Multi-modal optical microscopy image analysis and matching techniques for spatially encoded bead-based microarrays

Datta, Abhik

2017

Datta, A. (2017). Multi-modal optical microscopy image analysis and matching techniques for spatially encoded bead-based microarrays. Doctoral thesis, Nanyang Technological University, Singapore.

<http://hdl.handle.net/10356/70301>

<https://doi.org/10.32657/10356/70301>

NANYANG TECHNOLOGICAL UNIVERSITY
SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

**MULTI-MODAL OPTICAL MICROSCOPY IMAGE
ANALYSIS AND MATCHING TECHNIQUES FOR
SPATIALLY ENCODED BEAD-BASED
MICROARRAYS**

ABHIK DATTA
(M.Sc., Nanyang Technological University)

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

Acknowledgements

I would like to thank Nanyang Technological University for giving me the opportunity to pursue this journey in research and for providing a stimulating environment. I will cherish the time I have spent on campus and the good friends that I have made here.

I am grateful to Prof. Yow Kin-Choong for encouraging me to embark upon this journey. Even though he was my advisor only for a year, he has continued to guide and support me throughout this journey.

I would like to express my sincerest gratitude to my advisor Assoc. Prof. Dr. Adams Wai-Kin Kong for his invaluable guidance and motivation. His strive for excellence has been a source of inspiration to each of his students, including me.

I would also like to thank the team at Ayoxxa for their kind cooperation in allowing me to use their experimental data. I would like to thank Assoc. Prof. Dr. Dieter Trau for his feedback and comments.

I would like to thank my father Mr. Biswanath Datta for imbibing in me the drive for scientific pursuit from a young age. His relentless encouragement and earnest persuasion to keep going have kept me from giving up during difficult times. I am grateful to my sister Pushpita Datta for her endless support which provided me the strength to keep going forward.

At every step of this journey as in any journey of this nature there have been challenges. I would not have been able to scale any of those challenges if I did not have the company of my best friend and wife, Soumita Ghosh. She has shared with me every failure and every success with equal grit and I cannot thank her enough for that. Also, I would like to thank her for proof reading this thesis.

I treasure the memories of my late mother Mrs. Bijali Datta whose life and death continue to inspire me.

This work is dedicated to my father Mr. Biswanath Datta.

Table of Contents

Acknowledgements.....	ii
Table of Contents.....	iii
Abstract	vi
List of Tables.....	viii
List of Figures	ix
List of Abbreviations.....	xiv
Chapter 1 – Introduction	1
1.1 Bead-based Microarrays.....	2
1.2 Bead Encoding.....	4
1.3 Contributions	5
1.3.1 Robust Spatial Bead Encoding	6
1.3.2 Multi-modal Optical Microscopy Image Analysis and Matching	7
1.4 Motivation	9
1.5 Thesis Map	10
Chapter 2 - Robust Spatially Addressable Bead Encoding	11
2.1 Existing Bead Encoding Methods	11
2.1.1 Optical Encoding.....	11
2.1.2 Chemical Encoding	16
2.1.3 Spectrometric Encoding	17
2.1.4 Graphical Encoding.....	18
2.1.5 Electronic Encoding	19
2.1.6 Physical Encoding	20
2.1.7 Spatial Bead Encoding	20
2.2 Summary of Existing Bead Encoding Methods	21
2.3 Robust Spatial Bead Encoding	22
Chapter 3 - Review of Microarray Image Analysis Techniques	27
3.1 Microarray Gridding.....	28
3.1.1 Microarray Gridding Techniques for Hexagonal Grids.....	35
3.2 Spot Segmentation.....	36
3.2.1 Comparisons of Segmentation Techniques	44
3.3 Spot Filtering	45
3.4 Image Enhancement	47
3.5 Direct Quantification	47
3.6 Background Estimation	48
3.7 Summary of Microarray Image Analysis Techniques.....	50

Chapter 4 - Review of Point Pattern Matching Techniques	53
4.1 Types of Point Pattern Matching Algorithms.....	55
4.2 Review of Point Pattern Matching Algorithms	56
4.2.1 Rigid Matching.....	56
4.2.2 Non-Rigid Matching.....	74
4.3 Summary of Point Pattern Matching Techniques.....	78
Chapter 5 - Bright and Dark Field Image Processing.....	81
5.1 Bright and Dark Field Image Gridding.....	82
5.1.1 Grid Parameter Estimation	84
5.1.2 Grid Fitting	87
5.1.3 Grid Refinement	93
5.1.4 Grid Merging and Checking.....	95
5.2 Micro-well Classification	96
5.2.1 Template Banks	98
5.2.2 Similarity-Based Feature Extraction	101
5.2.3 Object-based Feature Extraction	103
5.2.4 k -Nearest Neighbor Graph Construction	104
5.2.5 Community Detection based Clustering.....	107
5.2.6 Positive Cluster Identification	108
5.3 Experiments and Results	113
5.3.1 Bright and Dark Field Image Gridding Results.....	114
5.3.2 Micro-well Classification Results	119
5.4 Discussion	125
Chapter 6 - Fluorescence Image Processing.....	127
6.1 Fluorescence Image Segmentation	128
6.1.1 Block Segmentation.....	129
6.1.2 Image Segmentation	131
6.1.3 Region Tree Growing.....	133
6.1.4 Inference on Region Trees.....	135
6.1.5 Region Probability Estimation using Gaussian Mixture Model with Markov Random Field Priors.....	136
6.1.6 Community Detection based Initialization	139
6.2 Fluorescence Image Gridding.....	140
6.2.1 Candidate Generation	142
6.2.2 Grid Alignment.....	143
6.2.3 Hypergeometric Test for Grid Alignment Quality Estimation	144
6.2.4 Thin-plate Spline based Non-linear Grid Refinement	145

6.3	Experiments and Results	147
6.3.1	Comparison with Other Segmentation Algorithms	156
6.3.2	Discussion	163
Chapter 7 - Point Pattern Matching for Bead Encoding		168
7.1	Affine Invariant Point Pattern Matching	168
7.1.1	Sampling.....	170
7.1.2	Local Feature Extraction	170
7.1.3	Local Feature Matching.....	175
7.1.4	Candidate Transformation Estimation.....	177
7.1.5	Candidate Transformation Pruning	177
7.1.6	Candidate Transform Ranking.....	178
7.1.7	Candidate Transform Evaluation.....	178
7.1.8	Correspondence Estimation.....	179
7.2	Grid-based Point Pattern Matching	180
7.2.1	Binary Descriptor Matching	182
7.3	Experiments and Results	183
7.3.1	Comparison of Affine Invariant Matching with Existing Methods.....	190
Chapter 8 - Conclusions and Future Work.....		199
8.1	Contributions and Conclusions.....	199
8.2	Future Work	200
References		203

Abstract

Bead encoding is a key problem central to all multiplexed bead-based microarrays. Most existing bead-based microarrays require complicated and costly bead fabrication and/or sophisticated bead decoding hardware to achieve high multiplexing. Recently, the development of spatial bead encoding techniques have opened up the possibility of using pattern matching and image processing to develop highly multiplexed, high-throughput bead-based arrays. In this thesis, we improve the existing spatial bead encoding scheme, and develop a pipeline of computational methods, which allows for automated spatial bead encoding. Six novel computational methods, that automate the proposed bead encoding scheme, are developed in this thesis. The proposed scheme improves the previously reported spatial encoding schemes by making them better suited for use in real world scenarios.

In the spatial bead encoding scheme, a sequential bead deposition method is used to capture the identities of the beads in bright or dark field images of the array. While, a second fluorescence image of the array is used to quantify the target analyte concentrations associated with the beads. By aligning the two images using the patterns formed by the beads, both the bead identities (i.e. the target analyte associate with the bead) and the analyte concentrations are decoded. The six computational methods developed in this thesis, are grouped into three categories: bright and dark field image processing, fluorescence image processing, and bead pattern matching. These methods are developed such that they require minimal parameter tuning, and are able to deal with noisy images acquired in uncontrolled environments

As part of the bright and dark field image processing we have developed two novel methods: a fully automatic method for detecting the underlying micro-well grid structure in the images, and an unsupervised learning based method to classify the micro-wells as either empty or containing a bead. We show the ability of the proposed methods to deal with extreme amounts of noise and distortions using multiple datasets. While, microarray image gridding is a well-studied problem in the context of fluorescence images of planar microarrays, to the best of our knowledge

no methods have yet been developed for processing either bright or dark field images of bead-based microarrays.

Unlike in bright field images, where all the beads are visible, in fluorescence images only those beads that are expressed in the sample are visible. Therefore, in fluorescence image analysis the first step is to detect the beads in the image. To this end, we have developed an unsupervised, simultaneous bead detection and segmentation method which uses the coherence in shape and size of the beads to circumvent the need for parameter optimization. The main challenge in this segmentation is the large dynamic range across which the beads are expressed. We have also developed a probabilistic non-linear gridding method that attempts to establish the micro-well grid structure from the detected beads. While similar gridding techniques have been developed in the past, we make two contributions in this regard: first, we put the gridding method in a probabilistic framework, and second, we show that in certain cases it is impossible to ascertain if the correct grid has been detected without using additional information about the physical properties of the array.

The final step in the proposed bead encoding scheme is to establishing one-to-one correspondences between the beads detected in the bright field and fluorescence images. We have developed two methods for this matching. When the micro-well grid structure in the fluorescence image is successfully established, the grid information can be used to reduce the matching problem to a binary grid alignment problem. A grid matching method that is very fast and can handle a large number of beads ($>50,000$) has been developed for this purpose. Finally, to perform matching when the fluorescence image grid is not available, a novel point pattern matching method that is affine invariant and robust to small non-linear distortions has been developed. The point pattern matching method addresses the problem of matching large point sets in the presence of large amounts of outliers and occlusion. The proposed method is evaluated using several real and simulated datasets.

List of Tables

Table 2.1: SUMMARY OF BEAD ENCODING SCHEMES	22
Table 4.1: SUMMARY OF POINT PATTERN MATCHING ALGORITHMS	79
Table 5.1: SUMMARY OF THE FIVE EVALUATION DATASETS	113
Table 5.2: GRIDDING ACCURACY ACHIEVED ON THE FIVE DATASETS	117
Table 5.3: RUNNING TIMES OF PROPOSED GRIDDING AND MICRO-WELL CLASSIFICATION METHODS	119
Table 5.4: MICRO-WELL CLASSIFICATION ACCURACY ON THE FIVE DATASETS	122
Table 5.5: IMPROVEMENTS IN ACCURACY DUE TO EVOLUTIONARY ALGORITHM BASED GRID PARAMETER ESTIMATION AND ICP BASED LOCAL OPTIMIZATION WITHIN REJSAC	125
Table 6.1: DETAILS OF THE DATASET USED TO EVALUATE THE PROPOSED BIOASSAY IMAGE SEGMENTATION AND GRIDDING METHODS	151
Table 6.2: BEAD DETECTION ACHIEVED USING THE PROPOSED FOREGROUND SEGMENTATION METHOD	151
Table 6.3: BIOASSAY IMAGE GRIDDING ACCURACY BEFORE AND AFTER THIN-PLATE SPLINE BASED REFINEMENT	155
Table 6.4: RUNNING TIME OF THE BIOASSAY IMAGE SEGMENTATION AND GRIDDING METHODS	156
Table 6.5: ACCURACY OF VARIOUS OVER-SEGMENTATION METHODS	160
Table 6.6: COMPARISON OF BIOASSAY BEAD DETECTION ACHIEVED USING VARIOUS SEGMENTATION TECHNIQUES	161
Table 7.1: RATIOS r_1 AND r_2 FOR THE NINE INTERSECTION TYPES SHOWN IN FIG. 7.2	174
Table 7.2: DETAILS OF THE DATASET USED TO EVALUATE THE PROPOSED MATCHING METHODS	184
Table 7.3: MATCHING ACCURACIES ACHIEVED ON REAL DATA	185
Table 7.4: SIMULATION PARAMETERS AND COMBINATIONS OF THEIR VALUES USED TO GENERATE SIMULATED TEST DATA	185
Table 7.5: MATCHING ACCURACIES ACHIEVED ON THE SIMULATED DATA	186
Table 7.6: PARAMETERS USED TO GENERATE SIMULATED DATA FOR MATCHING COMPARISONS	191

List of Figures

Fig. 1.1. A diagram of a biochemical assay. Shown here is a sandwich assay; one of many types of biochemical assays.	2
Fig. 1.2. A spotted planar array.	3
Fig. 1.3. An on-substrate bead-based microarray. (a) depicts a typical on-substrate bead-based microarray chip with multiple wells for performing multiple, multiplexed assays simultaneously, (b) a dark field image of a section of a well after bead deposition, (c) a fluorescence image of roughly the same section of the well acquired after the assay.	4
Fig. 1.4. The bead encoding problem.	6
Fig. 2.1. The bead encoding problem and the spatial bead encoding scheme.	23
Fig. 2.2. The proposed <i>robust spatially addressable bead encoding</i> pipeline.	25
Fig. 4.1. Examples of occlusions, outliers and missing beads. (a) a point set showing the outline of a fish, (b) examples of occluded regions marked using solid boxes and (c) examples of occluded regions, outliers and missing points.	55
Fig. 4.2. Taxonomy of point pattern matching algorithm. The point pattern matching algorithms are grouped into three categories: algorithms that are predominantly used for rigid matching (blue box), those that are predominantly used for non-rigid matching (yellow box), and those that are used for both rigid and non-rigid matching. Here we emphasize that these are not hard classifications, rather the approaches are categorized based on their most common usage.	56
Fig. 5.1. Small regions of bright field images. (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.	82
Fig. 5.2. Overview of the gridding method.	83
Fig. 5.3. (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.	87
Fig. 5.4. The bootstrapped and least trimmed squares ICP algorithm. Two point sets are shown using orange dots and blue circles. (a), (b) and (c) are 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.	93
Fig. 5.5. 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 are marked with red dots and the detected modes are marked with blue circles.	94
Fig. 5.6. A pair of micro-well images from two different regions of the same image.	96
Fig. 5.7. Two pairs of micro-well images. Each pair has a negative image on the left and a positive image on the right. The positive and negative images in each pair are from image	97

regions that are close to each other.

- Fig. 5.8. Gaussian shaped positive template bank images. 99
- Fig. 5.9. Elliptical and circular positive template bank images. 100
- Fig. 5.10. Negative template bank images. 100
- Fig. 5.11. Principal components analysis based normalization of the similarity-based features. (a) a small region of a bright field image, (b) similarity-based representation of the micro-wells in the region shown in (a) (the red dots are positive samples and the blue dots are negative samples), and (c) the data points in (b) after PCA based normalization. 103
- Fig. 5.12. Community detection using k -NN graphs of varying sparsity. (a) and (b) are two microarray image blocks. (c) PCA normalized similarity-based representation of the micro-wells in (a). The data points are colored based on the clustering achieved by the community detection algorithm [214] using a k -NN graph constructed with $k = 249$. (d) PCA normalized similarity-based representation of the micro-wells in (b). The data points are colored based on the clustering achieved by the community detection algorithm and a k -NN graph constructed with $k = 95$. (e) the same data points shown in (c) but colored based on the clustering achieved by the community detection algorithm using a k -NN graph constructed with $k = 21$. (f) the same data points shown in (d) but colored based on the clustering achieved by the community detection algorithm using a k -NN graph constructed with $k = 6$. 105
- Fig. 5.13. Effect size of Student's t-test to retain power $\Pi=0.99$ at significance level $\alpha=0.001$ for samples of equal size (N_s) and equal variance $\sigma^2=0.25$. 111
- Fig. 5.14. An illustration of linkage trees generated by hierarchical clustering. 111
- Fig. 5.15. The gridding achieved on a dirty image from the LRD dataset using the proposed method. (a) the original bright field image. The blue boxes are the approximate regions of the image from which the insets shown in figures (b), (c), (d) and (e) are extracted. (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. 115
- Fig. 5.16: The gridding achieved on the images presented in Fig. 5.1 116
- Fig. 5.17. The distributions of d_{err} achieved using the proposed method on the 5 datasets. (a), (b), (c), (d) and (e) are the distributions of d_{err} for the HRD, HRB, CLUST, LRC and LRD datasets respectively. The blue and the red curves show 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. 118
- Fig. 5.18. 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 (a) the top left block's grid points can be clearly seen as being skewed towards the top-right of the beads. 119
- Fig. 5.19: Classification output on LRD images. Left column shows image blocks from the LRD dataset and the right column shows the classification output overlaid on those images. The micro-wells detected as positive are marked using a blue box. The amount of artifacts in the images progressively increases from top row to bottom row. 123
- Fig. 5.20: Classification output on HRB and HRD images. The top-left image block is from the HRB dataset and bottom left image is from the HRD dataset. The right column shows the classification output overlaid on the images shown on the left. The micro-wells 124

detected as positive are marked using a blue box. The two images have starkly different bead density but the proposed method is able to accurately classify positive beads in both images. The bottom image also has irregular illumination.

Fig. 6.1. A sparsely populated fluorescence image where only a few beads are present. The micro-well grid structure cannot be readily discerned from such images. 127

Fig. 6.2. Regions of fluorescence image where bead segmentation is challenging. The two beads in (a) have a large difference in their intensities. The lowly expressed bead which is faintly visible is pointed to by the red arrow. In (b) a few beads are stacked with each other. The red arrow points to a pair of stacked beads where one of the beads is very lowly expressed. In (c) there are bright noise particles. 128

Fig. 6.3. Breaking the bioassay image into blocks. (a) a grid placed on the bioassay image. The grid in (a) is deformed to obtain the grid shown in (b). (c) and (d) show close-ups of two regions of (b). (e) the graph structure used to deform the grid lines. The green (dotted) and red lines show two possible cuts. The red cut is preferred over the green cut because it has fewer turns. (f) a loop like formation of two grid lines that is undesirable. 132

Fig. 6.4. Shows regions trees grown from the image segmentation output 133

Fig. 6.5. Two images demonstrating the over gridding problem. The red (filled) dots indicate the segmented bead regions and the blue (hollow) dots indicate the fitted grid points. Notice that the pattern of red dots is the same in both the images but different grids can fit this pattern. Here, a fit is estimated as a transformation of an ideal hexagonal grid that minimizes the sum of squared distance between the segmented bead regions and their nearest grid points. In images with few segmented bead regions it is often possible to fit grids with smaller spacing between grid points (finer grids), than that of the true grid, as finer grids are more likely to result in smaller sum of squared distances. We refer to this as the over-gridding problem. 141

Fig. 6.6. Candidate generation for matching an ideal hexagonal grid to the fluorescence image (see text for details). 143

Fig. 6.7. Two different grids aligned onto the same image. The grid in (b) produces a better fit. The grid in (a) can be rejected using the hypergeometric test. The detected beads are shown as blue dots. 146

Fig. 6.8. Segmentation output of various bioassay image regions. The detected foreground regions are overlaid (in red) over the segmented image. 149

Fig. 6.9. Grid detection output on various bioassay images. The detected grid centers are drawn as blue dots. 153

Fig. 6.10. The first row shows four histogram stretched images that were used for detectability and over-segmentation comparisons. The second to sixth rows (in that order) show the outputs of MSER, MS, GC, HC, and KM based over-segmentations applied on the images shown in the top row. 158

Fig. 6.11. The running times of the four over-segmentation methods. k-means clustering was performed for 10 clusters. Publicly available implementations of KM, MSER [224], MS [225], and GC [226] were used for the comparisons. All implementations are in C++. 160

Fig. 6.12. The first row shows four histogram stretched images that were used for detectability and over-segmentation comparisons. The second row shows the segmentation achieved using the proposed method. The third and fourth rows show the output of Hough transform based circle detection on Otsu's threshold based binary segmentation output for different sensitivities. Here sensitivity is a parameter of Hough 162

transform. Even at high sensitivities all the beads cannot be segmented. The fifth and sixth rows show the output of Hough transform based circle detection on k-means based binary segmentation output.

Fig. 6.13. The first row shows four manually labelled histogram stretched images that were used for detectability and over-segmentation comparisons. The second row shows the output of CVAC based binary segmentation initialized using the manual labelling shown in the first row. The third, fourth and fifth rows show the binary segmentation outputs obtained using OT, HCBS, and KMBS respectively. 163

Fig. 6.14. The first row shows four histogram stretched images used for evaluation. The second, third and fourth rows show the k-means segmentation output for different values of N_L (the number of clusters). The fifth row shows the segmentation achieved using the proposed method using $N_L = 10$. 165

Fig. 7.1. Invariant ratios extracted from 4 points. Two patterns of four points that are equivalent under affine transformation are shown in (a) and (b). The pattern in (c) is obtained from the pattern in (a) by flipping points p_0 and p_2 . An ordered arrangement of the points from the perspective of point p_m is shown in (d) and an ordered arrangement of the points from the perspective of point p_i is shown in (e). Notice that the cross product between the vectors $\overrightarrow{p_m p_i}$ and $\overrightarrow{p_m p_j}$ in (d) and between the vectors $\overrightarrow{p_i p_m}$ and $\overrightarrow{p_i p_n}$ in (e) are positive. 171

Fig. 7.2. The nine intersection types of two directed lines. 173

Fig. 7.3. Line Context feature extraction. 174

Fig. 7.4. An illustration of the ratio range calculation for soft matching of FPAI features. 176

Fig. 7.5. Directed path for traversing a hexagonal grid. Three orderings of the 6-NNs of a grid point in a hexagonal grid are shown in (a), (b), and (c). The NN ordering shown in (a) is used to generate the directed path shown in (d). The ordering of the NNs in (b) can be used to generate a directed path similar to the one shown in (d). However, the ordering of the NNs in (c) cannot be used. 180

Fig. 7.6. The sequence of moves used to generate the directed path shown in Fig. 7.5(d) 182

Fig. 7.7. Correspondence accuracies of the affine invariant and grid-based matching methods. (a) and (b) show the Rank 1 and Rank 10 correspondence accuracies of the affine invariant matching method, and (c) shows the Rank 1 correspondence accuracy of the grid-based matching method. 187

Fig. 7.8. The running time of the matching methods. (a) time taken by the affine invariant matching method to match point sets of different sizes using $n_c = 5\%$ of n_b . (b) time taken by the grid-based matching method to match point sets of different sizes using $n_c = 15\%$ of n_b . 188

Fig. 7.9. Examples of matching achieved using the proposed method. (a) and (d) are two query points and (b) and (e) are two template points. (c) and (f) show the query and template points aligned. 189

Fig. 7.10. The robustness of CPD, TPS-EM, LLAH, LGH, and the proposed method (PM) at different levels of outliers (points in query that do not have a match in template). LLAH could be run only till 80% outlier level due to its high memory requirements. 191

Fig. 7.11. Number of invariants extracted by LLAH, GH and the proposed method (PM) to find a match at 95% outlier level, for increasing point set sizes. LGH requires the same number of invariants as the proposed method for the same size of local neighbourhood. 192

Fig. 7.12. The effect of jitter noise on the correspondence matrix generated using LGH 195 when matching two identical patterns. The columns from left to right show correspondence matrices at 0%, 5%, and 10% jitter levels respectively. The rows from top to bottom show correspondence matrices for 10^3 , 10^6 , and 10^9 quantization bins respectively.

Fig. 7.13. The effect of jitter noise on the matching matrix generated using the proposed 196 affine invariant matching method when matching two identical patterns. The columns from left to right show the counts, the scores, and the top 5% ranked cells in the matching matrix respectively. Rows from top to bottom represent 0%, 5%, and 10% jitter noise levels respectively.

Fig. 7.14. Comparison of robustness of the proposed affine invariant matching method 197 and LGH under jitter noise.

Fig. 7.15. Comparison of robustness of the proposed affine invariant matching method 197 and LGH under jitter noise for point sets of varying densities.

List of Abbreviations

4CS	4-Correspondences Set
4PCS	4-Point Congruent Sets
ARS	Attributed Relational Structures
B&B	Branch And Bound
BGM	Balanced Graph Matching
BIC	Bayesian Information Criterion
CC	Connected Component
CCD	Charge-Coupled Device
cDNA	complementary Deoxyribonucleic Acid
CLUST	Clustered
Copalis	Coupled Particle Light Scattering
CPD	Coherent Point Drift
CT	Computed Tomography
CV	Coefficient Of Variation
CVAC	Chan and Vese Active Contour Model
DNA	Deoxyribonucleic Acid
DSLR	Digital Single-Lens Reflex
EA	Evolutionary Algorithm
ELISA	Enzyme Linked Immunosorbent Assay
EM	Expectation Maximization
F-19 NMR	Fluorine-19 Nuclear Magnetic Resonance
FDR	False Discovery Rate
FFT	Fast Fourier Transform
FN	False Negative
FNR	False Negative Rate
FP	False Positive
FPAI	Four Point Affine Invariant
FPR	False Positive Rate
FTIM	Fourier Transform Infrared Microspectroscopy
GA	Genetic Algorithm
GC	Graph Cut
GED	Graph Edit Distance
GH	Geometric Hashing
GLM	Generalized Linear Model
GMM	Gaussian Mixture Model
GMM-MRF	Gaussian Mixture Model Based Markov Random Field
GPU	Graphical Processing Unit
HC	Hierarchical Clustering
HCBS	Hierarchical Clustering based Binary Segmentation
HRB	High Resolution Blurry
HRD	High Resolution Dense

HT	Hough Transform
ICM	Iterative Conditional Modes
ICP	Iterative Closest Point
ICP-REJSAC	Iterative Closest Point-Rejection Sampling And Consensus
IR	Infrared
KC	Kernel Correlation
KDE	Kernel Density Estimation
KM	k -Means
KMBS	k -Means Clustering based Binary Segmentation
k-NN	k -Nearest Neighbor
LAP	Linear Assignment Problem
LGH	Local Geometric Hashing
LLAH	Locally Likely Arrangement Hashing
LMA	Levenberg-Marquardt Algorithm
LM-ICP	Levenberg-Marquardt-Iterative Closest Point
LO-RANSAC	Locally Optimized-Random Sampling And Consensus
LRC	Low Resolution Clean
LRD	Low Resolution Dense
MAP	Maximum a Posteriori
MCMC	Markov Chain Monte Carlo
MCS	Maximum Common Subgraph
ML	Maximum Likelihood
MLC	Metal Ligand Complex
MLP	Multi-layer Perceptron
MMP	Magnetic Micro-particle
MPE	Most Probable Explanation
MRF	Markov Random Field
MRI	Magnetic Resonance Imaging
mRNA	messenger Ribonucleic Acid
MS	Mean-shift
MSER	Maximally Stable Extremal Region
MST	Minimal Spanning Tree
NCC	Normalized Cross Correlation
NMR	Nuclear Magnetic Resonance
NN	Nearest Neighbor
NP	Nanoparticle
OMT	Optimal Multilevel Thresholding
OMTG	Optimal Multilevel Thresholding Gridding
OT	Otsu's Thresholding
PAM	Partition Around Medoids
PC	Principal Component
PCA	Principal Component Analysis
PET	Positron Emission Tomography
PROSAC	Progressive Sampling And Consensus

QAP	Quadratic Assignment Problem
RANSAC	Random Sampling And Consensus
RE	Rare Earth Ion
RET	Resonance Energy Transfer
RF	Radio Frequency
RICP	Robust Iterative Closest Point
ROC	Receiver Operator Characteristic
RPM	Robust Point Matching
R-RANSAC	Randomized-Random Sampling And Consensus
RT	Radon transform
SCP	Structural Correspondence Probabilities
SERS	Surface Enhanced Raman Scattering
SVD	Singular Value Decomposition
SVM	Support Vector Machine
TN	True Negative
TP	True Positive
TPR	True Positive Rate
TPS	Thin-Plate Spline
TPS-EM	Thin-Plate Spline-Expectation Maximization
TPS-RPM	Thin-Plate Spline-Robust Point Matching
UV	Ultraviolet
WMW	Wilcoxon-Mann-Whitney

Chapter 1 – Introduction

Microarrays are a vital tool for molecular detection and quantification in biomedical research and disease diagnosis. They are widely used to detect and quantify antibodies, oligonucleotides and peptides in biological samples. While there are many types of microarrays, deoxyribonucleic acid (DNA) microarrays, immuno-arrays, enzymatic arrays, etc. to quantify the different types of molecules, they work in a similar way. The biochemical reactions and steps involved in a typical microarray experiment are shown in Fig.1.1. To capture the target molecules in the sample, capture or probe molecules are attached to a solid substrate. The biological sample is then brought in contact with the solid substrate surface and the target molecules bind to the arrayed probe molecules. A solution with fluorophore conjugated detection molecules is then added to the mix and the detection molecules bind to the captured target molecules. When the fluorophores are excited with light of a particular wavelength they re-emit certain portion of the incident energy back as light. The intensity of this light is proportional to the concentration of the target molecule in the sample. Using a control whose concentration in the sample is known, the relative intensities of the control and the target molecules are used to estimate the true concentration of the target molecule in the sample.

While single molecule quantification techniques, such as western blotting for complementary deoxyribonucleic acid (cDNA) quantification and enzyme linked immunosorbent assay (ELISA) for immunoassays, had already been well established by the 1980s, the introduction of spotted microarrays allowed for parallel detection of thousands of different molecules in a single experiment (or assay). This parallel detection is referred to as multiplexing. Spotted microarrays are fabricated by placing (or synthesizing) thousands of different types of probe molecules at specific locations (spots) on a planar substrate. Each spot contains thousands of probe molecules of the same type and different spots are made up of different probe molecules. Thus, a single assay can be used to quantify the concentration of thousands of target molecules simultaneously. Multiplexing gave researcher the ability to study the interactions between large sets of genes and proteins, and discover patterns and relationships between them. The further development of robotic

Fig.1.1. A diagram of a biochemical assay. Shown here is a sandwich assay; one of many types of biochemical assays.

spotting methods, automated scanning, and sophisticated computational analysis methods in the 1990s enabled high throughput analysis. However, the data produced by spotted microarrays is of limited quality. They are affected by various kinds of noise and artifacts and results between different arrays can vary significantly. This lack of reproducibility combined with the increasing demands for extracting greater amounts of molecular information from smaller sample volumes led to the development of the bead-based microarrays.

1.1 Bead-based Microarrays

Bead-based arrays use micrometer sized spherical beads (also referred to as microparticles and microspheres) as the solid support for the assay. The bead surface is coupled with the capture probe molecules and the assay illustrated in Fig. 1.2 takes place on the bead surface. Bead-based arrays have some significant advantages over traditional spotted arrays. The large surface to volume

Fig.1.2. A spotted planar array.

ratios of the spherical beads allow more probes to be immobilized, resulting in more accurate quantitation. The three dimensional structure of the beads lead to better diffusion and results in faster and better binding of the probe and target molecules. The spherical shape of the beads also allows more efficient mixing of the beads [1]. In spotted arrays, all probe molecules have to be attached to the same planar substrate. However, beads carrying different probes can be separately synthesized. This allows the beads to be conjugated with probes under conditions optimized for

their particular surface chemistry, resulting in a very high degree of reproducibility. Also bead-based microarrays can be easily customized for the specific set of probes being multiplexed by simply adding or removing beads carrying the desired probes.

1.2 Bead Encoding

In multiplexed spotted microarrays the target molecule associated with each spot can be easily identified using their positions in the fluorescence image (acquired after the assay), since the positions of the spots are known (fixed) before array fabrication. However, manipulation and assembly of the highly mobile and minute beads into an ordered pattern is challenging. Bead-based microarrays are therefore randomly assembled. There are two major formats of bead-based

microarrays: suspension bead arrays and on-substrate bead arrays. In suspension arrays the assay is performed with the beads suspended in fluid. While in on-substrate bead arrays the beads are assembled onto a substrate, using fluidic, magnetic, or electrostatic forces [1], before the assay. Fig. 1.3(a) shows an on-substrate bead-based array where the beads are deposited into micrometer sized wells (micro-wells) etched on the surface of a planar substrate. In either case the pre-conjugated beads are randomly distributed either on the substrate or in suspension. Therefore, in bead-based microarrays the identity of a bead (and the target molecule associated with it) must be separately encoded. This bead encoding problem is central to all multiplexed bead arrays. Fig. 1.4 illustrates the bead encoding problem.

Numerous bead encoding schemes have been developed in the past 25 years that range from simple colorimetric encoding schemes that label the beads using fluorescent dyes [2] to complicated graphical encoding schemes that attach graphical barcodes to beads. A detailed review of bead encoding schemes is presented in Chapter 2. These encoding schemes however, either offer limited multiplexing capacities or involve complicated (and costly) bead fabrication methods and/or decoding machinery. This makes highly multiplexed bead-based arrays inaccessible to the vast majority of academic and industrial laboratories.

1.3 Contributions

Recently, Ng et al. [3] and Trau et al. [4] developed a spatial bead encoding scheme to overcome some of the challenges of the traditional bead encoding schemes. In this thesis, a reformulation of the spatially addressable bead encoding scheme of [3] and [4] is proposed. The proposed method improves the practicality of the spatially addressable bead encoding scheme and makes it better suited for real world use. Further, in this thesis, a collection of image processing and pattern matching methods are developed to automate the proposed spatial bead encoding scheme so as to enable high-throughput analysis using it. The specific contributions made in this thesis are further elaborated in the following sub-sections.

Fig. 1.4. The bead encoding problem.

1.3.1 Robust Spatial Bead Encoding

The spatially encoding scheme of [3] and [4] works with on-substrate bead-based arrays and uses two images for bead encoding and decoding. One image is acquired during the on-substrate bead deposition, and the other image, a fluorescence image, is acquired after the assay. Using a sequential bead deposition method, the identities of the beads are recorded in the first image, while the second image gives the fluorescence intensities (and therefore target analyte concentrations) of each bead. By aligning the two images, using the patterns formed by the beads, both the bead identities and the analyte concentrations are decoded. In [3] and [4] the viability of the spatial encoding scheme was demonstrated for various types of biochemical assays. However, the bead decoding itself was performed manually. The beads in the images were manually identified and the images were manually aligned. While such manual decoding is possible for a 3-plex experiment with a few (≈ 50) beads, it is hard to envision such a manual decoding being used for a 250-plex

experiment involving 50,000 or more beads, in a high-throughput environment where 100s of experiments are being performed each day.

We re-formulate the spatial encoding strategy of [3] and [4] so as to eliminate some of the restrictions on how the images are acquired. We also show how a set of positive control beads can be used for image matching to achieve very high multiplexing. The proposed scheme is referred to as *robust spatial bead encoding*. It is important to note that the proposed method is not a novel bead encoding scheme, rather, it refers to a specific computational pipeline that outlines a certain way in which a specific set of computational methods can be used to improve the robustness of the encoding scheme of [3] and [4].

1.3.2 Multi-modal Optical Microscopy Image Analysis and Matching

Six computational methods have been developed in this thesis to solve various sub-tasks involved in *robust spatial bead encoding*. The six methods are grouped into three categories: bright and dark field image processing, fluorescence image processing, and bead pattern matching.

An automated high throughput bead encoding system can afford minimal manual intervention, if any at all. The image processing and matching methods have therefore been developed such that they require minimal parameter tuning. At the same time one of the objectives of this research is also to expand the reach of the bead-based array technology to smaller commercial and academic laboratories where any microscope or scanner already available can be used for the assay. The image analysis methods therefore have been developed keeping in mind that the images can be acquired using a multitude of different devices in uncontrolled environments and can have large amounts of noise, artifacts and distortions.

Bright and Dark Field Image Processing

Bright and dark field image processing involves two sub-tasks, detecting the grid structure in the deposition images and then locating the beads (see Fig. 1.3(b)). To this end, we have developed a novel gridding method to detect the grid of micro-wells and a semi-supervised classification method that classifies the micro-wells as empty or containing a bead. To the best of

our knowledge, no algorithms have yet been developed for grid and bead detection in bright and dark field images of bead-based microarrays. We show the ability of the proposed methods to deal with extreme amounts of noise and distortions using multiple datasets of varying characteristics. Apart from their use in the *robust spatial bead encoding* scheme, the bright and dark field image processing methods can also be useful in quality control during microarray fabrication.

Fluorescence Image Processing

Unlike in the bright and dark field images, the grid structure is not clearly evident in the fluorescence image (see Fig. 1.3(c)). Fluorescence image processing therefore involves first segmenting the beads and then using the bead locations to try and establish the micro-well grid structure. The main challenge in fluorescence image segmentation is the large dynamic range across which the beads are expressed. The existing bead segmentation and generic blob detection methods either require parameter optimization or result in sub-optimal detection and/or segmentation. We have developed an unsupervised, simultaneous detection and segmentation method which utilizes the coherence in the physical shape and size of the beads to circumvent the need for parameter optimization. While fluorescence image gridding is a well-studied problem, very few algorithms have been developed for bead-based microarrays, which present certain unique challenges. While some algorithms similar to our proposed fluorescence gridding method have been developed in the past, we make two novel contributions: 1. we put the gridding method within a rigorous probabilistic framework, and 2. we also show that in certain cases it is impossible to ascertain if the correct grid has been detected without using additional information about the physical properties of the array.

The fluorescence image segmentation and gridding methods developed here have wide applicability in various other types of microarray image analysis as well, including in traditional planar microarrays image analysis.

Bead Pattern Matching

We have also developed two novel bead pattern matching methods. The first method

referred to as *grid matching*, uses the grid information for matching the two images. This method is used when the grid structures in both images have been established. The other matching method referred to as *affine invariant matching*, uses only the bead location information to match the images. We show that positive control beads can be used for bead pattern matching in 250-plex or higher multiplex experiments and we establish the number of positive control beads required for robust multiplexing. The bead pattern matching is an inherently difficult problem to solve as it involves extremely large amounts of mismatch between the bead patterns. The *grid based matching* uses the grid information to simplify the problem and solves a binary grid alignment problem. It can therefore be used to match very large patterns very quickly. The *affine invariant matching* however, addresses a harder problem. The problem of matching random, uncorrelated point patterns under affine transformation and with large amounts of outliers and occlusions. While this pattern matching problem is very well studied and has broad applicability in various areas of computer vision, no existing algorithm adequately addresses all the requirements that an automated bead encoding system entails. The proposed *affine invariant matching* method is evaluated using a number of real and simulated datasets to establish its efficacy in solving the bead encoding problem.

Throughout the rest of this thesis, we use the terms bioassay image and fluorescence image interchangeably. Similarly, the terms deposition image and bright or dark field image are also used interchangeably.

1.4 Motivation

The rapidly increasing cost of health care is a major concern that affects every individual. In 2014, the United States spent 17.5% of its GDP on healthcare, nearly \$3.0 trillion (\$9,523 per person). Almost 50% of this was spent on clinical services and hospital care. It is estimated that approximately 700 million immunodiagnostic tests were run in 2006 in the United States alone. Microarrays are a mainstay in both clinical diagnostics, and biomarker and drug discovery. The computational methods developed in this thesis as part of the *robust spatial bead encoding* scheme have the potential to increase the reach of the bead-based microarray technology by making it more affordable. The higher accuracy and reproducibility offered by bead-based microarrays can be a

critical difference in clinical diagnostics, while their higher multiplexing capacities can add much needed speed in biomarker and drug discovery. Given the ubiquity of microarrays in almost all areas of biomedical sciences, any improvement in the microarray technology has the potential to significantly impact various areas of biomedical sciences.

Despite significant advances in biological sciences the number of new drugs and therapies being introduced into commercial use has seen a steady decline over the last 20 years [5]. This trend can easily be attributed to the fact that introduction of a new drug today requires an investment of approximately a billion dollars. Introduction of new state-of-the-art technology is commonly cited as the predominant factor in increasing healthcare costs. For years the trend has been to build sophisticated hardware to solve difficult problems in medicine. Indeed even in bead-based microarrays the trend has been to develop sophisticated beads such as those with nanostructured shells, various types of barcodes, and complex optical properties [2],[6]. This work is motivated based on the idea of developing computational methods that are robust enough to replace (or reduce the need for) sophisticated hardware.

1.5 Thesis Map

The thesis is organized as follows: various existing bead encoding schemes are reviewed in Chapter 2, the proposed *robust spatial bead encoding* strategy is also further discussed in Chapter 2, existing microarray image analysis methods are reviewed in Chapter 3, existing point pattern matching algorithms are reviewed in Chapter 4, the proposed bright and dark field image processing methods are presented in Chapter 5, the proposed fluorescence image processing methods are presented in Chapter 6, the *grid based matching* and *affine invariant matching* methods are presented in Chapter 7, and finally, in Chapter 8 conclusions are drawn and some future directions of this work are discussed.

Chapter 2 - Robust Spatially Addressable Bead Encoding

Over the past 25 years as the demand for molecular information has grown exponentially, bead encoding schemes have also evolved rapidly from simple colorimetric encoding schemes to complex barcoding schemes, in the pursuit of ever higher-multiplexed assays. In this chapter, we first review some of the existing bead encoding schemes and then present the proposed *Robust Spatial Bead Encoding* scheme.

2.1 Existing Bead Encoding Methods

The reproducibility and the simplicity of the encoding and decoding process and the cost of implementing the scheme are often the deciding factors on the choice of encoding method [7],[8]. Some bead encoding schemes also restrict the size and the type of beads that can be used. For instance, graphical encoding schemes ([9],[10],[11]) require larger beads to accommodate their graphical barcodes. However, with larger beads efficient mixing can be hindered. Some organic dye and quantum dot based optical encoding schemes require polystyrene beads, which have poor compatibility with organic solvents and higher background fluorescence[12]. The existing bead encoding schemes are therefore evaluated based on: 1. their reproducibility and stability, 2. their multiplexing capacity, 3. the cost and complexity of bead fabrication, 4. the cost and complexity of decoding, and 5. the restrictions on the types of beads (such as size or material) that can be used with the encoding. The schemes are grouped into seven categories following the classification of Wilson et al.[7].

2.1.1 Optical Encoding

Optical encoding uses additional compounds attached to the beads, whose absorption or emission spectra are used to identify the beads. Optical encoding is by far the most commonly used encoding strategy. Several well-established optical encoding methods are discussed here.

Fluorescent Dye based Optical Encoding

The most common encoding mechanism is to dope beads with one or more fluorescent

organic dyes. The dye molecules are trapped onto the beads at different concentrations to produce different intensity levels. When combined with the different emission spectra (color) of the different dyes, a large number of unique spectral codes can be generated. Theoretically, the number of available codes C is $C = N^{(m-1)}$ where N is the number of intensity levels (dye concentrations) used and m is the number of colors (dyes). In practice however, the multiplexing capacity is limited by the number of compatible dyes whose emission spectra have limited overlap. Moreover, one (reporter) dye must be dedicated for analyte concentration reporting. By far the most widely used commercial encoding mechanism available today is Luminex Corporation's xMAP technology [13] which uses $5.5\ \mu\text{m}$ diameter beads encoded with two organic dyes at ten different concentrations, allowing 100-plex assays.

Organic dye encoded beads can be inexpensively decoded using fluorescence microscopy imaging. Multiple images acquired using different optical filters (one for each encoding dye's spectrum) are used to locate and decode the bead identities, while a separate image acquired using an optical filter suitable for the reporter dye's spectrum is used to quantify the analytes. An alternative and slightly more robust approach that is primarily used with suspension arrays, is to use a flow cytometer. Flow cytometers use two lasers, one laser excites the reporter fluorophores while the other laser excites the encoding fluorophores. Kellar and Iannone [14] demonstrated an 8-plex immunoassay using a suspension array with Luminex [13] beads decoded using a flow cytometer.

The broad emission spectra of organic dyes and the spectral overlap between dyes is a major limiting factor. In practice, most applications can use only up to 3 to 4 dyes. Another major concern with organic dyes is photobleaching; a phenomenon in which the dye molecules are permanently unable to fluoresce due to non-specific reactions with other compounds in the sample.

Optical Encoding with Quantum Dots

A more robust alternative to organic dyes are fluorescent semiconductor nanocrystals known as quantum dots. Quantum dots are highly luminescent fluorophores whose emission spectra can be continuously tuned by changing their size. In contrast to the organic fluorescent dyes that

have narrow excitation spectra, quantum dots have a continuous excitation spectrum and therefore can be simultaneously excited. Moreover, quantum dots have narrower emission spectra and are also more stable and less prone to photobleaching than organic dyes.

In theory, quantum dots of different colors and different intensities can be used to generate $C = N^m - 1$ unique codes, where N is the number of intensity levels and m is the number of colors. However, spectral overlap is an issue with quantum dots as well, and in practice the number of unique codes that can be generated is much less. In particular, while the number of distinguishable colors is higher for quantum dots than organic dyes, the number of distinguishable intensity levels are still low. Han et al. [15] developed techniques to incorporate multiple quantum dots onto small (1.2 μm) polymer beads in carefully controlled ratios. Such a technique allows the development of an encoding strategy where ratios of quantum dots of different colors can be used to identify a bead. A 3-plex experiment was performed to show the viability of the approach where 3 different colored quantum dots were incorporated onto beads in ratios of 1:1:1, 1:2:1 and 2:1:1. However, even when using quantum dots it is still necessary to use a dye for analyte quantification and the region of the spectrum used by the dye is unavailable for multiplexing. Moreover, quantum dots can be of bigger dimensions and have specific surface chemistries, which makes incorporating them into beads more complicated [16], [17].

Time Resolved Optical Encoding

When two light-sensitive (dye) molecules with overlapping emission spectra are in close spatial proximity one molecule (the donor) may transfer energy to another molecule (the acceptor) through resonance energy transfer (RET). RET can be used to modulate the luminescence decay times of dyes, by changing the ratios in which the dyes are mixed. Keij and Steinkamp [18] used the varying luminescence decay times to encode beads incorporated with different ratios of orange and red dyes. They showed the viability of the method by encoding 13 batches of beads. However, the hardware implementation costs of flow-cytometers that can measure the luminescence decay times of beads remains high.

Kurner et al. [19] developed phosphorescent nanoparticle composed of phosphorescent

ruthenium metal ligand complex (MLC) and cyanine dyes where the MLC acted as the donor and the cyanine dye acted as the acceptor. Compared to the Luminex beads used in [18] more efficient RET could be achieved because of the close (nanoscale) spatial proximity of the MLC and the cyanine dye and also because of the broader emission spectrum of MLC which allowed better spectral overlap between the MLC and cyanine dye. By varying the concentration of the cyanine dye the efficiency of the RET could be controlled and nanoparticles with distinct fluorescence emissions and decay times were constructed. They demonstrated that 16 distinguishable combinations of fluorescence emissions and decay times could be achieved using four different cyanine dyes.

