

# A New Geometric Biclustering Algorithm based on the Hough Transform for Analysis of Large-Scale Microarray Data

Hongya Zhao<sup>§</sup>, Alan W.C. Liew, Xudong Xie and Hong Yan

H. Zhao is with the Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong. E-mail: hyzhao@ee.cityu.edu.hk.

A.W.C. Liew is with the School of Information & Communication Technology, Griffith University, Brisbane, Australia. E-mail: a.liew@griffith.edu.au.

X. Xie is with the Department of Electronic Engineering, Tsinghua University, Beijing, China. E-mail: xudongxie@hotmail.com.

H. Yan is with the Department of Electronic and Engineering, City University of Hong Kong, Kowloon, Hong Kong and with the School of Electronic and Information Engineering, University of Sydney, NSW2006, Sydney, Australia. E-mail: h.yan@cityu.edu.hk.

<sup>§</sup>Corresponding author

# A New Geometric Biclustering Algorithm based on the Hough Transform for Analysis of Large-Scale Microarray Data

Hongya Zhao, Alan W.C. Liew, Xudong Xie and Hong Yan

**Abstract** — Biclustering is an important tool in microarray analysis when only a subset of genes co-regulates in a subset of conditions. Different from standard clustering analyses, biclustering performs simultaneous classification in both gene and condition directions in a microarray data matrix. However, the biclustering problem is inherently intractable and computationally complex. In this paper, we present a new biclustering algorithm based on the geometrical viewpoint of coherent gene expression profiles. In this method, we perform pattern identification based on the Hough transform in a column-pair space. The algorithm is especially suitable for the biclustering analysis of large-scale microarray data. Our studies show that the approach can discover significant biclusters with respect to the increased noise level and regulatory complexity. Furthermore, we also test the ability of our method to locate biologically verifiable biclusters within an annotated set of genes.

**Index Terms**— Microarray data analysis, gene expression profiles, biclustering, the Hough transform



## 1 INTRODUCTION

DNA microarray technology is a high-throughput and parallel platform that can provide expression profiling of thousands of genes in different biological conditions, thereby enabling the rapid and quantitative analysis of gene expression patterns on a global scale [1, 32]. These experimental data are usually arranged in a matrix, where each row corresponds to a gene and each column an experimental condition. Each entry

in the matrix records the expression level of a gene as a real number, which is usually derived by taking the logarithm of the relative abundance of the mRNA of that gene in a specific condition [27].

Clustering techniques can be applied to either gene or condition direction to investigate the underlying structure of gene expression datasets [9, 10, 11, 15, 25, 29, 38]. Nevertheless, most of them only consider global similarity between expression profiles. They typically assume that the related genes behave similarly across all measured conditions and the conserved condition patterns run across all measured genes. The clusters produced by these methods reflect the global patterns of expression data, but an interesting cellular process for most cases may be only involved in a subset of genes co-expressed only under a subset of conditions. Discovering such local expression patterns may be the key to uncovering many genetic pathways that are not apparent otherwise. Therefore, it is highly desirable to move beyond the clustering paradigm, and to develop approaches capable of discovering local patterns in microarray data [8, 26, 30, 34].

Inspired by Hartigan's so called "direct clustering" [17], the termed biclustering was first introduced to gene expression analysis by Cheng and Church [8]. In general, biclustering refers to the simultaneous clustering on the row and column dimensions of the data matrix [26]. The technique is more compatible with our understanding of cellular process: the related genes are considered to be regulated in a synchronized fashion and under certain conditions, but to behave almost independently under other conditions. The biclustering methods may be useful in recognizing reusable genetic "modules" that are mixed and matched in order to create more complex genetic response.

The research literature on biclustering has been booming in recent years. Existing biclustering methods include  $\delta$ -biclusters [8, 42], flexible overlapped biclustering algorithm (FLOC) [43] gene shaving [18], order-preserving sub-matrix (OPSM) [3],

spectral biclustering [23], interrelated two-way clustering (ITWC) [35], double conjugated clustering (DCC) [6], coupled two-way clustering (CTWC) [14], statistical-algorithmic method for bicluster analysis algorithm (SAMBA) [33], iterative signature algorithm (ISA) [19, 20], xMOTIF [28, 39], plaid model [24, 37], a fast divide-and-conquer algorithm (Bimax) [30], and maximum similarity biclustering (MSBE) [26]. Comprehensive surveys about the biclustering algorithms can be found in [27] and [34]. A systematic comparison of some biclustering methods is made in [30]. In general, existing algorithms perform biclustering by adding or deleting rows and/or columns in the data matrix in optimal ways such that a merit function is improved by the action [12].

A different viewpoint of the biclustering can be formulated in terms of the spatial geometrical distribution of points in data space. The biclustering problem is tackled as the identification and division of coherent sub-matrices of data matrices into geometrical structures (lines or planes) in a multidimensional data space [12]. Such perspective makes it possible to unify the formulations of different biclusters and detect them using the algorithms of detecting geometric patterns, such as lines and planes. This approach is radically different from all other biclustering approaches that are based on optimization of merit functions.

