(x, y) : x \in C^i, y \in C^j\}$$

Complete linkage was chosen since cores are circular and therefore ideally suitable for this linkage type. We find a preliminary estimate of the number of cores by varying the number of clusters K , whereby

for every value of K , the clustering result is assessed by the Davies-Bouldin index which is an ideal metric for circular cluster validation. To derive the final index for cluster validation, we first define an intermediate metric as follows. For a given K , resulting in clusters C^1, C^2, \dots, C^K , we consider every pair of clusters C^i, C^j and compute the following metric.

$$R_{ij} = \frac{\sigma_i + \sigma_j}{\|\bar{\mu}_i - \bar{\mu}_j\|}$$

where $i, j = 1, 2, \dots, K$, and $i \neq j$.

The quantities $\bar{\mu}_i$ and $\bar{\mu}_j$ are the mean vectors of the pixel locations in clusters C^i and C^j respectively, and σ_i, σ_j are the standard deviations representing the within-scatter of clusters C^i and C^j , respectively. That is,

$$\sigma_i = \sqrt{\frac{1}{n_i} \sum_{\vec{p}_l \in C^i} \|\vec{p}_l - \bar{\mu}_i\|^2}$$

$$\bar{\mu}_i = \frac{1}{n_i} \sum_{\vec{p}_l \in C^i} \vec{p}_l,$$

where n_i is the number of pixels in C^i , and \vec{p}_l is a vector corresponding to a pixel location in the image. The measure R_{ij} between two clusters C^i, C^j gives a measure of the clustering efficacy. The numerator of R_{ij} may be regarded as the within-cluster scatter representing the compactness of the clusters, whereas the denominator may be viewed as the between-cluster scatter representing their separation. Hence, lower values of R_{ij} signify that the clusters are ideally compact and well separated. As mentioned, every cluster C^i may be paired with $K-1$ different clusters resulting in $K-1$ metrics, namely R_{ij} $j = 1, 2, \dots, K$ and $j \neq i$. Yet to represent a single cluster with one such metric, the largest value for this measure is chosen, representing a conservative or worst-case value among the $K-1$ values. Therefore for a given cluster C^i , we compute $R_i = \max_{j: j \neq i} R_{ij}$, repeat this for all remaining clusters, and consider the average value over all the K clusters, I_C , as an indication of cluster validity in terms of within-cluster scatter and between-cluster scatter,

$$I_C = \frac{1}{K} \sum_{i=1}^K R_i.$$

For every choice of K , we obtain a single scalar index indicating the validity of the clustering, and as K is

varied we obtain a curve of indices in which the optimal choice of K is the value where the curve attains a minimum as shown in Fig. 2c. Note that the value of K serves only as a rough estimate of the number of cores in the image. Once K is estimated, the total area in the original binary image (consisting of the cores) is divided by K to obtain an estimate of the surface area of a single core from which we obtain an estimate of the radius of the circular cores, while noting that this radius varies slightly from core to core which in turn may slightly deviate from circular form. In summary, an approximation of K using the method described above, allows us to determine in an automated manner the range of radii values required as input for the next stage.

2.2. Stage 3: Automated morphological granulometry

Once a preliminary radius estimate is obtained as described in Stage 2, a refined radius estimate is computed using morphological granulometry which is used for determining the distribution of the size of particles present in an image [17, 18]. In our application, we use the method to apply a series of morphological openings to the binary microarray image using a disk with sequentially increasing radius as structuring element. Particles that are smaller than the structuring element disappear completely by morphological opening. We compute the total surface area of the cores remaining by summing the number of pixels in the binary image after every step of morphological opening, and we plot these values against the radius value of the structuring element as shown in Fig. 2d. Aside from the shape of the structuring element, the only input parameter is the range of values the radius of the structuring element may assume. The granulometry function or curve obtained as in Fig. 2d is always monotonically decreasing, and the regions of interest are the locations at which large jumps occur in the curve signaling the disappearance of an entire group of particles with a certain size given by the radius of the structuring element. To observe the jumps in the curve, we consider the first derivative or finite first difference of the discrete curve (Fig. 2e) and search for prominent peaks. Ideally we would expect only one main peak since microarray images consist of only one type of particles, namely discs with one fixed size. However in practice the cores vary slightly in radius and shape, resulting in curves

as illustrated in Fig. 2e. We automatically detect the very last peak in these derivative curves from which we obtain a refined radius estimate, R^* , corresponding to the largest core in the image. We choose the last

peak because it corresponds to the microarray cores with largest radius which means that in a later stage when reconstructed circles are assigned over the core locations, these circles will cover any given core by its