Active Encoding using Split-and-Mix Method

The split-and-mix method [20] is a widely used technique for generating large combinatorial libraries of molecules. The technique involves three steps: 1. randomly splitting the beads into batches, 2. depositing a specific type of monomer onto each batch of beads and 3. mixing the beads back together and repeating steps 1 to 3. This process sequentially deposits a random set of monomers onto each bead, forming a unique polymer on each bead. By repeating the steps multiple times, millions of polymers (each on a different bead) can be synthesized using combinations of a small set of monomers. While the combinatorial libraries generated using this method play a vital role in pharmaceutical industry and drug discovery, encoding such large libraries has been challenging. Others have also used smaller (0.2-3.0 μm diameter) dye doped silica particles to barcode the larger support beads (2.0-500 μm diameter) in a way that captures the history of the monomers that were deposited onto the bead. They added an additional step in the split-and-mix method where in each cycle, the beads were first made to absorb a few (approximately 50-100) small dye doped silica reporter particles, before the monomers were deposited onto them. Each type of reporter particle was constructed by absorbing multiple dyes and therefore had a unique spectra. By using a different silica reporter particle for each monomer in each cycle, a unique collection of colored silica particles (barcode) was absorbed onto each bead. Decoding was achieved using fluorescence microscopy imaging of the beads using multiple optical filters. A

library containing 20 different amino acids was generated in 2 cycles of the split-and-mix method where the beads were split into 10 batches in each cycle. The beads were encoded using 20 different silica reporters with each reporter containing a combination of 2 or 3 different dyes.

Nanostructured Shells for Optical Encoding

Microbeads with a polymeric core and multiple concentric nanostructured outer shells have also been constructed for optical encoding. The nanostructured outer shells allow developing more complicated (and therefore more diverse) emission spectra. Bladderer and Vrij [21] developed beads with concentric shells of fluorescent dye conjugated polymers. The overlapping emission spectra and close spatial proximity of the dye molecules in the concentric shells resulted in a complicated overall emission spectra due to resonance energy transfer (RET). However with increasing number of concentric shells the method became less and less stable. Also the RET resulted in broader emission spectra. Yang et al. [22] further improved the technique by alternatively depositing shells made up of fluorescent dye conjugated positively charged polymers and negatively charged dye free polymers. The dye free polymer layers acted as a spacing shell and limited the RET between the dye conjugated polymer shells. Beads with as many as 13 layers were constructed. Their results show that beads with at least 4 non-overlapping intensity ranges can be constructed using a single dye and when combined with 3 different dyes, up to 12 distinct types of beads could be generated.

Optical Encoding using Raman Spectroscopy

Surface enhanced Raman scattering (SERS) is a phenomenon where the Raman scattering intensity of molecules that are in the vicinity of rough nanostructured metallic surfaces is enhanced 10^{10} to 10^{11} times, thereby allowing single molecule detection and quantifications in the femtomolar and attomolar range.

Jin et al. [23] used composite Raman dye doped gold nanoparticles (NPs) (≈ 0.5 nm diameter) as detectors by conjugating them with detection molecules in a sandwich assay (similar to the one shown in Fig. 1.1). Glass beads (≈ 300 μm diameter) conjugated with probe molecules

were used to capture the target molecules while detection molecules conjugated with gold NPs bound to the captured target molecules. To identify the beads, the gold NPs for each type of detection molecule were doped with a different composite multiple Raman dye, generated by mixing multiple dyes in varying ratios. For further SERS enhancement the beads with the gold NPs bound to them were immersed in silver solution. The silver deposit on the gold NPs produced SERS signals strong enough to accurately distinguish the dye ratios in the NPs. The multiple dye doped NPs produced a complicated composite Raman spectra. However, since the wave number of the dominant peak in each individual dye's Raman spectra is known, the ratio of the dyes could be computed as the ratio of intensities found at these wave numbers in the composite spectra of the bead. They showed the efficacy of the method using a 9-plex assay where a combination of 2 and 3 dyes were mixed to generate the 9 ratios.

Fenniri et al. [24] developed polystyrene beads encoded by combining their infrared (IR) and Raman spectroscopic signatures. The peaks in the spectroscopic signatures were used to define barcodes. Unique spectral properties of the beads were achieved by absorbing two or three co-monomers at different concentrations. They identified six co-monomers based on their inertness, spectral overlaps and stability, which can be combined to generate twenty four unique barcodes.

Doering [25] developed a core-shell bead, with a gold core and a silica shell. Raman dyes were absorbed onto the gold core surface for SERS analysis based decoding, while the silica outer shell acted as both a protective layer for the core as well as a surface for absorbing capture molecules. The use of the Raman dyed metallic nanoparticle as the core, instead of using them as detectors (as was done in [23]) improved data reproducibility. While they identified a class of Raman dyes that could be stably absorbed onto gold nanoparticles, multiplexed analysis was not reported.

2.1.2 Chemical Encoding

Stoeva et al. [26] developed a "bio-barcode" based encoding system for immunological analysis using capture probe (antigen) conjugated magnetic micro-particles (MMP) and gold nanoparticles (NPs) functionalized with oligonucleotide molecules as detectors. The

oligonucleotide molecules acted as the bio-barcodes and multiplexing was achieved by conjugating each type of detector NP with a different oligonucleotide molecule. After the assay, the MMPs were separated from the gold NPs using a magnetic field. A de-hybridization step then cleaved the oligonucleotides from the NPs and a conventional spotted microarray chip was used to quantify the cleaved DNA, which indirectly gave a measure of the antigen specific target's concentration. They performed a 3-plex experiment and demonstrated detectability in the attomolar level. A drawback of this method is that decoding can take several hours as it involves the spotted microarray based quantification. Oh et al. [27] further improved the method's sensitivity by using fluorophore tagged DNA conjugated polystyrene beads as detectors instead of gold NPs so that quantification could be done by direct fluorescence microscopy imaging. Chemical encoding techniques present two problems: first they might interfere with compound synthesis, and second, the decoding methods are difficult to automate [7].

2.1.3 Spectrometric Encoding

Spectrometric encoding methods involve directly placing the beads into a spectrometer and identifying the chemical encoding tags by mass spectroscopy. Brummel et al. [28] used direct mass spectrometry to decode polymer beads conjugated with trimers synthesized using the split-and-mix method. They used time-of-flight secondary ion mass spectrometry to identify the molecular weights of the trimers conjugated to the beads. The main advantage of the method was that it allows for quantitation at very low (femtomolar level) concentrations. However, the sensitivity of the technique depends on molecular characteristics of the sample being studied.

Rahman et al. [29] combined Fourier transform infrared microspectroscopy (FTIM) and direct Raman spectroscopy to perform a 7-plex experiment. They used relatively larger beads (≈ 45 μm diameter) tagged with active IR and/or Raman labels. The beads were flattened into disks and inspected individually. The spectroscopic signatures obtained from direct FTIM and Raman spectroscopy were used to decode them.

Hochlowski et al. [30] used Fluorine-19 nuclear magnetic resonance (^{19}F -NMR) spectroscopy to encode a library of beads generated using the split-and-mix method. Commercially

available fluorine-carboxyl acids were attached to the beads as tags. Each tag had a unique, distinguishable fluorine chemical shift. They further showed that a combination of multiple fluorine tags can be used to generate a large number of unique codes. Beads were individually decoded using their NMR spectra.

2.1.4 Graphical Encoding

Dejneka et al. [9] developed cylindrical glass beads with "micro-barcode": micrometer sized barcodes containing patterns of fluorescent materials. They used rare earth ion (RE) doped glass compositions to create barcodes of different colors. REs have emission spectra that are half as wide as those of quantum dots and distinct from those of the reporter dyes. Moreover, RE emissions are extremely resistant to photo bleaching. Glass beads also have some advantages over polymeric beads, such as better compatibility with organic solvents and lower background fluorescence. Another advantage of using REs is that a common ultraviolet (UV) excitation source can be used to interrogate all barcode elements. The emission spectra being in the visible range, the barcodes can be visibly resolved. Using N bars and C colors it is possible to create approximately $C^{N/2}$ usable barcodes. However the micro-barcode are ten times larger than traditional beads and their relative density is 2.5 times the density of water. Further, the complicated manufacturing involved in developing the micro-barcode means that the method is economically feasible only for very high volume manufacturing. At reported rates it would take about a month to produce as many micro-barcode as there are beads in a single milliliter of commercially available products [1].

Nicewarner-Pena et al. [10] developed micrometer sized metal rods encoded with sub-micrometer sized metallic strips. Predefined stripping patterns were generated by sequential electrochemical deposition of metal ions onto template membranes with predefined pores. Patterns of different metals and lengths were generated by controlling the size of the pores, the sequence in which metal ions were introduced, and the charge passed. To decode the beads, the metallic strips were identified based on their differential reflectivity using conventional light microscopy. A 2-plex DNA hybridization and immunological assay was performed to show the viability of multiplexed assays. While in [10] only gold and silver were used to generate multiple binary barcodes based on

length of the stripes, the authors in their previous works had deposited a total of 7 different elements and generated 13 different barcodes. Theoretically, it was shown that up to 4160 binary barcodes could be generated by varying strip lengths of just 2 metals. Decoding the beads based on reflectivity however, can be challenging and the use of multiple metallic surfaces requires careful consideration of their surface chemistries.

2.1.5 Electronic Encoding

Moran et al. [31] developed glass encased microchips that are pre-tuned to emit unique binary codes when pulsed with electromagnetic radiation. The number of unique binary codes that are available is virtually unlimited. The decoding of the beads is unambiguous and simultaneous. 25 different types of beads were generated using a 2-step split-and-mix method. The radio frequency (RF) tags of the beads were recorded during the split-and-mix steps to maintain a database of RF tags associated with each of the 25 conjugated polymers.

Nicolaou et al. [32] used a microchip capable of not only receiving and emitting radio frequencies (RF) but also storing RFs. The chip could be activated from a distance from 75-100 mm and allowed for instant inspection of the beads. The chips were embedded into polymeric beads coated with an inert layer of porous support. Here also a split-and-mix method was used to generate 24 different polymer conjugated beads. Each successive reaction in the split-and-mix method was recorded and stored in the microchips, allowing querying of the history of chemical reactions that a bead has undergone.

A major problem with this encoding strategy is the size of the RF devices, which require larger beads, and therefore restrict their usage. However, recent developments have indicated the use of relatively smaller radio frequency tags. PharmaSeq Inc., have developed “microtransponders” of the dimensions $250\mu\text{m}\times 250\mu\text{m}\times 250\mu\text{m}$ which are integrated circuits consisting of photovoltaic cell, electronic circuitry and an antenna [33]. The microtransponders are decoded in a capillary or dedicated scanner where the laser excites the fluorescent reporter molecules simultaneously prompting transmission of its code as a radio frequency signal.

2.1.6 Physical Encoding

Physical characteristics of the beads such as size, shape, composition and refractive index have also been used for bead encoding. Benecky et al. [34] developed the Coupled particle light scattering (Copalis) technique which used the light scattering patterns to distinguish between beads of different sizes. Beads with a difference of 0.1 μm or more in their diameters could be distinguished using the method. Both gold and latex beads could be used for the assay. Decoding was done using a flow-cytometer where the beads were sequentially bombarded with a laser light and the unique forward- and light-angle scatters generated by the beads were used to identify them.

Various physical traits can also be combined to achieve higher multiplexing. Modern flow cytometers are equipped in decoding beads based on the differences in their size, refractive index and photoluminescence. Hence, a combination of all these properties can form a unique signature of a bead, which could be a more viable method for encoding high-plex assays. Lawrie et al. [35] used the fluorescence emission spectra, fluorescence intensity, particle size and refractive index to decode beads synthesized by coating silica cores with up to six different dye shells. Each dye shell was separated by a silica spacer shell to reduced RET. The decoding was done using a high-performance flow cytometer with multi-laser beam excitation. The different optical signatures and the bead sizes were achieved by varying the sequence in which the concentric dyes shells were coated and the thickness of the shells.

The main drawback of these methods is that they require customized flow-cytometers. Physical encoding methods such as these are most often used in conjunction with other encoding strategies.

2.1.7 Spatial Bead Encoding

In the spatial bead encoding techniques developed by Ng et al. [3] and Trau et al. [4] bead deposition is done sequentially, and after each batch of beads is deposited an image of the chip is acquired. These images capture the spatial location of the beads in each batch. The bead's identities, their spatial locations in the images, are maintained in a bead table that associates the batch (of the

bead) with its spatial location. After the bioassay, the fluorescence image is captured and the spatial locations of the beads in the fluorescence image are used to look up the bead table and identify them (their batches). Spatial encoding has three main advantages: 1) it does not require any specialized bead fabrication, 2) it does not need any specialized equipment like flow-cytometers to decode the bead identities, and 3) it allows easy customization, where different kinds of beads can be added to the chip during the bead deposition process.

2.2 Summary of Existing Bead Encoding Methods

In the spatial bead encoding technique developed by Ng et al. [3] and Trau et al. [4] bead deposition is done sequentially. After each batch of beads is deposited an image of the chip is acquired. This image captures the spatial location of the beads of each batch. The bead's identities are their spatial locations in the image and a bead table that associates the batch (of the bead) with its spatial locations is maintained. After the bioassay, the fluorescence image is captured and the spatial locations of the beads in the fluorescence image are used to look up the bead table and identify them (their batches). Spatial encoding has three main advantages: 1) it does not require any specialized bead fabrication, 2) it does not need any specialized equipment like flow-cytometers to decode the bead identities, and 3) it allows easy customization, where different kinds of beads can be added to the chip during the bead deposition process.

The advantages and disadvantages of the bead encoding schemes discussed above are summarized in Table 2.1. While chemical and spectrometric encoding schemes offer reasonable multiplexing and reproducibility, they have high bead decoding complexity. Graphical and electronic encoding schemes offer very high multiplexing capacities and good reproducibility but manufacturing the encoded beads is complicated and costly. Spatial encoding offers high multiplexing capacity at low cost. Optical encoding schemes are the most widely used encoding scheme due to the simplicity of their bead decoding mechanism. Spatial bead encoding schemes therefore have the potential to be widely adopted, given that they have similar bead encoding / decoding complexity as optical encoding schemes but offer much higher multiplexing capacity. Note that the multiplexing capacities listed in the table indicate theoretical limits of each type of

encoding scheme. In practice most methods achieve much lower multiplexing. For instance, using graphical encoding it is possible to keep increasing the multiplexing capacity, provided the beads can be made larger. Nicewarner-Pena et al. [10] developed 4160 binary barcodes by varying strip lengths two metals. Theoretically, much greater number of barcodes can be produced using three or more metals metals, even though it is not trivial to do so.

Table 2.1: SUMMARY OF BEAD ENCODING SCHEMES

Encoding Scheme	Multiplexing capacity	Reproducibility and Stability	Cost and complexity of bead fabrication	Cost and complexity of bead encoding/decoding
Optical encoding	Low (10 - 100)	Low	Moderate (requires multiple dye doping steps)	Low
Chemical encoding	Moderate (100 - 1000)	Moderate	Moderate (requires both antigen conjugated micro-particle as well as oligonucleotide conjugated nanoparticle)	Very High (requires nanoparticle micro-particle separation followed by dehybridization to cleave oligonucleotides and spotted microarray assay)
Spectrometric encoding	Moderate (100 - 1000)	Moderate to High	Low	Moderate to High (some methods require specialized mass spectrometers)
Graphical encoding	Theoretically unlimited.	High	High (requires specialized equipment for barcoding the beads)	Low to Moderate (while most barcodes can be resolved using software, some require specialized lighting during imaging)
Electronic encoding	Theoretically unlimited	High	High (requires specialized techniques for embedding electronic devices inside beads)	Moderate
Physical encoding	Low to Moderate (10 - 1000)	Moderate	Low	High (require flow-cytometers for decoding)
Spatial encoding	Theoretically unlimited	High	Low	Low

2.3 Robust Spatial Bead Encoding

We envision two scenarios in which the spatial bead encoding scheme is likely to be used. In the first scenario, the bead chip is fabricated and the bead deposition is performed in a (possibly commercial) manufacturing facility. The chip is then shipped to the laboratory where the bioassay is performed and the bioassay images are sent back to the commercial vendor for analysis. The bioassay images are matched to the deposition images in the vendor's database and the experiment's results are sent back to the laboratory. In the second scenario, the bioassay is performed in the same laboratory where the bead deposition is performed. In this scenario, the necessary pre-conjugated

beads and the chip are purchased separately from any of the many available commercial vendors and the bead deposition and image acquisition are manually performed in the laboratory.

In either scenario it is unreasonable to expect that the individual performing the bioassay will always be able to acquire the fluorescence image such that it is perfectly aligned with the deposition image. It is likely that the fluorescence image will be a transformed version of the deposition image; and that the two images would have to be aligned to decode the bead identities. This idea is illustrated in Fig. 2.1. However, when high-throughput is desired manual alignment of fluorescence and deposition images is infeasible. An automated spatially addressable bead encoding scheme would therefore require a robust image matching method to align the deposition and bioassay images.

Fig. 2.1 The bead encoding problem and the spatial bead encoding scheme.

In the first usage scenario, although the deposition images may be acquired in a well-controlled environment, the bioassay images can be acquired using a large range of microscopes or scanners under varying illumination conditions. In the second usage scenario both the deposition and bioassay images will be acquired in uncontrolled environments. The computational methods developed to process these images should therefore be robust enough to handle large amounts of noise, artifacts (like pieces of left over scaffolding, dirt, clustered beads, etc.) and image distortions (like spherical aberrations, blurring, occlusions, etc.). At the same time a high throughput automated

encoding mechanism can allow minimal manual intervention and therefore should require minimal (if any) parameter optimization.

Figure 2.2 shows the proposed computational pipeline, the sequence in which the six computational methods are used for automated, high throughput analysis. We refer to this pipeline as the *robust spatial bead encoding* pipeline.

To use the spatial bead encoding scheme in an automated high-throughput bead encoding system, some mechanism is necessary to ensure that the fluorescence and deposition images can always be matched. Consider as an example a 100-plex assay with 10 beads per batch. Now, consider the case where only one of the batches is expressed, such that the fluorescence image contains only 10 beads. In such a case, it would be difficult to match the fluorescence image to the final deposition image, which contains 1000 beads. Therefore, some way of ensuring that the fluorescence image has sufficient number of beads, for the affine invariant and grid based matching algorithms to be able to match the images, is necessary. We propose using positive control beads for this. All bioassays require positive control beads and using them as landmarks (for matching) does not impinge on the flexibility of choosing any commercially available chip that best suits the pre-conjugated beads to be used in the assay. In the *robust spatially addressable bead encoding* pipeline, we assume that the first batch of beads deposited on the chip are positive controls. Therefore, the fluorescence images have to be matched only with the first deposition image. In Chapter 7 it is shown that as long sufficient positive control beads are used (ranging between 5% and 20% of the total number of beads, depending on the size of the experiment) the images can be aligned with very high accuracy even under large amounts of occlusions.

An obvious approach towards automating the spatial bead encoding scheme would be to place markers on the chip that can be used to align the fluorescence images. However, this apparently simple approach has several problems. If markers are used to align the images, these markers would have to be distinctive. It is unlikely that a universal marker would be used by all chip manufacturers, therefore specialized software would be needed to match the images for each type of marker. Moreover, designing stable markers that are visible under fluorescence can be

challenging as their surface chemistries can interfere with the assay chemistry. Therefore, we propose using the pattern formed by the positive control beads, an intrinsic property of the array, as the marker. This pattern is also inherently, robust to occlusions as the beads are typically uniformly distributed across the deposition image.

Fig. 2.2. The proposed *robust spatially addressable bead encoding* pipeline

The robust spatial bead encoding pipeline shown in Fig. 2.2 was briefly alluded to in Section 1.3. The pipeline is divided into two branches: 1) an offline processing branch in which (deposition) images acquired prior to the assay are processed and 2) an online processing branch in which the fluorescence images acquired during the assay are processed. The offline and online processing branches can be thought of as bead encoding and bead decoding steps respectively. As mentioned before the pipeline constitutes of six computational sub-tasks: bright and dark field

image gridding, micro-well classification (in bright and dark field images), microsphere segmentation (in fluorescence images), fluorescence image gridding, pattern based correspondence estimation (also referred to as affine invariant point pattern matching), and grid based correspondence estimation (also referred to as grid-based point pattern matching). The proposed computational methods developed for these sub-tasks are logically grouped into three groups: bright (and dark) field image processing, fluorescence image processing, and pattern matching. Bright and dark field image processing is discussed in Chapter 5, fluorescence image processing is discussed in Chapter 6, and pattern matching is discussed in Chapter 7.

In the next chapter we review existing microarray image analysis techniques in detail.

Chapter 3 - Review of Microarray Image Analysis Techniques

Microarray image analysis involves three primary tasks: microarray image gridding, spot segmentation and quantification. Spotted microarrays are fabricated by placing (or synthesizing) thousands of different types of probe molecules at specific locations (spots) on a planar substrate. To pack the most number of spots on the planar surface they are usually placed in either a square or a hexagonal a grid pattern. Gridding refers to the process of determining this underlying grid structure in a microarray image. Gridding partitions the image into small rectangular or hexagonal regions (henceforth referred to as grid patches) such that each grid patch contains one spot. Gridding is also often referred to as spot addressing.

Microarray experiment are mainly used to quantify the concentration of target analytes associated with the spots, which is directly proportional to the intensities of the pixels associated with that spot. Therefore, to perform quantitation, the pixels belonging to the spot (referred to as foreground pixels) have to be identified. This is referred to as spot segmentation. As segmentation directly affects the quantification step, accurate segmentation of the spots is of utmost importance. Many of the segmentation techniques rely on the grid patch's center to initialize the segmentation and quantification process. Therefore, accurate gridding is also critical. In the following sections, we review some of the existing microarray gridding and spot segmentation algorithms.

Quantification involves estimating the relative concentration of fluorophores at the spots based on the foreground and background pixel intensities. Microarray images however can be affected by various types of noise. The imaging apparatus itself introduces some amount of background noise. At the same time, the biochemical reactions that occur on the microarray are not deterministic. Non-specific bindings and cross-reactivity between capture probes and sample also introduces noise. To identify the true signal of a spot, that reflects the actual concentration of the target analyte associated with it, various algorithms have been developed. Image enhancement algorithms for instance aim at reducing the background noise, whereas background subtraction algorithms aim at accurately estimating the noise, and spot filtering algorithms try to identify

contaminated spots so that they can be discarded to maintain the sanctity of the data. However, since quantitation is not one of the main objectives of this work, we will only briefly review these algorithms. In particular, we will review only those image enhancement, background subtraction, and spot filtering techniques that are either closely tied with spot segmentation or use interesting features and classification techniques. Spotted microarray typically contain multiple grids printed as blocks. A block segmentation step that identifies and separates these blocks is usually performed prior to gridding. Bead-based microarray images however do not have such blocks, therefore we do not consider the block segmentation techniques in this review.

Due to the crucial role of image analysis in the microarray technology, they have received much attention over the past two decades. Numerous gridding, spot segmentation, image enhancement, background subtraction, and spot filtering algorithms have been developed. However, even today there is no consensus about the best algorithms. Different laboratories continue to use different standards, and this has led to major problems with reproducibility.

3.1 Microarray Gridding

The most common approach for spotted microarray image gridding is based on axis projections. It uses two 1-d signals 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 assumes that the grid is aligned to the horizontal and vertical axes. The many axis projection based techniques that have been developed, differ in the way they detect the crests or troughs in the axis projections. Li et al. [36] for instance use a sliding window of user-specified width to detect the troughs in the axis projections. Bergmann et al. [37] use the first derivative of the smoothed axis projections to detect the crests. Angulo and Serra [38] propose using morphological operators to separate the peaks (crests) in the axis projections. Bidaut et al. [39] used the fast Fourier transform (FFT) of the autocorrelation of the axis projections to detect the crests. Bajcsy [40] used the output of horizontal and vertical edge detection to construct the two axis-projections. Blekas et al. [41] used an additional local adjustment step to refine an initial axis projection based grid estimate. Each horizontal (and vertical) grid

boundary separating two adjacent grid patches is shifted such that the sum of pixel intensities along the boundary is minimized.

Radon transform [42] (RT), which computes the line integral of pixel intensities along a particular direction, has been used to account for arbitrary rotations of the grid. Brandle et al. [43] use RT in conjunction with axis projections to correct for grid rotations. Even in the presence of small non-linear distortions the RT at the true grid rotation angle is expected to have well-separated troughs that correspond to grid lines. Brandle et al. estimate the grid by simultaneously searching for local maxima in two nearly perpendicular RTs. A pair of maxima in the two RTs corresponds to two lines, one through the spot rows and one through the spot columns and the intersection of two such lines will be at a spot center. A small region centered at such an intersection should contain the spot and therefore have high mean intensity. They developed a greedy algorithm that maximizes the sum of the mean intensities at the intersections to simultaneously search for local maxima in the two RTs. The method requires approximate estimates of spot size and distance between spots as input.

Katzer et al. [44, 45] perform a segmentation of the image prior to gridding. For images where sufficient numbers of spots are segmented as foreground regions the distribution of nearest-neighbor distances will have a significant peak corresponding to the distance between spots. The distribution will also have peaks corresponding to the diagonal distance between spots and integer multiples of the distance between spots. A sum of Gaussians is used to model this distribution, where the mean of each Gaussian is related to the actual spot distance d . For instance, the mean of diagonal spot distances is $\sqrt{2}d$. The sum of Gaussians is heuristically fitted to the distribution to estimate an approximate value of d . Fragments of the grid are then re-constructed by grouping together appropriately spaced foreground regions. These re-constructed grid are further refined using a sequence alignment algorithm that finds equally spaced peaks in the axis projections. Multiple such grid hypotheses are generated for each grid block in the image and a Markov random field (MRF) model is used to find the best hypothesis (label) for each grid block based on the agreement between the various hypotheses.

When very few spots are present in the image, the actual spot distance d cannot be reliably estimated from the distribution of nearest neighbor distances. Such sparsely populated images are particularly common in bead-based microarrays, where only a fraction of all micro-wells are occupied by beads and only a few of these beads are expressed sufficiently enough to be segmented as foreground regions.

Lawrence et al. [46] perform Bayesian inferencing to refine an initial grid hypothesis generated using a semi-automatic axis-projection based gridding technique [47]. The Bayesian inference starts with the assumption of a perfectly spaced grid with oval shaped spots and updates the belief based on the observed image data. A likelihood function, which represents the expectation about how the image should look, given a particular set of grid parameters and spot radius, is computed as:

$$p(I_{GP}|\mathbf{x}, \mathbf{d}) = \prod_{i \in S} p(I_i|S) \prod_{i \in \sim S} p(I_i|\sim S) \quad (3.1)$$

where I_{GP} is an image grid patch, $\mathbf{x}$ is a spot center, $\mathbf{d}$ is a 2-d vector specifying the spot radii, S represents the spot foreground region and $i \in S$ represents that the i^{th} pixel in I_{GP} belongs to the spot foreground region S . The individual pixel's likelihoods $p(I_i|S)$ and $p(I_i|\sim S)$ are computed based on global pixel intensity distributions of foreground and background regions, found through the initial grid estimate. Notice that the pixels in this model are assumed to be independent of each other. The prior probability, is computed assuming the two features, grid centers and spot radius, follow Gaussian distributions and are independent:

$$p(\mathbf{x}, \mathbf{d}|G) = \mathcal{N}(\mathbf{x}|\mathbf{m}_c, \Sigma_c) \mathcal{N}(\mathbf{d}|\mathbf{m}_r, \Sigma_r), \quad (3.2)$$

where, $\mathbf{m}_c$ is the expected grid center as per the initial grid G , $\mathbf{m}_r$ is the expected spot radius specified by the user, Σ_c and Σ_r are covariance matrices of circular Gaussians with diagonal elements σ_c^2 and σ_r^2 respectively, and $\mathcal{N}$ is the Gaussian function. Here σ_c^2 and σ_r^2 are free parameters that are empirically determined and represent the confidence in the prior belief. The posterior probability, the updated belief, is computed using Bayes' theorem:

$$p(\mathbf{x}, \mathbf{d} | G, I_{GP}) = \frac{p(I_{GP} | \mathbf{x}, \mathbf{d}) p(\mathbf{x}, \mathbf{d} | G)}{p(I_{GP})}. \quad (3.3)$$

Finally, quantification is done by computing the expression level as the expectation of the grid patch's intensity under the posterior distribution. In practice the normalization term in Bayes' rule is intractable and an approximation of the expectation under the true posterior is computed using a technique referred to as variational importance sampler.

Rueda and Rezaeian [48] developed an axis projection based gridding, referred to as optimal multilevel thresholding gridding (OMTG). They perform optimal multilevel thresholding (OMT) of the axis projections to detect the troughs corresponding to the grid lines. An exhaustive search for the optimal number of spots in a row or column (say k) is performed based on an α score metric. The α score for a given k is computed by combining Fisher's linear discriminant criteria and a heuristic that penalizes large axis projection values at the k detected troughs. To correct for rotation of the grid, the OMT based grid detection is iterated several times and RT is used to get the axis-projection in each iteration. The angle for which the OMT based grid detection has the minimum entropy is considered as the true rotation angle of the grid. A grid refinement similar to that of Blekas et al. [41] is also performed, where each horizontal (and vertical) boundary separating two adjacent grid patches is shifted to minimize the sum of pixel intensities along the boundaries.

Bariamis et al. [49] developed a max-margin classifier based gridding algorithm. They used RT to first correct for grid rotation. At the correct rotation angle the spots will be perfectly aligned and the peaks in the intensity projections will be large. Consequently, the sum of the projections at this angle will be high. Therefore, the optimal rotation angle is determined as the angle at which the sum of RT is maximum. The optimal row and column widths are estimated by iteratively searching through a range of possible values. For each value, the image is broken into a set of row (and column) images. At the true value of row width, the row images will have most of the spots well-aligned inside them and therefore the average of these row images will have many bright and many dark pixels. Therefore, the value at which the average row (or column) image has the highest variance is identified as the optimal row (or column) width. These estimates are used to generate

an initial grid which is refined by identifying the best separating line (max-margin line) between each pair of adjacent rows (and columns). To estimate the max-margin line between two adjacent rows of spots, say rows k and $k + 1$, a soft-margin support vector machine (SVM) classifier is trained using the 2-d spatial coordinates of the spots as features. All spots on or above row k are labelled positive samples and all spots on or below row $k + 1$ are labelled negative samples. The classification boundary learnt by the SVM classifier gives the max-margin grid line.

Charalambous et al. [50] also uses RT in a similar way to that of Bariamis et al. [49] to estimate the rotation angle. The axis projections at the optimal rotation angle is thresholded to get an estimate of the spot size. Normalized cross-correlation (NCC) between the microarray image and a circular template of the estimated spot size is used to detect spot centers. Local maxima in the NCC values within rectangles of size similar to that of the circles are considered as spot centers. A 2-d potential field is then constructed, where the energy at the spot centers is the highest and falls off as one goes further away from the spot center. The potential at each spot center follows a rectangular pyramid like structure. Initial estimates of the row and column grid lines are computed as the local minima at the leftmost regions and the top most regions of the energy field. These initial grid lines (fronts) are then allowed to move through the energy field while the energy field induces repulsive forces onto the fronts and keeps them away from the spot centers. The path of the fronts through the energy field produces the grid lines.

Ceccarelli and Antoniol [51] used orientation matching transform, an extension of Hough transform [52] to detect spots. They parameterize the grid using a 6-tuple: the 2-d location vector of the top-left spot, the horizontal and vertical gaps between spots, and the rotation angles of the two axes. An ideal grid is transformed and fitted to the detected spots in the image by searching through the 6-d grid space using an evolutionary algorithm (EA). A MRF based model with a 1-clique MRF prior is then used to refine the fitted grid G . The 1-clique prior models the i^{th} grid location as a Gaussian distributed with mean $\mathbf{g}_i$, where $\mathbf{g}_i$ is the spatial coordinates of the i^{th} grid point. The likelihood function encodes the probability of observing the orientation matching transform's output, given G . The maximum a posteriori (MAP) estimate of the refined grid is

computed using a simulated annealing based search, through the n -d search space, where n is the number of spots).

Morris [53] used a 7-tuple to describe a grid: the 2-d location vector of the top-left spot, the rotation angles of the x-axis, the number of rows and columns and the height and width of the image. He also used an evolutionary algorithm to search for a grid that best fits a detected set of spots. The sum of Euclidean distances between the detected spots and their nearest grid points is used as the fitness score. Such an objective function is however biased towards denser grids, for instance if each pixel is considered as a grid point then the value of the objective function will be close to zero. To overcome this problem a penalty function that penalizes over-gridding is introduced. Over-gridding is a problem common to most non axis-projection based methods but it is rarely addressed. To detect potential spots in the image a correlation based approach is used, similar to that of [50]. However unlike most other techniques, the template used to calculate the correlation is not a predefined mask. Rather a representative binary spot mask that is found to correlate well with at least 60% of the remaining binary spot masks is used as the template. To find such a template the image is thresholded multiple times and the correlations between the binary spots are evaluated.

Bidaut et al. [39] used the fast Fourier transform (FFT) of the autocorrelation of the axis projections to detect the spacing between the grid lines. The autocorrelation is the correlation of the axis-projection with itself, computed by shifting one instance of it over the other. It reduces the effect of noise and reveals the smooth underlying periodic function. The maxima in the FFT of the autocorrelation signal correspond to the underlying fundamental frequencies that the signal is composed of. When the spots are well aligned the autocorrelation signal can be almost entirely explained by the periodic signal corresponding to the regular grid, and so the FFT of the autocorrelation has a significant maxima. The wavelength corresponding to such a significant maxima gives an estimate of the spacing between spots. To detect the actual grid rotation Bidaut et al. [39] search through a series of rotated images and identify the angle at which the amplitude of the significant maxima in the FFT is the highest. To detect potential spots in the image a wavelet

decomposition of the image is performed, which produces two images: a horizontal detail image H and a vertical detail image V . Based on user-defined spot size, templates matching is used to find spot centers in the H and V images. A dynamic programming based algorithm is then used to independently detect the row and column grid lines that best fit the detected spots.

Rueda and Vidyadharan [54] use three different masks, a uniform function, an exponential function, and a circular Gaussian function, as templates to detect spots in the image. The algorithm assumes the grid to be well-aligned with the horizontal and vertical axes. Templates of user defined sizes are centered on each image pixel and the correlation with the image is computed. The templates are shifted by 1 pixel horizontally or vertically in a sliding window manner, till a maxima of the correlation is reached. This process of detecting a local maxima is referred to as hill climbing. Multiple passes of the hill climbing algorithm, starting from different initial locations, are used to detect the approximate spot centers in the image. The projection of the density of these approximate spot centers, along the horizontal and vertical axes, are computed and hill-climbing is again used on these 1-d projections to detect the grid lines.

Zacharia and Maroulis [55, 56] developed a genetic algorithm (GA) for detecting the grid lines. The search space of the horizontal (and vertical) lines is parameterized using a 3-tuple: the start and end (y and x) coordinates of the topmost (and leftmost) line, and the gap d between subsequent lines. Two independent searches are performed to find the optimal set of horizontal and vertical lines. Each chromosome in the GA encodes the real-valued 3-tuple, representing one possible configuration of horizontal (or vertical) lines. The fitness function is evaluated based on how much of the lines pass through background, low intensity, regions of the image. The image is thresholded using empirically determined thresholds to get foreground and background regions. For each line, a rectangular region R obtained by dilating the line is found, and the percentage of pixels in R belonging to the background region of the image is considered as the fitness score for the line. The fitness of a chromosome is determined as the sum of the fitness of the lines it spawned. To ensure the entire grid is covered with lines an additional term, that rewards solutions with higher number of lines, i.e. smaller d , is added in the fitness function as well. The algorithm is naturally

invariant to grid rotations, however, it cannot deal with local non-linearity of the grid.

Ho et al. [57] formulate the gridding problem as a point pattern matching problem. They assume that a reference grid is available from the manufacturer and therefore the number of rows and columns and the distance between adjacent spots is known. They formulate the gridding problem as one of matching the reference grid to a set of putative spots detected by thresholding the image. Their objective function searches through the space of similarity transforms and constitutes of two terms: the first minimizes the mean squared Euclidean distance between matched pairs of putative spot centers and reference grid points, and the second term imposes a smoothness constraint that penalizes large differences in transform estimates between adjacent grid points. Geometric hashing [58, 59] is used to determine initial matches between the reference grid points and putative spot centers. The objective function is optimized, by equating the partial derivatives of the function, with respect to the transformation parameters, to zero. The resulting set of equations is solved using a Jacobi iteration scheme to determine the optimal transformation. To reduce the dependency on the threshold used to binarize the image and detect putative spots, the authors propose performing multiple passes of thresholding and matching.

3.1.1 Microarray Gridding Techniques for Hexagonal Grids

The algorithms discussed thus far are designed for square grids, which are more common in spotted microarrays. Galinsky [60] developed one of the first techniques for gridding fluorescence images of spotted microarrays with hexagonal grids. The three step process begins by segmenting the expressed spots using an adaptive template matching method. In the second step the detected spots are used to estimate the grid parameters, the distance between adjacent grid points and grid rotation. These grid parameters are used to interpolate the positions of the missing spots. Finally, in the third step, an ideal hexagonal grid (with unit spacing and zero rotation) is affine transformed and aligned to the detected and interpolated grid points. The grid parameter estimation in the second step relies on identifying pairs of adjacent spots. Grid point pairs that have the smallest nearest neighbor distances among all nearest neighbor grid point pairs are identified as adjacent spots. The algorithm will only succeed if at least a few adjacent spots are expressed. Galinsky

showed that the probability p_s of having at least one pair of adjacent grid spots expressed, given that the spots are uniformly distributed, can be computed as:

$$p_s = ((1 - R_s)^6)^N \quad (3.4)$$

where $R_s = N/M$ is the expression rate and N and M are the number of expressed spots and the total number of grid points respectively. Galinsky showed that the algorithm succeeds for expression rates as low as 1%. However, in bead based microarrays the expression rates can be much lower, which makes this algorithm unsuitable. The algorithm is further discussed in the next section.

Giannakeas et al. [61] developed a similar hexagonal grid detection algorithm, which also begins by segmenting the expressed spots. They estimate the distance between adjacent grid points as the mode of the distance between nearest neighbor grid points and assume the image has no rotation. Missing grid points are interpolated from the positions of the detected grid points using a method they refer to as concentric hexagon growing, but which is essentially equivalent to the interpolation performed in [60]. Using the mode of nearest neighbor distances as an estimate for the grid distance is even less robust than the grid distance estimation of [60]. Such a method is likely to fail even for moderate expression rates.

3.2 Spot Segmentation

One of the first microarray spot analysis algorithms was developed by Chen et al. [62]. A circular mask of user specified radius, large enough to enclose the entire spot foreground, is placed at each grid patch center. A Mann-Whitney test [63] is then repeatedly performed to test for the null hypothesis that samples (pixel intensities) drawn from within the mask and from outside the mask belong to the same distribution. When a pair of samples is found for which the null hypothesis can be rejected, the minimum pixel intensity among the sample drawn from within the mask is used as a threshold to segment the grid patch pixels into foreground from the background pixels.

A popular microarray image analysis software ScanAlyze developed by Eisen in 1999 [47]

simply places fixed size circles of a user specified radius at the center of the grid patches and assumed all pixels inside the circular region as foreground pixels. Another popular software Genepix software (Axon Instruments, Inc., Union City, CA) implemented a proprietary adaptive circle fitting technique which adjusted the size of the circles to better fit the spots. However, a spot's morphology is not always a circle and therefore fixed and adaptive circle based approaches can lead to sub-optimal results.

Rueda and Qin [64] developed an adaptive ellipse based method to account for non-circular spots. They consider each grid patch as a 2-d Gaussian function with a full covariance matrix. The sample mean vector and sample covariance matrix of the density function are estimated using the pixel intensities. A diagonalization and whitening of the density function is performed to transform the grid patch such that the spot becomes circular in the transformed image. The corrected (transformed) density function is then used to segment the circular spot using the Mann-Whitney test based segmentation approach of Chen et al. [62].

Buhler et al. [65] proposed a convolution based approach to detect spots in the image. They used edge images of circles (hollow circles) with varying radius as filters to convolve the Laplacian of the microarray image. The filter producing the highest response is considered as the matched filter and the location of the maxima of its response in the grid patch is considered as the spot center. Each grid patch is processed independently and therefore spots of varying sized can be detected. Further, they developed a supervised classification scheme for detecting low quality spots. The scheme requires an operator to generate training data for the classifier by manually labelling spots in the image. The relative brightness of the spot's foreground compared to the background, and the offset of the spot's center from the grid patch's center are used as features for the classification. The spots are grouped into three classes: accept, reject, and show. Linear, axis parallel, classification boundaries are learnt by simply estimating the threshold values for the two features (independently) that optimally separates the training data into the three groups based on a user specified loss function.

Wang et al. [66] developed an iterative spot segmentation algorithm. The algorithm is also

initialized using circular masks of a user specified radius, large enough to enclose the entire spot foreground, placed at the grid patch centers. Pixels inside the grid patch but outside the circle are considered local background pixels. Pixels in the grid patch having intensities greater than $\mu_{bg} + 2\sigma_{bg}$, where μ_{bg} and σ_{bg} are the mean and standard deviation of the intensities of the local background pixels, are considered as foreground pixels. The circle is then adjusted to enclose the new set of foreground pixels and the process is repeated. The algorithm will not work when there is significant overlap between the foreground and background intensity distributions, which is often the case for weakly expressed spots and for spots affected by bleaching of neighboring spots.

Yang et al. [67] used seeded region growing to segment spots in microarray images. Small square patches (2×2 pixel wide) centered on the maximum intensity pixel, within the grid patches are used as seeds for the foreground regions. Whereas the background regions are grown using the points of intersection of the grid lines as seeds. The algorithm assumes that the pixel with the highest intensity (inside a grid patch) belongs to the spot region, which may not be the case in the presence of noise.

Bozinov and Rahnenfuhrer [68] used partition around medoids (PAM) [69] and k-means clustering [70] to group each grid patch's pixel intensities into two clusters, representing the foreground and background. They performed a two-channel microarray experiment where the gene expression levels between two heterogeneous messenger ribonucleic acid (mRNA) samples (for instance, a sample from disease tissue and another sample from healthy tissue) are compared. The two samples were labelled with red and green dye, and two images were captured one for the red fluorophore and one for the green fluorophore. Since both the images contained the same spots, 2-d pixel intensity vectors (containing the intensities of a pixel in the red image and the green image) is used to describe each pixel. The 2-d pixel intensity vectors in a grid patch are clustered to produce foreground and background segments. Nagarajan [71] used the same technique to cluster 1-d pixel intensities of a one-channel experiment and showed that more differentially expressed genes could be identified using this method than using the seeded region growing method of Yang et al. [67], although at the expense of more false positives. A major drawback of these methods is that, weakly

expressed spots, whose intensities are close to that of the background intensity, cannot be detected. Also, the clustering can produce multiple disjoint regions as foreground. These algorithms assume the largest of these regions to be the spot region, an assumption that often does not hold true.

Angulo and Serra [38] used morphological filtering and watershed segmentation to detect spots in microarray images. They estimate the mean area of the spots based on the distribution of the number of particles that vanish by applying morphological opening with structuring elements of different sizes. The estimate of the mean area is used to filter out noise particles from the image through morphological opening. Morphological levelling is then applied to the filtered image and the difference between the levelled image and a dilation of the levelled image is used to construct a gradient function. Finally, the watershed algorithm is used to segment the gradient function and identify the spot foregrounds. A visual inspection of the results shows that the method produces oddly shaped spots, particularly for low signal spots, and therefore results in sub-optimal segmentation.

Li et al. [36] modelled the 1-d pixel intensity distribution (of a grid patch) as a Gaussian mixture model (GMM). The maximum likelihood (ML) estimates of the parameters of the GMM was found using the expectation maximization (EM) algorithm. The number of components in the GMM (k) can be one, two, or three depending on whether the grid patch is empty, has a spot, or has a spot and an artifact, respectively. The optimal number of components (k) in the GMM is determined using the Bayesian information criterion (BIC). This is based on the assumption that the intensity distribution for grid patches with no discernable spots present would have only one Gaussian component corresponding to the background. Similarly, when an artifact such as noise or scratch is present along with a spot, three Gaussian components, corresponding to the background, spot foreground, and artifact pixels, would be present. In reality such a strong assumption that the spot foreground pixels and artifact pixels come from two different Gaussian distributions are rarely true.

Katzer et al. [44, 45] developed an active contour model for segmenting spots, in which the contour of the spots are modeled as a chain of tension springs connected by torsion springs. Bending

is allowed only at the torsion springs and stretching is allowed only at the tension springs. The energy function for the model has three components: the first component models the stretching energy, the second component models the bending energy and the third component models the external energy acting on the spline. The external energy at a point on the contour is computed as the inner product of the image gradient and the contour's normal at that point. As the energy function has multiple local minima a simple gradient descent is not sufficient to find a good minima of the energy function. Therefore, a smoothed gradient descent technique was developed for the optimization. A problem with using active contours is that a segmentation may be achieved even when a spot is not present at the location. To overcome this problem a SVM based classification scheme was developed that uses the change in external energy, the agreement between contour normals, and image gradients as features to identify true spots.

Chan and Vese developed an active contour based segmentation [72] technique that segments foreground object regions by minimizing the variation in pixel intensities inside and outside the object region. Unlike other active contour based segmentation methods, this technique does not rely on the object having a sharp gradient (a distinct edge) at the boundary. This is desirable in microarray image segmentation where highly expressed spots can have a halo of smooth gradient around them while lowly expressed spots can have boundaries that are not easily discernible. Mendhurwar et al. [73] used Chan and Vese active countour (CVAC) for spotted microarray image segmentation with promising results.

Elguebaly and Bouguila [74] used a generalized Gaussian Mixture Model to cluster the 1-d pixel intensities into foreground and background clusters. A generalized Gaussian distribution is parameterized using mean μ , an inverse scale parameter α , and a shape parameter β . The shape parameter β gives generalized Gaussian distributions additional flexibility to more accurately model distributions with significant plateaus or sharp peaks. A Bayesian Markov Chain Monte Carlo (MCMC) simulation method was used to get the posterior distribution of the parameters of the generalized Gaussian mixture model. Visual comparison of the segmentation achieved, on a small set of spots, between the proposed method and k-means clustering, showed better

segmentation was achieved by the proposed method, particularly for low-signal spots and in the presence of small noise particles.