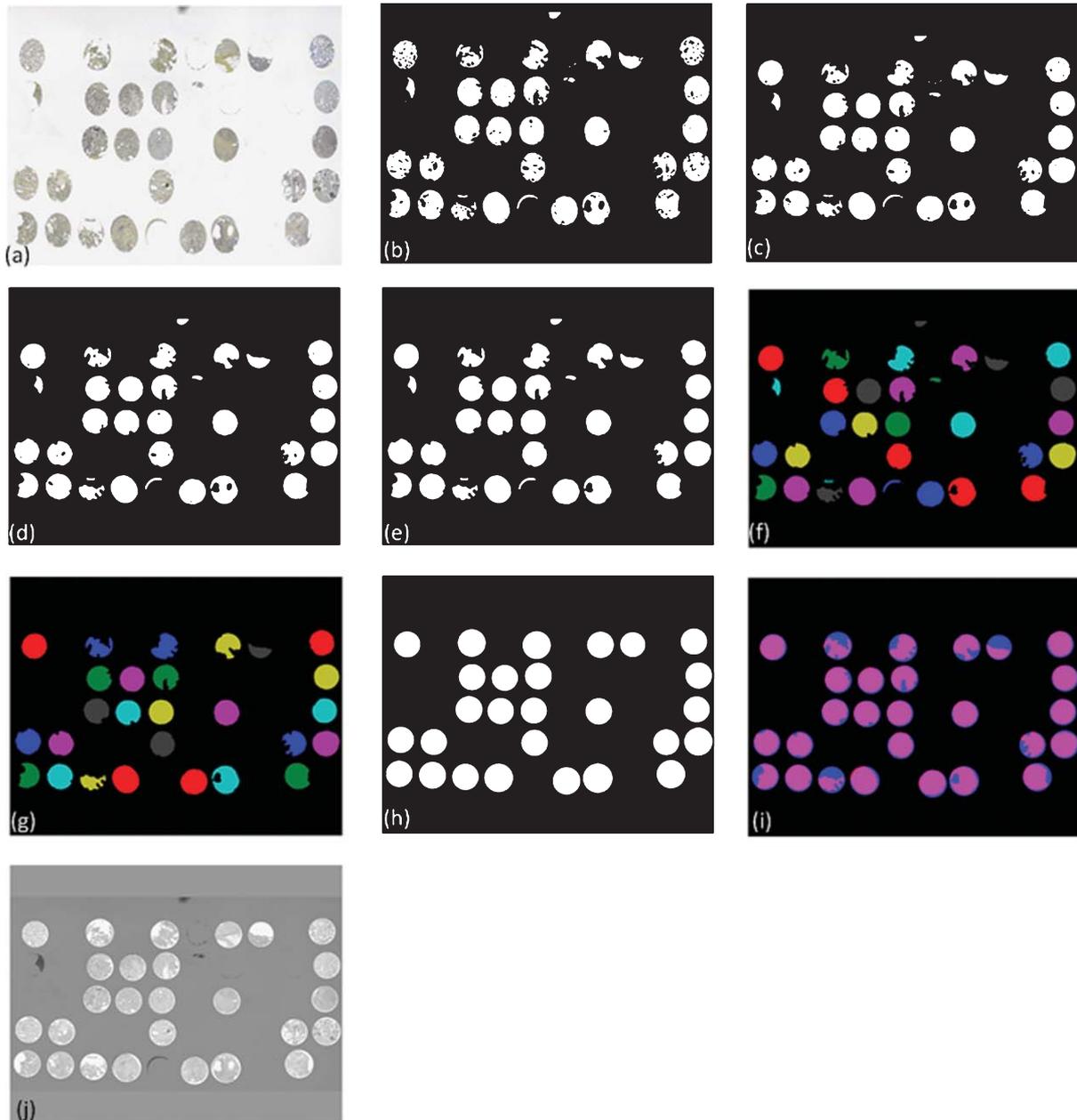


Fig. 3. An example showing the morphological operations (Stage 4) in sequential order and the result of circle reconstruction (Stage 5). (a) Original TMA image. (b) Global thresholding by Otsu's method. (c) Morphological closing. (d) Median filtering to remove speckles. (e) Region filling of holes. (f) Object labeling by 8-connectivity. (g) Filtering out any small objects by their respective areas using the size condition mentioned in Stage 4. (h) Reconstruction of circles from the objects in (g). (i) Superimposing the objects with their reconstructions, that is (i) = (g) + (h). (j) An overlay of the original TMA image with the reconstructed circles, that is (j) = (a) + (h).

entirety thereby ensuring that every core is detected in whole.

2.3. Stage 4: Morphological operations

Once a refined estimate of the radius, R^* , corresponding to the largest of the microarray cores is determined, this estimate is used to define the radius of the structuring element (disk) which is a fixed fraction of R^* . After a microarray image is transformed into a grayscale image, thresholded using Otsu's method, and inverted, the resulting binary image shows foreground cores against a dark background (see Fig. 3b). To remove isolated pixels, speckles, and insignificantly small structures from the binary image, we apply a median filter with a kernel whose side length is a quarter of R^* . However prior to median filtering, morphological closing is applied to connect the granulated core boundaries so that they are not eroded by the median filter. Fig. 3c, d shows the result of these two morphological processes on a microarray image. Thereafter, region filling is applied to close any existing holes within the cores (Fig. 3e), and the objects are labeled considering an 8-neighbor connectivity (Fig. 3f) in order to identify individual cores. Finally, the labeled structures may be subjected to the following size condition: if the area of an object is less than $1/3 \pi R^{*2}$, then the object is considered too small and is thereby removed from the image. This ensures that any cores that are broken into several small pieces are discarded automatically as shown in Fig. 3g. The sequence of morphological operations result in cores or their remainders from which we may recover the location of the entire core. Note that usually many of the cores in the original microarray image are already defective or have missing parts.

2.4. Stage 5: Circle reconstruction

The final stage of the algorithm determines the location of the cores by generating individual circles using the portions of these cores that remain after Stage 4. These circles should cover the complete core regions in the original microarray image.

Using the binary image with the objects labeled (as in Fig. 3g), the edge is derived for each object using Canny edge detection [7]. For each closed object boundary (edge) we derive the convex hull which will consist of a subset of boundary pixels. In Fig. 4b, e

sections of the boundary marked in blue correspond to concave regions that are not part of the convex hull, and the pixels from these sections of the boundary will be ignored. The convex hull determines valid (convex) sections of the boundary from which we may select three pixels in order to construct a circle encompassing the object boundary. Since a unique circle passes through any three non-collinear points, three pixels from the convex hull may serve as vertices through which a circle will pass. We choose these pixels to be as far apart as possible within the convex hull, which means that the shortest distance between any of the three vertices is one third the cardinality of the convex hull set. Once the three vertices are selected, a circle passing through these vertices is generated analytically as follows. Regard the three pixels or vertices as lying in a Cartesian system coinciding with the image coordinate system. Then suppose the three vertices have coordinate vectors:

$$A = \begin{bmatrix} x_a \\ y_a \end{bmatrix}, \quad B = \begin{bmatrix} x_b \\ y_b \end{bmatrix}, \quad C = \begin{bmatrix} x_c \\ y_c \end{bmatrix}$$

We augment the coordinate system by an additional dimension yielding:

$$A = \begin{bmatrix} x_a \\ y_a \\ z_a \end{bmatrix}, \quad B = \begin{bmatrix} x_b \\ y_b \\ z_b \end{bmatrix}, \quad C = \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix},$$

$$z_a = z_b = z_c = 0.$$

It is possible to find the centre S and radius r of the circle using dot product and cross product notation as follows:

$$r = \frac{\|A - B\| \|B - C\| \|A - C\|}{2 \|(A - B) \bullet (B - C)\|}$$

$$S = \alpha A + \beta B + \gamma C$$

$$\alpha = \frac{\|B - C\|^2 (A - B) \bullet (A - C)}{2 \|(A - B) \otimes (B - C)\|^2}$$

$$\beta = \frac{\|A - C\|^2 (B - A) \bullet (B - C)}{2 \|(A - B) \otimes (B - C)\|^2}$$

$$\gamma = \frac{\|A - B\|^2 (C - A) \bullet (C - B)}{2 \|(A - B) \otimes (B - C)\|^2}$$

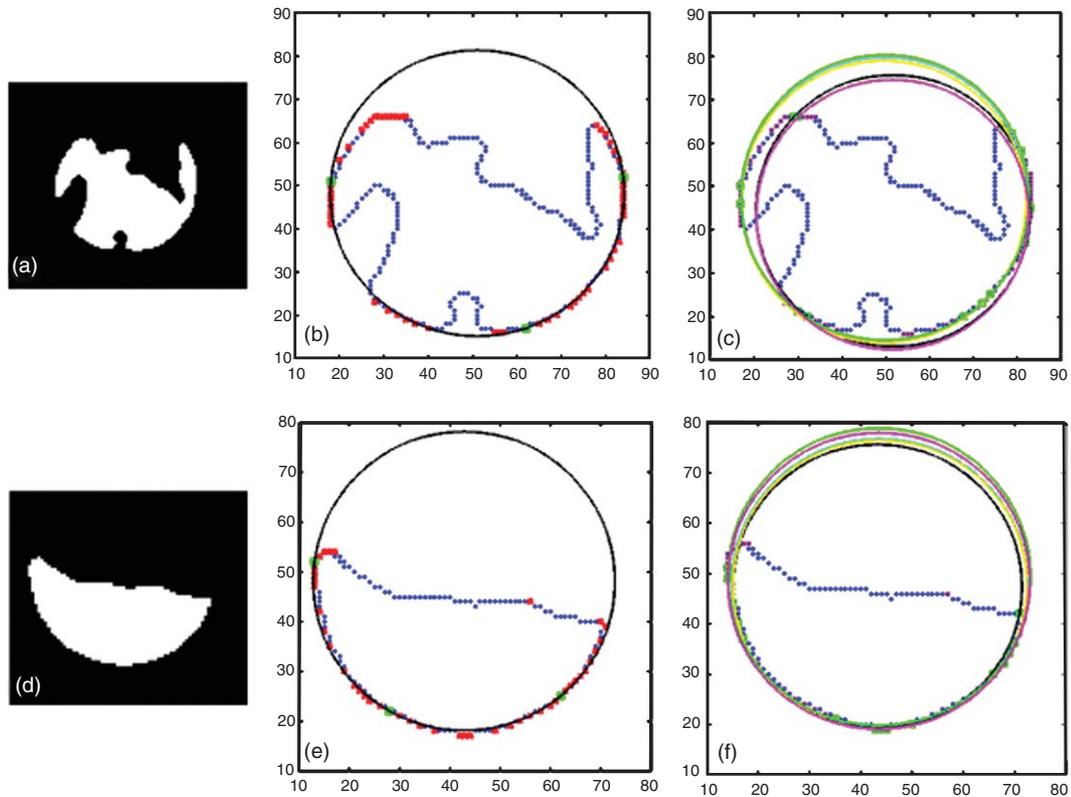


Fig. 4. The circle restoration algorithm (Stage 5) applied over two defective cores. (a)/(d) Object to be reconstructed into a perfect circle. (b)/(e) A single circle reconstructed from three vertices (marked in green squares) chosen randomly from the convex hull which consists of pixels marked in red. (c)/(f) Multiple circles are generated (shown in different colors) by randomly varying the initial three vertices; only the largest of the circles is retained. (Colours are visible in the online version of the article; <http://dx.doi.org/10.3233/ACP-2012-0067>)

where $\|\cdot\|$ represents magnitude in Euclidean space, and \bullet , \otimes represent the dot product and cross product operations, respectively.