A major limitation of pixel intensity clustering based segmentation approaches is that they ignore the spatial relationship between adjacent pixels. Consider a spot image I and another image I^* generated by randomly shuffling the pixels of I . GMM and k-means clustering of pixel intensities will not distinguish between I and I^* . Nagarajan and Upreti [75] developed a segmentation technique that incorporates the spatial relationship between pixels, by considering the correlation between pixels of adjacent rows and columns. The algorithm proceeds as follows: a correlation statistic is calculated for all pairs of adjacent rows and columns within the grid patch region. Student's t-distribution is then used to model the distribution of correlation statistics and rows and columns with p -value greater than a significance level α are identified as significant rows and columns. Finally, pixels that belong to both a significant row and a significant column are identified as foreground pixels. Grid patches in which no foreground pixels can be found are considered to be blank. A comparison with k-means, PAM, and seeded region growing, based segmentation methods showed that the correlation based method produced much fewer false positives.

Demirkaya et al. [76] developed a Markov random field (MRF) based model for segmenting grid patches. The conditional probabilities of the pixel's intensities given their labels (i.e. foreground or background) are estimated from the distribution of foreground and background pixels obtained through a binary segmentation of the patch. The author's argue that an exponential model better fits the pixel intensity distributions for both labels, compared to Gaussian distributions which have been widely used. In their MRF model each pixel is connected to its 8-nearest neighbor pixels so that the prior probability (MRF prior) of a pixel taking a particular label is influenced by its 8-neighboring pixels. The MRF prior essentially incorporates a smoothing constraint such that neighboring pixels are more likely to have similar labels and acts as a regularization term that penalizes assigning different labels to neighboring pixels. The posterior probabilities, of labels given the pixel, are computed using Bayes's theorem given the MRF prior and the conditional probabilities. An approximate maximum a posteriori (MAP) estimate of the posterior probabilities

is computed using the iterative conditional modes (ICM) algorithm [77]. The ICM algorithm re-computes the MRF priors in each iteration based on the MRF priors of the previous iteration, essentially allowing the pixel label information to propagate through the network, and allowing the network to reach a steady state, corresponding to some local minima of the MAP estimate.

Blekas et al. [41] also used the MRF model to incorporate pixel's spatial adjacency information in a GMM clustering based segmentation technique. The MRF model is used to compute the prior probabilities in a similar way to that of Demirkaya et al. [76]. The prior probabilities are used as additional parameters (replacing the mixing weights) within the GMM framework. The MAP estimates of the parameters of the GMM are found using the expectation maximization (EM) algorithm. Similar to Li et al. [36], the authors set the number of components in the GMM (k) as one, two, or three depending on whether the grid patch is empty, has a spot, or has a spot and an artifact, respectively. However, unlike Li et al. [36], the optimal number of components is found using cross-validated likelihood.

Baek et al. [78] propose using a gamma-t mixture model to cluster the 2-d pixel intensity distributions of a two-channel experiment. They show through empirical analysis that the proposed model better fits the bivariate pixel intensity (comprising of red and green channel intensities) distribution, compared to GMMs. The gamma distribution, because of its greater shape versatility, can better model the wide range of background intensity distributions. Whereas, the t-distribution because of its longer tail is less prone to outliers and therefore can model the foreground intensity distributions better. They developed an EM algorithm to obtain the ML estimates of the parameters of the mixture model. As discussed above such intensity based clustering algorithms ignore the spatial relationship between adjacent pixels, to overcome this problem a non-parametric Kernel smoothing of the pixel labels was performed in a subsequent step. The BIC was used to identify blank spots, similar to Li et al. [36]. However, unlike in the case of Li et al. [36], detecting artifacts in this method is not as straightforward, as only up to two components can be present in the mixture model. Therefore, an additional outlier detection step based on interquartile range is proposed to identify grid patches with artifacts.

To circumvent the strong assumptions about the shape of the distributions made by the parametric techniques like GMM and the gamma-t mixture model Chen et al. [79] used a non-parametric kernel density estimation (KDE) approach to cluster the 1-d pixel intensity distributions. The KDE based smoothing of the empirical distribution was performed using a Gaussian kernel with automatic bandwidth selection. The threshold separating the foreground and background pixels was identified as the first local minima found in the smoothed density function. They compared the proposed method with GMM based clustering, and the circle fitting based approach, and showed small improvements in correlation between replicate arrays in eight microarray experiments.

Ho and Hwang [80] used the Snake-Fisher model that combines level set based contour detection with region based segmentation for microarray image segmentation. They formulate the segmentation objective as an energy minimization problem with two components. The first component is the active contour energy function, which balances the external energy that pulls it towards the image contour and an internal energy that resists deformation. The second component encodes how well separated the foreground and background region's intensity distributions are, and is computed using Fisher's linear discriminant function, which minimizes the within class distance and maximize the between class separation. Snake based edge detection however, requires good initialization to reach a global or even a good local minima. Ho and Hwang used the MCMC based climber algorithm to approximately determine local maxima of the gradient, which corresponds to the edges. Several climbers are initialized where each climber can move a unit step in the gradient direction and randomly in the tangent direction in every iteration. A simulated annealing scheme, where a temperature associated with the climber decreases after each step, is used for optimization. The climber converges when the temperature reaches zero. Once all the climbers have converged, their locations are used to define the initial contours. From the initial contour estimate, the objective function (combining the snake and region energy functions) is minimized using an iterative approach. A t-test which tests the null hypothesis that both the background and foreground means belong to the same distribution is used to test for convergence. The iterations stop when the t-test

value increases (instead of decreasing) between two subsequent iterations.

Athanasiadis et al. [81] used a stationary wavelet transform to estimate a gradient magnitude image and a de-noised version of the original image. A MRF based segmentation, where both images were used to compute the prior and conditional probabilities of the model, was used to segment the image. K-means clustering was used to generate an initial rough clustering. The foreground and background pixels estimated from the k-means clustering were used to compute the prior and conditional probabilities of the model. The joint prior and conditional probabilities (combining the de-noised and magnitude images) were computed as weighted sum of the prior and conditional probabilities estimated from each image. The optimal weight was empirically found and the EM algorithm was used to obtain the MAP estimates. They showed that the proposed method produced better correlation in replicate experiments than Fuzzy c-means clustering and standard MRF based segmentation.

3.2.1 Comparisons of Segmentation Techniques

A number of studies have also been published that compare some of the more popular spot segmentation techniques available at the time. Yang et al. [67] compared four different spot segmentation algorithms: fixed circle based segmentation implemented in ScanAlyze [47], adaptive circle based segmentation of Genepix software (Axon Instruments, Inc., Union City, CA) , the seeded region growing based segmentation algorithm they proposed, and Chen et al.'s [62] algorithm, using various parameter settings and showed that seeded region growing outperformed the other techniques.

Most of the studies we have reviewed thus far focus on improving one or two steps in the microarray image analysis pipeline. Microarray image analysis however involves a large number of steps and different approaches used at different stages in the pipeline can lead to very different results in downstream processes. For instance, many of the segmentation approaches reviewed in the previous section do not consider how different types of smoothing (mean filtering or median filtering) can affect the segmentation. Glasbey and Ghazal [82] attempted to address this issue by studying combinatorial combinations of three segmentation techniques, two smoothing techniques,

a square root data transformation and a background subtraction. They analyzed 4464 combinations (including different parameters of each technique) to find the combination that produces the most consistent quantification, among 3 replicates of thousands of genes, across 9 experiments. The highest reproducibility was achieved when the image is smoothed using a 21×21 top-hat filter and the mean of the largest 20% of pixel values in the grid patch is used as the foreground. Square root transformation of the pixel values and background subtraction using the mean of the smallest 70% of pixel values as background intensity was also performed.

3.3 Spot Filtering

Tran et al. [83] propose using the correlation between mean and median intensities of segmented foreground pixels as a quality measure for the spots. They show that this simple method is powerful enough to detect distorted and oddly shaped spots and suggest using only those spots that have greater than 85% correlation for further downstream analysis.

Wang et al. [66] developed a composite quality scoring technique for each spot. This score combines multiple scores based on: (1) agreement between the area of a spot and the average area of all spots, (2) signal-to-noise ratio, (3) variability in background pixels within the grid patch, (4) difference between the local background of the spot and global background and (5) percentage of saturated foreground pixels. They showed that the variability in the quantification is closely correlated to the composite quality score. In particular, they show that the variability drops exponentially with increasing composite quality scores.

Lan et al. [84] developed a technique for classifying segmented spots, as good and bad, based on simple features such as area, eccentricity, solidity. They compared the classification accuracy of a multi-layer perceptron (MLP), a k -nearest neighbor (k -NN) classifier, and a generalized linear model (GLM) and showed that the MLP classifier was able to achieve error rates as low as 0.25%. Such a classification system would need to be trained whenever the image data changes. The additional operator effort required in training the system may limit the applicability of the technique.

Hautaniemi et al. [85] developed a Bayesian network based spot quality estimation algorithm that uses intensity, size, shape and alignment of the spots as features to classify them. A 2-d Gaussian function is fitted to each grid patch, such that the fitted Gaussian function is:

$$F(r) = AG(\boldsymbol{\mu}_c, \Sigma, r) + B, \quad (3.5)$$

where $\mathcal{G}$ is a 2-d Gaussian function centered at $\boldsymbol{\mu}_c$, r is the Euclidean distance from $\boldsymbol{\mu}_c$ and Σ is the covariance matrix. Seven features are extracted using the fitted Gaussian function: (1) the spot intensity A , (2) the background intensity B , (3) the alignment error, measured as the Euclidean distance between $\boldsymbol{\mu}_c$ and the center of the grid patch, (4) spot's roundness, computed as the ratio σ_1/σ_2 , where σ_1 and σ_2 are the diagonal elements of Σ , (5) size of the spot, computed as $\sigma_1 * \sigma_2$, (6) background noise, computed as the root mean square of the fitted Gaussian function, and (7) bleaching if any, computed as the percentage of outlier pixels based on the fitted Gaussian function. The features were extracted for each spot in both channels of a two-channel experiment, producing 14 features for each spot. Human labelled data was used to learn the graphical model using a cross-validation based structure learning technique. They show high levels of classification agreement between the graphical model and human experts. Although a small dataset of only 155 spots extracted from a single image was used.

Bicego et al. [86] used the same dataset and the same features to train a SVM classifier. They achieved a leave one out cross validation accuracy of 97.4%. The same dataset and features were further investigated by Nanni et al. [87], who argue that the correlations between the features should be accounted for to improve classification accuracy. They use a random subspace ensemble of unstable classifiers, which are known to perform well when the features are correlated, and achieve 99.7% leave one out cross validation accuracy.

Sauer et al. [88] compared three methods for scoring spot qualities: Wang et al.'s [66] composite quality score, Tran et al.'s [83] mean-median correlation, and the coefficient of variation of spots (CV_{spot}), computed as the ratio of standard deviation to mean of foreground pixels. The reproducibility and efficiency of the three methods were evaluated using gene expression levels in

four replicate arrays. The correlation between the expression levels of the genes calculated from pairwise comparisons of the four arrays show that filtering out bad spots increases the correlation for all three methods. The thresholds used for the three methods are computed to maximize the area under the receiver operator characteristic (ROC) curve. Results achieved using composite quality score were similar to those achieved using mean-median correlation and CV_{spot} , but at a much higher computational cost. Mean-median correlation and CV_{spot} performed better than composite quality score for lowly expressed spots. While it is common to reject such spots, the authors point out that these spots contain important information and are essential for detecting rarely expressed genes.

3.4 Image Enhancement

Ekstrom et al. [89] studied three data transformation approaches for enhancing the image signal: logarithmic, Box-Cox, and inverse hyperbolic sine transformations. The transformations are evaluated on 16 images, acquired across four different photometric gain levels. A subjective visual inspection of the transformed images shows that the logarithmic and Box-Cox transforms lead to similar results and provide better signal amplification than the inverse hyperbolic sine transformation. A more objective evaluation is performed on 100 manually selected spots by fitting four 2-d parametric functions to them: (1) a cylindrical function, (2) a Gaussian function, (3) a difference of two Gaussian functions, for modelling donut shaped spots, and (4) a polynomial-hyperbolic function. The functions were fitted to the grid patch using a maximum likelihood approach. The log-likelihood values achieved while fitting, for all combinations of the three transformations and the four functions were compared. The Box-Cox transformation provided the best enhancement when modelling the spots using the polynomial-hyperbolic function.

3.5 Direct Quantification

Steinfath et al. [90] bypass the segmentation step altogether and directly perform quantification. They estimate the relative concentrations at spots by fitting 2-d Gaussian functions using (1) to the grid patches. However, they consider only circular 2-d density functions such that

Σ is a diagonal covariance matrix with both diagonal elements equal to the σ^2 . The spot is assumed to be centered at the center of grid patch i.e., parameter μ_c is not estimated but assumed to be the grid patch's center. While the parameter σ^2 is the empirical variance of the pixel intensity distribution. A and B are estimated using a least squares fitting approach that minimizes the error between the observed pixel intensities in the grid patch and the fitted surface. As in [85], A corresponds to the actual intensity of the spot and B corresponds to background noise. The relative values of A for the spots can therefore be used to directly quantify the relative concentrations at the spots. A major drawback of this method is that it assumes that the grid patches follow a bivariate symmetric normal distribution. This is often not true, for instance, when the spots have elliptical or donut shapes, or when there is a scratch or large dust particle inside the grid patch.

Brandle et al. [43] also perform a similar 2-d Gaussian surface fitting for quantification, but using a full covariance matrix that allows better fitting of elliptical shaped spots. They use an M-estimator based parameter estimation is implemented as an iterative re-weighted least squares fitting problem, using Tukey's bi-weight function to reduce the influence of outliers.

Ekstrom et al. [89] also perform direct quantification using the four fitted parametric models. Interestingly, they show that the simple cylindrical model along with the inverse hyperbolic sine transformation of the image produces superior results than the widely used Gaussian and difference of Gaussian models.

3.6 Background Estimation

Most microarray quantification approaches suggest subtracting the local background intensity from the foreground intensity before computing relative concentrations at spots. Proper estimation of the background intensity is therefore critical for accurate quantification. Varying illumination can introduce systematic errors in the estimates of spot concentrations. Consequently, a number of techniques have been developed for background estimation. Here some of these techniques are briefly introduced.

Yang et al. [67] used morphological opening, with a structuring element large enough to

remove the spots from the image, to generate a background image. Kooperberg et al. [91] take a Bayesian approach to background estimation and developed an empirical Bayes model for estimating the true signal. The foreground and background pixel intensities X_f and X_b are modelled as:

$$X_b \sim N(\mu_b, \sigma_b^2), \quad (3.6)$$

$$X_f \sim N(\mu_t + \mu_b, \sigma_f^2), \quad (3.7)$$

where μ_b , and σ_b are the mean and standard deviation of the background pixels, σ_f is the standard deviation of the foreground pixels and μ_t is the mean of the true signal. They develop a closed form solution for estimating the parameters of the posterior distribution of $\mu_t, p(\mu_t | \sigma_f, \sigma_b, X_f, X_b)$.

Brandle et al. [43] proposed an iterative technique that alternates between a background estimation step and a fitting step. In the estimation step, the original image and the fitted spot surface image are rescaled to lower resolutions till the spots in the fitted surface image merge and are indistinguishable. The difference between the rescaled original image and the rescaled spot surface image is then rescaled back to the original resolution to get the background image. In the fitting step, a parametric surface is then fitted to the background subtracted original image. They show that both the estimate of background and the estimate of spots can be improved using an iterative re-estimation approach.

Daskalakis et al. [92] propose that the observed microarray image is the result of two types of degradation of the underlying signal: (1) a degradation of the image caused by cell-population effects and the image acquisition process, and (2) a degradation caused by biological errors and measurement errors. They propose that the first degradation can be modelled as a convolution of the actual underlying signal, while the second degradation can be modelled as additive noise. An initial binary segmentation of the grid patch is performed using Fuzzy C-means clustering to get a rough estimate of the standard deviation of the background distribution σ_b . The Weiner restoration algorithm, which uses the background variance σ_b^2 , is then used to get an estimate of the true

underlying signal.

3.7 Summary of Microarray Image Analysis Techniques

Axis projection and Radon transform (RT) are probably the most widely used techniques in microarray gridding algorithms. Axis projections are used in almost all gridding algorithms, either to get the final grid or to get an approximate estimate of it in an initial or an intermediate step. Axis projection requires a rotation corrected image where the grid rows and columns are aligned with the horizontal and vertical axes. Therefore, RT is used by most methods for rotation correction before using axis projections. Many approaches, such as hill climbing based gridding [54] and optimal multi-level threshold gridding [48], have been developed for robust mode detection in the axis projections. However, axis projection based methods are inherently prone to failure when significant distortions are present in the image. Axis projection based methods also cannot deal with non-linear deformation of the grid. The evolutionary algorithm based approaches ([51, 55, 56]) that search the grid parameter space and the max-margin gridding [49] are among the more robust approaches, that can deal with significant image distortions. However, these algorithms also cannot handle non-linear deformations. To handle non-linear deformation of the grid Markov random fields have been used ([44], [45]) that smoothly deform a fitted grid, estimated using evolutionary algorithms. These methods have shown good results in spotted microarray images. Although, they are computationally expensive. Gridding bead-based microarray images presents several challenges that are not encountered in spotted microarray images. These make it difficult to directly apply the existing gridding methods to bead-based microarray images. These challenges and how they are addressed are further discussed in Chapters 5 and 6.

As discussed above the only two gridding methods developed for hexagonal grids, to the best of our knowledge, are [60] and [61]. In both these algorithms the success of gridding depends on correctly identifying adjacent grid spots. Even though the method in [60] is much more robust than the method proposed in [61] it is still not suitable for bead based microarray images. In bead based microarrays, particularly in protein quantification assays, it is often the case that no pair of adjacent spots are simultaneously expressed. Consider the case where the grid has 40000 micro-

wells and a 5-plex assay with 50 beads per batch is being performed. Of the 5 batches if only two batches are well expressed, then the fluorescence image will have 100 beads. Using (3.4) the probability of not having any adjacent spots expressed can be computed to be as high as 0.3. From (3.4) we also see that the probability of failure is high when the grid is small i.e. when the number of grid points in the image (M) is small. Such a case can occur when only a small section of the chip is imaged to capture the beads at high-resolution. See Fig. 6.9(d) for an example of this situation. Further, the method proposed in [60] requires a good estimate of the number of adjacent grid point pairs needed for computing the grid parameters. This requires prior knowledge of the expected expression rates. Finally, the method assumes that the pairs of (detected) adjacent grid points used to compute the grid parameters are real grid points and not noise, and that they are not affected by non-linear distortions. The proposed fluorescence image gridding method discussed in Chapter 6 does not require any such assumptions. In Chapter 6 we also develop a method for computing the probability of an estimated grid being the true grid. Such a measure can be used in conjunction with these gridding algorithms as well.

As discussed in section 3.2 a large number of microarray spot segmentation techniques have been developed over the years. Attempts have also been made to compare these algorithms. However, there is no consensus on which of these are the best methods. This is probably due to the fact that the best segmentation algorithm often depends on the imaging device used to produce the images and the experimental conditions. Indeed, most commercial application use their proprietary segmentation algorithms. Nevertheless, three dominant approaches towards spot segmentation have emerged. The first is intensity clustering based binary segmentation algorithms that often use techniques like k-means, mixture models, PAM, etc. to cluster the 1-d pixel intensities into foreground and background clusters. The second is graph based image segmentation algorithms, which use graph cut, MRF, and CRF models for segmentation. The third approach is to use active contour based segmentation.

All these approaches benefit from the fact that in almost all cases of spotted microarray image analysis, gridding is performed before segmentation. Therefore, each spot region can be

segmented locally and independently. However, for bead-based microarray image segmentation, the full image has to be segmented before gridding. The suitability of these approaches is therefore different for bead-based microarray images.

In Chapter 6 the suitability of a number of these popular approaches (k-means, graph cut, etc.) for bead-based microarray image segmentation are evaluated.

Chapter 4 - Review of Point Pattern Matching Techniques

In a point pattern matching problem, given a query point set and a template point set, the objective is to find a transformation that best aligns the points of the query set to the points of the template set. The points in the two sets that are aligned with each other are called corresponding points and the pair is referred to as a correspondence. The objective in matching is to find a transformation that maximizes the number of correspondences, the best or optimal transformation.

The transformation can be a similarity, affine, projective or a non-rigid transformation. A similarity transform preserves distances and angles between points and involves translation, rotation and scaling. An affine transform preserves ratios of distances, collinearity and parallelism but not angles. It involves a linear combination of translation, rotation, scaling and shear. Similarity and affine transformation invariant matching have a large number of applications and have been extensively researched. They are also important in terms of the other transformations: projective transform can be approximated locally using affine transform and many non-rigid matching algorithms use affine transforms to generate initial local estimates of the transformation.

A projective transform is involved whenever the viewing position of the observer changes with respect to the object. Matching under projective transform is therefore critical in almost all areas of computer vision. While projective transform preserves collinearity, under a non-rigid transform even this is lost. Many real life matching problems also involve highly articulated and elastic objects. In such cases a non-rigid transform is essential. Consequently non-rigid point matching also has a vast array of applications ranging from object recognition to medical imaging.

The dimensionality of the transformation space depends on the transformation used. The more articulated the transformation, the higher the dimensionality of the transformation space. For instance, in the case of 2-d point sets, affine transform has six unknowns while projective transform has nine unknowns.

Depending on the application, the point matching problem may require a one-to-one matching between the points in the two point sets or it may require only the overall shapes of the

point sets to match, in which case each point in one set is allowed to match many points in the other. In the bead pattern matching problem, we are interested in finding one-to-one correspondences between the beads in the fluorescence image and the bright or dark field image.

If both the point sets have same the number of points (say N) and every point in one set has a corresponding matching point in the other set, the correspondences can be represented using an $N \times N$ permutation matrix. A permutation matrix is a square matrix that has exactly one 1 in each row and each column and all the other elements are 0s. Since $N!$ such permutation matrices of size $N \times N$ exist, the two point sets can be matched in $N!$ ways.

An exhaustive search through the real valued transformation space and the high dimensional correspondence space is impractical for almost any non-trivial application. The objective in designing a point pattern matching algorithms is therefore to devise effective alternative strategies for searching the transformation and/or correspondence spaces.

In most applications the problem is further complicated by the presence of outliers, occlusions and jitter noise. Outliers are points in one set that do not have a valid correspondence in the other set. Both point sets may contain outliers. Outliers in one set is often referred to as missing beads to distinguish them from outliers in the other set. Occlusions occur when portions of a point set are completely missing or hidden. Further, it is also possible that the corresponding points in the two point sets do not align perfectly because of small perturbations, called jitter noise. Fig. 4.1 shows examples of occlusion, outliers and missing beads. Fig. 4.1 (a) shows a point set containing the contour of a fish (let this be the template point set). Fig. 4.1 (b) shows an example of occlusion; the solid boxes are the regions of the point set that are completely occluded. Fig. 4.1 (c) shows the top region of the point set as occluded (the solid box) while the lower part of the point set (shown as the dotted box) is corrupted with missing points and outlier. The outliers, points that do not have a corresponding match in Fig. 4.1 (a), are shown as unfilled circles. The visible region of Fig 4.1 (c) (the dotted box) also does not have some the points present in the corresponding region of Fig. 4.1 (a). These are the missing points. By convention we refer to points that are present in the fluorescence image but not in the bright or dark field image as outliers.

Fig. 4.1. Examples of occlusions, outliers and missing beads. (a) a point set showing the outline of a fish, (b) examples of occluded regions marked using solid boxes and (c) examples of occluded regions, outliers and missing points.

4.1 Types of Point Pattern Matching Algorithms

A natural way of classifying point pattern matching algorithms is to group them based on their search strategy. The most common search strategies are listed in Fig. 4.2. They are grouped into two categories based on the kind of transformations under which these strategies are most commonly used. Those that are used to match under similarity, affine or perspective transformation are grouped under rigid matching. While those that can be used to match articulated objects are grouped under non-rigid matching. Note that some strategies are suitable for both rigid and non-rigid matching.

Apart from the above, many other classifications of point pattern matching algorithms have also been proposed. Chui et al. [93] solves the point pattern matching problem by determining two unknowns: the transformation between the point sets and the point-to-point correspondences. A correspondence based algorithm solves the matching problem using correspondences alone and does not use the information in the transformation space. A transformation based algorithm on the other hand solves only for the overall transformation between the point sets. While a joint algorithm simultaneously or alternatively searches both the correspondence and transformation spaces.

Fig. 4.2 Taxonomy of point pattern matching algorithm. The point pattern matching algorithms are grouped into three categories: algorithms that are predominantly used for rigid matching (blue box), those that are predominantly used for non-rigid matching (yellow box), and those that are used for both rigid and non-rigid matching. Here we emphasize that these are not hard classifications, rather the approaches are categorized based on their most common usage.

Yet another classification of point pattern matching algorithms is possible based on the dimensionality of the point sets. Algorithms can be classified as those that can match a pair of 2-d point sets, a 2-d point set with a 3-d point set, a pair of 3-d point sets, and those that can match n-d point sets. Since the bead pattern matching problem involves matching a pair of 2-d point sets we will focus primarily only on such techniques.

4.2 Review of Point Pattern Matching Algorithms

4.2.1 Rigid Matching

String Matching

These approaches define the point set as a string of inter-point distances and polar angles with respect to a reference point. Often global properties like centroids are used as reference points. The point pattern matching problem is then reduced to the problem of matching the distance-angle strings. This is often achieved by comparing and matching the string elements based on some predefined similarity or error threshold, as done in [94, 95].

To better account for spurious points and outliers some algorithms (such as [96] and [97])

take a more local approach and define only the local neighbourhood of each point as distance-angle strings. Global matching is then guided by the local matching scores. Alt et al. [96] for instance use directed weighted graph matching. Ghosh et al. [97] use a modified version of the Smith-Waterman algorithm [98] to perform the local string matching.

String matching based algorithms are primarily used for matching point sets of similar sizes under similarity transform.

Sampling Based

Random sampling and consensus (RANSAC) [99] is a commonly used algorithm for model fitting in the presence of a large number of outliers. Unlike other sampling techniques which use as much of the data as possible, the idea in RANSAC is to use the minimum number of data points required to estimate the parameters of the problem. In case of affine invariant matching we need a minimum of three 2-d point correspondences to estimate the six unknowns. The algorithm randomly selects three correspondences and estimates the parameters of the affine transform (a candidate solution); this is referred to as the hypothesis generation step. The candidate transform is then applied to the entire query point set and a check is performed to see how well the remaining transformed query points are registered with the template points; this is referred to as the hypothesis evaluation step. If the point sets are aligned within an acceptable tolerance of error, the algorithm terminates; otherwise it draws a new set of samples (correspondences) from the point sets and repeats the process. Repetitions continue until samples have been drawn sufficient number of times to ensure that the probability of at least one sample of all inlier points being drawn is below a certain acceptable threshold.

The success of RANSAC therefore depends on finding at least one good sample. In the case of affine point matching it means that at least one sample of true correspondences has to be drawn. The number of iterations required (say k) to achieve a certain probability of drawing such a sample is a function of the number of points in the two point sets and the number of true correspondences present. It can be calculated as:

$$k = \frac{\ln(1 - p_{min})}{\ln\left(-\frac{r^3 - m^3 n^3}{m^3 n^3}\right)} \quad (4.1)$$

where m and n are the query and template point set sizes respectively, r is the number of true correspondences and p_{min} is the probability of success that we want to achieve [100].

In practice the number of iterations required to achieve a good match is greater than the theoretical estimate obtained using (4.1). This is because of the basic assumption in RANSAC that any sample of all inlier points can be used to estimate the transformation that is consistent with the rest of the points. However, this is usually not the case. Some of the inlier samples may not produce the optimal transformation, for instance due to jitter noise. To overcome this issue Chum et al. [101] developed locally optimized RANSAC (LO-RANSAC). They proposed various schemes to perform local optimization including an inner RANSAC scheme, where a second step of sampling and model updating is performed after the usual RANSAC hypothesis evaluation step. This second model update step uses all of the inlier data (instead of a minimal set) and therefore can produce a better model.

Although for large data sets the hypothesis evaluation step can be computationally expensive, however if a sufficiently large fraction of randomly selected data points do not agree with the hypothesis, it is possible to reject the hypothesis without the need for further evaluation. Based on this observation Matas and Chum [102] proposed a randomized RANSAC (R-RANSAC). They replace the original hypothesis evaluation step with a two-step process: a pre-verification step that uses a sample of the data followed by an evaluation using all the data. The second evaluation is performed only if the first pre-verification step is successful.

In another variation that aims at reducing the time complexity of the algorithm progressive sampling and consensus (PROSAC) [103] was proposed by the same authors. PROSAC uses local similarity measures to first order the potential correspondences in decreasing order of quality of match. Samples are then drawn from a small set (hypothesis generation set) of very high quality correspondences instead of all possible correspondences. The size of the hypothesis generation set is gradually increased to include lower quality correspondences as the algorithm proceeds.

PROSAC however requires sufficiently descriptive and distinctive local descriptors.

Geometric Hashing

Geometric hashing was originally developed by Lamdan and Wolfson [58] and popularized by Rigoutsos and Hummel [59]. It is a widely used technique for matching point sets under rigid transform. The algorithm involves two stages of processing: the offline phase and the online phase. In the offline phase, given the template points, the algorithm encodes the points using a basis set, which is a minimal set of points that can be used to encode the remaining points in a transformation invariant manner. Each encoded point and the basis set is stored as a key-value pair in a hash table. The algorithm repeats until all possible combinations of points that can be used as basis sets have been enumerated. While matching under similarity transform, for instance, all pairs of points in the template point set will be used as basis sets.

In the online phase, given the query point set, again a basis set is extracted and the rest of the points are encoded using this basis set. Each encoded point now is used as a key to access and vote for a hash table bin. After processing all the encoded points and accumulating all the votes the hash table entries that received votes higher than a pre-defined threshold are selected. Each of these selected hash table entries represents a possible match for the current query basis set. For each of these matches the least squares transformation is then estimated and the transformation's registration quality is evaluated. If a sufficiently good match is found, the algorithm terminates. Otherwise a new basis set is selected from the query point set and the process is repeated.

Theoretically the technique can be used for affine invariant matching, however the number of possible bases is too large ($O(n^3)$) when n is even moderately large. Even under similarity transform for relatively large point sets, the hash table's memory requirements are too large and even enumerating all the basis combinations becomes unrealistic. To reduce the complexity often the selection of basis sets is restricted using some application specific techniques, as in Jiang et al. [104]. Nevertheless, geometric hashing is still very widely used in critical applications where an optimal match needs to be guaranteed or when the number of outlier points is simply too large for other algorithms to handle. Geometric hashing is also inherently highly parallelizable.

Consequently, many different flavours of geometric hashing (such as a probabilistic version by Hoffman and Lindenbaum [105] and a randomized version by Irani and Raghavan [106]) have been developed. Robust image feature descriptors such as maximally stable extremal regions (MSERs) [107] have also been combined with geometric hashing [108]. Nakai et al. [109] developed a locally likely arrangement hashing (LLAH) scheme where a point p is encoded using its n neighboring points. Each combination of three points from n are used as basis, resulting in C_3^n bases. The barycentric coordinates of p for all the C_3^n bases are used as a descriptor for p . The method relies only on the local neighbourhood of p and is therefore robust to occlusion. To further increase robustness against missing points and outliers, they select all combinations of n points from a larger set of m points and create $C_n^m * C_3^n$ descriptors for point p .

Our proposed affine invariant matching method borrows some traits from geometric hashing. However geometric hashing is exhaustive in nature and cannot be used when the transformation space or the point sets are very large. Our approach is therefore to look for local matches that can be used to guide the search in the transformation space. Geometric hashing also requires large amounts of memory to store the hash tables. In our problem, the template database is also likely to grow very fast. A single bioassay experiment can result in up to 60 template point patterns, with typical labs performing hundreds of such experiments each day. Therefore, our algorithm must be able to balance between the amount of offline and online processing required for each image pair. While at the same time use less memory to store the pre-processed data.

Pose Clustering

Pose Clustering or Hypothesis Accumulation is another technique of determining the transformation or pose between two point sets related by a rigid transform. The idea is to first determine likely transformations by considering all possible correspondences and then find the transformation with the highest support. For instance, when matching two point sets of size m and n under similarity transformation, $t = l_m^2 l_n^2$ likely transformations would be estimated, where $l_m = \frac{m(m-1)}{2}$ and $l_n = \frac{n(n-1)}{2}$ are the number of point pairs in the two point sets. Now to find the

transformation with the strongest support, the t likely transformations are clustered in the transformation space. The transformations estimated from the true correspondences are likely to be very similar to each other. When there is no jitter noise they would be the same. Therefore, when a large number of likely transformations are densely clustered together in the transformation space, it is an indicator of a possible (locally or globally) optimal transformation being present in the region.

For complex rigid transformations such as affine and projective transforms the parameter space becomes too large and clustering in these higher dimensional transformation spaces becomes computationally expensive. The technique is therefore typically used only when similarity transformations are involved. Stockman [110] presents a comparison of pose clustering with other techniques.

Generalized Hough Voting

This is a closely related approach to pose clustering. In this case, instead of clustering a voting mechanism is used to determine the transformation with the most support. Stockman et al. [111] define an accumulator array of slightly overlapping bins of predetermined sizes in the rotation, scale and translation space. The optimal transformation is determined to be contained in the bin with the most votes. In a similar way Chang et al. [112] uses a discrete accumulator array in the scale-rotation space to perform brute-force matching and find a provably optimal match. Manninen et al. [113] further improve Chang's accumulator array by dynamically determining the accumulator bin widths based on transformation space density.

In a slightly different approach, Lipman and Funkhouser [114] developed a randomized correspondence space voting technique (called Mobius Voting). They used it to non-rigidly register two 3-d point sets extracted from meshes. Given two triplets of 3-d points a Mobius transformation is estimated, which is used it to transform and align the point sets. From the aligned point sets, correspondences are extracted using a nearest neighbour approach. Each correspondence thus found is recorded as a vote in a correspondence matrix. By iterating this process multiple times a fuzzy correspondence matrix is generated. Finally a greedy approach is used to find the set of high quality

correspondences from the fuzzy correspondence matrix.

An advantage of voting based approaches is that the process of determining the set of likely transformations can be parallelized. Moreover the techniques can be exhaustive and therefore inherently robust to outliers and occlusions. These techniques are also able to handle regularly structured point sets, which other algorithms often fail to match optimally. However, because of the dependence on clustering and voting, the techniques are computationally expensive and are mostly used to match small point sets under similarity transform.

We use some key ideas from these approaches. While we take a slightly more discriminative approach to find the set of likely transformation, we use the idea of transformation space density to perform pruning. We also use voting to determine the best transformation. Our discriminative correspondence generation technique results in a much smaller set of possible transformations than the set of all possible transformations. Most of these transformations are further pruned using a density based non-maxima suppression approach.

Iterative Refinement

Iterative Closest Point

The most popular iterative refinement algorithm is the Iterative Closest Point (ICP) algorithm [115] introduced by Besl and McKay. ICP iterates over two steps: a correspondence step and a transformation step. In the correspondence step the point correspondences are estimated based on a nearest neighbour scheme, where the template point closest to a query point is assumed to be the query point's corresponding match. Using these correspondences the least squares transformation is estimated in the transformation step. The correspondences are then updated again in the next correspondence step, this time using the transformed query point set. Iterations continue till either the change in the least squares error is not below a threshold or the set of correspondences don't change anymore. Since both the correspondence and transformation steps reduce the matching error, the algorithm is guaranteed to converge to a local minimum. Zhang [116] further proposed using a maximum distance threshold while estimating the correspondences to better handle outliers

and occlusions. He showed how this simple technique widens the algorithm's convergence basin and makes it more robust.

Given the nearest neighbour based correspondence estimation, ICP requires the corresponding points in the two point sets to be close to each other initially in order to achieve a good registration. However, because of its speed and simplicity it is very popular and various attempts have been made to improve it.

In many cases additional invariants are used to improve correspondence estimation. Johnson and Kang [117] use colour to match 3-d models, Pulli [118] uses surface normals to align 3-d range data, while Feldmar and Ayache [119] combine surface curvatures with ICP to perform non-rigid registration of surfaces. Several variants of ICP itself have also been developed.

Trucco et al. developed robust-ICP (RICP) [120] which uses random sampling to select a subset of the query points as control points that are used for transformation estimations. They also replace the least squares transformation estimation of the original ICP with least median squares transformation estimation. Consequently they are better able to cope with outliers.

Zinßer et al. developed Picky-ICP [121] which further improves the control point selection mechanism using a hierarchical approach. It uses only the 2^h th points as control points in the $(h + 1)$ th iteration. This significantly reduces the computation required and allows faster convergence, allowing them to match very large point sets. Both Picky-ICP and RICP are able to handle up to 20% outliers in the query point sets.

The original ICP and all these variants have so far assumed that the jitter noise associated with the points have an isotropic distribution. Maier-Hein et al. developed Anisotropic-ICP [122] which extends the original ICP to account for anisotropic jitter noise.

Fitzgibbon developed Levenberg-Marquardt-iterative closest point (LM-ICP) [123] which uses the Levenberg-Marquardt algorithm (LMA) [124] to minimise the sum of squared residuals in ICP. LMA being a damped least squares method is more robust than the original ICP's ordinary least squares approach and thus allows for better optimization. While LM-ICP achieves some

improvements in results compared to the original ICP, it still converges to a local minimum.

A detailed comparison of the various variants of ICP can be found in [125].

Robust Point Matching – Soft Assign

Robust point matching (RPM) was originally developed by Rangarajan et al. [126, 127] and further improved by Gold et al. [128]. The algorithm was also extended to perform non-rigid matching by Chui et al. [129]. As mentioned before the point pattern matching problem can be formulated as a combination of a combinatorial optimization problem that estimates the correspondences and a continuous optimization problem that estimates the transformation. The main contribution of RPM is in formulating the combinatorial optimization problem as a continuous optimization problem. To achieve this the correspondences are relaxed from being hard binary relations to soft assignments. Thus relaxing the correspondence matrix from being a permutation matrix to a doubly stochastic matrix, a non-negative matrix where each row and column sums to unity. The ingenuity of their approach is in formulating these discrete constraints as a continuous function.

Sinkhorn showed that continuous iterative row and column normalization of any non-negative matrix is guaranteed to produce a doubly stochastic matrix [130]. Rangarajan et al. perform a *softmax* followed by iterative row and column normalization to produce the doubly stochastic matrix. *Softmax*, which involves an exponentiation followed by a normalization ensures that each element is positive and that each row and column sums to unity. This process is referred to as *softassign*.

The algorithm alternates between a *softassign* based correspondence estimation and a least squares transformation estimation. To achieve robust matching the scale of exponentiation (β), used in *softmax*, is updated using a deterministic annealing schedule [131]. *Softmax* using a large β assigns very large output values to the row or column elements with the maximum values, while assigning very small values to the remaining elements. β therefore controls the discreteness of the matrix obtained after exponentiation and the convexity of the function. The algorithm begins with

a small β and as it moves closer to a local minimum, the distances between the query and template points becomes a more reliable estimate of the true correspondences and β is increased.

Dual Step EM Algorithm

The expectation maximization (EM) algorithm [132] is widely used for point pattern matching. Cross and Hancock [133, 134] estimate the correspondences and transformation using a hierarchical EM framework. In the E-step of the EM algorithm the *a posteriori* probabilities of the points being outliers are computed using the transformation estimated in the previous M-step. The EM algorithm is known to converge to locally optimal solutions. To avoid this, structural constraints are imposed on the contribution of correspondences to the likelihood function using structural correspondence probabilities (SCPs). The SCPs of query and template point pairs are calculated by matching supercliques, which represent point neighbourhoods, of graphs obtained through Delaunay triangulation [135] of the point sets. During the correspondence estimation step the contributions of the query and template point pairs are weighted by their SCPs.

Each template superclique constructed using k -nearest neighbours has $k!$ permutations and to calculate the SCPs each query superclique needs to be matched with $nk!$ supercliques where n is the number of template points. Further Delaunay triangulation can be heavily affected by outliers, therefore for problems with large number of outliers a large k is needed. The M-step of the EM algorithm involves re-building the template superclique dictionary and matching the query and template supercliques. Given the high time complexity of the M-step, the algorithm can only be used to match small point sets.

Relaxation Matching

Relaxation labelling was introduced by Rosenfeld et al. in their seminal paper [136], originally for classification problems. The idea behind relaxation labelling is to use contextual information available from other objects when classifying a particular object.

Relaxation labelling was first used for point pattern matching by Ranade and Rosenfeld [137]. Their algorithm employed an exhaustive search technique for matching images under

translation. The technique popularly called *relaxation matching* is an iterative algorithm which begins by assigning merit scores to each query-template point pair, based on how well their registration aligns the rest of the point sets (i.e. the contextual information). Given two query-template point pairs a similarity transformation is estimated and the merit score of these two point pairs is computed as the average of inversely proportioned Euclidean distances between the transformed query points and their corresponding matches in the template point set. The corresponding match for a transformed query point being the nearest template point.

Ton and Jain [138] improved the algorithm to allow rotation along with translation by using ratios of relative distances. They also improve the running time complexity using a merge-sort technique to find the closest points while computing support. Thereby reducing the complexity of the original algorithm from $O(m^2n^2)$ to $O(mn(m+n))$, where m and n are the sizes of the query and template point sets. This nearest neighbour search can now be achieved using k-d tree in $O(m^2n\log n)$ time.

In another improvement, Li [139] uses attributed relational structures (ARSs) constructed using similarity transform invariant features of point triplets to perform the matching. A neighbourhood based approach is used to construct the ARSs thus incorporating the contextual information. The combinatorial optimization problem of matching the nodes of the ARS is formulated as a continuous relaxation labelling process that maximizes a global gain function using gradient ascent and graduated non-convexity. The gain function computes the similarity between two ARSs based on the similarity of their corresponding nodes.

Relaxation labelling based methods are dependent on good initialization to find the optimal matching.

Spectral Methods

Scott and Longuet-Higgins [140] developed a non-iterative matching algorithm that uses the spectral properties of a proximity matrix; a matrix whose elements are Gaussian weighted Euclidean distances between query and template point pairs. A Singular Value Decomposition

(SVD) of the proximity matrix produces two orthogonal matrices (left and right matrices) and a (central) diagonal matrix. The columns of the left orthogonal matrix (the eigenvector matrix) are the eigenvectors of the proximity matrix and each row of this matrix projects a query point on to the eigenspace of the proximity matrix. Similarly the columns of the right orthogonal matrix project the template points on to the eigenspace of the proximity matrix. Since both the query and template points are now defined in the same eigenspace, they can be directly compared. The inner product of the rows of the left orthogonal matrix with the columns of the right orthogonal matrix produces a doubly stochastic matrix (the association matrix) that encodes the strength of association between the points in the two point sets. The optimal permutation matrix is approximated from this doubly stochastic matrix. This permutation matrix effectively minimizes the sum of squared distances between the corresponding points in the two point sets. Since the inter-point distances are minimized this algorithm can match only under scaling, translation and shear but not rotation.

Shapiro and Brady [141] modified the algorithm to allow matching under rotation using intra-point distances, i.e. distance between points in the same set, to define two proximity matrices. Singular value decomposition (SVD) of the two proximity matrices produces two left orthogonal matrices whose rows project the points in their respective eigenspaces. Since the two point sets are likely to have similar structures their eigenspaces can also be expected to be similar, and therefore the points defined in their respective eigenspaces can be directly compared to obtain the association matrix. However, when there are significant structural changes such as those caused by a combination of large number of outliers, occlusions or even large affine transformations, the eigenspaces of the two point sets can be significantly different.

To improve robustness against such structural changes Wang and Hancock [142] use a kernel principal components analysis (PCA) like approach where the subspace spanned by only the most dominant (2 or 3) eigenvectors are used for matching. They show that using the polynomial kernel to calculate the proximity matrices produces better results than the Gaussian kernel. They also developed an iterative version of the algorithm [143] where labelling consistency constraints are used to compute the proximity matrices and perform articulated matching.

Li and Hartley [144] showed that two gram matrices: an inner product matrix and an outer product matrix (of the eigenvectors) can be used to estimate the transformation and correspondences respectively. The iterative correspondence and transformation estimations are implemented as an iterative Newton-Schulz factorization.

Additional invariants such as colour and intensity have also been used to calculate the proximity matrices. Srinivasan and Kankanhalli [145] use colour histograms of pixels in the neighbourhood of corner points in addition to Euclidean distance to measure the proximity between corner points in two images. Pilu [146] uses correlation between patches, centred over corner points, to weigh the Gaussian distances between corner points of two images to estimate stereo correspondence.

While spectral techniques provide a non-iterative method for matching, the SVD step is computationally expensive. SVD of a $n \times n$ matrix requires $O(n^3)$ time. Further, since the eigenspaces of the point sets need to be similar, spectral methods cannot be used when the two point sets have significantly different global structures, for instance when the query point set is a small subset of the model point set.

Tree Search

Zahn [147] used minimal spanning trees (MSTs) constructed from the complete graphs of the two point sets to compare and find regions of the point sets that match under similarity transform. Ratio of lengths of the two edges forming the smallest angle at the node was used as the descriptor for the node. Lavine et al. [148] also used MSTs constructed from complete graphs of the two point sets to do matching. They compared sorted adjacent edge distances of the two MSTs to find global similarity between the two point sets. These methods can only match when there are no occlusions, outliers, or missing points.

In a different approach, Li and Holstein [149] used k-d tree to match two point sets with equal cardinality. They developed a new k-d tree that partitions the point set such that the inter-point distances, projected onto the axis perpendicular to the partitioning hyper-plane, are

maximized. A model k-d tree is constructed from the template point set where each node encodes the number of points in its left child and the axis of splitting. The centred and rotation corrected query point set is split using the information available in the model k-d tree to produce a similar tree. The point-to-point correspondences are obtained using the leaf ordering of the two trees. The technique is able to match point sets under small local non-rigid deformations as long as the point sets have overall global similarity, same cardinality, and contain no outliers.