The result of generating one such circle using three vertices from the convex hull is shown in Fig. 4b, e. However in order to reduce sensitivity to the selection of the three vertices and to ensure that the largest possible circle is generated, the location of the vertices is randomized and the process is repeated ten times whilst preserving the relative distances among the vertices as described above. Finally the largest of the reconstructed circles is selected as shown in Fig. 4c, f. The randomization of the vertices and the selection of the largest resulting circle ensure that the underlying core is covered in its entirety.

In summary, by using morphological operations followed by circle reconstruction, we are able to construct perfect circles covering each and every core in the microarray image, and as a result also detect the

location of each core. Any core that remains after the morphological operations and the imposed size condition, will allow for perfect circle reconstruction around that core which when superimposed with the original microarray image indicates the location of the corresponding core (see Fig. 5 of the *Results* section).

The generation of circles from the three-point method of reconstruction is simple, analytic, and therefore computationally also highly efficient.

2.5. Elliptic reconstruction and generalization to analytic shapes

For our purpose, it is sufficient to describe TMAs as having circular or disc-like form since in practice they deviate only slightly from the ideal case, and the circle reconstruction method is therefore sufficient to cover the region of every core as the *Results* section shows. However we could instead assume a more general

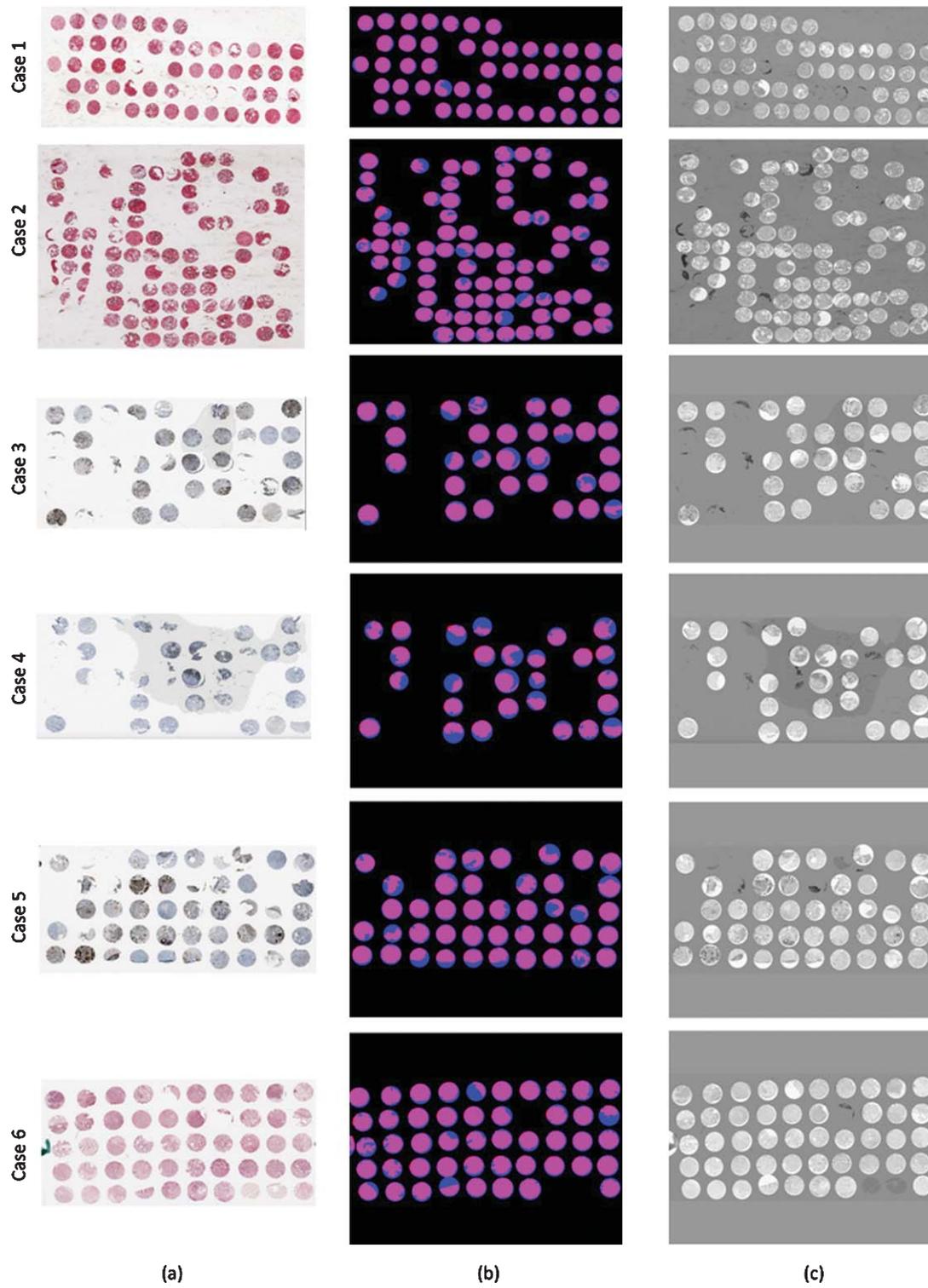


Fig. 5. (a–c). The application of our method to tissue microarray images. (a) Original TMA image. (b) Objects following morphology (Stage 4) overlaid against reconstructed circles (Stage 5). (c) Original image overlaid against reconstructed circles.

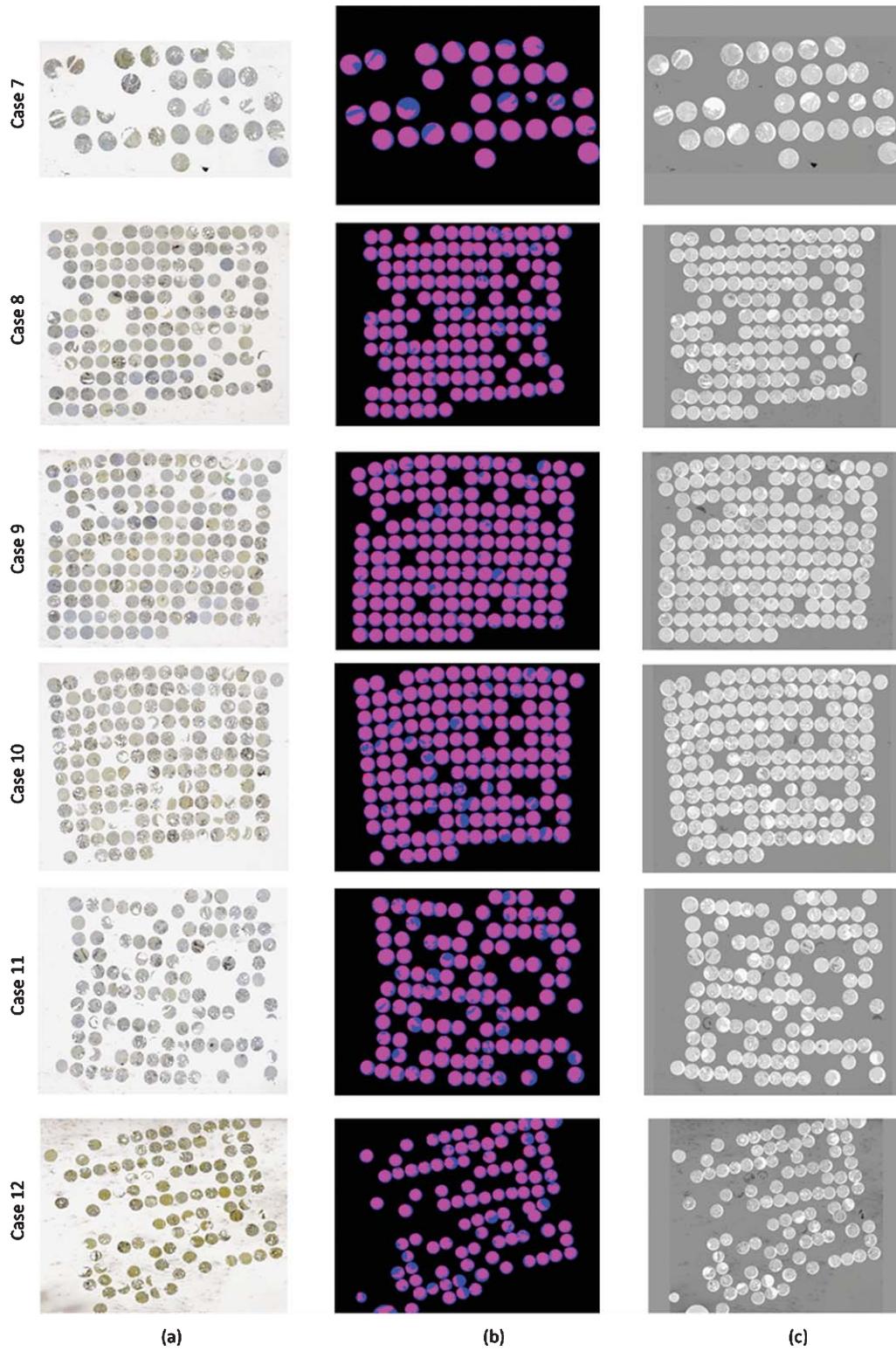


Fig. 5. Continued

elliptic shape. For the sake of completeness, we demonstrate how to analytically reconstruct such shapes. An ellipse, as any conic, may be described by an analytic equation, and similarly as in the case of the three-point reconstruction method for a circle, any given five points define a unique ellipse provided that no subset of these is collinear. Analytically, the ellipse passing through five points (x_a, y_a) , (x_b, y_b) , (x_c, y_c) , (x_d, y_d) , (x_e, y_e) may be described by the equation derived from solving the following determinant:

$$\begin{vmatrix} x^2 & xy & y^2 & x & y & 1 \\ x_a^2 & x_a y_a & y_a^2 & x_a & y_a & 1 \\ x_b^2 & x_b y_b & y_b^2 & x_b & y_b & 1 \\ x_c^2 & x_c y_c & y_c^2 & x_c & y_c & 1 \\ x_d^2 & x_d y_d & y_d^2 & x_d & y_d & 1 \\ x_e^2 & x_e y_e & y_e^2 & x_e & y_e & 1 \end{vmatrix} = 0$$

Thereafter, the method proceeds as for circles by finding the convex hull consorted with randomized repetitions of vertex selection.