Another common tree search technique, the branch and bound (B&B) framework, involves recursively sub dividing the search space into smaller sub-spaces and calculating lower bounds of the cost function in these sub-spaces. Sub-spaces whose lower bounds on the cost function are higher than that of the most optimal solution that has been found so far can be pruned immediately, thereby reducing the overall search space. The key challenge in developing such an algorithm is to formulate the objective function such that lower bounds of the cost function within a bounded search space can be efficiently computed.

The B&B framework for point pattern registration was first developed by Huttenlocher and Rucklidge [150] and was further improved by Mount et al. [151]. Mount et al. developed a geometric B&B technique to search the six dimensional affine transformation space. To calculate the lower bounds of the cost function they define an “uncertainty region” of a point, which is the region in the image space to which a query point can be transformed under a bounded interval of the transformation space. The lower bound of matching cost in the bounded transformation space is computed as the Hausdorff distance between the uncertainty regions and the template point set.

Pfeuffer et al. [152] further improved the technique using the same concept of “uncertainty regions” using Chui and Rangarajan’s RPM energy function [93, 127] as their objective function. They also developed a B&B strategy to search over the correspondence space. Although, for larger point sets the correspondence space becomes very large and searching this space is very time consuming even using a B&B search strategy.

Other Techniques

Graphical Models

Caetano and McAuley [153, 154, 155, 156] developed a series of globally rigid graph representations of point sets, on which probabilistic graphical model based inferencing can be performed to do matching. In their framework, template points represent graph nodes and query points represent possible assignments (class labels). Maximizing the joint probability distribution of the node-label assignments is then equivalent to finding a match for query points in the template point set. The binary edge potentials in the model, which encode the probability of matching a pair of template points (t_1, t_2) with a pair of query points (q_1, q_2) , are inversely related to the difference $|l_t - l_q|$, where l_t and l_q are the distances between t_1 and t_2 and q_1 and q_2 respectively.

They first developed a globally rigid chordal 3-tree graph representation on which junction tree belief propagation [157] based exact inference can be performed. The graph has a maximal clique size of 4 and therefore the junction tree based matching has a running time complexity of $O(nm^4)$, where n is the number of nodes (template points) and m is the number of labels per node (query points). McAuley et al. [156] further reduced the running time complexity to $O(nm^3)$ using a different global rigid graph with a maximal clique size of 3. While this graph does not have a chordal tree structure, the clique graph extracted from it is cyclic, and it was shown that loopy belief propagation in a cyclic clique graph converges to the optimal MAP assignment when iterated multiple times. Caetano and McAuley [154] then modified the graph representation into a chain graph structure, with a clique size of 2, augmented with an additional binary variable to encode reflections. This eliminated the need for multiple iterations of loopy belief propagation. While these graphical models allow jitter noise and outliers in the query point set, they cannot match with outliers and missing points in the template point set. To account for outliers, they developed a method [155] similar to geometric hashing where two points (conditioning points) are used to rotation and reflection invariantly encode and match the remaining points in the set. To account for missing points they suggest running the algorithm multiple times, using different pairs of points as conditioning points.

Genetic Algorithms

Ansari et al. [158] developed a Genetic Algorithm (GA) that searches the correspondence space to match point patterns with missing beads and outliers. The chromosomes are used to encode correspondences. For instance, based on some specific indexing of the template point set a chromosome “203” represents that the first template point matches the second query point, the second template point is an outlier, and the third template point matches the third query point. They ensure that only one-to-one correspondences are achieved during crossovers and mutations. However, large point sets would result in large chromosomes and very slow optimization. The technique is therefore only suitable for matching small point sets.

Zhang et al. [159] developed a GA based optimization technique which minimizes the Hausdorff distance between two point sets related by an affine transformation. They define a “feature ellipse” of a point set, as an ellipse centred over the point set and having a shape similar to that of the convex hull of the point set. Triplets of points lying on the query and template feature ellipses, are randomly selected and encoded as the initial population of chromosomes. Each chromosome therefore encodes an affine transformation and its fitness is calculated as the inverse bidirectional partial Hausdorff distance between the template point set and the transformed query point set. While using points lying only on the feature ellipses limits the search space, such a scheme cannot be used for matching under large occlusions.

Shape Context

Belongie et al. [160] developed shape context, a point descriptor, which coarsely describes the distribution of the points in a set with respect to a given point (say p). A coarse histogram of the spatial coordinates of the points in the set, with point p as origin, is used as a descriptor for p . To make the descriptor more sensitive to points closer to p log-polar sampling is used to construct the histogram. The descriptor is inherently invariant to translation and is partially robust to occlusions and outliers. To achieve scale invariance the distances of points from p are normalized, typically by dividing them with the mean. To achieve rotational invariance the authors suggest using the tangent at p as the reference x -axis. While such tangents are often defined for points extracted from

contours which are piecewise continuous, for randomly structured point sets such tangents may not be defined at all points. The χ^2 statistic between the shape contexts of a query and a template point is used as the cost of matching them. The correspondence cost matrix constructed using shape contexts can be used to find the optimal correspondences by solving the assignment problem. Finally, with the correspondences established, a non-linear transformation is estimated using a thin-plate spline (TPS) formulation.

Given two sets of points and a few known correspondences (control points), TPS generates a smooth interpolation such that the two sets of control points are perfectly aligned, while minimizing the deformation energy. TPS has the nice property that the mapping function can be decomposed into an affine and a non-linear part. This has led to many techniques where an affine transform is first estimated and the non-linear deformation is then approximated using the well registered points.

4-Point Congruent Sets

Aiger et al. [161] also uses a local shape descriptor called 4-point congruent sets (4PCS) to register 3-d point clouds. The 4PCS shape descriptor describes the arrangement of 4 coplanar points in an affine invariant manner. Given a set of 4 co-planar points from the query point set, they use a fast technique (the 4PCS algorithm) to extract all quadruple of points from the template point set that are potentially congruent to the given 4 points. For each potential match, the optimal transformation (in a least squares sense) is computed and its registration quality is estimated based on how well the rest of the points are aligned. This 4PCS algorithm is run within a RANSAC framework, where in each iteration a quadruple of approximately co-planar points is randomly chosen from the query point set and the best possible registration is estimated using it. We use a variant of the 4PCS descriptor in the proposed affine invariant matching algorithm.

Graph Matching

The point pattern matching problem can also be formulated as a weighted graph matching problem, where the points are the nodes of the graph and edges of the graph encode the distance

between points. In graph matching the objective is to estimate the correspondences between the nodes and the edges of the two graphs. This makes it a much harder problem as not only correspondences between nodes (points) have to be established, but also correspondences between edges have to be established. Point pattern matching is a linear assignment problem (LAP), given the transform. Graph matching however, is an instance of the quadratic assignment problem (QAP) where a quadratic term encodes the structural (edge) similarity and a linear term encodes local (node) similarity. The QAP is known to be NP-complete [162]. Further, when there are occlusions, outliers and missing beads the problem becomes that of finding the maximum common subgraph (MCS) between two graphs.

The bulk of graph matching algorithms can be categorized into three types: trees search based [163-168], spectral decomposition based [169-173] and continuous optimization based [174-178] algorithms. The tree search and spectral approaches for graph matching are very similar to their point pattern matching counterparts.

Spectral methods embed the graph in a higher dimensional eigenspace, spanned by the eigenvectors of the adjacency matrix, where the actual matching is performed. Spectral methods for graph matching were pioneered by Umeyama [169]. Among convex optimization techniques, Graduated Assignment and Relaxation Labelling based techniques ([176, 177]) are some of the most commonly used. Continuous optimization based techniques are however even more sensitive to outliers and occlusions than their point pattern matching counterparts. Optimization schemes that are more robust to local minima, such as convex-concave optimization [178], have been developed in recent years. However, these algorithms have high running time complexities and can be used for only small graphs. Tree search based methods are often provably optimal, and therefore are widely used in real applications. However, they can have worst case time complexities that scale exponentially. The first tree search based algorithm proposed by Ullman [163] still continues to be one of the most popular graph matching algorithm.

Error-tolerant graph matching schemes compute a matching score between two graphs based on their structural similarities. A widely used error-tolerant measure is the graph edit distance

(GED) [179] which quantifies the dissimilarity between two graphs as the minimum number of edits (such as edge deletion/insertion, node deletion/insertion) needed to transform one graph into another.

Dissimilarity based graph embedding techniques [180, 181] use such structural dissimilarity scores between a given graph and a database of prototype graphs to embed the given graph in a dissimilarity space. Matching two graphs in the dissimilarity spaces reduces to computing the dot product between two vectors. Various graph kernels [182-184] have also been developed which allow direct computation of graph similarity and are often used as local descriptors for graph nodes, allowing direct matching of the nodes of two graphs.

Genetic algorithms [185] and neural networks [186] have also been used to perform graph matching. Cour et al. developed balanced graph matching (BGM) [187] which uses spectral analysis to find the most dominant cluster in a weighted bipartite graph. Such approaches reduce the complex MCS problem to a much simpler bipartite graph matching problem. They are also better able to deal with outliers, occlusions and missing point. However, they require pre-computed node similarity scores as input. Such node similarity scores are often calculated using graph kernels or using GED.

For a detailed review of graph matching algorithms we refer the reader to [188].

4.2.2 Non-Rigid Matching

Piecewise Approximation

When the individual parts of a point set undergoes (approximately) rigid transformation while the overall object undergoes a non-rigid transformation, piecewise approximation based matching is used. These methods approximate the global non-rigidity with a connected network of locally rigid transformations. The global non-rigid transformation has to be smooth and continuous enough so that it can be locally approximated as rigid. This makes piecewise approximation based matching methods susceptible to occlusions. Li and Holstein's k-d tree based method [149], discussed previously, is an example of piecewise approximate matching.

Hinton et al.[189] and Revow et al. [190] used Gaussian mixture models (GMM) and b-

splines to match contour like point sets of handwritten characters. In their model, the contours of the characters are modelled as b-splines passing through a set of control points, while the control points themselves are modelled as Gaussians. Matching was achieved by balancing the deformation energy of the splines with the energy required to fit the mixture of Gaussian control points to the image pixels. The optimization is performed using an EM algorithm. The E-step of the algorithm is used to compute the probabilities of the image pixels belonging to the Gaussian control points, given the control points. In the M-step of the algorithm the local affine transformation parameters are estimated and the control points are transformed using the b-spline formulation.

Pappu et al. [191] used an objective function similar to that of RPM to develop a non-rigid matching framework which they refer to as “Affine Quilts”. The template point set is broken down into regions (say k regions), such that each region can undergo only affine transformations. The algorithm iterates over two steps. In the first step the correspondences are estimated by solving the assignment problem using *softassign*. In the second step k affine transformations are estimated for the k regions. However, unlike other piecewise approximation methods, the continuity of transformations between adjacent regions is not necessary in this algorithm.

Chui [192] extended the RPM algorithm to perform non-rigid matching using thin-plate splines (TPS) to parameterize the non-rigid transformation. The affine deformation and the non-linear deformations of the TPS mapping are controlled by two regularization parameters that are optimized using simulated annealing.

Hao et al. [193] used piecewise approximations for fingerprint matching. They used additional features, such as ridge information, to establish some initial correspondences. The initial correspondences and their neighbouring points are used to form the pieces in this piecewise approximation method. The pieces in the two point sets are individually matched, and the matching errors associated with them are used to weigh their contributions when computing the global transformation.

Pitiot et al. [194] developed a piecewise approximate technique for registering images using pixel intensities alone. It involves dividing both query and template images into sub-images using

a lattice structure and then computing a correspondence field between pairs of query-template sub-images. The correspondence field is clustered using hierarchical clustering [195] to break the query image into rigid pieces. The optimal affine transforms between the rigid pieces of the query image and the template image are then estimated. Finally the affine transformed query pieces are interpolated to obtain a smooth non-rigid transformation. While Pitiot et al. use only pixel intensities to register the images, similar landmark based techniques have also been developed [196].

Deformable models

Aggarwal et al. [197] classify deformable models into two categories: parametric models and physically based models. In parametric deformable models the correspondence problem is transformed into a parameter estimation problem, while in the physically based models the non-rigid transformation is constrained based on prior knowledge of the true physical transformation present in the application. Physically based models can also be combined with parametric models, such as in [198].

Joshi and Lee [199] used elastic nets [200] to solve the point pattern matching problem. The original elastic net algorithm was developed by Durbin and Willshaw to solve combinatorial problems involving geometric structures, such as the travelling salesman problem. The elastic net optimization framework minimizes an energy function that is akin to deforming a contour, such that neighbouring points on the original contour remain close to each other on the deformed contour. Joshi and Lee simplified Durbin and Willshaw's energy function and re-parameterized it using the transformation parameters, thereby transforming the correspondence problem into a continuous optimization problem. The optimization was done using gradient descent and they showed promising results under limited occlusion. Although the method is able to handle occlusion or missing points in any one of the two point sets.

Luo and Hancock [201] developed a joint transformation and correspondence estimation technique for matching a sequence (or frames) of a point set undergoing non-rigid distortion. They represent the non-rigid deformation as the change in each point's affiliation to the significant clusters in the point set. A sequence of labelled point sets (the training data) is used to compute a

mean point set representation and the covariance matrix of the displacement of points in the sequence of labelled point sets is also estimated. The un-labelled points are then represented as a linear combination of the eigenvectors of this covariance matrix. Each point's affiliation to the eigenvectors are referred to as the modal coefficients. Given a query point set, alignment is achieved by estimating the modal coefficients for the query point set that minimizes the sum of squared-error between the query point set and the mean point set.

Chui and Rangarajan [129] developed an EM based matching technique where the point pattern matching problem is formulated as a maximum a posteriori (MAP) estimation problem. In this model, the point sets are represented as mixtures of Gaussians and each point is represented as a Gaussian. The matching problem can then be formulated as MAP problem, which maximizes the probability of the transformation parameters, given the query and template point set mixture parameters. The EM algorithm is used to iteratively estimate the MAP transformation parameters and in the E-step and update the mixture parameters in the M step. Simulated annealing is also incorporated within the EM framework to update the covariance of the Gaussians. Missing points and outliers are modelled using an additional Gaussian in the mixtures. McNeill and Vijayakumar [202] have also used a similar approach to match shapes under similarity transform.

Tsin and Kanade [203] extended the well-known correlation based image matching technique to perform point pattern matching. They use symmetric, non-negative kernel functions such as those used for density estimation to calculate a kernel correlation (KC). The KC for a point set is defined as the sum of affinities between pairs of points in the set. They show that KC is a function of the entropy of the point set. The problem of registering the two point sets is formulated as that of finding the transformation that maximizes the kernel correlation (or minimizes the entropy) of the joint point set, the union of the query and template sets. While the technique is inherently capable of handling jitter noise, it is unable to handle extreme occlusions.

Jian and Vemuri [204] also use Gaussian mixture model (GMM) based model for matching. Similar to Chui and Rangarajan [129] the two point sets are represented as two GMMs, with the Gaussians centred over the points. One of the point sets is parameterized using the transformation

parameters and matching is achieved by minimizing L_2 norm between the mixture distributions. To realize this, they derived a closed form solution for L_2 norm between two Gaussian mixtures.

Myronenko and Song [198] further improved the GMM based matching by incorporating a physically based regularization term to constrain the deformation. The regularization term forces the points to move coherently, such that the point set's topology is preserved. They refer to this as coherent point drift (CPD). The regularization essentially forces the points close to each other to have similar deformations, thereby keeping the non-rigid deformation smooth across the point set. One of the main advantages of this method is that it generalizes GMM based point pattern matching to higher dimensional spaces, whereas the methods of [203] and [204] are restricted to 2 and 3 dimensional cases.

4.3 Summary of Point Pattern Matching Techniques

In the previous section a large number of point pattern matching techniques have been discussed in detail. In this section we identify the most widely used and the current state-of-the-art in point pattern matching techniques, and summarize their strengths and weaknesses. In Table 4.1 nine point pattern matching techniques are compared based on seven factors. Here the first factor, sensitivity to initial transformation, indicates how likely the method is of failing if the initial transformation between the two point sets is large.

ICP [115] and RANSAC [99] are probably the most widely used point pattern matching techniques. ICP has a low running time complexity and can be used to match large point sets. However, ICP only succeeds in finding the global optimal match if the initial transformation between the two point sets is small. This makes it unsuitable for the bead pattern matching problem, where the initial transformation can be a combination of a random un-constrained similarity transform, and small affine and non-linear transforms.

RANSAC offers relatively good robustness to data distortions such as occlusions, outliers and missing beads. Recall that using (4.1) the number of iterations required by RANSAC to achieve a minimum probability p_{min} can be computed, given the query and template point set sizes m and

n , and the number of true correspondences r . As outliers increase, m and n increase without increasing r and when occlusions and missing beads increase, r decreases. Both these cause a rapid reduction in the denominator of (4.1) and therefore increase the number of iterations required. In the bead pattern matching problem, the number of outliers, occlusions and missing beads is very high while p_{min} also has to be kept very high. The number of iterations required in such cases can become impractical.

Table 4.1: SUMMARY OF POINT PATTERN MATCHING ALGORITHMS

Matching Algorithm	Sensitivity to Initial Transformation	Robustness to Occlusion	Robustness to Outliers, Missing Points	Robustness to Jitter Noise	Transformation Estimated	Time Complexity	Space Complexity
ICP	High	Low	Low	Moderate	Small non-linear, Rigid	Low	Low
RANSAC	None	High	High	Low	Rigid	Moderate to High	Low
TPS-RPM	Moderate	Moderate	Low	High	Non-linear, Rigid	Low to Moderate	Low
CPD	Moderate	Moderate	Low	High	Non-linear, Rigid	Low	Low
GMM	Moderate	Moderate	Low	High	Non-linear, Rigid	Low	Low
GH	None	Moderate	High	Low	Rigid	High	High
LGH	None	High	High	Low	Small non-linear, Rigid	Moderate	Moderate
LLAH	None	High	High	Low	Small non-linear, Rigid	Very High	High
GA [158]	None	High	Moderate	Low	Rigid	Very High	Low

TPS-RPM [192] was one of the early convex optimization based non-linear point pattern matching algorithms. It has therefore been widely adopted in many applications. Due to its deterministic annealing based optimization TPS-RPM offers moderate robustness against occlusions and outliers. Although, simulated annealing can take a significantly long time to converge if a very small registration error is being sought.

CPD [198] and GMM [204] are two closely related methods that represent the current state-of-the-art in non-linear point pattern matching. Given the EM based convex optimization used in these methods they are prone to getting stuck in local minima and therefore offer limited robustness against outliers and missing points.

In geometric hashing (GH) [58, 59], when estimating an affine transformation, each combination of 3 points in the dataset is used as basis. If a point set has m points then C_3^m basis would be used. For each of these basis the barycentric coordinates of $m - 3$ points (all points excluding the points used to form the basis) are stored as keys in the hash table. For large point sets the number of entries in the hash table can be prohibitively large. For $m = 500$, for instance, the number of basis will be over 200 million and the number of entries in the hash table will be over 100 billion. Therefore, GH cannot be directly used for the bead pattern matching problem, in which thousands of beads have to be matched. However, it is possible to match large point sets by performing GH locally. When performed locally, the basis are formed only by points that are in the local neighbourhood of each other (the k -nearest neighbours of each other). Also, a basis is encoded in the hash table using the barycentric coordinates of only the k points that are in its local neighbourhood. This significantly reduces the time and space complexity of GH and also improves its robustness to occlusions. To draw a distinction between the original definition of GH and this local variant of it, we refer to this local variant as local geometric hashing (LGH).

LLAH [109] another local variant of GH is included in this summary as it is closely related to the proposed affine invariant matching method which is presented in Chapter 7.

In the bead pattern matching problem, where the number of points in the two patterns can be in the thousands, the correspondence space can be extremely large. For two point sets of sizes m and n there are C_n^m valid correspondence matrices. GA based matching methods that search the correspondence space [158] are therefore not very well-suited for this problem. Also, GA based methods cannot match under non-linear deformations [159] which makes them unsuitable for the bead pattern matching problem where small non-linear deformations can be present.

TPS-RPM, CPD, GMM, GH, LGH, and LLAH are further evaluated in the context of the bead pattern matching problem in Chapter 7. In the next chapter we discuss the proposed bright and dark field image gridding and microarray classification methods.

Chapter 5 - Bright and Dark Field Image Processing

Deposition image processing involves two sub-tasks: image gridding and micro-well classification. In this chapter we develop two novel methods for these two sub-tasks. The objective of the gridding step is similar to the gridding techniques discussed in Chapter 3, i.e. to detect the spatial location (x and y coordinates) of all the micro-wells in the image. The objective of the micro-well classification step is to classify the detected micro-wells as either empty or containing a bead, thus identifying the spatial locations of the deposited beads in the image.

In Section 5.1 a fully automatic method for gridding bright and dark field images of bead-based microarrays is presented. In Chapter 3 numerous techniques for gridding fluorescence images of traditional spotted microarrays were discussed, but to our best knowledge, no algorithm has yet been developed for gridding bright or dark field images of bead-based microarrays, which have very different characteristics. Apart from being a key component of the proposed robust spatial bead encoding scheme, the method can also be used 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 and is therefore well-suited for high-throughput analysis. It is capable of identifying the grid structure even in the presence of extreme amounts of artefacts and distortions. Evaluation results on a variety of images, with different characteristics, are presented.

In Section 5.2 a novel unsupervised method for classifying the micro-well regions, as either containing a bead or as empty, is presented. The main challenge in this classification problem is the large variations in the appearance of the micro-wells across images and even within the same image. To deal with this problem, the proposed algorithm uses local contextual information to guide the

clustering algorithm which identifies the clusters of empty and bead containing micro-wells.

5.1 Bright and Dark Field Image Gridding

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 (4) blurring of images due to poorly calibrated image acquisition (see Fig. 5.1(c)). The 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. 5.2.

Fig.5.1. Small regions of bright field images. (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.

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 5.1.1.

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 [205] is used to approximately locate the wells in the blocks. A variant of the Iterative Closest Point [115] pattern matching algorithm is then used within an Inner-RANSAC [101] 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 Inner Iterative Closest Point-Rejection Sampling and Consensus (Inner ICP – REJSAC) based

grid fitting. The method is detailed in section 5.1.2.

Fig.5.2. Overview of the gridding method.

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 [206] 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 5.1.3.

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 5.1.4.

5.1.1 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. 5.3(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 [207] 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 $[x_t, y_t, s_w, s_h]$, 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 s_w and 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 s_w and 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 [208] 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}) [209] 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:

$$\psi_{\text{fitness}} = \frac{n_b}{\sigma_a \sigma_n |1 - I_{\text{disp}}| + 1}. \quad (5.1)$$

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. 5.3(c) shows an image after rotation correction. The Radon transform gives the 1-D projection of the image along a projection line as:

$$R(\theta, d) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} b(x, y) \delta(d - x \cos \theta - y \sin \theta) dx dy, \quad (5.2)$$

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 $R(90^\circ, d)$ for the image in Fig. 5.3(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. 5.3(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 $[0^\circ, 70^\circ]$. The step size of the search is set to 1° . Let $R_\theta(d) = R(\theta, d)$, where $\theta = 0^\circ, 1^\circ, \dots, 70^\circ$ 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 $d = 0, \dots, N - 1$, the Fourier spectrum is computed as:

$$F_\theta(K) = \sum_{d=0}^{N-1} R_\theta(d) e^{-i2\pi K \frac{d}{N}}, \quad K = 0, \dots, N - 1. \quad (5.3)$$

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. 5.3(c). The location of the crest c_i in $R_\theta(d)$ is determined by $c_i = -\phi_\theta + (i + 0.75)\lambda_\theta$. Here, $-\phi_\theta$ 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:

$$\psi_\theta = \frac{((L\mathbf{R})^T \mathbf{1}_p) * \left(\left((\mathbf{1}_{pN} - L)\mathbf{1}_N \right)^T \mathbf{1}_p \right)}{((L\mathbf{1}_N)^T \mathbf{1}_p) * \left(\left((\mathbf{1}_{pN} - L)\mathbf{R} \right)^T \mathbf{1}_p \right)}. \quad (5.4)$$

where p is the number of crests detected in R_θ , $\mathbf{R} = [R_\theta(0), \dots, R_\theta(N - 1)]^T$, $L(i, j) = \begin{cases} 1, & |c_i - j| \leq 0.25\lambda_\theta \\ 0, & |c_i - j| > 0.25\lambda_\theta \end{cases}$ 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:

$$\alpha = \arg \max_{\theta \in [0^\circ, 70^\circ]} \psi_\theta. \quad (5.5)$$

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. 5.3(c).

Fig.5.3. (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.

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. 5.3(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.

5.1.2 Grid Fitting

To make the proposed gridding method robust against non-linear distortions, $\mathbb{B}'$ is broken

down into m^2 overlapping blocks ($\mathcal{B}'$) in a $m \times m$ 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 $\mathbf{M}_{n \times 2} = [\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_n]^T$, 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 $\mathbf{M}$ with the set of detected guide points.

Throughout the rest of this chapter, 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 $H(\mathbf{x}, r)$.

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 [205, 206] 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:

$$P(\mathbf{x}) = \sum_{\mathbf{x}_i \in H(\mathbf{x}, r, r)} K(\mathbf{x}_i - \mathbf{x}) \mathbb{I}_i. \quad (5.6)$$

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 $\mathbf{x}$,

$$\mathbf{x}_{\text{shift}} = \frac{\sum_{\mathbf{x}_i \in H(\mathbf{x}, r, r)} K(\mathbf{x}_i - \mathbf{x}) \mathbb{b}_i(\mathbf{x}_i - \mathbf{x})}{\sum_{\mathbf{x}_i \in H(\mathbf{x}, r, r)} K(\mathbf{x}_i - \mathbf{x}) \mathbb{b}_i}, \quad (5.7)$$

points to the opposite direction of the gradient of the convolution surface P [210].

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

$$\underset{\mathbf{x}_i \in H(\mathbf{x}, x_{\text{gap}}, x_{\text{gap}})}{\operatorname{argmax}} P(\mathbf{x}_i) \quad (5.8)$$

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 (5.6). 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 $(x_{\text{gap}}/4)$:

$$K(z) = \frac{1}{x_{\text{gap}}^2 \sqrt{2\pi \left(\frac{x_{\text{gap}}}{4}\right)^2}} e^{-\frac{\|z\|^2}{2\left(\frac{x_{\text{gap}}}{4}\right)^2}} \quad (5.9)$$

is used as the kernel function.

Transformation Estimation

To estimate an affine transformation between the model grid $\mathbf{M}$ and the guide points $\mathbf{S}_{\ell \times 2} = [\mathbf{s}_1, \dots, \mathbf{s}_\ell]^T$, 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 $\mathbf{S}_{3 \times 2}^k = [\mathbf{s}_1^k, \mathbf{s}_2^k, \mathbf{s}_3^k]^T$. The selected sample $\mathbf{S}^k$ 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 $\mathbf{M}'_{3 \times 2} = [\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3]^T$, 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 $\mathbf{M}'$ needs to be chosen at the start.

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

$$\mathbf{S}^k = \mathbf{A}^k \mathbf{M}' + \mathbf{L}^k, \quad (5.10)$$

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

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. [101]. 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 [211], 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 $\mathbf{M}'$ and $\mathbf{S}^k$ was already established. The neighborhoods of $\mathbf{M}'$ and $\mathbf{S}^k$ 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 $\mathbf{S}^k$. Correspondences between $\mathbf{s}_i$ and $\mathbf{t}_i \in \mathbf{T}^k$ 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. 5.4 illustrates the bootstrapping method. The two point sets are shown as blue empty circles and orange filled circles. Fig. 5.4(a) shows the initial alignment obtained using the three corresponding pairs (points inside the red boxes). In Figs. 5.4(b) and 5.4(c), the bootstrap region (illustrated as points inside boxes) is extended and the alignment quality is also improved.

The Trimmed ICP [212] 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. 5.4 (b) and 5.4 (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 $\{s_i, t_i\}$ be a correspondence between a guide point s_i and a transformed model point $t_i \in T_h^k$ where 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 $d_i = (s_i - t_i)^T(s_i - t_i)$ is calculated. The correspondence pairs, for which d_i is less than the median of the d_i s, are used to estimate the new similarity transformation matrices A^{hk} and L^{hk} and the new transformed model grid point set T_{h+1}^k , for the $(h + 1)^{\text{th}}$ inner iteration.

The local optimization (inner iterations) continues till the bootstrap region covers the entire image block $\mathcal{B}'$. The final transformed model grid T_f^k is used to calculate the mean squared error (e_{mse}^k) for the sample S^k as:

$$e_{mse}^k = \frac{\text{tr} \left((LT_f^k - S)(LT_f^k - S)^T \right)}{\mathbf{1}_n^T L^T \mathbf{1}_\ell}, \quad (5.11)$$

where $L(i, j) = \begin{cases} 1, & \text{if } \{s_i, t_j\} \text{ is a correspondence} \\ 0, & \text{otherwise} \end{cases}$ is an $\ell \times n$ logical matrix, $\text{tr}(\cdot)$ 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 [99] the number of iterations (k) required to achieve a probability of success η is:

$$k = \frac{\log(1 - \eta)}{\log(1 - \epsilon^3)} \quad (5.12)$$

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 T_f^k with the minimum mean squared error e_{mse}^k is selected as the rigidly aligned grid $\mathcal{G}$ for the image block $\mathcal{B}'$.

Fig.5.4. The bootstrapped and least trimmed squares ICP algorithm. Two point sets are shown using orange dots and blue circles. (a), (b) and (c) are 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.

5.1.3 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 $\mathbf{g} \in \mathcal{G}$ 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 (5.7). 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. 5.5 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.

Fig. 5.5. 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 are marked with red dots and the detected modes are marked with blue circles.

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:

$$\hat{\mathbf{g}} = \mathbf{g} + \text{median} \left([(\mathbf{x}_1 - \mathbf{g}_1), \dots, (\mathbf{x}_j - \mathbf{g}_j)] \right), \quad (5.13)$$

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 $\text{median}(\mathbf{X})$ returns the $n \times 1$ median vector of the $n \times m$ matrix $\mathbf{X}$. A smoothness score ψ_{smooth} is then computed for each smoothed $\hat{\mathbf{g}}$ as:

$$\psi_{\text{smooth}} = \frac{1}{\text{tr}(\text{cov}([(\hat{\mathbf{g}}_1 - \mathbf{g}_1), \dots, (\hat{\mathbf{g}}_j - \mathbf{g}_j)]))}, \quad (5.14)$$

where $\mathbf{g}_i \in H(\mathbf{g}, 2x_{\text{gap}})$, $\hat{\mathbf{g}}_i$ is the smoothed version of $\mathbf{g}_i$, $\text{tr}(\cdot)$ returns the trace of a matrix and $\text{cov}(\mathbf{X})$ returns the $n \times n$ covariance matrix of a $n \times m$ matrix $\mathbf{X}$. The points in $\mathcal{G}$ whose corresponding ψ_{smooth} scores are lower than the median of the ψ_{smooth} scores are selected as the control points ($\mathcal{G}'$) 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 $\mathcal{G}' = [\mathbf{g}_1', \dots, \mathbf{g}_q']^T$, and their corresponding smoothed versions $\hat{\mathcal{G}}' = [\hat{\mathbf{g}}_1', \dots, \hat{\mathbf{g}}_q']^T$, an affine transformation matrix $\mathbf{A}$ is computed by solving the system of linear equations:

$$\hat{\mathcal{G}}_1' = \mathbf{A}\mathcal{G}_1', \quad (5.15)$$

where $\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & t_1 \\ a_{21} & a_{22} & t_2 \\ 0 & 0 & 1 \end{bmatrix}$, $\mathcal{G}_1' = \begin{bmatrix} \mathbf{g}_1' & \mathbf{g}_2' & \mathbf{g}_3' \\ 1 & 1 & 1 \end{bmatrix}$ and $\hat{\mathcal{G}}_1' = \begin{bmatrix} \hat{\mathbf{g}}_1' & \hat{\mathbf{g}}_2' & \hat{\mathbf{g}}_3' \\ 1 & 1 & 1 \end{bmatrix}$. $\mathcal{G}$ is then

transformed using $\mathbf{A}$ to generate $\hat{\mathcal{G}} = \mathbf{A}\mathcal{G}$. $\hat{\mathcal{G}}$ is then set as the new $\mathcal{G}$. The bootstrap region is grown and the process is repeated until the entire block $\mathcal{B}'$ has been covered by the bootstrap region.

5.1.4 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.

5.2 Micro-well Classification

Similar to gridding, micro-well classification is also performed by breaking the image into smaller blocks and processing each block independently. As noted above this process has two advantages: first, it makes the method robust to illumination changes in the image, and second, it allows for parallel processing of the blocks. Let $\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_n$ be n micro-well image regions in an image block, centered at grid centers $\mathcal{G} = [\mathbf{g}_1, \dots, \mathbf{g}_n]^T$. The objective of the micro-well classification is to assign each micro-well image $\mathbf{g}_i$ a label (also referred to as class) l_i which can be either positive or negative. A positive label indicates that the micro-well contains a bead and a negative label indicates that the micro-well is empty. The main challenge in this classification problem is that the visual appearance of the micro-wells varies significantly between and even within an image. Fig 5.6 shows a pair of micro-well images. It is difficult to predict just by looking at the pair which class they belong to. However, an informed guess could be made based on the brightness (intensity) of the micro-wells. The micro-well in Fig 5.6(a) has higher foreground pixel intensity than the micro-well 5.6(b), therefore one could be inclined to label the former as a positive and the latter as a negative micro-well.

Fig. 5.6: A pair of micro-well images from two different regions of the same image.

Fig. 5.7 shows two more pairs of micro-well images. Using a similar logic as above one could guess that the left images in these two pairs are negative samples while the right images are positive samples. However notice that the left image in the pair shown in Fig. 5.7(a) is the same image that is shown in Fig. 5.6(a) and the right image in Fig. 5.7(b) is the same image that is shown Fig. 5.6(b). While the images in Fig. 5.6(a) and (b) seem like they belong to positive and negative classes respectively, they are in fact negative and positive micro-well images. The two pairs in Fig. 5.7 are negative and positive micro-well images that are close to each other in the image, whereas

the two images in Fig. 5.6 belong to different image neighborhoods. This shows the large variations in the appearance of micro-wells across an images. It also highlights the importance of using local neighborhood information in during classification.

Fig. 5.7: Two pairs of micro-well images. Each pair has a negative image on the left and a positive image on the right. The positive and negative images in each pair are from image regions that are close to each other.

It is difficult even for humans to distinguish between the two classes of micro-well images in isolation. However, when presented as pairs of neighboring images, the contextual information (i.e. the appearance of the micro-wells relative to each other) makes it comparatively easier to identify which class the images belong to.

There are two popular approaches to represent objects: a feature based representation, where an object's features such as its area, circularity, etc. are used to characterize it, and a dis(similarity) based representation, where an object is characterized based on its relationship to some reference objects [213]. In the proposed method, a similarity-based representation is combined with object specific features. There are two reasons for choosing this approach: (1) as shown above it is easier to classify the micro-well images when another micro-well image is used as a reference and therefore similarity based representation is an effective way to describe the micro-well images, and (2) the deposition images can contain tens of thousands of micro-wells and only simple features (such as area, intensity, etc.) can be extracted efficiently enough for such large number of objects. Such simple features are not powerful enough to discriminate between the two classes of images on their own.

Henceforth, the similarity-based representations are referred to as similarity-based features and object's features are referred to as object-based features.

The proposed classification method begins by generating a similarity-based representation

of the micro-well images using a bank of template images. The relationships between the micro-well images in the similarity-based features space is captured using a k -nearest neighbor (k -NN) graph and a community detection algorithm [214] is used to extract clusters of strongly connected micro-wells. The clusters are merged further using agglomerative hierarchical clustering and at each merge a non-parametric test using object-based features is performed to identify positive clusters.

5.2.1 Template Banks

Based on visual analysis of the micro-well images it is apparent that the majority of clean micro-well images can be modelled using three shapes: (1) the Gaussian function, (2) the uniform distribution with circular or elliptical support, and (3) the 2-d torus or donut shape. However, the scale, rotation and deformation of these three basic shapes varies widely among the micro-wells. Nevertheless, because the shapes of the micro-wells are derived from these three basic shapes, template matching can be used to characterize them. To account for the variability within each shape when performing template based matching a positive template bank containing 104 template images and a negative template bank containing 118 template images are created.

Half of the positive template bank images (52 images) emulate 2-d Gaussian functions, while the other half consists of images where the foreground region has the shape of a uniform distribution with circular or elliptical support. A 2-d Gaussian function is usually described using two parameters, the mean and the covariance matrix, where the covariance matrix encodes the scale, orientation and shape features of the function. An equivalent parameterization is also possible by explicitly specifying the three features: orientation θ (the angle between the 1st principal component and the x -axis), scale σ_a (the standard deviation along the 1st principal component), and an elongation deformation factor $e_{ab} = \sigma_a/\sigma_b$ where σ_b is the standard deviation along the 2nd principal component. 48 positive template images emulating 2-d Gaussian functions are generated by combining 6 orientations ($\theta = 0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ$), 4 scales ($\sigma_a = 1s, \sqrt{2}s, 2s, 2\sqrt{2}s$), and 2 elongation deformations ($e_{ab} = 0.7, 0.85$). Additional 4 template images are generated at the 4 scales with $e_{ab} = 1$ to produce a total of 52 images. These 52 images are shown in Fig. 5.8. The parameter s is also a scaling parameter, whose value was determined by

fitting Gaussian functions to a set of manually labelled positive and negative micro-well images.

The remaining 52 positive template images with circular and elliptical foregrounds are created in a similar way, with the difference that for elliptical (circular) shapes the scale and elongation deformation are specified using the major and minor axis radii, r_a and r_b respectively, instead of σ_a and σ_b . The same combination of parameters as above is used to produce the 52 uniform distribution shaped positive templates. These 52 images are shown in Fig. 5.9.

The pixel intensities in the negative templates images follow a donut shaped function which is parameterized using four parameters: (1) major axis length of the inner ellipse r_{in} , (2) major axis length of the outer ellipse r_{out} , (3) elongation deformation $e_{ab} = r_{out_b}/r_{out}$, where r_{out_b} is the minor axis length of the outer ellipse, and (4) orientation θ (angle between major axis of outer ellipse and x -axis). The orientation of the inner ellipse is the same as that of the outer ellipse. The minor axis lengths of the inner and outer ellipses are $e_{ab} * r_{in}$ and $e_{ab} * r_{out}$ respectively. For each combination of r_{in} and r_{out} , the template bank consists of 13 images: 1 image with $e_{ab} = 1$, and 12 images generated by combining 6 orientations, $\theta = (0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ)$, and 2 elongation deformations, $e_{ab} = (0.7, 0.85)$. Nine combinations of r_{in} and r_{out} , representing different scales and thickness of the donut are used to produce a total of $(13 \times 9 =)$ 118 images. The negative template bank images are shown in Fig. 5.10.

Fig. 5.8: Gaussian shaped positive template bank images.

Fig. 5.9: Elliptical and circular positive template bank images.

Fig. 5.10: Negative template bank images.

While the scale parameter s was estimated from a small set of training samples, we still consider the proposed classification method unsupervised. It can be seen in Figs 5.8 and 5.9 that the template images account for a wide range of variations and are suitable for not only the microarray images used in this study but also for most other microarray images. Moreover,

matching a template with a micro-well image requires computing only a dot product (discussed below). Therefore, a much larger template bank with finer scales can also be envisioned.

5.2.2 Similarity-Based Feature Extraction

Given the vector of pixel intensities T_i of a template image $\mathcal{T}_i$ and the vector of pixel intensities P_j of a micro-well image $\mathcal{P}_j$, the normalized cross-correlation ncc_{ij} between the two images, at the center pixel is:

$$ncc_{ij} = \left\langle \frac{P_j}{\|P_j\|_2}, \frac{T_i}{\|T_i\|_2} \right\rangle \quad (5.16)$$

where $\langle \cdot, \cdot \rangle$ is the inner product and $\|\cdot\|_2$ is the L_2 -norm. Traditionally, when performing template matching using NCC, the template is centered over every pixel of the image and the NCC value at each pixel is computed to identify the best alignment between the template and the image. Here, however, the images are assumed to be aligned and the NCC is computed only at the center pixel of the image. In this case the NCC is simply the dot product of the normalized vectors. Since the normalized vectors can be pre-computed, during matching only a dot product has to be computed.

The NCC between the n micro-well images and the 222 template images are computed. For each positive and negative template image T_i , the ncc_{ij} scores are summed over all the P_j micro-well images to get the sum of scores, $sos_i = \sum_{j=1}^n ncc_{ij}$. The positive template with the highest sos_i among the positive templates and the negative template with the highest sos_i among the negative templates are selected as the prototype positive and negative templates. The similarity-based representation of the j^{th} micro-well image is the 2-d feature vector $[ncc_j^+ \ ncc_j^-]^T$, where ncc_j^+ and ncc_j^- are the NCC scores between the j^{th} micro-well image and the positive and negative prototype templates, respectively. Because the classification is performed separately for each image block (and therefore different prototype templates are selected for different image blocks), the variance in the images is limited and one prototype per class is enough to generate a similarity-based representation that can sufficiently discriminate the two classes.

To label the micro-well images as positive or negative, the 2-d similarity-based features are

clustered into two classes. The two features ncc^+ and ncc^- are however likely to be highly correlated as they both represent correlation with similar circular shapes. Indeed, this is evident in the data as well. Fig. 5.11(b) shows the scatter plot of the similarity-based features of the micro-wells in the image block shown in Fig. 5.11(a). The correlation between the features is clearly evident. When clustering such highly correlated data, the clustering can be dominated by the direction in which the data has a high variance. In Fig. 5.11(b) the positive and negative samples are represented using red and blue dots, respectively. As can be seen, the direction along which (when projected onto) the clusters are separable, is not the same as the direction in which the data has high variance. This problem can be further compounded when the number of positive samples is very small compared to the number of negative samples and the separation between the clusters is less obvious. There are two ways to deal with such data: the first way is to transform (normalize) the data so that the data has similar variance in all directions, the second approach is to use a distance metric like Mahalanobis distance that inherently takes into account the shape of the distribution. We take the former approach and use principal components analysis (PCA) to transform (rotate and scale) the data.

The principal components (PCs) of the data are computed as the eigenvectors of the covariance matrix. The data points are then projected onto the PCs and thus rotated (axis aligned). The rotation corrected data will have the highest variance along the x -axis. The new similarity-based features (PCs) are the projections of the data points onto the PCs and therefore are some linear combination of the original features ncc^+ and ncc^- . Finally, the data is scaled by dividing the new feature values by the standard deviation of the new features. The resulting normalized data will have the identity matrix as its covariance matrix. Fig. 5.11(c) shows the result of PCA based normalization of the data points shown in Fig. 5.11(b).

Fig. 5.11: Principal components analysis based normalization of the similarity-based features. (a) a small region of a bright field image, (b) similarity-based representation of the micro-wells in the region shown in (a) (the red dots are positive samples and the blue dots are negative samples), and (c) the data points in (b) after PCA based normalization.

5.2.3 Object-based Feature Extraction

Based on visual inspection of the micro-wells it can be concluded that two micro-wells, that are close to each other in the image and are not affected by noise or artifacts, are likely to belong to two different classes if: (1) one of the micro-wells has significantly higher foreground intensity than the other, or (2) if one of the micro-wells has a significantly larger hole in its foreground region than the other. It is important to note that only micro-wells that are close to each other in the image can be compared in this way, as foreground intensities and sizes of micro-wells vary across the image.

To extract the object based features the micro-well images are segmented using Otsu's

thresholding. For each micro-well image $\mathcal{P}_i$ the foreground regions are used to compute four features: (1) contrast (ct_i), computed as the ratio of median intensity of the foreground pixels to the median intensity of the background pixels, (2) percent hole (ph_i), computed as the ratio between the area of holes in the foreground region and the area of the foreground region, (3) the isoperimetric quotient of the foreground region (iq_i), and (4) the convexity of the foreground region (cx_i), computed as the ratio of the foreground area and its convex hull.

5.2.4 k -Nearest Neighbor Graph Construction

The 2-d similarity-based feature space is one of many possible representations of the data (albeit a good representation from the point of view of classification), and it is likely that the data points actually lie on some higher-dimensional, possibly smooth, surface (a manifold). To accurately partition the data therefore it is necessary to capture the true relationships, the geodesic distances, between the data points on this underlying manifold. The most common way to learn local neighborhood structures of the manifold is to use k -nearest neighbors. The rationale being that local neighborhood structures are preserved across various (approximate) representation of the true underlying manifold. That is two samples that are nearest neighbors of each other on the manifold are likely to be close to each other in some other (approximate) representation of the manifold as well.

Let $F^{cc}_{n \times 2} = [f_1^{cc}, f_2^{cc}, \dots, f_n^{cc}]^T$ be the PCA normalized data, where $f_i^{cc} = [\lambda_i^1, \lambda_i^2]^T$ is the i^{th} data point in the PCA normalized feature space. While several constructions of k -NN graphs are possible, here a mutual k -NN graph is used. Formally, a mutual k -NN graph is an undirected graph $\mathbb{N}^k = (F^{cc}, E^k)$, whose nodes are the data points f_i^{cc} and whose edges $E^k(f_i^{cc}, f_j^{cc})$ connect the data points f_i^{cc} and f_j^{cc} , if $f_i^{cc} \in H^k(f_j^{cc})$ and $f_j^{cc} \in H^k(f_i^{cc})$, where $H^k(f_i^{cc})$ is the list of k -nearest data points of f_i^{cc} . The distance between the data points is computed using Euclidean distance. The graph $\mathbb{N}^k$ captures the structure of the data points on the underlying manifold well, if the data points that are connected by an edge lie close to each other on the manifold. The notion of what is a “local” neighborhood however is subjective and is controlled by the parameter k .

Fig. 5.12: Community detection using k -NN graphs of varying sparsity. (a) and (b) are two microarray image blocks. (c) PCA normalized similarity-based representation of the micro-wells in (a). The data points are colored based on the clustering achieved by the community detection algorithm [214] using a k -NN graph constructed with $k = 249$. (d) PCA normalized similarity-based representation of the micro-wells in (b). The data points are colored based on the clustering achieved by the community detection algorithm using a k -NN graph constructed with $k = 95$. (e) the same data points shown in (c) but colored based on the clustering achieved by the community detection algorithm using a k -NN graph constructed with $k = 21$. (f) the same data points shown in (d) but colored based on the clustering achieved by the community detection algorithm using a k -NN graph constructed with $k = 6$.