It is worth noting that the method may be extended further to any convex shape that may be described by an analytic equation in analogy with how the Hough transform [4, 22] may be generalized to detect any analytically defined geometrical shape. Thus general shape restoration may be addressed by this method following morphology. One such application could be the restoration of shapes that originally appear defective when acquired from a source.

3. Results

We tested our method on over 2300 cores belonging to 32 tissue microarray images. The TMAs were prepared from whole mount paraffin blocks from patients operated with total prostatectomy due to localized prostate cancer. This was carried out at the Department of Pathology, University Hospital, Uppsala, Sweden. The Aperio ScanScope XT Slide Scanner (Aperio Technologies, Vista, CA) system was used to capture digital whole slide images with a 40X objective. Most of these TMA images presented difficulties ranging from defective or partial cores to non-regular array spacing with geometric distortions (as evident in Cases 8–12 of Fig. 5). The results show that our method is immune to these problems. Figure 5 shows the results of twelve sample cases. Next to each TMA image

in Fig. 5a, we show two results: the objects after Stage 4 overlaid with the reconstructed circles or discs (Fig. 5b), and the original TMA image overlaid with the same reconstructed discs (Fig. 5c).

One important conclusion is that the circle restoration method is able to reconstruct perfect core locations in all test images from the defective objects that pass through the morphological operations. Thus whatever core portions that pass through the morphological operations of Stage 4 may be completely and perfectly restored by Stage 5. This is however not the case with the Hough transform and methods that rely upon it (such as [14]) as Fig. 6 shows in the following section. Another important conclusion is that cores that are defective or partially missing were still recovered completely by our method. The intention is to recover all of the cores in the original image and leave the choice to the pathologist to reject or accept some of these cores. Lastly we note that in Case 6 and Case 12 of Fig. 5, the original TMA image contains some unusual artifacts, namely a green marking from a pen at the left border in the former case and some dark patches at the lower left corner in the latter. In both cases, the algorithm identifies those regions as microarray cores as would be expected.

4. Comparison with other methods

4.1. Template matching for core detection

Direct template matching for microarray core detection [3, 6, 9] suffers from several drawbacks. Template matching assumes that a rectangular grid may be fitted over the microarrays; this is not always justifiable as Cases 8–12 above demonstrate. When selecting a certain rectangular grid as template, one has to decide a priori on the distance between the grid points and the size of each grid cell. Such knowledge prevents the method from being fully automated in the sense that different templates would have to be predefined for microarray images that differ in size and in core spacing. Furthermore, accounting for a wide range of translations, rotations, and other transformations makes the method slow and burdensome. Because template matching attempts to fit several cores in a rectangular grid at once, the method may achieve high detection rates if the main assumption is fulfilled; however, it may on the other hand lead to drastically low