Smaller values of k produce uneven manifolds and can cause over-partitioning of the data points. Whereas larger values of k produce smoother manifolds but risk introducing spurious and noisy edges between data points that lie far apart on the manifold, and likely belong to different classes. Numerous sparsification strategies have been proposed [215, 216] to remove spurious edges from k -NN graphs. In the proposed method, sparsification is achieved by pruning edge $E^k(f_i^{cc}, f_j^{cc})$ if:

1. $|ct_i - ct_j| > idr^{ct}$ or $|ph_i - ph_j| > idr^{ph}$ where idr^{ct} and idr^{ph} are the interdecile ranges of the contrast and percent hole features, or if
2. $\|\mathbf{g}_i - \mathbf{g}_j\|_2 > bw/2$ where $\mathbf{g}_i$ and $\mathbf{g}_j$ are the grid centers of the i^{th} and j^{th} micro-well images respectively, $\|\cdot\|_2$ is the L2-norm distance, and bw is the width of the image block.

The second condition removes edges between micro-well images that are far from each other in the image. The k -NN graph is used to perform label propagation based clustering in the next stage. Labelling a micro-well image based on other micro-well images that are far away from it is not desirable, hence edges between micro-wells that are far apart are pruned.

One of the challenges in the micro-well clustering problem is that the number of clusters inherently present in the data can vary and is not known a priori. For instance, when some of the micro-well images are affected by noise, they can form a cluster of their own. Even in clean images, the large number of negative micro-well images can be separated into more than one cluster. Fig. 5.12(a) shows such a scenario. Due to large variation in illumination the appearance of the negative micro-wells at the left and right extremities are quite different. Fig. 5.12(c) shows the PCA normalized features F^{cc} for the micro-well images in Fig. 5.12(a) where the negative micro-well images can be separated into two clusters. Another challenge in the clustering problem is that the number of data points in the positive cluster can vary from zero to a few hundred. Fig. 5.12(b) shows a sparsely populated (by beads) image block which has 21 positive micro-wells whereas the image block in Fig. 5.12(a) has 279 positive micro-wells. This makes defining a good local neighborhood very difficult. For sparsely populated image blocks the number of good (similar) neighbors can be much smaller than that of densely populated image blocks. The problem is

illustrated in Fig. 5.12: Figs. 5.12(c) and (e) show the clustering achieved on the same data by a community detection algorithm (discussed in the next section) using k -NN graphs constructed using $k = 249$ and $k = 21$ respectively. Figs. 5.12(d) and (f) show the two clustering achieved on another data by the community detection algorithm using k -NN graphs constructed with $k = 95$ and $k = 6$ respectively. For Fig. 5.12(b) $k = 95$ creates neighborhoods that are too large and the positive cluster is affected by spurious edges that produce incorrect labelling. The image block in Fig. 5.12(a) has 1656 micro-wells and the image block in Fig. 5.12(b) has 475 micro-wells. For the micro-wells in Fig. 5.12(a) $k = 249$, which is 15% of the data, is a good measure of local neighborhood, for the micro-wells in Fig. 5.12(b) $k = 6$, which is 1.25% of the data, is a good measure of local neighborhood.

Even though sparsification can help prune some noisy edges, there is no single measure of locality that works for all images. Therefore, in the proposed method the clustering is performed six times, using $k = 0.20n, 0.15n, 0.10n, 0.05n, 0.025n$, and $0.0125n$, where n is the number of micro-wells in the image block. In each iteration, a new k -NN graph is constructed and pruned, community detection is performed using the pruned graph, and the clusters found through community detection are analyzed to find a positive cluster.

5.2.5 Community Detection based Clustering

As discussed above, the true underlying number of partitions in the data are not known. The label propagation based community detection algorithm of Raghavan et al. [214] is therefore a natural choice for this problem, as it relies only on the network structure and does not require the number or size of clusters to be known a priori. Moreover, the algorithm has a near-linear running time, which is desirable because the clustering has to be performed six times for each image block.

The algorithm begins by assigning unique labels to each node in $\mathbb{N}^k$. In each iteration the algorithm picks nodes (data points) from $\mathbb{N}^k$ in a random order, and assign each node the label that is most popular among its k -nearest neighbors. The algorithm terminates when all nodes have the labels that the majority of their k -nearest neighbors have. During the initial iterations small dense

clusters with same labels begin to form. As these clusters become larger, their influence increases and they begin to attract more nodes. When two large cluster's boundaries come close they start competing with each other for the nodes. Finally, when stable cluster boundaries are found an equilibrium is reached and the algorithm terminates.

5.2.6 Positive Cluster Identification

Wilcoxon-Mann-Whitney U-Test

The clusters identified using the community detection algorithm are further clustered using agglomerative hierarchical clustering to produce a binary linkage tree. The hierarchical clustering is performed using Euclidean distance and complete linkage. The binary linkage tree is then traversed from top-down and at each split a Wilcoxon-Mann-Whitney (WMW) U-test is performed to check if either of the child nodes represents a cluster of positive micro-well images (samples). It is assumed that the cluster of all positive samples will be one single node in the linkage tree. Which means that in case that the positive samples were partitioned into multiple clusters by the community detection algorithm, these clusters merge with each other before merging with other clusters in the hierarchical clustering.

Given two clusters C_i and C_j , two independent WMW U-tests are performed to check if one of them is a cluster of positive samples. One of the tests is based on contrast while the other is based on hole size. Micro-well images that are affected by artifacts have non-circular foreground regions. Therefore, to exclude such micro-well images, only those micro-well images that have $iq_i > 0.85$ or $cx_i > 0.9$ are considered, when generating the samples to perform the WMW U-tests.

The WMW U-test is a non-parametric test for the null hypothesis that the two samples come from the same population. The alternate hypothesis being that the two samples come from different populations. When comparing the two clusters, we are however interested in evaluating only whether one of the clusters (say C_i) is a positive or a negative cluster, assuming that the other cluster (C_j) is a negative cluster. We refer to such an evaluation as C_i vs C_j . To test if C_j is a positive cluster a separate evaluation C_j vs C_i is performed, assuming C_i to be a negative cluster. The reason for this

will become evident later. Let $\mathbb{s}_i = \{s_i^1, s_i^2, \dots, s_i^{N_i}\}$ and $\mathbb{s}_j = \{s_j^1, s_j^2, \dots, s_j^{N_j}\}$ be two samples (of contrast or hole size depending upon the test being performed) drawn from micro-wells belonging to clusters C_i and C_j respectively. For the evaluation of C_i vs C_j based on contrast, the only outcome that we are interested in is whether the contrasts in sample $\mathbb{s}_i$ are significantly greater than the contrasts in sample $\mathbb{s}_j$. Whether the contrasts in the two samples are equal or whether the contrasts in $\mathbb{s}_i$ are less than the contrasts in $\mathbb{s}_j$ does not change our outcome. Therefore, a one-sided (right-tailed) WMW U-test is performed. A similar reasoning holds true for the percent hole feature ph_i as well. Except that we expect the percent hole to be less in the positive micro-well images compared to the percent hole in negative micro-well images. Therefore, for the percent hole feature a one-sided (right-tailed) WMW U-tests is performed.

While the WMW U-tests give us a principled way to compare the contrasts and holes of the micro-well images in the two clusters, several factors need to be considered before confidently concluding that one of the clusters is indeed a positive cluster, even if the result of the tests are significant. Factors such as the size of the two samples N_i and N_j , the significance level α , the power of the test Π , and the effect size d^{obs} must all be considered to meaningfully use the outcome of the test. A major challenge in this clustering problem is that the number of positive sample in a block can vary from zero to hundreds. Since, only well-segmented micro-well images are considered for the tests, the number of positive samples available for the tests can often be small. This presents several challenges because for small samples (<5 for instance) discerning the shape of the underlying distribution is difficult and estimates for mean and variance are not very meaningful. While the non-parametric WMW test is better suited for small sample sizes (than the paired t-test), the power of the WMW test is severely compromised when the sample sizes are small. The power (or sensitivity) Π of a test is the probability of correctly rejecting the null hypothesis when the alternative hypothesis is true. A lower power therefore means that some positive clusters will not be identified. At the same time the significance level α , is the probability of rejecting the null hypothesis when it is true, i.e. the probability of incorrectly identifying a cluster as a positive cluster when it is not. For the proposed method it is necessary to retain sufficient power while

evaluating the hypothesis at very low significance levels.

The left-tailed WMW U-test evaluates the following effect: the distribution of positive samples is shifted to the right compared to the distribution of the negative samples. The magnitude of the shift, taking into account the spread of the samples, is known as the effect size. For instance, the gap between the means of the two samples, taking into account the standard deviations of the samples, is an indicator of the effect size when the distributions are normal. For two samples to be considered significantly different at high significance level and power, their distributions should have minimal overlap and the gap between them or the effect size has to be large. Similarly, for smaller sample sizes, only weak assumptions can be made about the population distributions and therefore a larger effect size is required to achieve sufficient power at low significance levels. The effect size d^t required to retain sufficient power at a given significance level and for samples of a certain size can be computed if the distributions are assumed to follow normal distributions [217].

Unfortunately, no well-established method exists for computing d^t for non-parametric distributions, to the best of our knowledge. However, Lehmann [218] suggests that to retain the same power, significance level, and effect size as a Student's t-test, a WMW U-test requires 15% more samples. Therefore, we compute the minimum required effect size d^t , as the effect size required in a Student's t-test to retain power $\Pi = 0.99$ at significance level $\alpha = 0.001$ for samples of sizes $0.85N_i$ and $0.85N_j$ with standard deviations $\sigma_i = \sigma_j = \max(\sigma_i, \sigma_j)$ (Student's t-test assumes equal variance of the samples). Fig. 5.13 shows the required effect size d^t with changing sample sizes $N_s = N_i = N_j$ at significance level $\alpha = 0.001$ and power $\Pi = 0.99$, assuming equal variance ($\sigma^2 = 0.25$) of the two samples. To compute the observed (sample) effect size d^{obs} , we use the method proposed by McGraw and Wong [219] for non-parametric distributions:

$$d^{obs} = \frac{|\mu_i - \mu_j|}{\sqrt{\rho_1 \sigma_1^2 - \rho_2 \sigma_2^2}}, \quad (5.17)$$

where μ_i and μ_j are the means, σ_i and σ_j are the standard deviations of samples s_i and s_j respectively, and ρ_1 and ρ_2 are the proportion of samples i.e., $\rho_1 = N_i/(N_i + N_j)$, and $\rho_2 = 1 - \rho_1$.

When comparing two clusters C_i vs C_j first the observed effect size d^{obs} and the minimum required effect size d^t are computed, and only if $d^{obs} \geq d^t$ the exact WMW U-test is performed. The WMW U-test, because of the combinatorial nature, has a high running time complexity. Therefore, the effect sizes are used first to check for significant shift under the normal approximation. If a significant shift is found, the exact WMW U-test is performed using samples of size $N_i = N_j = 50$ (generated by further uniformly randomly sampling from $\mathcal{S}_i$ and $\mathcal{S}_j$ respectively). If the p -value of the exact WMW U-test is found to be significant at $\alpha = 0.001$, C_i is identified as a positive cluster. The tests for the two features, contrast and percent hole, are performed independently and if either one is found to be significant C_i is identified as a positive cluster.

Fig. 5.13: Effect size of Student's t-test to retain power $\Pi = 0.99$ at significance level $\alpha = 0.001$ for samples of equal size (N_s) and equal variance $\sigma^2 = 0.25$.

Hierarchical Clustering based Cluster Analysis

Fig. 5.14: An illustration of linkage trees generated by hierarchical clustering.

Consider the binary linkage tree shown in Fig. 5.14, which shows a hierarchical clustering

of 3 clusters (C_1 , C_3 and C_4). Each internal node C_i represents a cluster of data points generated by combining other clusters. The binary linkage tree is traversed from top-down and at each split, the two child nodes are compared with all the other nodes using the WMW U-tests. At the second split of the tree two clusters (C_1 and C_2) are produced. Two samples are drawn from the data points belonging to the two clusters following the method discussed above. The pair of WMW U-tests are performed to evaluate C_1 vs C_2 and if either of the two tests reveals that C_1 is a positive cluster, the tree traversal terminates. Otherwise, tests to evaluate C_2 vs C_1 are performed. If C_2 is found to be a negative cluster the traversal continues and the next split where cluster C_2 is partitioned into clusters C_3 and C_4 is considered. To test if C_3 is a positive cluster a sample is drawn from C_3 and another sample is drawn from all data points that do not belong to C_3 , i.e. from $C_1 \cup C_4$. Tests to evaluate C_3 vs $C_1 \cup C_4$ are then performed and if C_3 is found to be a positive cluster, the tree traversal terminates. Otherwise, tests to evaluate C_4 vs $C_1 \cup C_3$ are performed. If C_4 is also found to be negative then no positive micro-well images could be found in the block using the current k -NN graph.

Top down traversal of the tree ensures that the largest possible cluster of positive data points can be identified. Consider the scenario that C_4 is the true positive cluster and at the second split C_3 vs $C_1 \cup C_4$ is being evaluated. If a symmetric test is used where the null hypothesis is that either of the samples, from C_3 or $C_1 \cup C_4$, is significantly greater than the other, then $C_1 \cup C_4$ may be falsely identified as a positive cluster as it contains C_4 . Hence, two one-sided tests, the first evaluating C_3 vs $C_1 \cup C_4$ assuming $C_1 \cup C_4$ is a negative cluster and the second evaluating C_4 vs $C_1 \cup C_3$ assuming $C_1 \cup C_3$ is a negative cluster, are necessary to identify the true positive cluster C_4 .

The k -NN graph construction, clustering, and tree traversal steps are repeated six times using the six values of k . Each iteration may return a positive cluster, if found. The largest positive cluster obtained after the six iterations is identified as the true positive cluster. If in none of the iterations any positive clusters are identified then no positive micro-well images are present in the block.

An obvious question at this point is: why not use hierarchical clustering to directly generate

the binary linkage tree instead of using the community detection algorithm to generate initial clusters? The main reason for using the community detection algorithm is that it relies on the k -NN graph to generate the initial clusters and therefore by changing the k -NN graph structure a different initial clustering can be produced.

5.3 Experiments and Results

The proposed gridding and micro-well classification methods were 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 5.1. The High Resolution Dense (HRD) dataset contained relatively clean high resolution images with tightly packed wells, where the distance between the wells is 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.

Table 5.1: 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

All the images were acquired using a digital single-lens reflex (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 MATLAB and all the experiments were performed on a standard desktop computer with an Intel Xeon Hexa Core 3.2 GHz processor with 16GB RAM. Multithreading was used with up to 12 threads.

5.3.1 Bright and Dark Field Image Gridding Results

The proposed gridding 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.

Fig. 5.15 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. 5.15(c) and 5.15(e) are less sharp than the insets shown in Figs. 5.15(b) and 5.15(d). Fig. 5.16 shows the gridding achieved on the challenging images presented in Fig. 5.1.

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 d_{err} between the manually labelled well center and the corresponding well center detected using the proposed method.

Fig. 5.15. The gridding achieved on a dirty image from the LRD dataset using the proposed method. (a) the original bright field image. The blue boxes are the approximate regions of the image from which the insets shown in figures (b), (c), (d) and (e) are extracted. (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.

Fig. 5.16: The gridding achieved on the images presented in Fig. 5.1

Table 5.2 shows the mean, median and standard deviation of the 500 d_{err} values (25×2 labels for each image $\times 10$ 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 5.2 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 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 5.2: GRIDDING ACCURACY ACHIEVED ON THE FIVE DATASETS

Dataset	Statistic	d_{err} (pixels) [$d_{err\%}$]		
		Overall	Manual Label 1	Manual Label 2
HRD	Mean	<u>0.560 [3.73]</u>	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	<u>0.764 [5.09]</u>	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	<u>0.328 [3.64]</u>	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	<u>0.316 [5.26]</u>	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	<u>0.389 [6.48]</u>	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]

The receiver operator characteristic (ROC) curves of the proposed method are given in Fig. 5.17. 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. 5.18. The image shows a region of the grid where four blocks overlap. Fig. 5.18(a) shows the grids detected before the refinement. For each well, only one of the block's grid is shown. Fig. 5.18(b) shows the grid in the same region after refinement. The grid in Fig. 5.18(b) is smooth whereas the grid points in Fig. 5.18(a) belonging to the four blocks are distinguishable. In particular, the grid points in Fig. 5.18(a) belonging to the top-left corner block can be easily identified, as they are heavily skewed towards

Fig. 5.17. The distributions of d_{err} achieved using the proposed method on the 5 datasets. (a), (b), (c), (d) and (e) are the distributions of d_{err} for the HRD, HRB, CLUST, LRC and LRD datasets respectively. The blue and the red curves show 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.

the top right of the wells.

The run-times of the proposed gridding method are presented in Table 5.3. 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.

Fig.5.18. 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 (a) the top left block's grid points can be clearly seen as being skewed towards the top-right of the beads.

Table 5.3: RUNNING TIMES OF PROPOSED GRIDDING AND MICRO-WELL CLASSIFICATION METHODS

Task	Running Times	Dataset				
		<i>HRD</i>	<i>HRB</i>	<i>CLUST</i>	<i>LRC</i>	<i>LRD</i>
Gridding	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
Micro-well Classification	Avg. (secs)	65.41	39.58	27.91	50.83	43.33
	Std. Dev. (secs)	5.70	2.91	2.05	4.14	3.81

5.3.2 Micro-well Classification Results

The proposed micro-well classification method was also run five times for each image and the classification output was visually analyzed. Because of the non-deterministic nature of the community detection algorithm small differences in the classification results are expected. Nevertheless, visual inspection showed that the classification scheme was able to identify the correct positive cluster in all cases.

Fig. 5.19 shows the classification output achieved on three image blocks from the LRD dataset. The amount of artifacts in the image increases progressively from the top row to bottom row but the classification accuracy remains similar. The top and bottom rows of Fig. 5.20 show the classification outputs achieved on image blocks from the HRB and HRD datasets, respectively. Even though there is an extreme difference in density of beads between the two images, the algorithm is still able to classify the micro-wells accurately.

To objective evaluate the classification accuracy manual labeling was performed by two individuals who were familiar with the data and were involved in performing the biological experiments. To manually label positive and negative micro-wells rectangular regions (test blocks) were first marked by drawing boxes of a specific color in randomly chosen locations in the microarray images. All positive micro-wells inside these regions were then manually tagged by setting one pixel inside the micro-well region to a specific color. All micro-well regions inside the marked rectangular region that were not tagged were considered as negative micro-wells. The images were thresholded based on the specific colors associated with the test blocks and positive beads to automatically extract the positive and negative samples. Due to the differences in densities of beads in the images, to achieve sufficient number of positive samples, varying number of images (and test blocks) were labelled for each dataset. The number of images used, the total number of test blocks marked, and the total number of positive and negative micro-wells that were manually identified in each dataset are presented in the first four columns of Table 5.4.

The average classification rates obtained in five runs are presented in Table 5.4. The average number of true positive (TP), false positive (FP), true negative (TN), and false negative (FN) classifications for each of the dataset, are respectively presented in columns 5 to 8 of Table 5.4. The false positive rate, defined as $FPR = FP/(FP + TN)$, the false discovery rate, defined as $FDR = FP/(FP + TP)$, and the false negative rate, defined as $FNR = FN/(FN + TP)$, are also presented in columns 9, 10, and 11 of Table 5.4, respectively. The FPR is the rate at which empty micro-wells were detected as containing beads, while the FNR is the rate of not being able to detect a bead. The FDR shows what percentage of the micro-wells classified as positive by the proposed

method actually contained beads.

Overall the proposed method had a *FPR* of only 0.003, meaning that in an image with 50,000 micro-wells, approximately 150 empty micro-wells would be falsely classified as containing a bead. The number of false positives can be further reduced by comparing successive deposition images. If a bead is present in a micro-well in the n^{th} deposition image but not in the $(n + 1)^{\text{th}}$ deposition image, the bead was either falsely detected in the n^{th} micro-well image or was washed off during deposition of beads from the $(n + 1)^{\text{th}}$ batch. As more beads are added and the number of positive samples increases the *FPR* reduces, (the *FPR* for the HRD dataset for instance is the lowest) and therefore a *FP* detection in the n^{th} image can be correctly classified in later deposition images as more beads are added.

The case of false negative detection is however different. If a bead is missed in the n^{th} deposition image but detected in the $(n + 1)^{\text{th}}$ image, it will get associated with the $(n + 1)^{\text{th}}$ batch of beads. This will affect quantitation. Therefore, the minimum number of replicate beads required in each batch should be decided based on the expected *FNR*. As the image quality degrades the *FNR* naturally increases and more replicate beads are needed. For good images the *FNR* achieved using the proposed method is very low, for instance, the mean *FNR* for the HRD dataset is 0.01. Even in the worst case the mean *FNR* of the proposed method is 0.058 (for Clust dataset) which means that for an experiment with 30 replicate approximate only 2 beads per batch can be expected to be missed. Some of which may be detected in a later deposition image. The *FNR* therefore is small enough that even in the worst case, robust statistics (e.g. mean of the interdecile range) computed from the replicates will not be affected. The need to ensure that the *FNR* stays low is the primary reason why the proposed classification method tries to detect the largest cluster of positive samples.

A significant percentage of the misclassifications happen at the peripheral regions of the image where the largest illumination and shape variations (due to spherical aberration) are present. For the HRD dataset for instance the classifications in the central regions of the image are near

perfect. Therefore, better quantitation accuracies can be achieved by weighing the contributions of beads when computing the quantitation related statistics, such that beads in the peripheral regions of the image have lower contributions.

Table 5.4: MICRO-WELL CLASSIFICATION ACCURACY ON THE FIVE DATASETS

Dataset	No. of Labelled Images	No. of Labelled Blocks	No. of Positive Beads	No. of Negative Beads	TP	FP	TN	FN	FPR	FDR	FNR	Statistic
HRD	5	30	2772	10676	2743.80	16.20	10659.80	28.20	0.001	0.005	0.010	Mean
					2744	17	10659	28	0.001	0.006	0.010	Median
					2.28	2.38	2.38	2.28	0.000	0.000	0.000	Std. Dev.
HRB	7	50	390	6402	367	18.40	6383.60	23	0.002	0.047	0.058	Mean
					367	19	6383	23	0.002	0.048	0.058	Median
					4.58	2.70	2.70	4.58	0.000	0.006	0.011	Std. Dev.
CLUST	5	25	148	3155	141.60	15.80	3139.20	6.40	0.005	0.100	0.043	Mean
					145	14	3141	3	0.004	0.092	0.020	Median
					4.66	3.42	3.42	4.66	0.001	0.020	0.031	Std. Dev.
LRC	20	40	240	9466	229.09	21.90	9444.09	10.90	0.002	0.087	0.045	Mean
					229.69	21.30	9444.69	10.30	0.002	0.084	0.042	Median
					1.51	2.40	2.40	1.51	0.000	0.008	0.006	Std. Dev.
LRD	38	76	428	11628	414.43	81.96	11546.03	13.56	0.007	0.165	0.031	Mean
					415.23	82.76	11545.23	12.76	0.007	0.166	0.029	Median
					3.34	1.09	1.09	3.34	0.000	0.002	0.007	Std. Dev.

Fig. 5.19: Classification output on LRD images. Left column shows image blocks from the LRD dataset and the right column shows the classification output overlaid on those images. The micro-wells detected as positive are marked using a blue box. The amount of artifacts in the images progressively increases from top row to bottom row.

Fig. 5.20: Classification output on HRB and HRD images. The top-left image block is from the HRB dataset and bottom left image is from the HRD dataset. The right column shows the classification output overlaid on the images shown on the left. The micro-wells detected as positive are marked using a blue box. The two images have starkly different bead density but the proposed method is able to accurately classify positive beads in both images. The bottom image also has irregular illumination.

The running times of the proposed classification method are presented in Table 5.3. The running times are dependent mainly on the number of micro-wells in the image and is dominated by the time required to extract the object-based features. In fact approximately 50% of the time is spent in feature extraction. Better implementation of the feature extraction process (for instance GPU based implementation) can therefore significantly reduce the time required to process the images.

5.4 Discussion

To further evaluate the efficacy of some of the components of the proposed gridding algorithm three experiments were performed, in which the components are either replaced by some other technique or not used at all. In the first experiment the evolutionary algorithm based search for a clean image patch is not performed. Instead the grid parameters are estimated using entire image. In the second experiment, a random search was used instead of the evolutionary algorithm to search for a clean image patch. In the third experiment the Inner ICP based local optimization in Inner ICP-REJSAC is not performed. The percentage of times the gridding failed in the three experiment for the different datasets are presented in Table 5.5. The gridding was determined to have failed when 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.

Table 5.5: IMPROVEMENTS IN ACCURACY DUE TO EVOLUTIONARY ALGORITHM BASED GRID PARAMETER ESTIMATION AND ICP BASED LOCAL OPTIMIZATION WITHIN REJSAC

Dataset	Fraction of Times Gridding is Unsuccessful			
	Grid Parameter Estimation using Full Image	Random Search based Grid Parameter Estimation	REJSAC based Grid Fitting without Inner ICP based local optimization	Statistic
HRD	0.18	0	0.04	Mean
	0.16	0	0.05	Median
	0.05	0	0.02	Std. Dev.
HRB	0.37	0	0.02	Mean
	0.31	0	0	Median
	0.15	0	0.03	Std. Dev.
CLUST	0.54	0.06	0.68	Mean
	0.60	0	0.70	Median
	0.12	0.08	0.04	Std. Dev.
LRC	0.49	0	0.02	Mean
	0.47	0	0	Median
	0.07	0	0.02	Std. Dev.
LRD	1.00	0.12	0.33	Mean
	1.00	0.11	0.33	Median
	0	0.03	0.03	Std. Dev.

From the first experiment it can be seen that directly estimating the grid parameters using the entire image does not produce acceptable results for any of the datasets. This is primarily due

to the fact that almost every image has some level of non-linear distortion, particularly near the peripheral regions of the images. In the second experiment, random search using 200 samples, produced results similar to that of EA based search for HRD, HRB, and LRC datasets. For CLUST and LRD datasets EA based search produces better results. In both cases the increase in accuracy achieved using the evolutionary algorithm is significant. The EA based search is an important fail safe that improves the robustness of the proposed method at very little additional computational cost.

The third experiment showed that for all the datasets, gridding accuracy could be improved with the help of ICP based local optimization. For CLUST and LRD datasets there was a significant improvements. For these datasets the quick shift based mode detection output is often noisy, with many modes belonging to non-grid points. This results in REJSAC failing to find a grid that is close enough to the actual grid. Due to the noise and clustered beads in these two datasets, mean-shift based refinement gets stuck in a local minima and the gridding fails. Using the inner ICP based local optimization ensures that REJSAC aligns the ideal hexagonal grid as close to the true grid as possible using similarity transform.

Chapter 6 - Fluorescence Image Processing

Fluorescence or bioassay image processing involves two sub-tasks: image segmentation and image gridding. In this chapter we propose two novel methods for these two sub-tasks. The fluorescence image captures the light emitted back by the fluorophores attached to the beads. Therefore, only those beads whose corresponding target analytes are present in measurable quantities, are expressed in the image. Moreover, the intensity of the bead pixels is proportional to the quantity of the target analyte and therefore the bead intensities can vary widely. Because beads occupy only a fraction of the micro-wells and only a few of those are expressed in the bioassay image, it is not uncommon to have bioassay image that are relatively sparse. Fig. 6.1 shows a sparse fluorescence image. The grid structure is not easily discernable in such images and therefore the traditional gridding algorithms discussed in Chapter 4 and even the methods proposed in Chapter 5 cannot be directly used for such bioassay images.

Fig. 6.1. A sparsely populated fluorescence image where only a few beads are present. The micro-well grid structure cannot be readily discerned from such images.

While most traditional gridding approaches first establish the grid structure and then use the identified grid locations to simplify the analysis of individual bead regions, this is not possible for bioassay images. In bioassay image processing, the beads are first segmented and the bead's location information is then used to try to establish the grid. The proposed bead segmentation method is presented in Section 6.1 and the proposed bioassay gridding method is presented in Section 6.2.

6.1 Fluorescence Image Segmentation

As mentioned before one of the main challenges in bioassay image segmentation is the extreme difference in size and pixel intensity among beads. As shown in Fig. 6.2(a) and (b), beads associated with target analytes that are present in large quantities are highly expressed and can be over saturated. The bead in Fig. 6.2(a) for instance has a halo around it which makes it seem bigger than it actually is. Whereas beads associated with lowly expressed target analytes can have foreground intensities that are barely distinguishable from the background. Nevertheless, the underlying physical size of the beads is almost identical. The proposed image segmentation method uses this coherence in shape and size of the beads to identify and segment them.

Fig. 6.2. Regions of fluorescence image where bead segmentation is challenging. The two beads in (a) have a large difference in their intensities. The lowly expressed bead which is faintly visible is pointed to by the red arrow. In (b) a few beads are stacked with each other. The red arrow points to a pair of stacked beads where one of the beads is very lowly expressed. In (c) there are bright noise particles.

Another major challenge in the bioassay bead segmentation problem is that beads often cluster together. An example of such bead stacking is shown in Fig. 6.2(b). Traditional microarray segmentation techniques that perform a binary segmentation of local image regions often fail to

accurately separate the stacked beads. To overcome this problem, the proposed method thresholds the image at multiple levels and tries to identify coherent regions in the thresholded images.

The proposed segmentation method begins by breaking the bioassay image into smaller blocks. The proposed method for breaking the image into blocks (referred to as block segmentation) is discussed in Section 6.1.1. Each block image is then independently segmented (referred to as image segmentation) and the structure within each detected foreground region is encoded as a region tree. The image segmentation and region tree growing steps are discussed in Sections 6.1.2 and 6.1.3 respectively. The region trees are then considered as probabilistic graphical models which are used to identify the true foreground regions associated with the bead surfaces. To take into account the coherence in shape and size of the beads the inference on the tree-structured graphical models is performed within a clustering scheme. The proposed method for inferencing on the region trees is presented in Section 6.1.4 and the clustering scheme is presented in Section 6.1.5. Finally, a method for initializing the clustering process is presented in Section 6.1.6.

6.1.1 Block Segmentation

As in the case of deposition image processing, for bioassay image processing also, the image has to be first broken into smaller blocks to reduce the effects of varying illumination and to improve parallelizability. Breaking a bioassay image to regular rectangular blocks however can split a bead into multiple blocks. The proposed segmentation algorithm identifies beads based on region's shape properties and each block is processed independently. Therefore if a bead region is split into multiple blocks it may not be identified as belonging to a bead. To avoid this a more content aware method of splitting the image into blocks is required.

The proposed method uses a technique similar to the graph cut based superpixel lattice segmentation algorithm developed by Moore et al. [220]. To segment the image into blocks a regular rectangular grid is placed over the bioassay image and the grid lines are deformed so that they pass around beads rather than through them. To deform a grid line a regular 4-connected graph G_s is placed on an image strip around the grid line, and the max-flow/min-cut of the graph is identified as the deformed grid line. Fig. 6.3(a) shows the image strip (blue shaded region), around the top-

most horizontal grid line, over which the graph G_s is placed. The graph G_s is shown in Fig 6.3(e). Each node $v_i \in G_s$ is associated with a unary potential that is high if there is a bead nearby. Each edge $e_{ij} \in G_s$ is associated with a binary potential that encapsulates the probability of the edge e_{ij} passing through a bead. In the proposed method both v_i and e_{ij} are computed as the sum of the log transformed pixel intensities in a small rectangular region around them. The blue rectangular boxes in Fig 6.3(e) shows these regions. For horizontal image strips the nodes in the top and bottom row of G_s are connected to a source and sink respectively (as shown in Fig 6.3(e)). For vertical image strips the nodes in the left and right column of G_s are connected to source and sink respectively.

The max-flow/min-cut of the graph G_s minimizes the energy function

$$E_{cut} = \sum_{v_i \in G_s} S_i(b_i) + \sum_{e_{ij} \in G_s} S_{ij}(b_i, b_j) \quad (6.1)$$

where S_i and S_{ij} are the sums of log transformed pixel intensities in the small rectangular regions around node v_i and edge e_{ij} , and b_i is a binary label associated with node v_i . The max-flow/min-cut partitions G_s by assigning the binary labels b_i to all nodes. The path following the max-flow/min-cut passes through regions of the image that have low pixel intensity and therefore is likely to pass around the beads.

To recursively apply the graph cut method on images and generate superpixels, Moore et al. [220] proposed two modifications: (1) to discourage the max-flow/min-cut path from becoming too articulate such that the path can bend back on itself and have a zig-zag structure (like the dashed green path shown in Fig 6.3(e)) they propose increasing the energy potentials of the horizontal edges of G_s for vertical strips and increasing the energy potentials of the vertical edges of G_s for horizontal strips, (2) to prevent a horizontal (vertical) max-flow/min-cut path from intersecting a previously established vertical (horizontal) max-flow/min-cut path more than once, by bending over itself, they propose increasing the energy potentials of both the horizontal and vertical edges of G_s on and near the previously established max-flow/min-cut paths. In the proposed method the horizontal and vertical edges are scaled by a factor 1.25 when processing vertical and horizontal

strips, respectively, and the potentials of both vertical and horizontal edges near previously detected max-flow/min-cut paths are also scaled by a factor of 1.25.

Two important parameters of the graph cut based block extraction are the thickness of the image strips and the dimensions of the rectangular regions S_i and S_{ij} . The thickness of the image strips should be such that two adjacent vertical (or horizontal) dilated regions do not overlap, while ensuring that the image strip is wider than a bead. This is relatively simple to achieve since the width and height of the blocks are typically much larger than the diameter of a bead. However, image strips that are too thick will result in unevenly sized blocks and hamper parallelizability. Therefore, the thickness of image strips are set to $4r_{bead}$, where r_{bead} is an approximate estimate of the bead radius. r_{bead} is estimated as the median radius of circular connected components (isoperimetric quotient greater than 0.8) detected in binary images generated by thresholding (using Otsu's method) random regions of the bioassay image. The rectangular regions S_i and S_{ij} are set as squares with side length equal to $2r_{bead}$.

6.1.2 Image Segmentation

Each block identified in the bioassay image is independently segmented using k-means algorithm. The pixel intensity of a bead in a bioassay image is directly proportional to the number of fluorophores attached to the bead surface. Since the target analyte's concentration can vary widely, the concentration of the fluorophores and the pixel intensities also vary widely. As shown in Fig. 6.2(a) and (b), beads associated with highly expressed analytes can be over saturated and have a halo around them, making them seem bigger compared to beads associated with lowly expressed analytes. Nevertheless, the underlying physical size of the beads is almost identical. The objective in this step is to segment (possibly over-segment) the block image such that at least one region that approximately corresponds to the true size and shape of a bead is found.

The k-means clustering is performed on the 1-d pixel intensities of the image block to generate N_L clusters. If an image block is viewed as a 2-d surface such that the pixel intensities represent the height of the surface at that position, the clustering can be thought of as horizontal

planes (or layers) that cut the 2-d surface at specific heights. The second row in Fig. 6.4 shows the results of applying k-means clustering to the image regions in top row of Fig. 6.4.

Fig. 6.3. Breaking the bioassay image into blocks. (a) a grid placed on the bioassay image. The grid in (a) is deformed to obtain the grid shown in (b). (c) and (d) show close-ups of two regions of (b). (e) the graph structure used to deform the grid lines. The green (dotted) and red lines show two possible cuts. The red cut is preferred over the green cut because it has fewer turns. (f) a loop like formation of two grid lines that is undesirable.

Fig. 6.4. Shows regions trees grown from the image segmentation output

6.1.3 Region Tree Growing

Let $t_0, \dots, t_{N_L-1}$ be the $N_L - 1$ thresholds obtained using k-means clustering obtained in the previous section, ordered in increasing order. These thresholds are used to generate $N_L - 1$

foreground binary *layer* images, $L_0, \dots, L_{N_L-1}$, such that in the i^{th} binary *layer* image $L_i(x, y) = 1$ if $I(x, y) \geq t_i$, where I is the enhanced block image and (x, y) denotes the xy^{th} pixel of the image. That is, in the binary image L_i the foreground contains all pixels that have intensity greater than or equal to t_i in I . All pixels that have intensity less than t_0 in I are considered background pixels.

The connected components are identified in all the binary *layer* images to produce the set of N connected component regions $R = \{r_0^l, \dots, r_N^l\}$ where r_i^k denotes that the i^{th} connected component region was extracted from the k^{th} binary *layer* image. For each connected component region r_i^0 in L_0 , a region tree $\mathbb{T}_i$ is constructed with r_i^0 as its root. All the remaining nodes of $\mathbb{T}_i$ are connected component regions $r_i^k, k > 0$. Consider the i^{th} connected component region r_i^0 in the binary *layer* image L_0 and the same region in the subsequent binary *layer* image L_1 . There are two ways that the region r_i^0 can appear in L_1 : (1) the region can contain one or more connected components $r_{k1}^1, \dots, r_{kn}^1$ in L_1 , and (2) the region can be empty in L_1 . In the first case, the connected component regions $r_{k1}^1, \dots, r_{kn}^1$ are considered as the child nodes of r_i^0 in $\mathbb{T}_i$. In the second case the tree will have only a single node, the root node r_i^0 . The region trees $\mathbb{T}_i$ are grown by recursively comparing each connected component region in the i^{th} *layer* and the same region in the $(i + 1)^{\text{th}}$ *layer*.

The third row in Fig. 6.4 shows the regions trees grown from the segmentations shown in the second row of Fig. 6.4. Because the thresholds are ordered in increasing order, each child region in a tree is fully contained within its parent region. Moreover, no two regions in a tree can be overlapping. They will either be disjoint or one region will fully contain the other. Therefore, the trees can be grown efficiently, by uniquely labelling each connected component in each *layer*. For instance, let B^1 be a label image such that $B^1(x, y)$ contains the label of the connected component that the xy^{th} pixel belongs to in layer image L_1 . To identify r_i^0 's children in L_1 , the labels of all pixels contained in r_i^0 are looked-up in B^1 . The unique labels found in the look-up are the labels of the connected components that are r_i^0 's children in L_1 . In [221] Arteta et al. use MSERs [107] to detect cell. However, in our experiments, MSERs either produced sub-optimal detection, particularly of

lowly expressed beads, or took too long to process the image, depending on various parameter setting.

6.1.4 Inference on Region Trees

In this section, the region trees are used to construct graphical models, where each node represents the event that the region that is associated with the node is a foreground region. Given some evidence about how likely the nodes are of being the foreground regions, probabilistic inference on the trees can be performed to decode the optimal states (foreground or background) of each region. Given a tree $\mathbb{T}_i$, let the binary state variables associated with the k regions $r_i \in \mathbb{T}_i$ be $z_i, i = 1, \dots, k, z_i \in Z_{\mathbb{T}_i}$. Here, $z_i = 1$ indicates that region r_i is a foreground region and $z_i = 0$ indicates that it is a background region. In this section, we assume that the evidence is available in the form of the conditional probabilities $p(z_i = 0|r_i) = p(z_i^0|r_i)$ and $p(z_i = 1|r_i) = p(z_i^1|r_i)$. The method for computing these conditional probabilities is discussed in the next Section. Based on these conditional probabilities the Most Probable Explanation (MPE) of the state vector $Z_{\mathbb{T}_i}$ is computed. The MPE is the realization of $Z_{\mathbb{T}_i}$ that maximizes the joint probability distribution of the tree structured graphical model $\mathbb{T}_i$ [157]. For small trees the MPE, $Z_{\mathbb{T}_i}^*$, can be computed simply by enumerating through all possible realizations of $Z_{\mathbb{T}_i}$. For larger trees exact inference can be performed on tree structures in linear time using the max-product belief propagation algorithm [157].

To complete the tree structured graphical model $\mathbb{T}_i$ the costs associated with pairwise node labels $(z_p^j, z_c^k), j, k \in \{0,1\}$, have to be specified. Here, z_p and z_c are the states of regions r_p and r_c respectively and r_c represents a region contained inside r_p , i.e. r_c is a child node and r_p is its parent node in $\mathbb{T}_i$. For these binary labelling costs we use the formulation proposed by Arteta et al. [221]. The formulation is presented here for the sake of completeness. The cost associated with the labelling (z_p^0, z_c^1) is set to $p(z_c^1|r_c)$, the cost associated with the labelling (z_p^1, z_c^0) is set to $-\infty$, and the costs associated with the labellings (z_p^0, z_c^0) and (z_p^1, z_c^1) are set to 1.

The tree-structured graphical model captures the structure of the entire connected

component to which a region belongs and allows making a decision about the state of a region based not only on its own shape and size but also based on the shape and size of its children and parents. Here, size is represented using area whereas shape is represented using circularity and the region tree. The counts, relative arrangements, circularities, and areas of the regions contained within a given region provide useful information about the structure of the given region. The region tree efficiently encodes this structural information. In essence, every sub-tree in the region tree is a descriptor for the structure of a region. Setting the cost of labeling (z_p^1, z_c^0) to $-\infty$ adds a constraint, that whenever a region is selected all its sub-regions will also have to be selected. This is essential because without this constraint we would end up with foreground regions that have holes. Moreover, this constraint acts as a quality control mechanism. A region with high circularity and area similar to that of other bead foreground regions may not be identified as a foreground region, if its sub-regions have very odd shapes. Such odd shaped sub-regions can be due to noise or due to non-uniform binding between the conjugate, target analytes and/or fluorophores molecules. In either case, such beads are not a good indicators of the true concentration of the target analyte and should not be used for quantitation. This type of quality control mechanism can be particularly effective when many replicates are present.

6.1.5 Region Probability Estimation using Gaussian Mixture Model with Markov

Random Field Priors

In a supervised setting one could use labelled data to learn a regression or a classification boundary that separates the foreground regions from the background regions, to estimate the conditional probabilities $p(z_i^j | r_i)$. Arteta et al. [221] used a structured SVM to learn $p(z_i^j | r_i)$ for their cell segmentation problem. However, bioassay experiments produce large amounts of image data which can vary widely based on the imaging device (such as camera and microscope models) and image acquisition parameters (such as light setting, gain, focal length, etc.). A fully supervised scheme would require operator intervention whenever the settings change. Such a method is not suitable for a bioassay image segmentation method that is part of an automated bead encoding system. Moreover, unlike in the case of the micro-wells, the positive and negative classes of

fluorescence beads do not have distinctive visual appearances. Therefore, a template matching based scheme is also not suitable. While there is a lack of distinctive features that can be used to classify the regions, there is a lot of contextual information available in the form of coherence in shape and area of the beads. The proposed method uses these two simple features, circularity (computed as the isoperimetric quotient) and area of the regions, to cluster them.

A Gaussian mixture model based Markov random field (GMM-MRF) is used to capture the contextual information and identify the foreground regions. The GMM-MRF model has been widely used for image segmentation [222]. Here, the tree structured graphical model based inference described in Section 6.1.3 is incorporated into the GMM-MRF model. To describe this model we use the formulation and notation used by Nguyen [222].

Let $X_{N \times 2} = [x_1, \dots, x_N]^T$ be the feature matrix, where $x_i \in \mathbb{R}^2$ is the feature vector associated with region r_i . Recall that $z_i, i = 1, \dots, N, z_i \in Z$, is the binary latent variable, such that $z_i = 1$ indicates that region r_i is a foreground region and $z_i = 0$ indicates that it is a background region.

It is reasonable to assume that among the regions extracted for each bead, at least one region approximately represents the true bead region. Since the underlying physical shape and size of the beads are almost identical, a tight cluster of foreground regions that have similar area and circularity can be expected. Based on this assumption, the region's features, in the 2-d feature space, are modelled using a Gaussian mixture model (GMM) with two components. One component representing the tight cluster of true foreground regions (positive class) and the other component explaining the rest of the data (negative class). The GMM can be described as:

$$p(x_i|\theta, \Pi) = \pi_{i0}g(x_i|\theta_0) + \pi_{i1}g(x_i|\theta_1) \quad (6.2)$$

where g is a Gaussian distribution parameterized by the vectors $\theta_0 = (\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$ and $\theta_1 = (\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$, where $\boldsymbol{\mu}_0$ and $\boldsymbol{\mu}_1$ are the mean vectors and $\boldsymbol{\Sigma}_0$ and $\boldsymbol{\Sigma}_1$ are the covariance matrices of the two components respectively, and $\Pi = \{\pi_{ij}\}, i = 1, \dots, N; j \in \{0,1\}$ is the parameter set of prior

distributions π_{ij} that specify the probability of region r_i taking label j . The prior distributions follow the following conditions:

$$\pi_{i0} + \pi_{i1} = 1; 0 \leq \pi_{ij} \leq 1 \quad (6.3)$$

The GMM in (6.2) does not take into account the relationships between the regions; the structural information about the parent/child relationships is ignored. However, these relationships contain strong contextual information that can be useful. To incorporate this contextual information into the GMM model a Markov Random Field (MRF) prior, that influences the latent state z_i of a region based on the latent states of its connected regions, is used. The relationships between the regions is captured in a mutual NN graph, $G_{MRF} = (X, E)$, where $x_i \in \delta_j \Leftrightarrow x_j \in \delta_i$ and $x_i \notin \delta_i$. Here, δ_i is the neighborhood of x_i in the feature space and $E_{N \times N} = [e_{ij}]$ is a binary edge matrix where $e_{ij} = 1$ implies regions r_i and r_j are connected. In the proposed method, E is defined such that every region is connected to its parent, child, and sister regions. A MRF can then be defined using G_{MRF} , if and only if:

$$p(z_i^j | Z^j / z_i^j) = p(z_i^j | z_{\delta_i}^j) \quad (6.4)$$

Computing the likelihood of the MRF in (6.4) is impractical for arbitrary graphs as it would require enumerating all cliques in the graph. To overcome this problem Besag et al. [77] introduced the following pseudo-likelihood function:

$$p(Z|\beta) = \prod_{j=0}^1 \pi_{ij} = \prod_{j=0}^1 p(z_i^j | z_{\delta_i}^j; \beta), \quad (6.5)$$

where

$$\pi_{ij} = p(z_i^j | z_{\delta_i}^j; \beta) = \frac{e^{\beta u_i(j)}}{\sum_{k=0}^1 e^{\beta u_i(k)}}, \quad (6.6)$$

$u_i(j)$ is the number of neighbors of x_i with $z_i = j$ and β is a parameter whose higher values increases smoothness and encourages neighbors to have the same labels.

To estimate the parameters of this GMM-MRF model an iterative 2-step approach is used. In the first step, given an estimate of the parameters θ , the MPE estimate Z^* of latent state vector Z

is computed by independently estimating the MPE of each Z^k , where $Z^k \subset Z$ and $z_i \in Z^k \forall r_i \in \mathbb{T}_k$. To compute the MPE estimates using the tree-structured graphical models the conditional probabilities $p(z_i^j | r_i)$ is needed. This conditional probability is computed as the posterior

$$p(z_i^j | r_i) = p(z_i^j | X, \theta_j) = \frac{\pi_{ij} g(x_i | \theta_j)}{\sum_{k=0}^1 \pi_{ik} g(x_i | \theta_k)} \quad (6.7)$$

where the prior probabilities $\pi_{ij} = p(z_i^j | z_{\delta_i}^j; \beta)$ are computed using (6.6). This is the conditional expectation of the hidden variable z_i^j .

In the second step of the iteration the maximum likelihood estimate of the parameter vector θ is computed. Given the data X the likelihood and log-likelihood functions are:

$$L(\theta, \Pi) = p(X | \theta, \Pi) = \prod_{i=1}^N \sum_{j=0}^1 \pi_{ij} g(x_i | \mu_j, \Sigma_j) \quad (6.8)$$

$$\log L(\theta, \Pi) = \sum_{i=1}^N \log \left\{ \sum_{j=0}^1 \pi_{ij} g(x_i | \mu_j, \Sigma_j) \right\} \quad (6.9)$$

and the parameters θ_0 and θ_1 are updated as follows:

$$\mu_j^{(t+1)} = \frac{\sum_{i=1}^N h_i^j x_i}{\sum_{i=1}^N h_i^j} \quad (6.10)$$

$$\Sigma_j^{(t+1)} = \frac{\sum_{i=1}^N h_i^j (x_i - \mu_j)(x_i - \mu_j)^T}{\sum_{i=1}^N h_i^j} \quad (6.11)$$

where $h_i^j = p(z_i^j | X, \theta_j)$ computed using (6.7). The two steps alternate till either the state vector Z does not change between two iterations or till a maximum number of iterations is reached.

6.1.6 Community Detection based Initialization

To initialize the parameters of the GMM model the community detection based clustering described in Chapter 5 is used. Using a small value of k_{init} to construct the k -NN graph, small coherent clusters of regions are identified. The following cost function is then used to evaluate each

cluster:

$$cost_i = \frac{1}{p_{area} * \mu_{circ} * n_{utree}} \quad (6.12)$$

Where p_{area} is ratio of sum of foreground area of the regions in cluster i to the sum of area of all regions, μ_{circ} is the mean circularity of the regions in cluster i , and n_{utree} is the number of unique trees whose regions are present in cluster i . The regions in the cluster which minimizes the cost in (6.12) are identified as the initial foreground regions, and the parameters of the GMM are computed using them.

The proposed bead segmentation algorithm is further evaluated in Sections 6.3 and 6.4. In the next section we present a method for identifying the grid structure in the fluorescence image using the segmented bead regions.

6.2 Fluorescence Image Gridding

In this section a novel fluorescence gridding method is proposed. We particularly address the case of sparse fluorescence images, where the grid structure is easily not discernable and traditional gridding methods such as axis projection based methods cannot be used. In such cases, even when a grid can be established, it is may be impossible to ascertain whether the detected grid corresponds to the actual underlying micro-well structure (henceforth referred to as the true grid). Consider the two grids in Figs. 6.5(a) and (b) where the grid centers are shown as blue (hollow) dots and the beads are shown as red (filled) dots. The grids in both the images are fitted to the same set of beads and they both fit the beads accurately. It is impossible to ascertain which of these two grids if any, correspond to the true grid, without using any additional information. Some gridding methods (such as [53]) refer to this as the over-gridding problem and attempt to identify the sparsest grid that explains the data. However, there is no reason to believe that the sparsest grid is always the true grid, in fact it is often the contrary. Manufacturers often try to pack the micro-wells as tightly as possible to improve miniaturization. Therefore, to identify the true grid in sparse images some additional information about the physical properties of the chip is needed.

Fig. 6.5. Two images demonstrating the over gridding problem. The red (filled) dots indicate the segmented bead regions and the blue (hollow) dots indicate the fitted grid points. Notice that the pattern of red dots is the same in both the images but different grids can fit this pattern. Here, a fit is estimated as a transformation of an ideal hexagonal grid that minimizes the sum of squared distance between the segmented bead regions and their nearest grid points. In images with few segmented bead regions it is often possible to fit grids with smaller spacing between grid points (finer grids), than that of the true grid, as finer grids are more likely to result in smaller sum of squared distances. We refer to this as the over-gridding problem.

Here, we assume that approximate estimates of the physical distance between the micro-wells b_a and the physical radius of the micro-wells r_a are available. The median radius of the beads in the image space $\tilde{s}_I$ is known from the segmented spots. The beads are usually only slightly smaller than the micro-wells to ensure that they remain inside the micro-wells after deposition. Therefore, in the image space the micro-wells are assumed to have radius r_I that lies within the range $[\tilde{s}_I, 1.5\tilde{s}_I]$. Using b_a , r_a , and the range estimate for r_I , the range for the distance between micro-wells in the image space can be computed as $[d_I^{start}, d_I^{end}] = \left[\frac{\tilde{s}_I b_a}{r_a}, \frac{1.5\tilde{s}_I b_a}{r_a} \right]$.

The gridding algorithm begins by fitting an ideal hexagonal grid to the beads using two random beads selected from the bioassay image. The approximate micro-well distance d_I is used to identify pairs of points in the ideal grid that are likely correspondences of the two selected beads. This correspondence candidate generation step is discussed in Section 6.2.1. The candidate correspondences are then evaluated by aligning the ideal grid to the segmented beads using a method similar to the trimmed bootstrapped ICP method used in chapter 5. The alignment method is discussed in section 6.2.2. While in the case of deposition images rejection sampling could be performed, for bioassay image this is not possible. Therefore, all candidates have to be evaluated. To reduce the computation an early stopping strategy is proposed based on a hypergeometric test.

The hypergeometric test is also used to determine whether the gridding was successful. The hypergeometric test for estimating the probability that the established grid is the true grid is discussed in section 6.2.3.

6.2.1 Candidate Generation

Let the 2-d spatial coordinates of the centers of the bead foreground regions detected in the previous section be $Y = [\mathbf{y}_0, \mathbf{y}_1, \dots, \mathbf{y}_n]^T$. To initiate the gridding process two beads (say the m^{th} and n^{th} beads) are randomly selected as a basis pair. Let the Euclidean distance between $\mathbf{y}_m$ and $\mathbf{y}_n$ be d_{mn} . Consider an ideal grid $H = [\mathbf{h}_0, \mathbf{h}_1, \dots, \mathbf{h}_N]$, where $\mathbf{h}_i$ is the 2-d spatial coordinates of the i^{th} grid location. Further, let each concentric hexagon of grid points around the grid point $\mathbf{h}_0$ be indexed as $1, 2, \dots, k$. Our objective is to fit H to the bioassay image using beads $\mathbf{y}_m, \mathbf{y}_n$ and the input parameter range $[d_l^{\text{start}}, d_l^{\text{end}}]$.

Assuming that points $\mathbf{y}_m$ and $\mathbf{h}_0$ are aligned, to align the grid with the image, a suitable correspondence for $\mathbf{y}_n$ has to be found in H . If the distance between the micro-wells is indeed d_l^{start} , the match for $\mathbf{y}_n$ must be one of the grid points in the k^{th} concentric hexagon around $\mathbf{h}_0$, where $k = \frac{d_{mn}}{d_l^{\text{start}}}$. Therefore, the range of concentric hexagons that contains the match for $\mathbf{y}_n$ is $[k^{\text{start}}, k^{\text{end}}]$ where $k^{\text{start}} = \frac{d_{mn}}{d_l^{\text{start}}}$ and $k^{\text{end}} = \frac{d_{mn}}{d_l^{\text{end}}}$. The idea is presented in Fig. 6.6. The point $\mathbf{h}_0$ is the red dot at the center and the grid points in the 3rd, 4th and 5th concentric hexagons around $\mathbf{h}_0$ are the orange dots. In this example $k^{\text{start}} = 3$ and $k^{\text{end}} = 5$. Now, consider the two points $\mathbf{h}_i$ and $\mathbf{h}_j$ in Fig. 6.6. Assuming that H is sufficiently large, so that it covers the entire bioassay image, matching $\mathbf{y}_n$ to either $\mathbf{h}_i$ or $\mathbf{h}_j$ will result in the same gridding, as they are rotationally symmetric. Therefore, $\mathbf{y}_n$ only needs to be matched to the grid points that are highlighted by a green enclosing circle in Fig. 6.6. These are therefore the candidate matches of $\mathbf{y}_n$.

To find the optimal grid, given that both points $\mathbf{y}_m$ and $\mathbf{y}_n$ are true beads, these candidate matches need to be evaluated. Since only pairs of points are matched and only a similarity transform can be estimated to align H to the beads. While the similarity transform will align the grid points

and beads in immediate neighborhood of $\mathbf{y}_m$ and $\mathbf{y}_n$ quite well, it may not align the entire grid sufficiently well to the beads, due to local affine and non-linear deformations. Therefore, a trimmed bootstrapped ICP based grid refinement, similar to the one presented in Chapter 5, is used to optimize the alignment. The grid alignment is discussed in the next section.

Fig. 6.6. Candidate generation for matching an ideal hexagonal grid to the fluorescence image (see text for details).

6.2.2 Grid Alignment

Given the basis pair points $\mathbf{y}_m$ and $\mathbf{y}_n$ and their correspondences $\mathbf{h}_0$ and $\mathbf{h}_i$ a similarity transformation A is estimated by solving the system:

$$AH_{0i} - Y_{mn} = 0 \quad (6.13)$$

where $Y_{mn \times 2} = [\mathbf{y}_m^T, \mathbf{y}_n^T, 1_2^T]^T$, $H_{0i \times 2} = [\mathbf{h}_0^T, \mathbf{h}_i^T, 1_2^T]^T$, 1_2 is a 2×1 vector with all 1s and

$A = \begin{bmatrix} a_{11} & a_{12} & t_1 \\ a_{21} & a_{22} & t_2 \\ 0 & 0 & 1 \end{bmatrix}$ is a transformation matrix. The ideal grid H is then transformed using A to

get the coarsely fitted grid H' . The trimmed bootstrapped ICP method described in Section 5.1.2 is used to deform H' and generate H'' , the refined grid. In the case of deposition image gridding rejection sampling could be used and the number of candidates to be evaluated was smaller. Here a

large number of candidates have to be evaluated for each basis pair, most of which are likely to be false matches. Therefore, an early stopping criteria is used to detect false matches that are unlikely to produce a good alignment even if the ICP based refinement is continued further. The alignment proceeds as follows: the trimmed bootstrapped ICP is iterated a few times, such that the bootstrap region covers 25% of the image, an alignment quality check (described in the next section) is then performed to decide if the ICP based refinement should be continued further. If the quality check fails a new set of candidates is selected and evaluated.

6.2.3 Hypergeometric Test for Grid Alignment Quality Estimation

Let H^t be the current estimate of the aligned grid obtained after t iterations of the trimmed bootstrapped ICP based refinement. For each bead $\mathbf{y}_j$ in Y we define a *matching region* $M(\mathbf{y}_j, r_b)$, as a circular region of radius $r_b = 0.75\tilde{s}_l$ centered at $\mathbf{y}_j$. A grid point $\mathbf{h}_i \in H^t$ is considered to be matching a bead $\mathbf{y}_j$ if $\mathbf{h}_i$ lies inside $M(\mathbf{y}_j, r_b)$. In the alignment quality check a hypergeometric test is used to compute the probability of finding m or more matching grid point-bead pairs given Y and H^t .

The number of pixels that lie within the circular *matching regions* of the n detected beads in the image is $K = n * \pi r_b^2$. Let the total number of pixels in the image be N , the number of grid points in H^t that overlap the image be n_H and the number of grid points that have a match be k . Then the hypergeometric distribution can be used to compute the probability of finding at least k matches (successes) among n_H grid points (trials), given that there are K possible matching pixels (successes) among the N pixels (population) of the image, i.e.

$$p_k = P(X \geq k) = 1 - \sum_{i=0}^k \frac{\binom{K}{i} \binom{N-K}{n_H-i}}{\binom{N}{n_H}}, \quad (6.14)$$

where $\binom{a}{b}$ is a binomial coefficient. The trials n_H in this case are not random, as they are points of the same grid and are therefore spatially related. However, the above formulation still produces a good approximation of the quality of match. Fig 6.7(a) and (b) show two different grids overlaid on

the same bioassay image. There are 62 beads in the images shown in Fig. 6.7. The grid in Fig 6.7(a) has 11 matches and the grid in Fig 6.7(b) has 61 matches, using $r_b = 5$ pixels. The image has a total of 279312 pixels and 70136 matching pixels. The probability p_{11} for the grid in Fig 6.7(a) is 0.08 and the probability p_{61} for the grid in Fig 6.7(b) is < 0.0000001 .

The ICP based refinement is terminated if p_k is greater than 0.00001 otherwise it continues till the bootstrap region covers the entire image.

The bead pair $\mathbf{y}_m$ and $\mathbf{y}_n$ are randomly selected and therefore following the RANSAC scheme $k = \frac{\log(1-\eta)}{\log(1-\varepsilon^2)}$ iterations have to be performed using different randomly selected bead pairs, where η is the desired probability of success and ε is the number of inliers. Using $\eta = 0.9999$ and $\varepsilon = 0.75$, the value of k is 17 and therefore 17 iterations are performed. The probabilities p_k are again computed for the refined grids H'' obtained in the RANSAC iterations where trimmed bootstrapped ICP was performed on the full image (i.e. iterations where grid quality check was successful and early stopping was not performed). The refined grid for which the probability p_k is the lowest is identified as the true grid H^* . This true grid is then further improved using a thin-plate spline based non-linear refinement.

6.2.4 Thin-plate Spline based Non-linear Grid Refinement

To perform the thin-plate spline based refinement the image is broken into 25 blocks in a 5×5 grid pattern and q' random beads are selected from each block as control points. Let the selected $q = 25q'$ control points be $C = [\mathbf{c}_1, \dots, \mathbf{c}_q]^T$, where $C \subset Y$ and $\mathbf{c}_i = [x_i^c, y_i^c] \in \mathbb{R}^2$. Let the nearest neighbor grid point of $\mathbf{c}_i$ be $\mathbf{h}_i = [x_i^h, y_i^h] \in \mathbb{R}^2$, $\mathbf{h}_i \in H^*$. Now consider the points $T = [\mathbf{t}_1, \dots, \mathbf{t}_q]^T$, where $\mathbf{t}_i = [x_i^c, y_i^c, |x_i^c - x_i^h|] \in \mathbb{R}^3$. We refer to $|x_i^c - x_i^h|$ as the height of $\mathbf{t}_i$ since the point (x_i^c, y_i^c) is on the image plane. The thin-plate spline can then be interpreted as the 2-d surface with minimum bending energy that passes through all the $\mathbf{t}_i$ s. The height of this surface at any point $\mathbf{h}_i$ is the shift in x coordinate (Δx_i) needed to non-linearly deform the point. Another similar thin-plate spline over points $\mathbf{t}_i = [x_i^c, y_i^c, |y_i^c - y_i^h|]$ is used to compute Δy_i for $\mathbf{h}_i$. The

shifted grid points $\mathbf{h}'_i = [x_i^h + \Delta x_i, y_i^h + \Delta y_i]$ constitute the smoothly deformed grid. Since the 2-d surface passes through the $\mathbf{t}_i$ s corresponding to all the q control points, the control points perfectly aligned with their nearest neighbour grid points. The remaining grid points are smoothly deformed since the 2-d surface has minimum bending energy.

A detailed discussion of thin-plate spline based deformation can be found in [192].

Fig. 6.7. Two different grids aligned onto the same image. The grid in (b) produces a better fit. The grid in (a) can be rejected using the hypergeometric test. The detected beads are shown as blue dots.

6.3 Experiments and Results

The proposed bead segmentation method was evaluated using 202 fluorescence images. The images were acquired from 8 experiments with multiplexing ranging from 1-plex to 5-plex. The details of the experiments are presented in Table 6.1. The proposed bioassay gridding requires approximate estimates of the distance between micro-wells and the micro-well radius as input. These values were manually determined.

The bioassay experiments were performed using the Ayoxxa biochip (Ayoxxa, Singapore) and polystyrene beads from different providers. The deposition 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 bioassay images were acquired using an epifluorescent microscope (BX41, Olympus, Singapore) with a digital charge-coupled device (CCD) camera (Retiga-4000R, QImaging, Canada). The bioassay images were acquired at different zoom levels ranging from 10x to 30x. All the bioassay experiments were conducted by Ayoxxa, Singapore and the images were provided in kind cooperation.

The proposed bioassay segmentation and gridding methods were run five times for each image. The detected foreground regions and grid points were drawn on the image with super-pixel accuracy using a MATLAB script and the output was visually analyzed. Reasonable segmentation output was achieved for all images. Because the proposed segmentation method takes into account the shape of a foreground region as well as the shape of its sub-regions, it is biased towards selecting only those beads that present a somewhat consistent surface. Therefore, the method is likely to miss a few actual beads (i.e., produce false negatives) but it is unlikely to misclassify a noise region as bead (i.e., produce false positives). This is desirable from the point of view of quantitation, particularly when a large number of replicate beads are used. The proposed bioassay gridding method succeeds as long as reasonable estimates for r_a and b_a are provided.

The segmentations achieved on a number of different regions of bioassay images are shown in Fig. 6.8. The gridding achieved on a number of different sparse bioassay image regions are shown

in Fig. 6.9.

To objectively evaluate the bead detectability and gridding accuracies ground truth data was generated by manually labelling images. The ground truth data was generated in a manner similar to the manual labelling process discussed in Chapter 5. The ground truth data for evaluating detectability was generated using 32 randomly selected images. Within each image randomly selected rectangular regions were marked using a specific color. All beads within these rectangular regions (test blocks) were tagged by drawing rectangular regions of a specific color enclosing the beads. While care was taken to ensure that all beads present in the region are identified, some very lowly expressed beads could not be identified manually and were discovered through post-hoc analysis of the segmentation output.

The average foreground segment detectability results obtained in five runs of the proposed algorithm is presented in Table 6.2. In the case of bioassay images only positive labeled ground truth is available therefore the true positive rates $TPR = TP/(TP + FN)$ and the false discovery rates $FDR = FP/(TP + FP)$ are presented. Here, the true positives (TP) are defined as foreground regions that contain a manually labelled pixel, the false positives (FP) are defined as foreground regions that do not contain a manually labelled pixel.

(a)

(b)

(c)

Fig. 6.8. Segmentation output of various bioassay image regions. The detected foreground regions are overlaid (in red) over the segmented image.

Table 6.1: DETAILS OF THE DATASET USED TO EVALUATE THE PROPOSED BIOASSAY IMAGE SEGMENTATION AND GRIDDING METHODS

Experiment No.	Multiplex	No. of Bioassay Images Acquired	No. of Bioassay Images Manually Labelled for Bead Detectability Evaluation	No. of Bioassay Images Manually Labelled for Gridding Evaluation
1	1	42	7	10
2	2	8	2	5
3	3	6	2	5
4	3	10	2	5
5	4	74	9	10
6	4	8	2	5
7	5	28	4	5
8	5	26	4	5

Table 6.2: BEAD DETECTION ACHIEVED USING THE PROPOSED FOREGROUND SEGMENTATION METHOD

Experiment No.	TPR			FDR		
	Mean	Median	Std. Dev.	Mean	Median	Std. Dev.
1	0.941	0.933	0.056	0.008	0	0.021
2	0.878	0.878	0.021	0	0	0
3	0.937	0.937	0.062	0.045	0.045	0.045
4	0.914	0.914	0.014	0.050	0.050	0.050
5	0.960	1	0.045	0.007	0	0.020
6	0.944	0.944	0.055	0	0	0
7	0.980	1	0.033	0	0	0
8	0.902	0.901	0.028	0.031	0	0.054
Average	0.932	0.938	0.0395	0.017	0.011	0.024

As can be seen in Table 6.2, the proposed foreground segmentation method, as expected, has a very low *FDR*. In fact for many of the test blocks there were no false positive detections at all. Experiment 2 had the highest amount of bead clustering and therefore a few stacked beads could not be detected. The mean *TPR* was the lowest for this experiment at 0.87. In some cases when one of the beads in a cluster of stacked beads is highly expressed but the others are not, the halo of the bead that is highly expressed can make it difficult to segment the remaining beads even when multiple levels of thresholds are used. For Experiment 7, which had the cleanest images, the proposed method was able to achieve a *TPR* of 0.98.

The segmentation algorithm has two free parameters: (1) the number of clusters produced using k-means during over-segmentation, N_L , and (2) the number of nearest neighbors used to build the k -NN graph k_{init} in the GMM-MRF initialization step. The first parameter first parameter N_L

needs to be high enough so that lowly expressed beads can be separated from the background. However, increasing N_L beyond a certain value (above 8) has very little impact on the overall segmentation output. The effect of slightly changing the second parameter, k_{init} is also limited. Values for $k_{\text{init}} = 11, 15$, and 21 produced similar results. The small difference between the clusters obtained during initialization is easily overcome by the MRF prior. Moreover, even though the GMM parameter estimates can vary slightly between runs, because the robust MPE estimate is used the final segmentation result remains stable.

To generate the ground truth for evaluating the gridding, 50 randomly selected images were labelled. Beads were marked in the images by placing tight rectangular regions around them. Each of the 50 randomly chosen images were split into 25 blocks in a 5×5 grid pattern and one randomly selected bead was labelled in each block. All the labelling was done by an individual who is familiar with microarray image data and is involved in similar biological experiments. As in the case of deposition image gridding, the center of the marked rectangular region is assumed to be the center of the well (the reference well center). Both the absolute value of the distance between a reference well center and the grid point nearest to it, d_{err} , and the scale invariant $d_{\text{err}\%}$ are presented in Table 6.3. The resolution and scale of bioassay images within each experiment varies but only slightly. The median diameter of the beads in the bioassay images in Experiment 1 for instance ranges between 19 and 22 pixels. However, within each image the bead diameters can vary more, because beads of different batches are expressed at different levels. Therefore, the scale invariant value $d_{\text{err}\%} = \frac{d_{\text{err}}}{d_{\text{bead}}}$, is computed using d_{bead} which is the median diameter of the segmented bead foreground regions in the image.

(a)

(b)

(c)

(d)

Fig. 6.9. Grid detection output on various bioassay images. The detected grid centers are drawn as blue dots.

Table 6.3: BIOASSAY IMAGE GRIDGING ACCURACY BEFORE AND AFTER THIN-PLATE SPLINE BASED REFINEMENT

Experiment No.	d_{err} after TPS based Refinement			$d_{err\%}$ after TPS based Refinement			d_{err} before TPS based Refinement			$d_{err\%}$ before TPS based Refinement		
	Mean	Median	Std. Dev.	Mean	Median	Std. Dev.	Mean	Median	Std. Dev.	Mean	Median	Std. Dev.
1	2.32	2.35	0.16	0.131	0.133	0.009	8.22	8.19	1.06	0.467	0.465	0.060
2	2.57	2.57	0.17	0.106	0.106	0.007	6.6	6.45	1.99	0.272	0.266	0.082
3	1.79	1.78	0.14	0.113	0.113	0.008	6.47	6.46	1.33	0.412	0.411	0.084
4	2.24	2.23	0.17	0.174	0.173	0.013	6.64	6.41	1.18	0.516	0.498	0.091
5	1.97	1.93	0.19	0.093	0.091	0.009	7.63	7.76	0.85	0.363	0.369	0.040
6	1.97	1.95	0.11	0.115	0.113	0.006	5.7	5.66	0.8	0.332	0.330	0.046
7	1.87	1.86	0.14	0.102	0.102	0.007	5.58	5.57	1.05	0.306	0.305	0.057
8	2.13	2.19	0.22	0.119	0.122	0.012	8.47	8.52	1.98	0.474	0.477	0.111

In Table 6.3 it can be seen that after the TPS based refinement the $d_{err\%}$ values are close 0.17, i.e. 17% of the diameter of the bead for all experiments. This indicates the grid centers are well within the bead foreground region. Since the bioassay gridding does not affect quantitation and is only used for subsequent grid-based matching (discussed in the next chapter) the accuracy achieved is sufficient. Nevertheless, the value of $d_{err\%}$ is bound to be higher for bioassay images when compared to deposition images because the grid is estimated using bead regions and the bead regions may not always lie exactly at the center of the micro-well. The mean $d_{err\%}$ to median $d_{err\%}$ values for all experiments are similar indicating that there are no outliers. As can be seen in Table 6.3 the TPS based non-linear refinement leads to a very significant improvement in gridding. This is because in the case of bioassay image a single global affine transform is used to align the full grid to the image, unlike in the case of deposition images where local regions were separately gridded.

The running time of the proposed methods are presented in Table 6.4. The running time of the segmentation is mainly dependent on the number of beads in the bioassay image. Each experiment had images with multiple bead densities and therefore the average running times across the experiments is similar. The running time of the gridding step depends on the size of the range $[d_l^{start}, d_l^{end}]$. The number of candidates to be evaluated for each basis pair of points can grow very quickly with small increases in the range. However, the early stopping criteria introduced in Section 6.1.3 allows most unlikely grid alignments to be quickly rejected and prevents the running time from scaling too fast.

Table 6.4: RUNNING TIME OF THE BIOASSAY IMAGE SEGMENTATION AND GRIDDING METHODS

Task	Running Times	Experiment No.							
		1	2	3	4	5	6	7	8
Bioassay Image Segmentation	Avg. (secs)	43.582	48.429	39.173	39.621	40.964	42.557	37.778	47.624
	Std. Dev. (secs)	7.470	5.318	6.998	8.166	5.053	4.114	6.868	5.828
Bioassay Image Gridding	Avg. (secs)	30.160	28.485	30.639	28.185	35.097	31.377	33.763	28.347
	Std. Dev. (secs)	3.680	3.177	4.592	3.294	3.390	2.134	2.153	2.430

6.3.1 Comparison with Other Segmentation Algorithms

In this section the proposed bioassay image segmentation method and its components are evaluated against other existing alternatives. The bioassay image block segmentation method can be broken down into two steps: 1) k-means based over segmentation and 2) region tree and GMM-MRF based foreground segment identification. In the first step, the image is over-segmented to produce region trees and in the second step, foreground regions are identified from the set of over-segmented regions produced in Step 1. The output of the second step is therefore a binary image with bead region pixels set to 1. To distinguish between the two steps, comparisons evaluating the suitability of algorithms for over-segmentation are referred to as over-segmentation comparisons and comparisons evaluating the suitability of algorithms for detecting bead regions and generating binary segmentation output are referred to as detectability comparisons.

Over-segmentation can be achieved using a number of techniques. We evaluate five of the most widely used techniques: k-means clustering (KM) [70], maximally stable extremal regions [107] (MSER), mean-shift clustering [206] (MS), hierarchical clustering [195] (HC), and graph cut based segmentation [223] (GC).

Bead detection (binary segmentation) can also be achieved using k-means clustering with $k=2$ (KMBS), hierarchical clustering with two clusters (HCBS), and Otsu's thresholding [208] based segmentation (OT). Active contour based segmentation is also widely used for microarray image segmentation [44, 45, 73, 80]. Active contour based models require careful initialization to achieve good segmentation. When gridding is performed prior to segmentation the grid locations

can be used to initialize the active contours. However, in our problem the gridding is performed after bead segmentation and initializing the active contours can be challenging. Nevertheless, we evaluate the Chan and Vese active contour model [72] (CVAC), with manual initialization, to highlight the challenges in segmenting bead-based microarray images. The foreground segment identification step can also be thought of as a filtering step that identifies circular regions that are coherent in terms of their area and shape. Hough transform [52] (HT) is commonly used to detect circles and therefore it is also evaluated as an alternative for this step.

The quality of over-segmentation and detectability were qualitatively and quantitatively compared using the 32 manually labelled images. Detectability of the five methods was assessed in the same way as the detectability of the proposed algorithm was assessed in the previous section.

The over-segmentation achieved by the different methods on some of the image regions used for the evaluation are presented in Fig. 6.10. The top row of the figure shows four histogram stretched image regions from different bioassay images that were used for the qualitative comparisons. The first image region (first column) has an uneven background, the second image region (second column) has some beads that are lowly expressed, the third image region (third column) has clustered beads, and the fourth image region (fourth column) has a fiber near two of the beads. The subsequent rows of Fig. 6.10 show the over-segmentation achieved by the different methods.

MSER based over-segmentation results are presented in the second row of Fig. 6.10. MSER produces a hierarchy of regions, i.e. a region tree where each node represents a maximally stable region. A maximally stable region is a region whose area remains unchanged with significant changes in intensity. The color of each pixel in the output images in the second row of Fig. 6.10 represent the number of MSERs to which the pixel belongs. Hotter colors (red) indicate that a pixel belongs to many MSERs and cooler colors (blue) indicate the pixel belongs to few or no MSERs. MS based over-segmentation results are presented in the third row of Fig. 6.10. The pixels were clustered using MS in the 3-d feature space, where each pixel was represented using its intensity and its 2-d spatial location in the image space. Each segment is represented as a different gray level

Histogram stretched images

Maximally Stable Extremal Regions (MSER) detection output

Mean-shift segmentation output

Graph cut segmentation output

Hierarchical clustering output (10 clusters)

k-means clustering output ($N_L = 10$)

Fig. 6.10. The first row shows four histogram stretched images that were used for detectability and over-segmentation comparisons. The second to sixth rows (in that order) show the outputs of MSER, MS, GC, HC, and KM based over-segmentations applied on the images shown in the top row.

in the images. Note that while MSER, KM, and HC use only the image intensity as feature, MS and GC use both image intensity and the spatial location of pixels as features. MS based clustering can

be applied on 1-d pixel intensities too, however, better results were achieved using the 3-d feature space. GC on the other hand by design uses the spatial arrangement of pixels. In fact GC based segmentation can be viewed as a spatial smoothing applied to some initial labelling of the pixels. The results of GC based over-segmentation are presented in the fourth row of Fig. 6.10. Here, KM (with 20 clusters) is used to generate an initial labelling of the pixels. The over-segmentation results achieved using HC and KM based clustering of the pixel intensities, using 10 clusters for both, are presented in the fifth and sixth rows of Fig. 6.10 respectively.

MSER, MS, GC, HC, and KM produce similar segmentation results. The five methods are able to deal with the uneven background in the first image region. They are also able to segment the lowly expressed beads in the second image region into one or more segments. None of the methods are able to clearly segment individual beads from the clustered beads in the third image. For the fourth image region all five methods are able to identify segments representing the beads that are separated from the segments representing the fiber.

However, HC has a very high memory complexity and can only be run for image regions of size smaller than 500×500 pixels even with 16GB RAM. HC was therefore not considered for further evaluation. To further evaluate the quality of over-segmentation achieved using the remaining four methods the true positive rate (TPR) of over-segmentation is defined as the ratio of manually labelled rectangular bead regions that had at least one foreground region. During over-segmentation we are only concerned with the TPR because the subsequent foreground region detection step should be able to filter out false positives. As long as at least one bead region is detected as a foreground, the subsequent step should be able to identify it as a bead. Table 6.5 presents the TPRs of over-segmentation. All four methods have very similar accuracies with MSER and KM having only marginally higher accuracies than the other two. However, KM has the lowest running time complexity among the four methods. The running times of the four methods for different image sizes are presented in Fig. 6.11. MS and GC require nearly one and two orders of magnitude more time, respectively, than KM. For larger image regions MSER requires three times more time than KM. For images of dimension 1000×1000 pixels KM, MSER, MS, and GC

require 0.3, 1.12, 4.6, and 252.6 seconds respectively. Hence, we use KM for over-segmentation in the proposed bioassay bead segmentation method.

Table 6.5: ACCURACY OF VARIOUS OVER-SEGMENTATION METHODS

	TPR of Over-Segmentation		
	Mean	Median	Std. Dev.
MSER	0.996	0.996	0.003
MS	0.995	0.995	0.003
GC	0.994	0.993	0.003
KM	0.995	0.996	0.003

Fig. 6.11. The running times of the four over-segmentation methods. k-means clustering was performed for 10 clusters. Publicly available implementations of KM, MSER [224], MS [225], and GC [226] were used for the comparisons. All implementations are in C++.

The TPR and FDR of detectability achieved by the various algorithms are presented in Table 6.6. Hough transform requires binary images for circle detection, these binary images were generated using k-means and Otsu's thresholding. Both TPR and FDR are presented for the proposed methods, for Hough transform based circle detection on Otsu's thresholding output (HT + OT), and for Hough transform based circle detection on k-means based binary segmentation output (HT + KMBS). The TPR for HT is limited by the quality of binary segmentation output. While HT can achieve reasonable detectability in some of the images, using the same parameters, it detects large number of false positives in other images. This is evident from the high FDR of HT. The HT's parameters were optimized to produce the maximum accuracy across all 32 images. The proposed method out performs HT by a large margin. The detection outputs of Hough transform

and the proposed method on some of the images are presented in Fig. 6.12.

The FDR is not very meaningful for OT, KMBS, HCBS, and CVAC, since, they do not have any filtering mechanism. Any foreground region that has high intensity (such as noise, dirt) would be segmented as foreground regions in these methods. Therefore, only the TPR are presented for these methods. All four methods have much poorer TPR than the proposed method. This shows that binary segmentation is not sufficient to discriminate the lowly expressed beads from background. Hence there is a need to perform over-segmentation. OT, KMBS, and HCBS's outputs are nearly identical for most images. Whereas CVAC is slightly worse, even though it uses the manual initialization. The detection outputs of the four methods are presented in Fig. 6.13.

Table 6.6: COMPARISON OF BIOASSAY BEAD DETECTION ACHIEVED USING VARIOUS SEGMENTATION TECHNIQUES

	TPR			FDR		
	Mean	Median	Std. Dev.	Mean	Median	Std. Dev.
Proposed Method	0.932	0.938	0.0395	0.017	0.011	0.024
HT + OT	0.715593	0.730075	0.09704	0.249703	0.249518	0.1121
HT + KMBS	0.704324	0.715949	0.09386	0.256153	0.251386	0.11862
HCBS	0.790754	0.798952	0.08225			
OT	0.81182	0.817748	0.06937			
KMBS	0.802947	0.814474	0.07891			
CVAC	0.784085	0.802792	0.08679			

Histogram stretched images

Proposed method's segmentation output ($N_L = 10$, $k_{\text{init}} = 15$)

Hough transform circle detection output (sensitivity = 0.85) on Otsu's segmentation output

Hough transform circle detection output (sensitivity = 0.95) on Otsu's segmentation output

Hough transform circle detection output (sensitivity = 0.95) on k-means (k=2) segmentation output

Hough transform circle detection output (sensitivity = 0.98) on k-means (k=2) segmentation output

Fig. 6.12. The first row shows four histogram stretched images that were used for detectability and over-segmentation comparisons. The second row shows the segmentation achieved using the proposed method. The third and fourth rows show the output of Hough transform based circle detection on Otsu's threshold based binary segmentation output for different sensitivities. Here sensitivity is a parameter of Hough transform. Even at high sensitivities all the beads cannot be segmented. The fifth and sixth rows show the output of Hough transform based circle detection on k-means based binary segmentation output.

Manual labelling used to initialize Chan-Vese active contour based segmentation

Chan-Vese active contour based segmentation output

Otsu's thresholding output

Hierarchical clustering output (2 clusters)

k-means clustering output ($N_L = 2$)

Fig. 6.13. The first row shows four manually labelled histogram stretched images that were used for detectability and over-segmentation comparisons. The second row shows the output of CVAC based binary segmentation initialized using the manual labelling shown in the first row. The third, fourth and fifth rows show the binary segmentation outputs obtained using OT, HCBS, and KMBS respectively.

6.3.2 Discussion

The hexagonal gridding algorithm developed by Galinsky [60] (discussed in sections 3.1.1 and 3.7) interpolates the missing grid points using the neighboring detected grid points, based on the initial grid parameter estimates. The method relies on identifying a sufficient number of adjacent

grid points for the initial grid parameter estimation. In the proposed method, the Inner ICP-RANSAC based fitting of the ideal hexagonal grid to the detected points, identifies the missing points. The Inner ICP makes the proposed approach robust to small non-linear deformations and jitter noise. As explained in section 3.7 it is not reasonable to expect the presence of expressed beads in adjacent micro-wells in a bead based microarray. The proposed method therefore uses RANSAC, which is robust to missing points and outliers, to fit the ideal grid to the detected grid points.

A common approach for handling illumination variations during segmentation is to break the image into overlapping regular sized blocks and to segment each block independently. Such an approach is not suitable for our problem because we need to ensure that beads are not split into multiple blocks. This is due to the nature of the tree based segmentation algorithm which uses the structural information of the region to decide whether it is a bead region. When using regular sized blocks it is impossible to guarantee that beads will not be split into multiple blocks, particularly when bead density is high. Although, it is possible to ensure that the full region of a bead is contained in at least one block, when using overlapping blocks, the same bead region may be split in other blocks. This can negatively affect the segment tree based inferencing, which is done globally i.e. simultaneously for all beads in the image and relies on coherence of shape and area. Therefore, we use the graph cut based block segmentation method, which ensure that each bead is fully contained inside any one block. Because the graph cut is applied only on small image strips it has a low running time complexity.

Fig. 6.14 shows some outputs of over-segmentation achieved using k-means for different values of N_L . As the value of N_L increases the number of segmented foreground regions belonging to the beads also increases and greater amount structural information is captured for each region. However, even with $N_L = 7$ the lowly expressed bead regions can be segmented. By increasing N_L the accuracy of segmentation can be marginally improved, at the cost of an increased running time. However, even with $N_L = 7$ reasonable segmentation can be achieved.

Histogram stretched images

k-means clustering output ($N_L = 7$) k-means clustering output ($N_L = 10$) k-means clustering output ($N_L = 15$) Proposed method's segmentation output ($N_L = 10$, $k_{\text{init}} = 15$)

Fig. 6.14. The first row shows four histogram stretched images used for evaluation. The second, third and fourth rows show the k-means segmentation output for different values of N_L (the number of clusters). The fifth row shows the segmentation achieved using the proposed method using $N_L = 10$.

Identifying bead regions based on the coherence of their shapes and area, can be achieved by finding a coherent cluster of bead regions in the 2-d features space (circularity and area). However, the over-segmentation allows us to extract much more structural information, about each region, than just circularity and area. The count, the relative arrangement of the regions, the circularities, and areas of regions contained within a given region, capture important structural information. The region tree efficiently encodes this structural information. In essence, every sub-

tree starting at a node in the region tree is a descriptor for that region. To incorporate this structural information in to the segmentation process the tree based inference is used.

To initialize the GMM we first identify a set of foreground regions and estimate the initial parameters of the Gaussians representing the foreground and background regions from these foreground regions. GMM based clustering is sensitive to initializations. Therefore, during initialization a conservative approach is taken and a small set of regions that are very likely to be foreground regions are identified. This however, has a negative effect as well. Because the small set of foreground regions are not diverse enough (i.e. the covariance is small) the GMM clustering does not adapt to the data quickly. To alleviate this problem the MRF-GMM model is used. The Markov random field, over the data points in the feature space, has the effect of shifting the prior probabilities of the regions based on the labels of its parents, children, and siblings. Consider the scenario where a set of regions have been identified as foreground regions in the current iteration. Suppose the parents of these regions are also coherent in terms of their area and circularity. The MRF based label propagation increases the prior probabilities of the parent regions belonging to the class of foreground regions. Since, these parent regions are coherent in terms of area and circularity, they will also be relatively close to each other in the feature space. Their increased prior probabilities and their high density in the feature space will force the GMM parameters to adapt such that the posterior probabilities of these regions belonging to the foreground class increases in the next iteration. On the other hand if the parent regions are incoherent in terms of area and circularity then they will be randomly spread across the feature space. Therefore, even with higher prior probabilities, the GMM parameters will not be substantially affected.

The fluorescence gridding method developed here addresses the scenario where the image is so sparse that gridding can be impossible without some knowledge about the physical properties of the chip. This scenario has not been widely studied or addressed by existing gridding algorithms. Possibly because such sparse images are not common in spotted microarray experiments. In bead-based microarrays, however, such images are quite likely, given that some experiments involve only a few hundred beads while a well can have over hundred thousand micro-wells. The hypergeometric

test based gridding quality estimation provides an elegant heuristic for identifying this scenario, so that the bead decoding strategy can be adapted to handle it. That is, when gridding is unsuccessful, affine invariant matching can be used to decode the bead identities instead of grid based matching.

Chapter 7 - Point Pattern Matching for Bead Encoding

In this chapter two methods are proposed for finding bead correspondences between the deposition and bioassay image beads. The problem is first formulated as a point correspondence problem and an affine invariant point pattern matching method is developed in Section 7.1. The affine invariant matching method is capable of handling large amounts of occlusions and outliers and large affine deformations. It is also highly parallelizable and allows matching of large point sets (up to 30000 points).

A second matching method based on the grid information is proposed in Section 7.2. The grid-based matching method can be used when bioassay image gridding is successful. The method can be used to efficiently match very large point sets (50000 and more).

Given a bioassay image and a deposition image, our objective is to find the location of the bioassay image beads in the deposition image. We want to find a geometric transformation for the bioassay image that transforms the bioassay image and aligns it with the deposition image, in a way that the number of bioassay and deposition image beads that coincide with each other is maximized. The problem of finding this image transformation is formulated as a point pattern matching problem.

In the context of this matching problem, the deposition images are referred to as template images as they are acquired first, during bead deposition and stored in a database. While the bioassay image is referred to as a query image. Similarly, the 2-d spatial coordinates of beads detected in the deposition image are defined as the template points, and the 2-d spatial coordinates of beads detected in the bioassay image are defined as the query points. Throughout this chapter the terms deposition image/point and template image/point, and the terms bioassay image/point and query image/point are used interchangeably.

7.1 Affine Invariant Point Pattern Matching

Let $X_{2 \times m} = [x_1, x_2, \dots, x_m]$, $x_i \in \mathbb{R}^2$ be the spatial coordinates of beads detected in a

bioassay image and $Y_{2 \times n} = [y_1, y_2, \dots, y_n]$, $y_i \in R^2$ be the spatial coordinates of beads detected in a deposition image. Our objective is to find a binary correspondence matrix $P_{m \times n}$, whose element $p_{ij} = 1$ indicates that point x_i corresponds to point y_j .

Now if $m = n$ and P is an identity matrix, the relation between X and Y can be defined as:

$$Y_a = AX_a \quad (7.1)$$

where $Y_{a_{3 \times n}} = [Y^T, 1_n^T]^T$, $X_{a_{3 \times n}} = [X^T, 1_n^T]^T$, 1_n is an $n \times 1$ row vector with all 1s and $A = \begin{bmatrix} a_{11} & a_{12} & t_1 \\ a_{21} & a_{22} & t_2 \\ 0 & 0 & 1 \end{bmatrix}$ is a transformation matrix. Therefore, given X and Y the affine transformation

between them can be estimated through the least squares solution of the system:

$$AX_a - Y_a = 0 \quad (7.2)$$

i.e.,

$$\underset{A}{\operatorname{argmin}} \quad 1_n \operatorname{tr}((AX_a - Y_a)^T (AX_a - Y_a)) \quad (7.3)$$

where $\operatorname{tr}(M)$ is the trace of M .

In the matching problem however, the correspondences are not known and so we must evaluate and select a subset of all the possible correspondences. This objective can be defined as:

$$\underset{P, A}{\operatorname{argmin}} \quad 1_n d((AX_a P - Y_a P^T)^T (AX_a P - Y_a P^T)) \quad (7.4)$$

The general case where $m \neq n$ will be handled later. The optimization problem in (7.4) is hard because there are $n!$ possible permutation matrices, but given A , estimating P reduces to the well-known assignment problem [227]. Therefore the matching is done in two steps. First local patterns are matched to generate a set of candidate (or likely) estimates of A . These candidate A s are further evaluated to find the best A . Then using the best A the assignment problem is solved to find the bead correspondences.

7.1.1 Sampling

Bioassay images can have thousands of beads. To reduce the complexity of the matching algorithm we use at most $m_s = \max(0.1n_b, 200)$ bioassay beads for matching, where n_b is the total number of beads used in the experiment. We assume that the beads are deposited following a uniform distribution, which is typical for most bead deposition mechanisms. Given m uniformly distributed beads the area (a_s) that will contain m_s beads is calculated and all the beads found in a central region of the image with area a_s are extracted as sample beads. If the bioassay image has fewer than m_s beads, all the beads are used as samples.

Notice that the number of beads in the bioassay image may be fewer than n_b , for instance when all the batches do not produce a positive response. Nevertheless, as long as the total number of positive control beads used (n_c) is large, we expect to find sufficient positive control beads for matching within the m_s sampled beads. If all the batches produce positive responses, the expected number of control beads within the sample will be $0.1n_c$, and in all other cases we will have greater than $0.1n_c$ control beads among the m_s sampled beads. Henceforth, the spatial coordinates of the sampled beads is denoted as $X^+_{2 \times m_s}$ to distinguish them from the spatial coordinates of all the bioassay beads X .

7.1.2 Local Feature Extraction

Four Point Affine Invariants

Given four non-collinear points on a plane, it is always possible to form a pair of intersecting non-consecutive line segments. An example is shown in figure 7.1(a). Under affine transformation, the ratio of distances between co-linear points is preserved. Therefore, the two ratios $r_1 = \frac{p_0 p_e}{p_2 p_e}$ and $r_2 = \frac{p_1 p_e}{p_3 p_e}$ computed using the point of intersection p_e of the line segments $p_0 p_2$ and $p_1 p_3$ are invariant under affine transformation. These ratios form a descriptor for the pattern formed by the points p_0, p_1, p_2 , and p_3 . We call this descriptor the 4-points affine invariant (FPAI) feature.