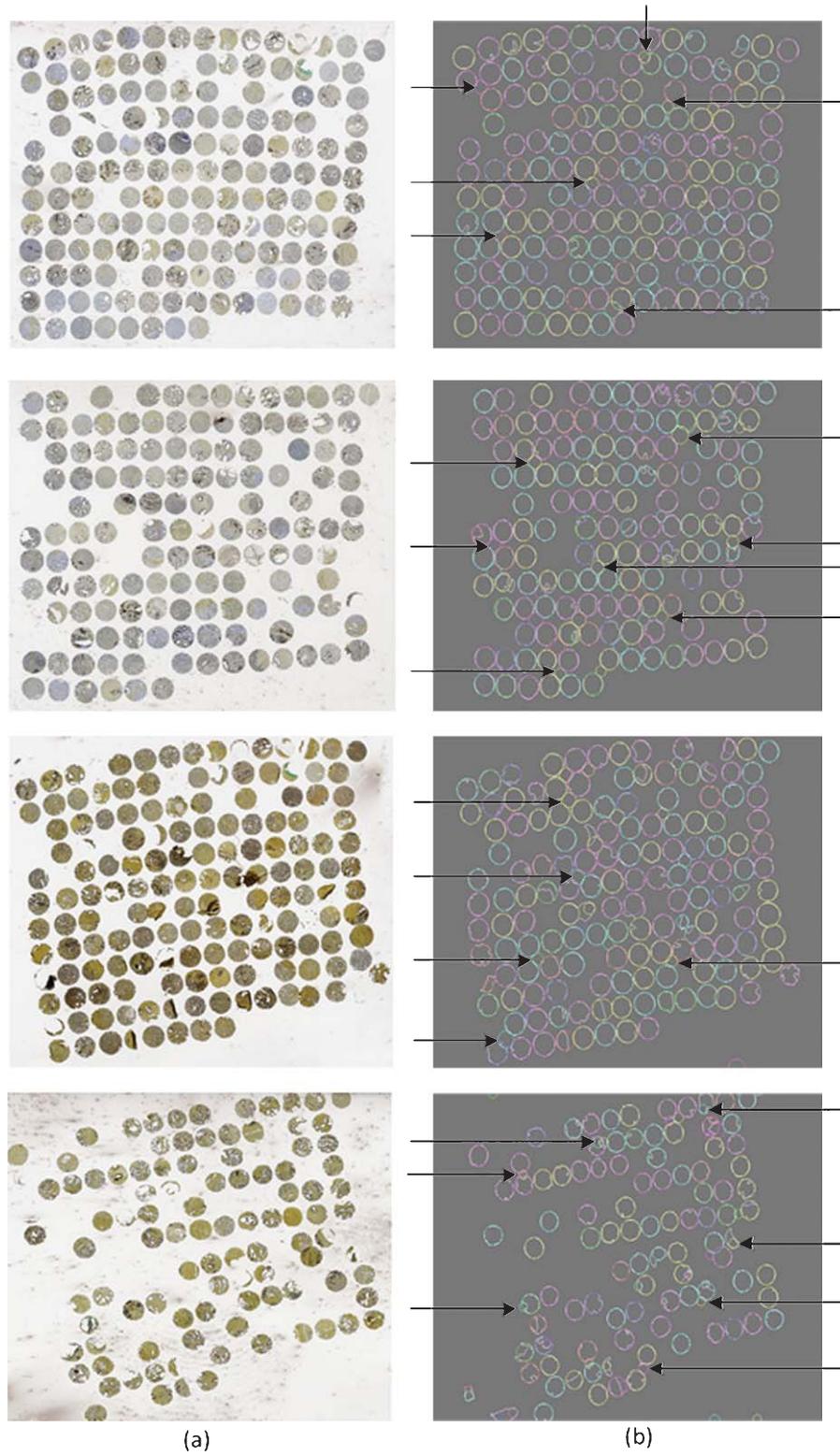


Fig. 6. Continued

detection rates if the matching is not successful due to the aforementioned reasons.

4.2. The hough transform for circle detection

As we have mentioned in the previous section, template matching on its own may not provide a completely automated procedure for high-throughput detection of microarray cores. However if template matching is combined with the Hough transform as is done in [14], the method may be used to detect the center of circular objects and their radii by virtue of the Hough transform which will return the center of the circle and its radius, both of which could be used to set the grid cell size for the template matching algorithm. The Hough transform [4, 22] for circle detection may be applied to the binary images we obtained previously in which the edge boundaries were derived following morphological operations (Fig. 6b). In addition, the Hough transform requires as input a possible range for the circle radii. However the method we have devised for determining the range for the core radii using hierarchical clustering and the Davies-Bouldin index may be utilized here as input for the Hough transform exactly as we have used it as input to morphological granulometry. Thus following the morphological operations of Stage 4, we could have used the Hough transform followed by template matching instead of circle restoration.

The main drawback of the Hough transform for circle detection is that the parameter space is three-dimensional wherein the intersection of cones counts toward extrema. This makes the method computationally expensive and gives considerable weight to optimizing the construction of the 3D accumulator. For a more detailed comparison in terms of computational time, refer to the following section on 'Processing Time'. Another drawback of the Hough transform is maxima detection in parameter space which is associated with several difficulties since peaks in parameter space are generally neither narrow nor convex, in addition to the fact that there is usually less evidence in parameter space for the case of defective microarray cores with missing parts than for complete cores. Furthermore, Fig. 6 demonstrates the sensitivity of the method in detecting maxima in parameter space and

its dependence on setting a suitable threshold, which in case the threshold is too low may lead to an excess number of detected circles. Yet if the threshold is too high, a deficit number of circles would be detected. Figure 6 also shows that many of the correctly detected circles do not actually cover the entire objects underneath. Therefore by this method the size of the reconstructed discs would be underestimated with respect to the true cores.

As we have shown, there is no need for direct shape detection using the Hough transform in tissue microarray images for the simple reason that we are not trying to find a certain object shape from among several others. Therefore instead, we have opted for simple and fast morphology in detecting microarray cores followed by a yet simpler, analytic and efficient shape restoration method.

5. Processing time

The algorithms described in *Materials and Methods* have been developed using Matlab version 7.12 with Image Processing Toolbox version 7.2 and DIPimage [11] version 2.3 running on a RedHat Linux 6.1 operating system, Intel Xeon CPU with 8 GB RAM, and 2.53 GHz clock frequency. In this section we compare the Hough transform for circle detection (which constitutes the first step in the Hough-Template matching procedure) with our proposed circle reconstruction method in terms of computation time. We assume here that both methods were already provided an input range of core radii by the hierarchical clustering and cluster validation methods described earlier. The average processing time over 32 tissue microarray images of size 640×816 , was 19.52 seconds using the Hough transform for circle detection (using DIPimage) versus 1.06 seconds using our analytic circle restoration method. Thus our method was around 18 times faster than the standard Hough transform for detecting circles. Stages 1 through 5 of the proposed algorithm required respectively around 3, 18, 28, 12, and 38 percent of the overall running time. Therefore the proposed method *as a whole* was around 7 times faster than the Hough transform which is often only one stage of several in algorithms for detecting microarray cores.