Fig. 7.1. Invariant ratios extracted from 4 points. Two patterns of four points that are equivalent under affine transformation are shown in (a) and (b). The pattern in (c) is obtained from the pattern in (a) by flipping points p_0 and p_2 . An ordered arrangement of the points from the perspective of point p_m is shown in (d) and an ordered arrangement of the points from the perspective of point p_i is shown in (e). Notice that the cross product between the vectors $\vec{p_m p_i}$ and $\vec{p_m p_j}$ in (d) and between the vectors $\vec{p_i p_m}$ and $\vec{p_i p_n}$ in (e) are positive.

The ratios r_1 and r_2 extracted from the point patterns in Figs. 7.1(a) and (b) will be same as the point patterns are equivalent under affine transformation, i.e. one pattern can be obtained by applying an affine transform to the other. In [228] the ratios are computed as $r_1 = \frac{\max(p_0 p_e, p_2 p_e)}{\min(p_0 p_e, p_2 p_e)}$ and $r_2 = \frac{\max(p_1 p_e, p_3 p_e)}{\min(p_1 p_e, p_3 p_e)}$. The ratios r_1 and r_2 thus extracted from the point patterns in Figs. 7.1(a) and (c) will be same, even though the pattern in 7.1(c) is obtained from the pattern in 7.1(a) by flipping points p_0 and p_2 . These ratios are therefore invariant to the ordering between points p_0 and p_2 , and between points p_1 and p_3 . However, the chances of obtaining a flipped microarray image is quite low in the bead pattern matching problem. Therefore, some ordering of the points can be considered to make the ratios more informative (in terms of the bead matching problem) by sacrificing the invariance to flipping transform. Any such ordering however has to preserve rotation invariance.

Given four points and a pair of directed lines formed by them, the following ordering convention is used to describe the points: $\{p_s^{L_1}, p_s^{L_2}, p_e^{L_1}, p_e^{L_2}\}$, where $p_s^{L_1}$, $p_s^{L_2}$, $p_e^{L_1}$, and $p_e^{L_2}$ are the start of the primary line (L_1), the start of the secondary line (L_2), the end of the primary line (L_1), and the end of the secondary line (L_2) respectively. Determining the ordering of four points therefore requires identifying one of the lines as the primary line L_1 and deciding the direction of the two lines. Given four points p_m, p_n, p_i and p_j , such that p_m and p_n form L_1 , and p_i and p_j form

L_2 , and given that L_1 is directed from p_m to p_n , the direction of L_2 can be fixed such that the cross product between the vectors $\overrightarrow{p_m p_i}$ and $\overrightarrow{p_m p_j}$ is positive. The direction of L_2 (i.e., the order of the points p_i, p_j) can therefore be decided in a rotation invariant way given the direction of L_1 . There is no rotation invariant way of fixing the direction of L_1 or of deciding which of the two lines will be considered as L_1 . Therefore, in order to achieve rotation invariance while matching, four descriptors (formed by combining the two choices of L_1 and its two directions) have to be used. Each descriptor can be viewed as representing the arrangement of the four points from the perspective of one of them. Figs. 7.1(d) and (e) for instance show the same pattern from the perspective of point p_m and p_i respectively. The primary line L_1 is shown as solid line and the secondary line L_2 is shown as dashed line. The corresponding point ordering in Figs. 7.1(d) and (e) are p_m, p_n, p_i, p_j and p_i, p_j, p_m, p_n respectively.

The two lines L_1 and L_2 can intersect with each other in nine ways, depending on where the point of intersection is. These are shown in Fig. 7.2. We refer to these as the nine *intersection types*. Notice that in each *intersection type* presented in Fig. 7.2 the direction of L_2 is such that cross the product between the vectors $\overrightarrow{p_0 p_1}$ and $\overrightarrow{p_0 p_3}$ is positive. The *intersection type* of two lines is also invariant under affine transformation. The ratios r_1 and r_2 are computed differently for each intersection type to ensure that they are within the range $[0,1]$. These are presented in Table 7.1.

As discussed above, to represent a set of four points, four descriptors are used. Each descriptor $[t, r_1, r_2]$, consists of the *intersection type* t , and the two ratios r_1 and r_2 , and describes the arrangement of the four points from the perspective of one of them. Extraction of these descriptors from point neighborhoods is discussed in the next section.

Fig. 7.2. The nine intersection types of two directed lines.

FPAI Feature Extraction

Consider the point pattern in Fig. 7.3 where the 8-nearest neighbor points of p_0 are used to describe the local neighborhood around it. FPAI features are extracted by considering each line from p_0 to its NNs, such as line p_0p_2 , as primary lines (shown as solid lines in Fig. 7.3) and all lines between the NNs, such as line p_1p_3 , as secondary lines (shown as dashed lines in Fig. 7.3). Considering p_0p_2 as the primary line 21 FPAI features can be extracted using the 21 lines formed between the 7 remaining NNs. Therefore, for the neighborhood centered on point p_0 a total of 168 FPAI features can be extracted using the 8 primary lines formed with the 8-NNs and the 21 FPAI features extracted per primary line. Similarly, 168 FPAI features will be extracted for each of the 9 points, considering each one at the center, producing a total of 1512 FPAI features.

Table 7.1: RATIOS r_1 AND r_2 FOR THE NINE INTERSECTION TYPES SHOWN IN FIG. 7.2

Intersection Type	r_1	r_2
(a)	$\frac{p_2 p_e}{p_2 p_0}$	$\frac{p_1 p_e}{p_1 p_3}$
(b)	$\frac{\max(p_0 p_e, p_2 p_e)}{\min(p_0 p_e, p_2 p_e)}$	$\frac{p_1 p_e}{p_1 p_3}$
(c)	$\frac{p_0 p_e}{p_0 p_2}$	$\frac{p_1 p_e}{p_1 p_3}$
(d)	$\frac{p_2 p_e}{p_0 p_2}$	$\frac{p_3 p_1}{p_3 p_e}$
(e)	$\frac{\max(p_0 p_e, p_2 p_e)}{\min(p_0 p_e, p_2 p_e)}$	$\frac{p_3 p_1}{p_3 p_e}$
(f)	$\frac{p_0 p_e}{p_0 p_2}$	$\frac{p_3 p_1}{p_3 p_e}$
(g)	$\frac{p_0 p_e}{p_0 p_2}$	$\frac{\max(p_1 p_e, p_3 p_e)}{\min(p_1 p_e, p_3 p_e)}$
(h)	$\frac{p_2 p_e}{p_2 p_0}$	$\frac{\max(p_1 p_e, p_3 p_e)}{\min(p_1 p_e, p_3 p_e)}$
(i)	$\frac{\max(p_0 p_e, p_2 p_e)}{\min(p_0 p_e, p_2 p_e)}$	$\frac{\max(p_1 p_e, p_3 p_e)}{\min(p_1 p_e, p_3 p_e)}$

Fig. 7.3. Line Context feature extraction.

Each line in Fig. 7.3 (solid and dashed) will be considered as the primary line exactly twice. For instance line $p_1 p_3$ will be considered as the primary line when the neighborhood is centered on point p_1 , the same line (but in opposite direction) will also be considered as the primary line when the neighborhood is centered on point p_3 .

In general, for a point pattern of N points described using a k -NN graph, N neighborhoods

have to be considered with one neighborhood centered on each point. In each neighborhood, k primary lines can be formed using the k NNs, and for each primary line $\frac{(k-1)(k-2)}{2}$ secondary lines can be formed using the remaining $(k-1)$ NNs. Therefore, a pattern of N points and a local neighborhood structure of k -NNs is represented using $\frac{Nk(k-1)(k-2)}{2}$ FPAI features.

When matching point patterns with large number of outliers and missing points, larger values of k have to be used. As k increases, the number of FPAI features grows at an order of k^3 . This is true for any affine invariant matching scheme because to describe a point in an affine invariant way at least three other points are required.

FPAI features are extracted from both, the sampled bioassay beads X^+ , and deposition beads Y . Since the parameter k depends on the amount outliers present in the set different k are used for the bioassay and deposition bead sets. These are referred to as k_b and k_d . To address the large number of outliers and missing beads in the bead pattern matching problem, values of k_b and k_d can range from 10 to 50. With such large values of k_b and k_d an efficient method is required to match the FPAI features extracted from X^+ and Y . One such matching method is discussed in the next section.

7.1.3 Local Feature Matching

One of the main advantages of the FPAI features is it quantizes the feature space using *intersection type* t , whose values are restricted to the set $\{0,1,2,3, \dots, 8\}$, corresponding to the 9 intersection types presented in Fig. 7.2. This reduces the matching complexity and allows matching larger point sets.

In order to account for the jitter noise a soft matching is performed, where for each bioassay FPAI feature multiple approximately matched deposition FPAIs are found. The ratios in each deposition FPAI feature (r_1, r_2) are used to calculate an orthogonal matching range. The actual bead is assumed to lie within a distance of $3\sigma_j$ from its estimated position. Fig. 7.4 illustrates the range calculation. To estimate bead location p_1 we assume that the actual bead lies within a circle of radius

$p_1 p_1'$. In the two extreme cases, the line segment $p_1 e$ can extend to $p_1'' e''$ (d_{max}) or contract to $p_1' e'$ (d_{min}) and the ratio $r = \frac{p_1 e}{p_2 e}$ can become $r_{high} = \frac{p_1'' e''}{p_2' e'} = \frac{p_1 e + 6\sigma_j}{p_2 e - 6\sigma_j}$ or $r_{low} = \frac{p_1' e'}{p_2'' e''} = \frac{p_1 e - 6\sigma_j}{p_2 e + 6\sigma_j}$.

Fig. 7.4. An illustration of the ratio range calculation for soft matching of FPAI features.

The features extracted from the deposition point set Y are maintained in nine 2-d segment trees, indexed by the *intersection type* t . Given the orthogonal feature range $(t, \{r_{1_{high}}, r_{1_{low}}\}, \{r_{2_{high}}, r_{2_{low}}\})$ extracted from a set of ordered deposition beads $\{y_m, y_i, y_n, y_j\}$, the range $\{r_{1_{high}}, r_{1_{low}}\}, \{r_{2_{high}}, r_{2_{low}}\}$ is inserted into the t^{th} segment tree.

Given a feature vector $d^Y = [t, r_1, r_2]^T$ extracted from a set of ordered bioassay beads $\{x_m, x_i, x_n, x_j\}$, a stabbing query for the point (r_1, r_2) is performed on the t^{th} segment tree. The ordered deposition four points whose corresponding orthogonal features ranges contain the point (r_1, r_2) are identified as the matches for the ordered set of bioassay beads $\{x_m, x_i, x_n, x_j\}$.

A $k_b m \times k_d n$ matching matrix M is constructed to store the matching bioassay and deposition ordered four points, where $k_b m$ and $k_d n$ are the number of primary lines used for X and Y respectively. The $(uv)^{\text{th}}$ element in M corresponds to the matching between the u^{th} primary line in X and the v^{th} primary line in Y . Each cell in M maintains a list, whose nodes are pairs of directed secondary bioassay and deposition lines. For each matching pair of ordered bioassay and deposition four point $\{x_m, x_i, x_n, x_j\}$ and $\{y_m, y_i, y_n, y_j\}$ the pair of directed secondary lines $\{L_{ij}^X, L_{ij}^Y\}$ are added to the $(uv)^{\text{th}}$ cell of M , where L_{ij}^X is the secondary line from the bioassay point x_i to x_j and L_{ij}^Y is the secondary line from template point y_i to y_j , and u is the index of the primary line in bioassay image from point x_m to x_n , and v is the index of the primary line in deposition image from point y_m to y_n .

The overall matching requires $O(n_{fb} \log^2 n_{fd})$ time and $O(n_{fd} \log^2 n_{fd})$ memory, where n_{fb} and n_{fd} are the number of FPAI features extracted from the sampled bioassay beads X^+ and the deposition beads Y , respectively

7.1.4 Candidate Transformation Estimation

Each pair of matched ordered four points $\{x_m, x_i, x_n, x_j\}$ and $\{y_m, y_i, y_n, y_j\}$, produces four point correspondences. This set of correspondences is referred to as a 4-correspondences set (4CS). Using the four correspondences in a 4CS, (7.3) is used to calculate the affine transformation A_C that best aligns the corresponding points, in a least squares sense. The affine transform A_C thus estimated is referred to as a candidate transformation.

7.1.5 Candidate Transformation Pruning

Soft-matching of FPAIs produces a large number of 4CSs, most of which are likely to be random local matches. Solving the assignment problem to obtain the correspondence matrix P from the candidate transform A_C is costly, therefore to identify the candidate transforms that are most likely to be the true transform the list of candidate transforms are first pruned and the remaining candidate transforms are then ranked.

The $(uv)^{\text{th}}$ cell in the matching matrix M contains the list of NN lines (secondary lines) that match when the u^{th} primary line in X^+ and the v^{th} primary line in Y are matched (aligned). For instance, if two matching FPAI features were found, one for the 4CS $S_1 = \{\{x_a, x_i, x_b, x_j\}, \{y_a, y_i, y_b, y_j\}\}$ and the other for the 4CS $S_2 = \{\{x_a, x_k, x_b, x_l\}, \{y_a, y_k, y_b, y_l\}\}$, then the $(uv)^{\text{th}}$ cell in the matching matrix M will contain two pairs of matched secondary lines $\{L_{ij}^X, L_{ij}^Y\}$ and $\{L_{kl}^X, L_{kl}^Y\}$. Here L_{kl}^X is the line from the bioassay point x_k to x_l and L_{kl}^Y is the line from template point y_k to y_l . The number of entries in the ij^{th} cell therefore can be thought of as the support for the hypothesis that the u^{th} primary line in X (the line from x_a and x_b) and the v^{th} primary line in Y (the line from y_a and y_b) are true correspondences. By choosing appropriate values of k_b and k_d it can be ensured that matching bioassay and deposition image neighborhoods

have at least h matching NN points. In such a case, the cells in M corresponding to true line correspondences can be expected to have at least $s_h = \frac{h(h-1)}{2}$ support and any cell with fewer than s_h support can be pruned.

7.1.6 Candidate Transform Ranking

The pairs of matched (corresponding) secondary lines $\{L_{ij}^X, L_{ij}^Y\}$ and $\{L_{kl}^X, L_{kl}^Y\}$ are however weak indicators in support of the u^{th} primary line in X and the v^{th} primary line in Y being true correspondences, as they need not necessarily be true correspondences themselves. It could be that S_1 and S_2 are random local matches and are related by two different transformations. In that case, either or both of the matched pairs can be false correspondences. However, it is unlikely that the transformations A_{c1} and A_{c2} estimated from S_1 and S_2 will be similar if they are random local matches. But if S_1 and S_2 are true candidate transforms, i.e. candidate transforms generated from CSs containing true correspondences, A_{c1} and A_{c2} will lie very close to each other in the transformation space. These properties are used to calculate a score for each cell of M and rank them based on the scores.

The score for the $(uv)^{\text{th}}$ cell in the matching matrix M which contains n_{uv} 4CSs is computed as:

$$score_{m_{uv}} = \frac{1}{\sum_{i=1}^{n_{uv}} \sum_{j=i+1}^{n_{uv}} \|\vec{A}_{ci} - \vec{A}_{cj}\|} \quad (7.5)$$

where $\vec{A}_{ci}$ is the 1×8 vector obtained by stacking the rows of the candidate transform A_{ci} and $\|\cdot\|$ is the L_2 -norm. The top 5% cells based on the rankings are further evaluated to find the best candidate transform.

7.1.7 Candidate Transform Evaluation

To evaluate a candidate transform, the transformed bead coordinates $\hat{X}^+$ are generated from X^+ . For each $x_i \in \hat{X}^+$, if there exists a $y_j \in Y$ such that $\|x_i - y_j\| < 6\sigma_j$ then $\{x_i, y_j\}$ is called a

candidate correspondence. The search for candidate correspondences is also done using a k-d tree constructed from Y .

The candidate transform with the most x_i s that have at least one candidate correspondence is determined as the best candidate transform.

7.1.8 Correspondence Estimation

The candidate correspondences found using the best candidate transform are then used to create a fuzzy correspondence matrix $F_{n_x \times n_y}$ whose elements

$$F(i, j) = \exp \frac{(x_i - y_j)^T (x_i - y_j)}{2\sigma_j^2} \quad (7.7)$$

n_x and n_y are the number of x_i and y_j respectively, for which candidate correspondence(s) were found.

The fuzzy correspondence matrix F is reduced to a permutation matrix using the Hungarian algorithm [227]. When $n_x < n_y$, $(n_y - n_x)$ dummy rows with very high correspondence costs are added to F . After the permutation matrix is found, the dummy rows and the columns corresponding to the dummy rows are discarded, producing a $n_x \times n_x$ permutation matrix. Given this permutation matrix, the optimal transformation A^* is estimated using (7.4). While the Hungarian algorithm requires $O(n_y^3)$ time, each x_i produces few (in most cases 1) candidate correspondences, therefore there are few matching conflicts and the Hungarian algorithm requires only a few iterations.

To estimate the final bead correspondences (P^*), X is transformed using A^* and the same process of candidate correspondence estimation, fuzzy correspondence matrix generation and permutation matrix estimation are repeated using $\hat{X}$ and Y .

While a number of techniques can be used to reduce the fuzzy correspondence matrix into a permutation matrix (for instance genetic algorithms) the Hungarian algorithm guarantees finding the optimal solution in polynomial time. GA based matching algorithms do not guarantee finding the global optima. Since, it is expected that the fuzzy correspondence matrix derived by ranking

and filtering the cells of the matching matrix would already be close to a permutation matrix, the Hungarian algorithm is used to derive the permutation matrix relatively quickly and without the need for any additional parameters.

7.2 Grid-based Point Pattern Matching

The matching can be greatly simplified if the grid in the fluorescence image is known. In this section a method for matching the point patterns that uses the grid structure is proposed. Let $G^b = [\mathbf{g}_0^b, \dots, \mathbf{g}_m^b]^T$ and $G^d = [\mathbf{g}_0^d, \dots, \mathbf{g}_n^d]^T$ be grids extracted from the bioassay and deposition images respectively, where $\mathbf{g}_i = [x_i, y_i]^T$ are the spatial coordinates of the grid points. Let, a binary state variable b_i be associated with each $\mathbf{g}_i$ such that $b_i = 1$ indicates that a bead is present at grid location $\mathbf{g}_i$.

Fig. 7.5. Directed path for traversing a hexagonal grid. Three orderings of the 6-NNs of a grid point in a hexagonal grid are shown in (a), (b), and (c). The NN ordering shown in (a) is used to generate the directed path shown in (d). The ordering of the NNs in (b) can be used to generate a directed path similar to the one shown in (d). However, the ordering of the NNs in (c) cannot be used.

In the previous section a bead was described by the pattern formed by the other beads in its neighborhood. When the grid locations are known, both the presence and absence of beads at the grid locations in a bead's neighborhood can be used to describe the bead. Therefore, a descriptor that captures the state of the grid (presence/absence of beads at grid locations) in a bead $\mathbf{g}_i$'s neighborhood in a transformation invariant manner can be used for matching. Fig. 7.5(d) shows a directed path starting from grid location $\mathbf{g}_i$ that can be used to traverse the neighborhood of $\mathbf{g}_i$.

Let the directed path P_i starting at grid location $\mathbf{g}_i$ be the sequence $P_i = (\mathbf{g}_i, \mathbf{g}_i^1, \mathbf{g}_i^2, \dots, \mathbf{g}_i^n)$. The vector $\mathbf{z}_i = [b_i, b_i^1, b_i^2, \dots, b_i^n]^T$, whose k^{th} element b_i^k is the state of the corresponding grid location $\mathbf{g}_i^k$ in P_i , can be used as the descriptor for $\mathbf{g}_i$.

Due to the regular nature of the hexagonal grid the directed path P_i can be efficiently extracted for a large number of beads. Consider the ordering of the 6-NNs of a grid location shown in Fig. 7.5(a). The directed path P_i starting at a grid location can be extracted using the move sequence $\Delta = (\delta_1, \delta_2, \dots, \delta_n)$ shown in Fig 7.6. The entry δ_k in Δ indicates that in the directed path P_i the $(k + 1)^{\text{th}}$ grid point is the δ_k^{th} NN of the k^{th} grid point, where the δ_k^{th} NN is defined as per the ordering in Fig. 7.5(b). For instance, $\delta_2 = 3$ implies that the 3rd grid point (i.e., the $(k + 1)^{\text{th}}$ grid point) $\mathbf{g}_i^2$ is the 3rd-NN (δ_2^{th} -NN) of $\mathbf{g}_i^1$.

The descriptor $\mathbf{z}_i$ is not rotation invariant, although it is scale and translation invariant. However, given the grid point $\mathbf{g}_i$ and the first grid point $\mathbf{g}_i^1$ in the directed path P_i , the descriptor is rotation invariant. Since, there are only six ways of selecting the first grid point $\mathbf{g}_i^1$, six binary descriptors are extracted for each grid point. Figs. 7.5(b) and (c) show two other possible ordering of the 6-NNs of $\mathbf{g}_i$. While the ordering in Fig. 7.5(b) will produce a valid traversal path P_i , the ordering in Fig. 7.5(c) will not. To achieve a valid traversal path the NNs have to be ordered sequentially with respect to the angle the line between them and $\mathbf{g}_i$ make with a reference line (say the x -axis). The directed path P_k starting from any grid location $\mathbf{g}_k$ can be established, as long as each grid point maintains the list of its 6-NNs. Therefore, each grid points needs to maintain only its local information even though the directed path can traverse the entire grid.

$$\begin{aligned}
\Delta &= (1,3,4,5,6,1,1,2,3,3,4,4,5,5,6,6,1,1,1,2,2,3,3,3,4,4,4,5,5,5,6,6,6,1,1,1,1,2,2,2,3,3,3,3,4,4,4,5,5,5,5,6,6,6,6, \dots) \\
&= (\Delta_1, \Delta_2, \Delta_3, \Delta_4, \dots) \\
\Delta_1 &= (1,3,4,5,6) \\
\Delta_2 &= (1,1,2,3,3,4,4,5,5,6,6) \\
\Delta_3 &= (1,1,1,2,2,3,3,3,4,4,4,5,5,5,6,6,6) \\
\Delta_4 &= (1,1,1,1,2,2,2,3,3,3,3,4,4,4,4,5,5,5,5,6,6,6,6,6) \\
\Delta_n &= (1^{(1)}, \dots, 1^{(n)}, 2^{(1)}, \dots, 2^{(n-1)}, 3^{(1)}, \dots, 3^{(n)}, 4^{(1)}, \dots, 4^{(n)}, 5^{(1)}, \dots, 5^{(n)}, 6^{(1)}, \dots, 6^{(n)})
\end{aligned}$$

Fig. 7.6. The sequence of moves used to generate the directed path shown in figure 7.5(d)

The move sequence Δ has a repetitive pattern and therefore can be computed to generate the directed paths of desired length. The repetitive pattern in Δ is shown in Fig.7.6, where Δ_n is the move sequence for traversing the n^{th} concentric hexagon.

7.2.1 Binary Descriptor Matching

Let $\mathbf{z}_i^b$ and $\mathbf{z}_j^d$ be the binary descriptors of length L extracted by traversing the grids around the i^{th} bead in the bioassay image and the j^{th} bead in deposition image. The binary descriptors are extracted for only those grid locations where beads are present, producing m descriptors for beads in bioassay image and n descriptors for beads in deposition image. Unlike in the case of affine invariant matching, sampling is not performed when matching using grid information. The descriptors $\mathbf{z}_i^b$ and $\mathbf{z}_j^d$ are matched produce a score $m_{ij} = \mathbf{z}_i^{bT} \mathbf{z}_j^d$. The score can be computed using the bit-wise AND operation followed by bit-counting and therefore can be computed very efficiently.

Given two binary descriptors $\mathbf{z}_i^b$ and $\mathbf{z}_j^d$ of length L with $c_i = \|\mathbf{z}_i^b\|_1$ and $c_j = \|\mathbf{z}_j^d\|_1$, where $\|\cdot\|_1$ is the 1-norm that gives the number of 1s in the binary vector, the probability that $\mathbf{z}_i^b$ and $\mathbf{z}_j^d$ will have at least a score of m_{ij} can be computed using Binomial theorem:

$$s_{ij} = Pr(x \geq m_{ij}) = 1 - \sum_{i=1}^{m_{ij}} \binom{c_i}{i} \left(\frac{c_j}{L}\right)^i \left(1 - \frac{c_j}{L}\right)^{c_j-i} \quad (7.8)$$

Let, $\mathcal{P}$ be a $m \times n$ matrix whose ij^{th} element is the probability s_{ij} computed using (7.8) for the i^{th} bead in the bioassay image and the j^{th} bead in deposition image. For each row i of $\mathcal{P}$ the

column j with the lowest probability gives the deposition image bead y_j that best matches the bioassay bead x_i . The pairs of best matching bioassay and deposition beads $\{x_i, y_j\}$ are then ranked (in increasing order) based on the probability scores s_{ij} .

The top 10 bead pairs are then used to generate 10 candidate transformations. Given a matching bead pair their binary descriptors are used to identify more corresponding matching beads (candidate correspondences) in the bioassay and deposition images. Given the candidate correspondences, the candidate affine transformation A_C is estimated using (7.4). The 10 candidate transforms are evaluated and the optimal correspondences are estimated as described in Section 7.1.7 and 7.1.8.

7.3 Experiments and Results

The proposed affine invariant matching method was evaluated using 202 deposition and bioassay image pairs from a database consisting of 148 deposition images and 202 bioassay images. The images were generated from a total of 8 experiments consisting of 56 sub-experiments. The grid-based matching was evaluated on 202 deposition and bioassay image pairs and 192 deposition image pairs. The bioassay experiments ranged from 1-plex to 5-plex. Details of the experiments and the images per experiment are presented in Table 7.2. Here sub-experiments are individual bioassays that are simultaneously performed using a multi-well chip. Experiment 1 in Table 7.2 for instance involved 24 sub-experiments using 8 dilution levels and 3 replicates of each.

The bioassay experiments were performed using the Ayoxxa biochip (Ayoxxa, Singapore) and polystyrene beads from different providers. The deposition 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 bioassay images were acquired using an epifluorescent microscope (BX41, Olympus, Singapore) with a digital CCD camera (Retiga-4000R, QImaging, Canada). The bioassay images were acquired at different zoom levels ranging from 10x to 30x. All the bioassay experiments were conducted by Ayoxxa, Singapore and the images were provided in kind cooperation.

Table 7.2: DETAILS OF THE DATASET USED TO EVALUATE THE PROPOSED MATCHING METHODS

Experiment No.	Multiplex	No. of Sub-experiments	No. of Deposition Images	No. of Bioassay Images	No. of Deposition - Bioassay Image Pairs Matched	No. of Deposition Image Pairs Matched
1	1	24	24	42	42	0
2	2	4	8	8	8	4
3	3	1	3	6	6	3
4	3	3	9	10	10	9
5	4	15	60	74	74	90
6	4	1	4	8	8	6
7	5	4	20	28	28	40
8	5	4	20	26	26	40

The proportion of incorrectly matched bioassay beads or the Hamming error is used as the performance measure for all the tests. The matching results were also visually inspected to check for sub-optimal matches. The proposed methods were implemented in Java and MATLAB and all the experiments were performed on a standard desktop computer with an Intel Xeon Hexa Core 3.2 GHz processor with 16GB RAM. Multithreading was used with up to 12 threads.

Table 7.3 shows the mean Hamming error achieved on the 56 sub-experiments. The number of beads in the deposition images ranged from approximately 200 to 800 and the number of beads in the bioassay images ranged from 30 to 500 beads. For such small point sets and limited multiplexing both the proposed affine invariant matching and grid-based matching algorithms are able to find the correct transformation between the point sets in all cases. The small hamming error is due to a few bead correspondences that could not be established even under the correct transformation. These are mainly due to beads that had moved out of their micro-wells (and therefore their position had changed) between bead deposition and the bioassay image acquisition. For the affine invariant matching $k_b=10$ and $k_d=20$ were used. All the batches (including the positive control batch) in these experiments have nearly the same size. Therefore, the percentage of positive control beads is $\frac{1}{u}$, where u is the number of batches (or multiplex). For this test σ_j was assumed to be 20% of the median bead radius.

The performance of both the affine invariant matching method and the grid-based matching methods are very similar because in all cases they are able to identify the same true transformation

and the estimated bead correspondences are nearly the same.

Table 7.3: MATCHING ACCURACIES ACHIEVED ON REAL DATA

Experiment No.	Dataset		Affine Invariant Matching			Grid-based Matching		
	Multiplex	No. of Deposition-Bioassay Image Pairs	Mean Hamming Error (HE)	HE (Std. Dev.)	% Times HE > 0.5	Mean Hamming Error (HE)	HE (Std. Dev.)	% Times HE > 0.5
1	1	42	0.005	0.004	0%	0.005	0.004	0%
2	2	8	0.009	0.007	0%	0.008	0.006	0%
3	3	6	0.004	0.001	0%	0.004	0.001	0%
4	3	10	0.002	0.001	0%	0.003	0.001	0%
5	4	74	0.005	0.003	0%	0.004	0.002	0%
6	4	8	0.001	0.001	0%	0.001	0.001	0%
7	5	28	0.002	0.001	0%	0.002	0.001	0%
8	5	26	0.006	0.003	0%	0.005	0.003	0%

To establish the matching performance on larger point sets 5400 simulated experiments were performed. The parameters and their values used to generate the simulated data is given in Table 7.4. For affine invariant matching, 96 combinations of parameters were used and 50 samples were generated for each combination, producing a total of 4800 simulations. For the grid-based matching, 6 combinations of parameters were used and 100 simulations per combinations were generated. The simulated deposition and bioassay bead patterns were generated by uniformly randomly sampling micro-wells from the grids detected in 10 real deposition images. All 10 deposition images had more than 50000 micro-wells.

Table 7.4: SIMULATION PARAMETERS AND COMBINATIONS OF THEIR VALUES USED TO GENERATE SIMULATED TEST DATA

Simulation Parameters	Values	
	Affine Invariant Matching	Grid-based Matching
Total number of beads used in the experiment (n_b)	5000, 10000, 20000, 30000	40000, 50000
Number of positive control beads (n_c)	5%, 10%, 15%, 20% of n_b	5%, 10%, 15% of n_b
% of total beads (n_b) present in bioassay image	75%, 100%	100%
Transformations	similarity transform, affine transform, and small non-linear distortion	similarity transform,

The matching results obtained on the simulated data are presented in Table 7.5. As the number of beads used in the experiment increases the number of control beads needed to achieve

successful matching also increases. For $n_b = 5000$, 5% positive control beads are enough to successfully match the bead patterns whereas for $n_b = 30000$, 15% positive control beads are needed. The number of NNs needed to match also increases as the number of beads increases. Nevertheless, the proposed method is able to match 30000 beads in a reasonable time. When 15% positive control beads are used in an experiment of 30000 beads, 25500 beads are available for multiplexing. Therefore, even when using 50 replicates per batch, a 510-plex experiment can be performed.

Table 7.5: MATCHING ACCURACIES ACHIEVED ON THE SIMULATED DATA

		% Times Hamming Error > 0.5							
Total number of beads used in the experiment (n_b)		$n_b=5000$		$n_b=10000$		$n_b=20000$		$n_b=30000$	
Number of positive control beads (n_c)	% of total beads (n_b) present in bioassay image	$k_b=100$ $k_d=10$	$k_b=100$ $k_d=20$	$k_b=100$ $k_d=20$	$k_b=150$ $k_d=20$	$k_b=100$ $k_d=25$	$k_b=150$ $k_d=30$	$k_b=100$ $k_d=25$	$k_b=150$ $k_d=30$
$n_c = 5\%$ of n_b	75%	0	0	0.02	0	0.09	0.04	0.35	0.21
$n_c = 5\%$ of n_b	100%	0.02	0	0.04	0.02	0.14	0.07	0.46	0.30
$n_c = 10\%$ of n_b	75%	0	0	0	0	0.03	0	0.13	0.08
$n_c = 10\%$ of n_b	100%	0	0	0	0	0.04	0	0.21	0.13
$n_c = 15\%$ of n_b	75%	0	0	0	0	0	0	0	0
$n_c = 15\%$ of n_b	100%	0	0	0	0	0	0	0.03	0
$n_c = 20\%$ of n_b	75%	0	0	0	0	0	0	0	0
$n_c = 20\%$ of n_b	100%	0	0	0	0	0	0	0	0

The grid-based matching achieved 100% matching accuracy in all the simulated tests. Fig. 7.7. show the utility of the candidate transformation ranking step. Fig. 7.7(a) shows the Rank-1 correspondence accuracy of the affine invariant matching algorithm. Here, Rank-1 correspondence accuracy is the proportion of times, the best estimated match for the query primary lines (that have a true correspondence in the template) is a true correspondence. Fig.7.7(b) shows the analogously defined Rank-10 correspondence accuracy of the affine invariant matching algorithm and Fig. 7.7(c) shows the Rank-1 correspondence accuracy of the grid-based matching algorithm. Note that in the case affine invariant matching the Rank-200 of all the estimated correspondences are evaluated.

Fig. 7.7. Correspondence accuracies of the affine invariant and grid-based matching methods. (a) and (b) show the Rank 1 and Rank 10 correspondence accuracies of the affine invariant matching method, and (c) shows the Rank 1 correspondence accuracy of the grid-based matching method.

The running times of the two matching methods are presented in Fig 7.8. Fig 7.8(a) shows the time required by the affine invariant matching method to match point sets of different sizes using $n_c = 5\%$ of n_b . Fig 7.8(b) shows the time required by the grid-based matching method to match point sets of different sizes using $n_c = 5\%$ of n_b . The affine invariant matching takes the similar amount of time to match 12000 points with 2000 points, that the grid based matching method takes to match 50000 points with 7500 points.

Fig. 7.8. The running time of the matching methods. (a) time taken by the affine invariant matching method to match point sets of different sizes using $n_c = 5\%$ of n_b . (b) time taken by the grid-based matching method to match point sets of different sizes using $n_c = 15\%$ of n_b .

Fig. 7.9. Examples of matching achieved using the proposed method. (a) and (d) are two query points and (b) and (e) are two template points. (c) and (f) show the query and template points aligned.

7.3.1 Comparison of Affine Invariant Matching with Existing Methods

The proposed affine invariant matching is evaluated against five other methods: coherent point drift (CPD) [198], thin-plate spline - expectation maximization (TPS-EM) [204], geometric hashing (GH) [58, 59], locally likely arrangement hashing (LLAH) [109], and local geometric hashing (LGH). In TPS-EM, the *softassign* based correspondence estimation of TPS-RPM [192] is replaced using an EM based correspondence estimation, which has been shown to be similar to *softassign* [204]. Gaussian mixture model (GMM) based registration [204] and kernel correlation (KC) [203] were also considered but produced very poor results during qualitative assessment and were not evaluated further.

To compare the five algorithms 300 query-template point pattern pairs were simulated. The combination of query and template point pattern sizes used to generate the 300 pairs is shown in Table 7.6. Twenty levels of outliers ranging from 0% to 95% were considered. Here an outlier is a point in the query that does not have a match in the template. These points are analogous to non-control beads in the bead pattern matching problem.

As discussed in Section 4.3, the main drawback of convex optimization based methods is that they are not robust against outliers, and missing points, particularly when the initial transformation is large. The first comparison experiment was designed to highlight the robustness of the proposed method (PM) in this regard. To simulate jitter noise, each query point was shifted in a random direction by a distance equal to 5% of the mean distance between query points. Geometric hashing could not be run for point sets of size greater than 70 points when using 16GB RAM and therefore was not included in this experiment.

Fig. 7.10 shows the proportion of times the algorithms were able to find the true transformation at increasingly greater levels of outliers in the query. Performance of CPD starts to fall near the 40% outlier level. By 80% outlier level both CPD and TPS-EM can find the correct match less than 10% of times. At 80% outlier level they can find the correct match in only those cases where the initial transformation between the query and template point set is very small. In

contrast PM, LGH and LLAH are able to achieve high matching accuracies even at 95% outlier level. However, LGH has slight random fluctuations at high outlier levels. This is due to its sensitivity to jitter noise

Table 7.6: PARAMETERS USED TO GENERATE SIMULATED DATA FOR MATCHING COMPARISONS

Parameter	Values
Number of beads in query (n_b)	100, 200, 300
Number of beads in template (analogous to positive control beads) (n_c)	0% 5%, 10%, 15%, 20%, ... , 95% of n_b
Number of replicate experiments for each combination of n_b and n_c	5
Transformations	random similarity transform, affine transform (with the absolute value of shear factor ranging between 0.5 and 1.0)
Jitter noise	5% of mean distance between points

Fig. 7.10. The robustness of CPD, TPS-EM, LLAH, LGH, and the proposed method (PM) at different levels of outliers (points in query that do not have a match in template). LLAH could be run only till 80% outlier level due to its high memory requirements.

The parameters of CPD and TPS-EM were optimized using grid search over the parameter space using all 300 query-template pairs. For PM, LGH, and LLAH the size of the local

neighbourhood (i.e. the number of nearest neighbours) was set so that the local neighbourhood contains at least 5 inlier (matching) points. That is, when 95% outliers are present, 100-nearest neighbours are used. As the amount of outliers increases the time and space complexity of these algorithms also increases. LLAH could be run only for point sets of size 100 till 80% outliers when using 16GB RAM. Fig 7.11 shows the number of invariants extracted by LLAH, GH, and PM for point sets of different sizes at 95% outlier level. For LLAH, LGH, and GH the invariants are barycentric coordinates and for PM the invariants are FPAIs. The number of invariants extracted by LGH is the same as PM.

To handle outliers, given a point p , LLAH first samples n points from m nearest neighbour points of p and then computes the barycentric coordinates of p using all combinations of 3 points in n . The C_3^n barycentric coordinates are used as a descriptor for p . Since it is not known if all the sampled n points are inliers, all possible combinations of n points (among m) must be considered. This ensures that as long as there are n inliers among the m nearest neighbours of p the correct match for p can be found. However, to encode one point $C_n^m * C_3^n$ features have to be extracted.

Fig. 7.11. Number of invariants extracted by LLAH, GH and the proposed method (PM) to find a match at 95% outlier level, for increasing point set sizes. LGH requires the same number of invariants as the proposed method for the same size of local neighbourhood.

The proposed method achieves the same robustness using the matching matrix. Consider a primary line L_{ij}^X , between the i^{th} and j^{th} points in the template point set X , and three secondary lines L_{kl}^X , L_{km}^X , and L_{ml}^X , between the k^{th} , l^{th} and m^{th} points in X . If these five points are inliers and if

the corresponding k^{th} , l^{th} and m^{th} points in the query point set Y are neighbors of the corresponding the i^{th} and j^{th} points in Y then these primary and secondary lines are guaranteed to be extracted from Y as well. This is because all pairs of nearest neighbors of a line are considered while constructing the secondary lines. In the matching matrix cell corresponding to the L_{ij}^X and L_{ij}^Y primary lines (where L_{ij}^Y is the primary line in query that matches L_{ij}^X) three pairs of matching secondary lines will be added. Since, the three transformations estimated using the three pairs of secondary lines (along with the pair of primary lines) will be in agreement, this cell will have a high score. Computing the number of transformations that are in agreement in the matching matrix cell tells us that local the neighborhoods of lines L_{ij}^X and L_{ij}^Y have greater than two matching points. Thereby allowing us to find matching patterns formed by more than four points, at little additional computational cost.

A major limitation of geometric hashing based techniques is the need to quantize the invariant space to account for jitter noise. A discrete partitioning of the invariant space can introduce boundary effects. Moreover, the granularity of quantization can have a significant effect on matching performance. Fig. 7.12 shows the correspondence matrices obtained using LGH when matching identical point patterns (such that the true correspondence matrix is the identity matrix) perturbed by jitter noise. The three rows in Fig. 7.12 correspond to quantization levels 10^3 , 10^6 and 10^9 from top to bottom respectively, where quantization level 10^3 means that the invariant space was quantized into a lattice of 10^3 equal sized rectangular bins. The three columns in Fig. 7.12 correspond to 0%, 5% and 10% jitter noise, where 5% jitter means each query point was shifted in a random direction by a distance equal to 5% of the mean distance between the query points. As jitter noise increases the estimated correspondence matrices deviate further from the identity matrix. Also, when jitter noise is 5% (second column) quantizing the invariant space into 10^6 bins (second row) produces the best correspondence matrix, among the three cases. However, when jitter noise is 10% (third column) quantizing the invariant space into 10^3 bins (first row) produces better a correspondence matrix than quantizing it into 10^6 bins. To avoid this need for quantization the proposed method uses the ratio ranges.

While the ratio range provides a way of avoiding quantization it has a downside as well. Large ratio ranges can increase the number of false positive matches. Fig. 7.13 shows the effect of jitter noise on the proposed algorithm. The first column of Fig. 7.13 shows the matching matrix counts for 0%, 5% and 10% jitter noise in the first, second and third rows respectively. As jitter noise increase the ratio ranges have to be increased as well. As a result some primary lines which form many large ratio ranges, have large number of matches. These rows can be seen as horizontal streaks in the matching matrix. The proposed method relies on the candidate transform ranking step to extract true correspondences from these matching matrices. In the candidate transform ranking step a score is assigned to each matching matrix cell based on the count of matches that have similar transformations. Only the top 5% matching matrix cells with the highest scores are kept for further, more exhaustive, evaluation. The second and third columns of Fig. 7.13 shows the matching matrix cell scores and the filtered top 5% cells respectively. Even under 10% jitter noise this strategy is able to identify a large number of true correspondences (see Fig. 7.13(i)).

To further evaluate the limits of the proposed algorithm and compare it to LGH, in term of robustness to jitter noise an experiment with increasing amounts of jitter noise was performed. The results are presented in Fig. 7.14. The proposed method is more robust to jitter noise than LGH at any of the three quantization levels. Moreover, it can be seen that the proposed method can match without any loss of accuracy till 6% jitter noise. The effect of distortion caused by jitter noise is however not just dependent on the absolute value of shift in the points but also on the density of the points. In a very dense point set small amounts of jitter can introduce ambiguity about correspondences. Whereas if the points are sparse and far away from each other the same amount of jitter may not affect the overall pattern formed by the points. Therefore, to study the limits of jitter noise in the context of point density another experiment was performed using point sets of increasing densities. Fig. 7.15 shows the matching accuracies of the proposed method and LGH for point sets of increasing densities.

Fig. 7.12. The effect of jitter noise on the correspondence matrix generated using LGH when matching two identical patterns. The columns from left to right show correspondence matrices at 0%, 5%, and 10% jitter levels respectively. The rows from top to bottom show correspondence matrices for 10^3 , 10^6 , and 10^9 quantization bins respectively.

Fig. 7.13. The effect of jitter noise on the matching matrix generated using the proposed affine invariant matching method when matching two identical patterns. The columns from left to right show the counts, the scores, and the top 5% ranked cells in the matching matrix respectively. Rows from top to bottom represent 0%, 5%, and 10% jitter noise levels respectively.

Fig. 7.14. Comparison of robustness of the proposed affine invariant matching method and LGH under jitter noise.

Fig. 7.15. Comparison of robustness of the proposed affine invariant matching method and LGH under jitter noise for point sets of varying densities.

A density of 0.05 in Fig. 7.15 implies that the query point set was generated by uniformly randomly sampling from an ideal hexagonal grid, such that on average one in twenty grid points was sampled. That is, the ideal hexagonal grid is assumed to have density 1. The proposed method outperforms LGH at all density levels. In fact the proposed method's accuracy at density 0.20 is comparable to that of LGH at density level 0.05. Also, when point density is low the proposed method can handle as much as 18% jitter noise.

As discussed earlier, the affine invariant matching algorithm is particularly useful for bead

pattern matching when the point sets are sparse. For dense point sets bioassay gridding is likely to succeed and grid-based matching can be used.

Chapter 8 - Conclusions and Future Work

8.1 Contributions and Conclusions

To harness the power of bead-based microarrays numerous bead encoding techniques, ranging from simple colorimetric to complex bead assembly techniques, have been developed. The robust spatially addressable bead encoding technique suggested here gives a cheaper and more efficient alternative to these methods. The algorithms and the pipeline developed in this research can be used for high throughput bead-based arrays that can achieve greater than 250-plex bead encoding in a completely unsupervised setting using as many as 100 replicates per target analyte. When some basic information about the physical properties of the bead chip is available the proposed method can achieve greater than 500-plex encoding. In comparison, currently available commercial bead-based systems are limited to 30 to 40-plex arrays.

The main contributions of this thesis are the six computational methods that have been developed. The gridding method presented in Chapters 5 extends the current state-of-the-art gridding techniques and is able to handle unprecedented levels of noise and image degradation. The micro-well classification method presented in Chapter 5 shows promising results and addresses a novel problem that has not been studied before.

The fluorescence image segmentation discussed in Chapter 6 presents a novel way of analyzing microarray images. It allows for a much more involved analysis while maintaining efficiency. The fluorescence image gridding method discussed in Chapter 6 considers an often overlooked problem of gridding fluorescence images with few expressed beads.

While numerous computational tools are available for spotted microarray images, few non-commercial tools have been developed for bead-based microarray. However, as has been presented in this theses, bead-based microarrays present unique challenges which must be adequately addressed. Microarrays are a unique technology where computation, particularly image processing, plays a critical role in the biological experiments. While this has allowed tremendous amounts of biological data to be generated, it has also introduced the problem of reproducibility. The image

processing methods introduced too many parameters, often without consensus on how to use them. These parameters can have a large impact on the final experimental results. Moreover, it is often difficult to go back and analyze the cause of variability when the results cannot be reproduced. Bead-based microarrays have tried to address this issue of reproducibility by making the biochemical processes of estimation more robust. However, the bead-based microarray technology also depends on similar computational tools as the spotted microarrays. It is a common misconception that the tools that have been developed for spotted microarrays can be readily used for bead-based microarrays. As has been discussed in this thesis, bead-based microarrays present unique challenges and robust computational tools are needed to specifically address these challenges. The computational tools developed in this thesis are a first step towards this endeavor.