Fig. 6. Circle detection in TMA images using the Hough transform. (a) Original TMA image. (b) Circles detected with the Hough transform. The results show an excess number of detected circles (some indicated by arrows) that are not present in the original images. Many of the correctly detected circles also do not cover the entire core region.

6. Conclusion

We have presented a method for robust and accurate core detection and localization in microarray images. The traditional problem of microarray gridding relies on finding linear, piece-wise linear, or curvilinear borders around the cores. All the approaches we have encountered in the literature do not alter these premises, but rather attempt to improve such gridding methods by employing different techniques. Instead, our approach relies upon reconstructing a circular border encompassing the underlying core without placing any assumptions on core arrangement or grid geometry, transforming the problem into a much simpler one, namely the detection of any portions of the cores through morphology, and the restoration of the core locations starting from these core portions. The method was tested on over 2300 cores. As demonstrated in the *Results* section, the simplicity, robustness, accuracy, and computational efficiency of the proposed method provide considerable advantage over other approaches in the literature. In conclusion the presented method allows for high-throughput analysis of microarray cores in the hope of facilitating accurate histological analysis of these cores.

References

- [1] Alphelys. SpotBrowser 3. <http://www.alphelys.com>.
- [2] Aperio. TMA Lab Microarray Analysis Tool. <http://www.aperio.com>.
- [3] Axon Instruments Inc. GenePix Pro. <http://www.moleculardevices.com/Products/Software/GenePix-Pro.html>.
- [4] D.H. Ballard, Generalizing the hough transform to detect arbitrary shapes, *Pattern Recognition* **13**(2) (1981), 111–122.
- [5] D. Bariamis, D.E. Maroulis and D.K. Iakovidis, Unsupervised SVM-based gridding for DNA microarray images, *Computerized Medical Imaging and Graphics* **34** (2010), 418–425.
- [6] J. Buhler, T. Ideker and D. Haynor, Dapple: Improved techniques for finding spots on DNA microarrays, *UV CSE Technical Report UWTR*, 2000-08-05.
- [7] J. Canny, A Computational Approach to Edge Detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **8**(6) (1986), 679–698.
- [8] D.L. Davies and D.W. Bouldin, A cluster separation measure, *IEEE Transactions on Pattern Analysis and Machine Intelligence-PAMI-1* **2** (1979), 224–227.
- [9] M.B. Eisen, ScanAlyze. <http://rana.lbl.gov/EisenSoftware.htm>.
- [10] T. Hastie, R. Tibshirani and J. Friedman, *The Elements of Statistical Learning-14.3.12 Hierarchical clustering*, 2nd ed., New York: Springer, 2009, pp. 520–528.
- [11] C.L.L. Hendriks, L.J. van Vliet, B. Rieger, G.M.P. van Kempen and M. van Ginkel, DIPimage: A scientific image processing toolbox for MATLAB, *Quantitative Imaging Group, Faculty of Applied Sciences*, Delft University of Technology, The Netherlands, 1999.
- [12] M. Katzer, F. Kummert and G. Sagerer, A Markov random field model of microarray gridding, *Proceedings of ACM Symp Applied Computing (SAC)*, Melbourne, FL, 2003, 72–77.
- [13] J. Kononen, L. Bubendorf, A. Kallioniemi, M. Barlund, P. Schraml, S. Leighton, J. Torhorst, M.J. Mihatsch, G. Sauter and O.P. Kallioniemi, Tissue microarrays for high-throughput molecular profiling of tumor specimens, *Nature Medicine* **4** (1998), 844–847.
- [14] B. Lahrmann, N. Halama, K. Westphal, C. Ernst, Z. El Sawaf, P. Sinn, F.X. Bosch, H. Dickhaus, D. Jäger, P. Schirmacher and N. Grabe, Robust gridding of TMAs after whole-slide imaging using template matching, *Cytometry A* **77**(12) (2010), 1169–1176.
- [15] Q. Li, C. Fraley, R.E. Bumgarner, K.Y. Yeung and A.E. Raftery, Donuts, scratches and blanks: Robust model-based segmentation of microarray images, *Bioinformatics* **21**(12) (2005), 2875–2882.
- [16] A.W.-C. Liew, H. Yang and M. Yang, Robust adaptive spot segmentation of DNA microarray images, *Pattern Recognition* **36** (2003), 1251–1254.
- [17] G. Matheron, *Éléments pour une Théorie des Milieux Poreux*, Masson, Paris, 1967.
- [18] G. Matheron, *Random Sets and Integral Geometry*, John Wiley and Sons Inc., New York, 1975.
- [19] N. Otsu, A Threshold selection method from gray-level histograms, *IEEE Transactions on Systems, Man, and Cybernetics* **9** (1979), 62–66.
- [20] J. Serra, *Image Analysis and Mathematical Morphology*, 1982.
- [21] Slidepath. OpTMA. <http://www.slidepath.com>.
- [22] R. Yip, P. Tam and D. Leung, Modification of Hough transform for circles and ellipses detection using a 2-dimensional array, *Pattern Recognition* **25**(9) (1992), 1007–1022.



Hindawi
Submit your manuscripts at
<http://www.hindawi.com>