These methods introduce a number of novel ideas that have wide applicability not only in microarray image analysis but also in various other computer vision problems. The micro-well classification scheme developed in Chapter 5 can be used for automatic quality control during bead fabrication, where the number of successfully deposited beads has to be counted. The Inner-ICP REJSAC based grid alignment and the hypergeometric test based grid alignment quality estimation can be used for various types of grid matching problems that often arise in computer vision. Finally, the affine invariant matching algorithm discussed in Chapter 7 addresses one of the most widely encountered problems in computer vision, that of affine invariant matching under extreme distortion.

8.2 Future Work

The main bottle neck in the proposed computational pipeline, in terms of the processing time required, are the micro-well gridding and classification methods. This is mainly because many deposition images have to be processed for each experiment. However, there are invariably some similarities between the deposition images of the same experiment. Therefore, in the future, it would be useful to investigate what knowledge can be transferred from one deposition image analysis step to the next.

One of the challenges in microarray image analysis, that distinguishes it from other computer vision tasks, is the sheer number of objects (micro-wells and beads) that have to be processed in each image. Both the micro-well classification method discussed in Chapter 5 and the bead foreground segmentation method discussed in Chapter 6, rely on simple features such as circularity and area to speed-up the task. In the future, one possible direction of research would be to investigate graphical processing unit (GPU)-based feature extraction.

One of the objectives of this research is to develop tools that can be used by academic labs to perform adhoc and customized multiplexed analysis. Where, the technicians purchase custom bead chips and commercially available beads and do bead deposition in the lab. To perform a 100-plex assay the technician would have to do the bead deposition step 100 times and acquire 100 deposition images. In practice, this limits the multiplexing capability of the bead encoding scheme. Therefore, from the perspective of the spatial bead encoding scheme a future direction of research would be to develop spatial bead encoding schemes that can circumvent the need for, or at least reduce the dependence on the sequential deposition process.

While numerous bead encoding techniques have been developed that can provide very high multiplexing capabilities (albeit using costly hardware), commercial bead encoding systems still mostly allow 30 to 40-plex bead encoding. One of the factors contributing towards the slow increase in the multiplexing capacity is that bio-molecules cross react with each other. Even when the surface chemistries of the beads are optimized, some non-specific binding (where molecules other than the target molecule bind with the bead) cannot be avoided. From the broader perspective of targeted biomolecule quantification a future direction of research would be to investigate how computational tools (such as bead segmentation) can accommodate such discrepancies in the data and may be even predict and take corrective actions.

While spatial encoding offers high multiplexing at low cost, it also has a drawback. It requires bead deposition to be performed multiple times. For a 100-plex experiment, bead deposition would have to be performed 100 times. This can be tedious and time consuming, if performed manually. Spatial bead encoding schemes would have to be further developed in the

future so that the need to perform multiple depositions can be removed or reduced.

References

- [1] Q. Zhu and D. Trau, "Multiplex detection platform for tumor markers and glucose in serum based on a microfluidic microparticle array," *Analytica Chimica Acta*, vol. 751, pp. 146-154, 2012.
- [2] S. Birtwell and H. Morgan, "Microparticle encoding technologies for high-throughput multiplexed suspension assays," *Integrative Biology*, vol. 1, pp. 345-362, 2009.
- [3] J. K. Ng, E. S. Selamat, and W.-T. Liu, "A spatially addressable bead-based biosensor for simple and rapid DNA detection," *Biosensors and Bioelectronics*, vol. 23, pp. 803-810, 2008.
- [4] D. Trau, W.-T. Liu, and J. K. Ng, "Microarray System and a Process for Producing Microarrays," US Patent US20100075865A1, Mar. 25, 2010.
- [5] P. M. LoRusso, "Phase 0 clinical trials: an answer to drug development stagnation?," *Journal of Clinical Oncology*, vol. 27, pp. 2586-2588, 2009.
- [6] K. Braeckmans, S. C. De Smedt, M. Leblans, R. Pauwels, and J. Demeester, "Encoding microcarriers: present and future technologies," *Nature Reviews Drug Discovery*, vol. 1, pp. 447-456, 2002.
- [7] R. Wilson, A. R. Cossins, and D. G. Spiller, "Encoded microcarriers for high - throughput multiplexed detection," *Angewandte Chemie International Edition*, vol. 45, pp. 6104-6117, 2006.
- [8] M. B. Meza, "Bead-based HTS applications in drug discovery," *Drug Discovery Today*, vol. 5, pp. 38-41, 2000.
- [9] M. J. Dejneka, A. Streltsov, S. Pal, A. G. Frutos, C. L. Powell, K. Yost, *et al.*, "Rare earth-doped glass microbarcodes," *Proc. of the National Academy of Sciences of the United States of America*, vol. 100, no. 2, pp. 389-393, 2003.
- [10] S. R. Nicewarner-Pena, R. G. Freeman, B. D. Reiss, L. He, D. J. Peña, I. D. Walton, *et al.*, "Submicrometer metallic barcodes," *Science*, vol. 294, pp. 137-141, 2001.
- [11] M. Evans, C. Sewter, and E. Hill, "An encoded particle array tool for multiplex bioassays," *Assay and Drug Development Technologies*, vol. 1, pp. 199-207, 2003.
- [12] K. L. Michael, L. C. Taylor, S. L. Schultz, and D. R. Walt, "Randomly ordered addressable high-density optical sensor arrays," *Analytical Chemistry*, vol. 70, pp. 1242-1248, 1998.
- [13] Luminex Corporation, "Luminex xMAP® Technology," 2016, <https://www.luminexcorp.com/clinical/our-technology/xmap-technology/>
- [14] K. L. Kellar and M. A. Iannone, "Multiplexed microsphere-based flow cytometric assays," *Experimental Hematology*, vol. 30, pp. 1227-1237, 2002.
- [15] M. Han, X. Gao, J. Z. Su, and S. Nie, "Quantum-dot-tagged microbeads for multiplexed optical coding of biomolecules," *Nature Biotechnology*, vol. 19, pp. 631-635, 2001.
- [16] A. Sukhanova and I. Nabiev, "Fluorescent nanocrystal-encoded microbeads for multiplexed cancer imaging and diagnosis," *Critical Reviews in Oncology/Hematology*, vol. 68, pp. 39-59, 2008.
- [17] S. Rauf, "Quantum dot encoded magnetic beads for multiplexed fluorescence biosensing," Ph.D. dissertation, Dept. of Electronics and Electrical Engineering, University of Glasgow, Scotland, 2010.
- [18] J. F. Keij and J. A. Steinkamp, "Flow cytometric characterization and classification of multiple dual - color fluorescent microspheres using fluorescence lifetime," *Cytometry*, vol. 33, pp. 318-323, 1998.
- [19] J. M. Kürner, I. Klimant, C. Krause, E. Pringsheim, and O. S. Wolfbeis, "A new type of

- phosphorescent nanospheres for use in advanced time-resolved multiplexed bioassays," *Analytical Biochemistry*, vol. 297, pp. 32-41, 2001.
- [20] L. Grøndahl, B. J. Battersby, D. Bryant, and M. Trau, "Encoding combinatorial libraries: a novel application of fluorescent silica colloids," *Langmuir*, vol. 16, pp. 9709-9715, 2000.
 - [21] A. Van Blaaderen and A. Vrij, "Synthesis and characterization of colloidal dispersions of fluorescent, monodisperse silica spheres," *Langmuir*, vol. 8, pp. 2921-2931, 1992.
 - [22] W. Yang, D. Trau, R. Renneberg, N. T. Yu, and F. Caruso, "Layer-by-layer construction of novel biofunctional fluorescent microparticles for immunoassay applications," *Journal of colloid and interface science*, vol. 234, pp. 356-362, 2001.
 - [23] R. Jin, Y. C. Cao, C. S. Thaxton, and C. A. Mirkin, "Glass - Bead - Based Parallel Detection of DNA Using Composite Raman Labels," *Small*, vol. 2, pp. 375-380, 2006.
 - [24] H. Fenniri, L. Ding, A. E. Ribbe, and Y. Zyrianov, "Barcoded resins: A new concept for polymer-supported combinatorial library self-deconvolution," *Journal of the American Chemical Society*, vol. 123, pp. 8151-8152, 2001.
 - [25] W. E. Doering and S. Nie, "Spectroscopic tags using dye-embedded nanoparticles and surface-enhanced Raman scattering," *Analytical Chemistry*, vol. 75, pp. 6171-6176, 2003.
 - [26] S. I. Stoeva, J.-S. Lee, J. E. Smith, S. T. Rosen, and C. A. Mirkin, "Multiplexed detection of protein cancer markers with biobarcode nanoparticle probes," *Journal of the American Chemical Society*, vol. 128, pp. 8378-8379, 2006.
 - [27] B. K. Oh, J. M. Nam, S. W. Lee, and C. A. Mirkin, "A Fluorophore - Based Bio - Barcode Amplification Assay for Proteins," *Small*, vol. 2, pp. 103-108, 2006.
 - [28] C. L. Brummel, I. N. Lee, Y. Zhou, S. J. Benkovic, and N. Winograd, "A mass spectrometric solution to the address problem of combinatorial libraries," *Science-AAAS-Weekly Paper Edition-including Guide to Scientific Information*, vol. 264(5157), pp. 399-402, 1994.
 - [29] S. S. Rahman, D. J. Busby, and D. C. Lee, "Infrared and Raman spectra of a single resin bead for analysis of solid-phase reactions and use in encoding combinatorial libraries," *The Journal of Organic Chemistry*, vol. 63, pp. 6196-6199, 1998.
 - [30] J. E. Hochlowski, D. N. Whittern, and T. J. Sowin, "Encoding of combinatorial chemistry libraries by fluorine-19 NMR," *Journal of Combinatorial Chemistry*, vol. 1, pp. 291-293, 1999.
 - [31] E. J. Moran, S. Sarshar, J. F. Cargill, M. M. Shahbaz, A. Lio, A. M. Mjalli, *et al.*, "Radio frequency tag encoded combinatorial library method for the discovery of tripeptide-substituted cinnamic acid inhibitors of the protein tyrosine phosphatase PTP1B," *Journal of the American Chemical Society*, vol. 117, pp. 10787-10788, 1995.
 - [32] K. Nicolaou, X. Y. Xiao, Z. Parandoosh, A. Senyei, and M. P. Nova, "Radiofrequency encoded combinatorial chemistry," *Angewandte Chemie International Edition in English*, vol. 34, pp. 2289-2291, 1995.
 - [33] D. A. Vignali, "Multiplexed particle-based flow cytometric assays," *Journal of Immunological Methods*, vol. 243, pp. 243-255, 2000.
 - [34] M. J. Benecky, D. R. Post, S. M. Schmitt, and M. S. Kochar, "Detection of hepatitis B surface antigen in whole blood by coupled particle light scattering (Copalis™)," *Clinical Chemistry*, vol. 43, pp. 1764-1770, 1997.
 - [35] G. A. Lawrie, B. J. Battersby, and M. Trau, "Synthesis of optically complex core-shell colloidal suspensions: pathways to multiplexed biological screening," *Advanced Functional Materials*, vol. 13, pp. 887-896, 2003.
 - [36] 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*, vol. 21, pp. 2875-2882, 2005.
 - [37] T. L. Bergemann, R. J. Laws, F. Quiaoit, and L. P. Zhao, "A statistically driven approach for image segmentation and signal extraction in cDNA microarrays," *Journal of*

- Computational Biology*, vol. 11, pp. 695-713, 2004.
- [38] J. Angulo and J. Serra, "Automatic analysis of DNA microarray images using mathematical morphology," *Bioinformatics*, vol. 19, pp. 553-562, 2003.
 - [39] G. Bidaut, F. J. Manion, C. Garcia, and M. F. Ochs, "WaveRead: automatic measurement of relative gene expression levels from microarrays using wavelet analysis," *Journal of Biomedical Informatics*, vol. 39, pp. 379-388, 2006.
 - [40] P. Bajcsy, "Gridline: automatic grid alignment DNA microarray scans," *IEEE Transactions on Image Processing*, vol. 13, pp. 15-25, 2004.
 - [41] K. Blekas, N. P. Galatsanos, A. Likas, and I. E. Lagaris, "Mixture model analysis of DNA microarray images," *IEEE Transactions on Medical Imaging*, vol. 24, pp. 901-909, 2005.
 - [42] A. G. Ramm and A. I. Katsevich, *The Radon transform and local tomography*, CRC press, 1996.
 - [43] N. Brändle, H. Bischof, and H. Lapp, "Robust DNA microarray image analysis," *Machine vision and Applications*, vol. 15, pp. 11-28, 2003.
 - [44] M. Katzer, F. Kummert, and G. Sagerer, "Methods for automatic microarray image segmentation," *IEEE Transactions on Nanobioscience*, vol. 2, pp. 202-214, 2003.
 - [45] M. Katzer, F. Kummert, and G. Sagerer, "A Markov random field model of microarray gridding," in *Proc. 2003 ACM Symposium on Applied Computing*, Melbourne, FL, USA, pp. 72-77, 2003.
 - [46] N. D. Lawrence, M. Milo, M. Niranjana, P. Rashbass, and S. Soullier, "Reducing the variability in cDNA microarray image processing by Bayesian inference," *Bioinformatics*, vol. 20, pp. 518-526, 2004.
 - [47] M. Eisen, "ScanAlyze," 1999.
 - [48] L. Rueda and I. Rezaeian, "A fully automatic gridding method for cDNA microarray images," *BMC Bioinformatics*, vol. 12, p. 1, 2011.
 - [49] D. Bariamis, D. K. Iakovidis, and D. Maroulis, "M 3 G: maximum margin microarray gridding," *BMC Bioinformatics*, vol. 11, p. 1, 2010.
 - [50] C. C. Charalambous and G. K. Matsopoulos, "A new method for gridding DNA microarrays," *Computers in biology and medicine*, vol. 43, pp. 1303-1312, 2013.
 - [51] M. Ceccarelli and G. Antoniol, "A deformable grid-matching approach for microarray images," *IEEE Transactions on Image Processing*, vol. 15, pp. 3178-3188, 2006.
 - [52] L. Shapiro and G. Stockman, *Computer Vision*, Prentice Hall, 2001.
 - [53] D. Morris, "Blind microarray gridding: a new framework," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, pp. 33-41, 2008.
 - [54] 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(1), pp. 72-83, 2006.
 - [55] E. Zacharia and D. Maroulis, "An original genetic approach to the fully automatic gridding of microarray images," *IEEE Transactions on Medical Imaging*, vol. 27, pp. 805-813, 2008.
 - [56] E. Zacharia and D. Maroulis, "Microarray image gridding via an evolutionary algorithm," in *Proc. 15th IEEE International Conference on Image Processing*, San Diego, CA, USA, pp. 1444-1447, 2008.
 - [57] J. Ho, W.-L. Hwang, H. H.-S. Lu, and D. Lee, "Gridding spot centers of smoothly distorted microarray images," *IEEE Transactions on Image Processing*, vol. 15, pp. 342-353, 2006.
 - [58] Y. Lamdan and H. J. Wolfson, "Geometric hashing: A general and efficient model-based recognition scheme," in *Proc. 2nd International Conference on Computer Vision*, Tampa, FL, USA, pp. 238-249, 1988.
 - [59] I. Rigoutsos and R. Hummel, "Implementation of geometric hashing on the connection machine," in *Proc. Workshop on Directions in Automated CAD-Based Vision*, Maui,

- Hawaii, pp. 76-84, 1991.
- [60] V.L. Galinsky, "Automatic registration of microarray images. II. Hexagonal grid," *Bioinformatics*, vol. 19(14), pp. 1832-1836, 2003.
 - [61] 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.
 - [62] Y. Chen, E. R. Dougherty, and M. L. Bittner, "Ratio-based decisions and the quantitative analysis of cDNA microarray images," *Journal of Biomedical Optics*, vol. 2, pp. 364-374, 1997.
 - [63] H. B. Mann and D. R. Whitney, "On a test of whether one of two random variables is stochastically larger than the other," *The Annals of Mathematical Statistics*, vol. 18(1), pp. 50-60, 1947.
 - [64] L. Rueda and L. Qin, "A new method for DNA microarray image segmentation," in *Proc. International Conference Image Analysis and Recognition*, Toronto, ON, Canada, pp. 886-893, 2005.
 - [65] J. Buhler, T. Ideker, and D. Haynor, "Dapple: improved techniques for finding spots on DNA microarrays," Dept. of Computer Science and Engineering, Washington University in St. Louis, Tech. Rep. UWTR 2000-08-05.
 - [66] X. Wang, S. Ghosh, and S.-W. Guo, "Quantitative quality control in microarray image processing and data acquisition," *Nucleic Acids Research*, vol. 29, pp. e75-e75, 2001.
 - [67] Y. H. Yang, M. J. Buckley, S. Dudoit, and T. P. Speed, "Comparison of methods for image analysis on cDNA microarray data," *Journal of Computational and Graphical Statistics*, vol. 11, pp. 108-136, 2002.
 - [68] D. Bozinov and J. Rahnenführer, "Unsupervised technique for robust target separation and analysis of DNA microarray spots through adaptive pixel clustering," *Bioinformatics*, vol. 18, pp. 747-756, 2002.
 - [69] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, Springer, 2001.
 - [70] J.B. MacQueen, "Some methods for classification and analysis of multivariate observations" in *Proc. of 5th Berkeley Symposium on Mathematical Statistics and Probability (University of California Press)*, vol. 1, pp. 281-297, 1967.
 - [71] R. Nagarajan, "Intensity-based segmentation of microarray images," *IEEE Transactions on Medical imaging*, vol. 22, pp. 882-889, 2003.
 - [72] T.F. Chan and L.A. Vese, "Active contours without edges," *IEEE Transactions on Image Processing*, vol. 10(2), pp. 266-277, 2001.
 - [73] K.A. Mendhurwar, R. Kakumani, V. Devabhaktuni, "Microarray image segmentation using chan-vease active contour model and level set method," in *Annual Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 3629-3632, 2009.
 - [74] T. Elguebaly and N. Bouguila, "Bayesian learning of generalized Gaussian mixture models on biomedical images," in *Proc. IAPR Workshop on Artificial Neural Networks in Pattern Recognition*, Cairo, Egypt, pp. 207-218, 2010.
 - [75] R. Nagarajan and M. Upreti, "Correlation statistics for cDNA microarray image analysis," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 3, pp. 232-238, 2006.
 - [76] O. Demirkaya, M. H. Asyali, and M. M. Shoukri, "Segmentation of cDNA microarray spots using markov random field modeling," *Bioinformatics*, vol. 21, pp. 2994-3000, 2005.
 - [77] J. Besag, "On the statistical analysis of dirty pictures," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 48, no. 3, pp. 259-302, 1986.
 - [78] J. Baek, Y. S. Son, and G. J. McLachlan, "Segmentation and intensity estimation of microarray images using a gamma-t mixture model," *Bioinformatics*, vol. 23, pp. 458-465,

- 2007.
- [79] T.-B. Chen, H. H.-S. Lu, Y.-S. Lee, and H.-J. Lan, "Segmentation of cDNA microarray images by kernel density estimation," *Journal of Biomedical Informatics*, vol. 41, pp. 1021-1027, 2008.
 - [80] J. Ho and W.-L. Hwang, "Automatic microarray spot segmentation using a snake-fisher model," *IEEE Transactions on Medical Imaging*, vol. 27, pp. 847-857, 2008.
 - [81] E. I. Athanasiadis, D. A. Cavouras, D. T. Glotsos, P. V. Georgiadis, I. K. Kalatzis, and G. C. Nikiforidis, "Segmentation of complementary DNA microarray images by wavelet-based Markov random field model," *IEEE Transactions on Information Technology in Biomedicine*, vol. 13, pp. 1068-1074, 2009.
 - [82] C.A. Glasbey and P. Ghazal, "Combinatorial Image Analysis of DNA microarray features," *Bioinformatics*, vol. 19(2), pp. 194-203, 2003.
 - [83] P. H. Tran, D. A. Peiffer, Y. Shin, L. M. Meek, J. P. Brody, and K. W. Cho, "Microarray optimizations: increasing spot accuracy and automated identification of true microarray signals," *Nucleic Acids Research*, vol. 30, pp. e54-e54, 2002.
 - [84] Y. Lan, G. Cawley, and R. Harvey, "Train-spotting: building classifiers for microarrays," in *Proc. International Joint Conference on Neural Networks*, Portland, OR, USA, 2003, pp. 2934-2939.
 - [85] S. Hautaniemi, H. Edgren, P. Vesanen, M. Wolf, A.-K. Järvinen, O. Yli-Harja, *et al.*, "A novel strategy for microarray quality control using Bayesian networks," *Bioinformatics*, vol. 19, pp. 2031-2038, 2003.
 - [86] M. Bicego, M. D. R. Martinez, and V. Murino, "A supervised data-driven approach for microarray spot quality classification," *Pattern Analysis and Applications*, vol. 8, pp. 181-187, 2005.
 - [87] L. Nanni, A. Lumini, and S. Brahnam, "Advanced machine learning techniques for microarray spot quality classification," *Neural Computing and Applications*, vol. 19, pp. 471-475, 2010.
 - [88] U. Sauer, C. Preininger, and R. Hany-Schmatzberger, "Quick and simple: quality control of microarray data," *Bioinformatics*, vol. 21, pp. 1572-1578, 2005.
 - [89] C. T. Ekstrøm, S. Bak, C. Kristensen, and M. Rudemo, "Spot shape modelling and data transformations for microarrays," *Bioinformatics*, vol. 20, pp. 2270-2278, 2004.
 - [90] M. Steinfath, W. Wruck, H. Seidel, H. Lehrach, U. Radelof, and J. O'Brien, "Automated image analysis for array hybridization experiments," *Bioinformatics*, vol. 17, pp. 634-641, 2001.
 - [91] C. Kooperberg, T. G. Fazzio, J. J. Delrow, and T. Tsukiyama, "Improved background correction for spotted DNA microarrays," *Journal of Computational Biology*, vol. 9, pp. 55-66, 2002.
 - [92] A. Daskalakis, D. Cavouras, P. Bougioukos, S. Kostopoulos, D. Glotsos, I. Kalatzis, *et al.*, "Improving gene quantification by adjustable spot-image restoration," *Bioinformatics*, vol. 23, pp. 2265-2272, 2007.
 - [93] H. Chui and A. Rangarajan, "A new point matching algorithm for non-rigid registration," *Computer Vision and Image Understanding*, vol. 89, pp. 114-141, 2003.
 - [94] P. M. Griffin and C. Alexopoulos, "Point pattern matching using centroid bounding," in *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 19(5), pp. 1274-1276, 1989.
 - [95] J. Hong and X. Tan, "A new approach to point pattern matching," in *Proc. 9th International Conference on Pattern Recognition*, Rome, Italy, pp. 82-84, 1988.
 - [96] H. Alt, K. Mehlhorn, H. Wager, and E. Welzl, "Congruence, similarity, and symmetries of geometric objects," *Discrete & Computational Geometry*, vol. 3, pp. 237-256, 1988.
 - [97] S. Ghosh, A. Schmidt, and D. Trau, "Fast bead detection and inexact microarray pattern matching for in-situ encoded bead-based array," in *Proc. International Conference on*

- Computer Vision Theory and Applications*, Rome, Italy, pp. 5-14, 2012.
- [98] T. F. Smith and M. S. Waterman, "Identification of common molecular subsequences," *Journal of Molecular Biology*, vol. 147, pp. 195-197, 1981.
 - [99] 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, pp. 381-395, 1981.
 - [100] J. A. Denton and J. R. Beveridge, "An algorithm for projective point matching in the presence of spurious points," *Pattern Recognition*, vol. 40, pp. 586-595, 2007.
 - [101] O. Chum, J. Matas, and J. Kittler, "Locally optimized RANSAC," in *Proc. Joint Pattern Recognition Symposium*, Magdeburg, Germany, pp. 236-243, 2003.
 - [102] J. Matas and O. Chum, "Randomized RANSAC with T d, d test," *Image and Vision Computing*, vol. 22, pp. 837-842, 2004.
 - [103] O. Chum and J. Matas, "Matching with PROSAC-progressive sample consensus," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, San Diego, CA, USA, pp. 220-226, 2005.
 - [104] X. Jiang, K. Broelemann, S. Wachenfeld, and A. Krüger, "Graph-based markerless registration of city maps using geometric hashing," *Computer Vision and Image Understanding*, vol. 115, pp. 1032-1043, 2011.
 - [105] M. Hoffman and M. Lindenbaum, "Some trade-offs and a new algorithm for geometric hashing," in *Proc. 14th International Conference on Pattern Recognition*, Washington, DC, USA, pp. 1700-1704, 1998.
 - [106] S. Irani and P. Raghavan, "Combinatorial and experimental results for randomized point matching algorithms," in *Proc. 12th Annual Symposium on Computational Geometry*, Philadelphia, PA, USA, pp. 68-77, 1996.
 - [107] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," *Image and Vision Computing*, vol. 22, pp. 761-767, 2004.
 - [108] O. Chum and J. Matas, "Geometric hashing with local affine frames," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, New York, NY, USA, pp. 879-884, 2006.
 - [109] T. Nakai, K. Kise, and M. Iwamura, "Use of affine invariants in locally likely arrangement hashing for camera-based document image retrieval," *International Workshop on Document Analysis Systems*, Springer, Berlin, Heidelberg, pp. 541-552, 2006.
 - [110] G. Stockman, "Object recognition and localization via pose clustering," *Computer Vision, Graphics, and Image Processing*, vol. 40, pp. 361-387, 1987.
 - [111] G. Stockman, S. Kopstein, and S. Benett, "Matching images to models for registration and object detection via clustering," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 229-241, 1982.
 - [112] S.-H. Chang, F.-H. Cheng, W.-H. Hsu, and G.-Z. Wu, "Fast algorithm for point pattern matching: invariant to translations, rotations and scale changes," *Pattern Recognition*, vol. 30, pp. 311-320, 1997.
 - [113] T. Manninen, R. Rönkkä, and H. Huttunen, "Point pattern matching for 2-D point sets with regular structure," in *Proc. Scandinavian Conference on Image Analysis*, Ystad Saltsjöbad, Sweden, pp. 156-165, 2011.
 - [114] Y. Lipman and T. Funkhouser, "Möbius voting for surface correspondence," in *ACM Transactions on Graphics (TOG)*, vol. 28(3). Article 72, 2009.
 - [115] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," in *IEEE Transactions of Pattern Analysis and Machine Intelligence*, vol. 14(2), pp. 239-256, 1992.
 - [116] Z. Zhang, "Iterative point matching for registration of free-form curves and surfaces," *International Journal of Computer Vision*, vol. 13, pp. 119-152, 1994.
 - [117] A. E. Johnson and S. B. Kang, "Registration and integration of textured 3D data," *Image*

- and Vision Computing*, vol. 17, pp. 135-147, 1999.
- [118] K. Pulli, "Multiview registration for large data sets," in *Proc. 2nd International Conference on 3-D Digital Imaging and Modeling*, Ottawa, Canada, pp. 160-168, 1999.
 - [119] J. Feldmar and N. Ayache, "Rigid, affine and locally affine registration of free-form surfaces," *International journal of computer vision*, vol. 18, pp. 99-119, 1996.
 - [120] E. Trucco, A. Fusiello, and V. Roberto, "Robust motion and correspondence of noisy 3-D point sets with missing data," *Pattern Recognition Letters*, vol. 20, pp. 889-898, 1999.
 - [121] T. Zinßer, J. Schmidt, and H. Niemann, "A refined ICP algorithm for robust 3-D correspondence estimation," in *Proc. IEEE International Conference on Image Processing*, Barcelona, Spain, pp. 695-698, 2003.
 - [122] L. Maier-Hein, A. M. Franz, T. R. dos Santos, M. Schmidt, M. Fangerau, H.-P. Meinzer, *et al.*, "Convergent iterative closest-point algorithm to accomodate anisotropic and inhomogenous localization error," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, pp. 1520-1532, 2012.
 - [123] A. W. Fitzgibbon, "Robust registration of 2D and 3D point sets," *Image and Vision Computing*, vol. 21, pp. 1145-1153, 2003.
 - [124] K. Levenberg, "A method for the solution of certain non-linear problems in least squares," *Quarterly of Applied Mathematics*, vol. 2, pp. 164-168, 1944.
 - [125] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," in *Proc. 3rd International Conference on 3-D Digital Imaging and Modeling*, Quebec City, QC, Canada, pp. 145-152, 2001.
 - [126] A. Rangarajan, E. Mjolsness, S. Pappu, L. Davachi, P. S. Goldman-Rakic, and J. S. Duncan, "A robust point matching algorithm for autoradiograph alignment," in *Proc. 4th International Conference Visualization in Biomedical Computing*, Hamburg, Germany, vol. 1131, pp. 277-286, 1996.
 - [127] A. Rangarajan, H. Chui, E. Mjolsness, S. Pappu, L. Davachi, P. Goldman-Rakic, *et al.*, "A robust point-matching algorithm for autoradiograph alignment," *Medical Image Analysis*, vol. 1, pp. 379-398, 1997.
 - [128] S. Gold, A. Rangarajan, C.-P. Lu, S. Pappu, and E. Mjolsness, "New algorithms for 2D and 3D point matching: Pose estimation and correspondence," *Pattern Recognition*, vol. 31, pp. 1019-1031, 1998.
 - [129] H. Chui and A. Rangarajan, "A feature registration framework using mixture models," in *Proc. IEEE Workshop on Mathematical Methods in Biomedical Image Analysis*, Hilton Head Island, SC, USA, pp. 190-197, 2000.
 - [130] R. Sinkhorn, "A relationship between arbitrary positive matrices and doubly stochastic matrices," *The Annals of Mathematical Statistics*, vol. 35, pp. 876-879, 1964.
 - [131] C. Peterson and B. Söderberg, "A new method for mapping optimization problems onto neural networks," *International Journal of Neural Systems*, vol. 1, pp. 3-22, 1989.
 - [132] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39(1), pp. 1-38, 1977.
 - [133] A. D. Cross and E. R. Hancock, "Recovering perspective pose with a dual step EM algorithm," *Advances in Neural Information Processing Systems*, pp. 780-786, 1998.
 - [134] A. D. Cross and E. R. Hancock, "Graph matching with a dual-step EM algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, pp. 1236-1253, 1998.
 - [135] B. Delaunay, "Sur la sphere vide," *Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk*, vol. 7, pp. 1-2, 1934.
 - [136] A. Rosenfeld, R. A. Hummel, and S. W. Zucker, "Scene labeling by relaxation operations," in *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 6, pp. 420-433, 1976.
 - [137] S. Ranade and A. Rosenfeld, "Point pattern matching by relaxation," *Pattern Recognition*,

- vol. 12, pp. 269-275, 1980.
- [138] J. Ton and A. K. Jain, "Registering Landsat images by point matching," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 27, pp. 642-651, 1989.
 - [139] S. Z. Li, "Matching: invariant to translations, rotations and scale changes," *Pattern Recognition*, vol. 25, pp. 583-594, 1992.
 - [140] G. L. Scott and H. C. Longuet-Higgins, "An algorithm for associating the features of two images," *Proc. of the Royal Society: Biological Sciences*, vol. 244, pp. 21-26, 1991.
 - [141] L. S. Shapiro and J. M. Brady, "Feature-based correspondence: an eigenvector approach," *Image and Vision Computing*, vol. 10, pp. 283-288, 1992.
 - [142] H. Wang and E. R. Hancock, "A kernel view of spectral point pattern matching," in *Proc. Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition and Structural and Syntactic Pattern Recognition*, Lisbon, Portugal, pp. 361-369, 2004.
 - [143] H. F. Wang and E. R. Hancock, "Correspondence matching using kernel principal components analysis and label consistency constraints," *Pattern Recognition*, vol. 39, pp. 1012-1025, 2006.
 - [144] H. Li and R. Hartley, "A new and compact algorithm for simultaneously matching and estimation," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Montreal, QC, Canada, 2004.
 - [145] S. Srinivasan and M. S. Kankanhalli, "Wide baseline spectral matching," in *Proc. International Conference on Multimedia and Expo*, Baltimore, MD, USA, pp. 93-96, 2003.
 - [146] M. Pilu, "A direct method for stereo correspondence based on singular value decomposition," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, San Juan, PR, USA, pp. 261-266, 1997.
 - [147] C. Zahn, "An algorithm for noisy template matching," CERN, Tech. Rep. CERN-DD-74-11, Mar., 1974.
 - [148] D. Lavine, B. A. Lambird, and L. N. Kanai, "Recognition of spatial point patterns," *Pattern Recognition*, vol. 16, pp. 289-295, 1983.
 - [149] B. Li and H. Holstein, "Using kd trees for robust 3D point pattern matching," in *Proc. 4th International Conference on 3-D Digital Imaging and Modeling*, Banff, AB, Canada, pp. 95-102, 2003.
 - [150] D. P. Huttenlocher and W. J. Rucklidge, "A multi-resolution technique for comparing images using the Hausdorff distance," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, New York, NY, USA, pp. 705-706, 1993.
 - [151] D. M. Mount, N. S. Netanyahu, and J. Le Moigne, "Efficient algorithms for robust feature matching," *Pattern Recognition*, vol. 32, pp. 17-38, 1999.
 - [152] F. Pfeuffer, M. Stiglmayr, and K. Klamroth, "Discrete and geometric branch and bound algorithms for medical image registration," *Annals of Operations Research*, vol. 196, pp. 737-765, 2012.
 - [153] T. S. Caetano, T. Caelli, D. Schuurmans, and D. A. C. Barone, "Graphical models and point pattern matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, pp. 1646-1663, 2006.
 - [154] T. S. Caetano and J. J. McAuley, "Faster graphical models for point-pattern matching," *Spatial Vision*, vol. 22, pp. 443-453, 2009.
 - [155] J. J. McAuley and T. S. Caetano, "Fast matching of large point sets under occlusions," *Pattern Recognition*, vol. 45, pp. 563-569, 2012.
 - [156] J. J. McAuley, T. S. Caetano, and M. S. Barbosa, "Graph rigidity, cyclic belief propagation, and point pattern matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, pp. 2047-2054, 2008.
 - [157] J. Pearl, *Probabilistic reasoning in intelligent systems: networks of plausible inference*, Morgan Kaufmann, 1988.

- [158] N. Ansari, M.-H. Chen, and E. S. Hou, "Point pattern matching by a genetic algorithm," in *Proc. 16th IEEE Industrial Electronics Society Annual Conference*, pp. 1233-1238, 1990.
- [159] L. Zhang, W. Xu, and C. Chang, "Genetic algorithm for affine point pattern matching," *Pattern Recognition Letters*, vol. 24, pp. 9-19, 2003.
- [160] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 509-522, 2002.
- [161] D. Aiger, N. J. Mitra, and D. Cohen-Or, "4-points congruent sets for robust pairwise surface registration," *ACM Transactions on Graphics (TOG)*, vol. 27(3) Article. no. 85, 2008.
- [162] M. R. Garey and D. S. Johnson, *Computers and intractability: a guide to the theory of NP-completeness. A Series of books in the mathematical sciences*, WH Freeman and Company, New York, NY, vol. 25, p. 141, 1979.
- [163] J. R. Ullmann, "An algorithm for subgraph isomorphism," *Journal of the ACM (JACM)*, vol. 23, pp. 31-42, 1976.
- [164] D. E. Ghahraman, A. K. Wong, and T. Au, "Graph optimal monomorphism algorithms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 10(4), pp. 181-188, 1980.
- [165] L. G. Shapiro and R. M. Haralick, "Structural descriptions and inexact matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 3, pp. 504-519, 1981.
- [166] W.-H. Tsai and K.-S. Fu, "Subgraph error-correcting isomorphisms for syntactic Pattern Recognition," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-13, pp. 48-62, 1983.
- [167] L. P. Cordella, P. Foggia, C. Sansone, and M. Vento, "Subgraph transformations for the inexact matching of attributed relational graphs," in *Graph based Representations in Pattern Recognition*, vol. 12, pp. 43-52, 1998.
- [168] S. Berretti, A. Del Bimbo, and E. Vicario, "Efficient matching and indexing of graph models in content-based retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, pp. 1089-1105, 2001.
- [169] S. Umeyama, "An eigendecomposition approach to weighted graph matching problems," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, pp. 695-703, 1988.
- [170] M. Carcassoni and E. R. Hancock, "Weighted graph-matching using modal clusters," in *Proc. International Conference on Computer Analysis of Images and Patterns*, Warsaw, Poland, pp. 142-151, 2001.
- [171] S. Kosinov and T. Caelli, "Inexact multisubgraph matching using graph eigenspace and clustering models," in *Proc. Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition and Structural and Syntactic Pattern Recognition*, Windsor, ON, Canada, pp. 133-142, 2002.
- [172] L. Xu and I. King, "A PCA approach for fast retrieval of structural patterns in attributed graphs," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 31, pp. 812-817, 2001.
- [173] A. Shokoufandeh and S. Dickinson, "A unified framework for indexing and matching hierarchical shape structures," in *Visual Form 2001*, vol. 2059, pp. 67-84, 2001.
- [174] A. Rangarajan, H. Chui, and E. Mjolsness, "A relationship between spline-based deformable models and weighted graphs in non-rigid matching," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Kauai, HI, USA, pp. 897-904, 2001.
- [175] R. Myers, R. Wison, and E. R. Hancock, "Bayesian graph edit distance," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 628-635, 2000.
- [176] S. Medasani, R. Krishnapuram, and Y. Choi, "Graph matching by relaxation of fuzzy assignments," *IEEE Transactions on Fuzzy Systems*, vol. 9, pp. 173-182, 2001.
- [177] A. Torsello and E. R. Hancock, "Computing approximate tree edit distance using relaxation

- labeling," *Pattern Recognition Letters*, vol. 24, pp. 1089-1097, 2003.
- [178] M. Zaslavskiy, F. Bach, and J.-P. Vert, "A path following algorithm for the graph matching problem," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, pp. 2227-2242, 2009.
 - [179] X. Gao, B. Xiao, D. Tao, and X. Li, "A survey of graph edit distance," *Pattern Analysis and Applications*, vol. 13, pp. 113-129, 2010.
 - [180] K. Riesen, M. Neuhaus, and H. Bunke, "Graph embedding in vector spaces by means of prototype selection," in *Proc. International Workshop on Graph-Based Representations in Pattern Recognition*, Alicante, Spain, pp. 383-393, 2007.
 - [181] L. Zou, L. Chen, M. T. Özsu, and D. Zhao, "Answering pattern match queries in large graph databases via graph embedding," *The VLDB Journal—The International Journal on Very Large Data Bases*, vol. 21, pp. 97-120, 2012.
 - [182] D. Haussler, "Convolution kernels on discrete structures," Genomics Institute, Univ. of California Santa Cruz, Tech. Rep. UCSC-CRL-99-10, Jul. 8, 1999.
 - [183] H. Kashima, K. Tsuda, and A. Inokuchi, "Marginalized kernels between labeled graphs," in *Proceedings of the Twentieth Conference on Machine Learning*, Washington DC, pp. 321-328, 2003.
 - [184] H. Kashima and A. Inokuchi, "Kernels for graph classification," in *Proc. IEEE ICDM International Workshop on Active Mining*, Maebashi City, Japan, 2002.
 - [185] K. Khoo and P. Suganthan, "Multiple relational graphs mapping using genetic algorithms," in *Proc. of the 2001 Congress on Evolutionary Computation*, pp. 727-733, 2001.
 - [186] P. Suganthan, "Attributed relational graph matching by neural-gas networks," in *Proc. IEEE Workshop on Neural Networks for Signal Processing*, Sydney, Australia, pp. 366-374, 2000.
 - [187] T. Cour, P. Srinivasan, and J. Shi, "Balanced graph matching," *Advances in Neural Information Processing Systems*, vol. 19, pp. 313-320, 2007.
 - [188] D. Conte, P. Foggia, C. Sansone, and M. Vento, "Thirty years of graph matching in pattern recognition," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 18, pp. 265-298, 2004.
 - [189] G. E. Hinton, C. K. Williams, and M. D. Revow, "Adaptive elastic models for hand-printed character recognition," in *Advances in Neural Information Processing Systems*, pp. 512-519, 1991.
 - [190] M. Revow, C. K. Williams, and G. E. Hinton, "Using generative models for handwritten digit recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, pp. 592-606, 1996.
 - [191] S. Pappu, S. Gold, and A. Rangarajan, "A Framework for Non-rigid Matching and Correspondence," *Advances in Neural Information Processing Systems*, pp. 795-801, 1996.
 - [192] H. Chui, "Non-rigid point matching: algorithms, extensions and applications," Ph.D. dissertation, Yale University, CT, USA, 2001.
 - [193] Y. Hao, T. Tan, and Y. Wang, "Fingerprint matching based on error propagation," in *Proc. IEEE International Conference on Image Processing*, Rochester, NY, USA, pp. 273-276, 2002.
 - [194] A. Pitiot, G. Malandain, E. Bardinet, and P. M. Thompson, "Piecewise affine registration of biological images," in *Proc. 2nd International Workshop on Biomedical Image Registration*, Philadelphia, PA, USA, pp. 91-101, 2003.
 - [195] L. Kaufman and P.J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*, 1st ed, John Wiley, 1990.
 - [196] G. Seetharaman, G. Gasperas, and K. Palaniappan, "A piecewise affine model for image registration in nonrigid motion analysis," in *Proc. International Conference on Image Processing*, Vancouver, BC, Canada, pp. 561-564, 2000.
 - [197] J. K. Aggarwal, Q. Cai, W. Liao, and B. Sabata, "Articulated and elastic non-rigid motion:

- A review," in *Proc. IEEE Workshop on Motion of Non-Rigid and Articulated Objects*, San Juan, Puerto Rico, USA, pp. 2-14, 1994.
- [198] A. Myronenko and X. Song, "Point set registration: Coherent point drift," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, pp. 2262-2275, 2010.
 - [199] A. Joshi and C.-H. Lee, "On the problem of correspondence in range data and some inelastic uses for elastic nets," *IEEE Transactions on Neural Networks*, vol. 6, pp. 716-723, 1995.
 - [200] R. Durbin, "An analogue approach to the travelling salesman," *Nature*, vol. 326, p. 16, 1987.
 - [201] B. Luo and E. R. Hancock, "Structural graph matching using the EM algorithm and singular value decomposition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, pp. 1120-1136, 2001.
 - [202] G. McNeill and S. Vijayakumar, "A probabilistic approach to robust shape matching," in *Proc. IEEE International Conference on Image Processing*, Atlanta, GA, USA, pp. 937-940, 2006.
 - [203] Y. Tsin and T. Kanade, "A correlation-based approach to robust point set registration," in *Proc. European Conference on Computer Vision*, Prague, Czech Republic, pp. 558-569, 2004.
 - [204] B. Jian and B. C. Vemuri, "A robust algorithm for point set registration using mixture of Gaussians," in *Proc. 10th IEEE International Conference on Computer Vision*, Beijing, China, pp. 1246-1251, 2005.
 - [205] A. Vedaldi and S. Soatto, "Quick shift and kernel methods for mode seeking," in *Proc. European Conference on Computer Vision*, Marseille, France, pp. 705-718, 2008.
 - [206] 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, pp. 32-40, 1975.
 - [207] C.V. Loan, *Computational frameworks for the fast Fourier transform*, Society for Industrial and Applied Mathematics, vol. 10, 1992.
 - [208] N. Otsu, "A threshold selection method from gray-level histograms," *Automatica*, vol. 11, pp. 23-27, 1975.
 - [209] D. Cox and P. Lewis, *The statistical analysis of series of events*, Methuen & Co., 1966.
 - [210] R. T. Collins, "Mean-shift blob tracking through scale space," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Madison, WI, USA, pp. II-234-240, 2003.
 - [211] 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, pp. 1379-1394, 2003.
 - [212] D. Chetverikov, D. Svirko, D. Stepanov, and P. Krsek, "The trimmed iterative closest point algorithm," in *Proc. 16th International Conference on Pattern Recognition*, Quebec City, QC, Canada, pp. 545-548, 2002.
 - [213] S. Martelli, M. Cristani, L. Bazzani, D. Tosato, and V. Murino, "Joining feature-based and similarity-based pattern description paradigms for object detection," in *Proc. 21st International Conference on Pattern Recognition*, Tsukuba Science City, Japan, pp. 2702-2705, 2012.
 - [214] U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," *Physical Review E*, vol. 76, p. 036106, 2007.
 - [215] V. Premachandran and R. Kakarala, "Consensus of k-nns for robust neighborhood selection on graph-based manifolds," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Portland, OR, USA, pp. 1594-1601, 2013.
 - [216] M. Pavan and M. Pelillo, "Dominant sets and pairwise clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, pp. 167-172, 2007.

- [217] J. Cohen, *Statistical power analysis: A Computer Program*, Taylor & Francis, 1988.
- [218] E. L. Lehmann and H. D'abrerera, *Nonparametrics: statistical methods based on ranks*, Springer, 1975.
- [219] K.O.McGraw and S.P.Wong, "A common language effect size statistic," in *Psychological Bulletin*, vol. 111(2), pp. 361-365, 1992.
- [220] A. P. Moore, S. J. Prince, J. Warrell, U. Mohammed, and G. Jones, "Superpixel lattices," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Anchorage, AK, USA, pp. 1-8, 2008.
- [221] C. Arteta, V. Lempitsky, J. A. Noble, and A. Zisserman, "Learning to detect cells using non-overlapping extremal regions," in *Proc. International Conference on Medical Image Computing and Computer-Assisted Intervention*, Nice, France, pp. 348-356, 2012.
- [222] T. Nguyen, "Gaussian Mixture Model based Spatial Information Concept for Image Segmentation," Ph.D. dissertation, Dept. of Electrical and Computer Engineering, University of Windsor, Ontario, Canada, 2011.
- [223] T. H. Cormen, C.E. Leiserson, R.L.Rivest, *Introduction to Algorithms*, MIT Press and McGraw-Hill, 1st ed., 1990.
- [224] A. Vedaldi and B. Fulkerson, "VLFeat : An open and portable library of computer vision algorithms," 2008, <http://www.vlfeat.org/>.
- [225] "Edge detection and image segmentation (EDISON) system," 2003, <http://coewww.rutgers.edu/riul/research/code/EDISON/index.html>.
- [226] V. Kolmogorov, "Max-flow/min-cut," 2010, <http://vision.csd.uwo.ca/code/>.
- [227] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2(1-2), pp. 83-97, 1955.
- [228] A. Datta, A. W.-K. 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, 2013.