In the geometric biclustering algorithm, the well-known Hough transform (HT) is employed to detect lines and planes. Statistical properties of the HT, such as robustness, consistency and convergence, make it more suitable for biclustering analysis of microarray data than the traditional methods [16]. Especially, it is noted for its ability to identify geometric patterns in noisy data because noise is one of the major issues in microarray data analysis [16, 21, 41].

A HT based algorithm, HoughFeature, has also been developed to analyze three-color microarray data, which involve three experimental conditions [46]. However, the original HT based algorithm becomes ineffective, in terms of both computing time and

storage space, as the number of conditions increases.

To overcome the difficulties, a novel strategy is proposed in this paper. In our algorithm, the HT is only performed in a two-dimensional (2-D) space of column pairs [45]. The coherent columns are then combined iteratively to form larger and larger biclusters. This reduces the computational complexity considerably and makes it possible to analyze large-scale microarray data.

The paper is organized as follows. In Section 2, we show that different types of biclusters can be formulated in the same way using linear equations in 2-D column-pair spaces. Based on this premise, a visualization tool, the additive and multiplicative pattern plot (AMPP), is proposed to separate sub-biclusters into different bicluster patterns. Then, Section 3 presents the complete biclustering algorithm based on the HT and the AMPP. In Section 4, the characteristics of the algorithm are studied using simulated data. In Section 5, we apply the algorithm to bicluster gene expression data of *saccharomyces cerevisiae* and multiple human organs. To compare our method with other biclustering algorithms, we use a quantitative measure based on gene ontology and KEGG pathway annotation, to assess the biological relevance of biclusters. The paper is concluded in Section 6.

## **2 BICLUSTER PATTERNS IN COLUMN-PAIR SPACES**

Biclustering of gene expression is of particular interest in microarray analysis. The underlying biological rationales for biclustering include (1) only a subset of genes participates in a cellular process, (2) a process is active only in a subset of conditions, and (3) a gene may participate in multiple pathways that may or may not co-act under all conditions. With biclustering algorithm, one can identify a subset of genes that are co-regulated under a subset of conditions [26, 27].

In the biclustering literature, various types of coherent patterns have been defined to capture important biological phenomena. In this paper, we focus on five coherent types

be inferred from their genes and conditions relations. For example, in a constant bicluster, a subset of genes always displays the same expression level under a subset of conditions. In an additive bicluster, expression levels of a subset of genes under one condition are always higher (or lower) by a constant than under another condition. The study of common fluctuations of the expression levels in these biclusters is useful in practical applications, such as cancer classification [29].

Table 1. Types of biclusters, from left to right: (a) constant, (b) constant rows, (c) constant columns, (d) additive coherent values, and (e) multiplicative coherent values.

$x_1$	$x_2$	$x_3$	$x_4$	$x_1$	$x_2$	$x_3$	$x_4$	$x_1$	$x_2$	$x_3$	$x_4$	$x_1$	$x_2$	$x_3$	$x_4$	$x_1$	$x_2$	$x_3$	$x_4$
10	10	10	10	50	50	50	50	10	25	35	50	15	12	20	35	2	6	4	12
25	25	25	25	50	50	50	50	10	25	35	50	20	17	25	40	5	15	10	30
35	35	35	35	50	50	50	50	10	25	35	50	25	22	30	45	6	18	12	36
50	50	50	50	50	50	50	50	10	25	35	50	30	27	35	50	8	24	16	48
$x_i=x_j$				$x_i=x_j=a$				$x_i=a_i, x_j=a_j (a_i \neq a_j)$				$x_j=x_i+b_{ij} (b_{ij} \neq 0)$				$x_j=a_{ij}x_i (a_{ij} \neq 1)$			

Now we reconsider the biclusters in Table 1 from the geometrical viewpoint. We denote the expression vector of the subset of genes under condition the as  $x_i$ . Then a constant bicluster, for example in Table 1, should satisfy the equation  $x_1 = x_2 = x_3 = x_4 = a$ , a constant row bicluster corresponds to  $x_1 = x_2 = x_3 = x_4$ , and an additive pattern satisfies  $x_1 = x_2 + b_{12} = x_3 + b_{13} = x_4 + b_{14}$ , where  $a$ ,  $b_{12}$ ,  $b_{13}$  and  $b_{14}$  are constant vectors. If the conditions span a high dimensional space, the expression of every gene corresponds to a point in the space. The five different patterns in Table 1 can be uniquely mapped to points, lines or planes in the space. In general when the biclusters are embedded in a larger data matrix, the points or lines defined by the bicluster would sweep out a hyperplane in the spatial space [12]. The perspective unifies the definition of all coherent biclusters. Thus, a generic plane finding algorithms can be applied to identify the biclusters in microarray data. For example, the well-known Hough technology in computer vision can be employed. However, computational complexity makes it difficult to deal with all genes and conditions at the same time in large-scale microarray

data.

To reduce the computational complexity, we start the Hough algorithm in every 2-D condition space, called the column-pair space. A splitting technique is successfully used to cope with the NP-hard problem of biclustering [38, 44]. Considering the number of conditions significantly less than that of genes in microarray matrix, we only split the conditions and form the maximal biclusters by combination in our algorithm. For example, instead of find a pattern satisfying  $x_1 = x_2 = x_3 = x_4$  in a 4-D space, we find patterns satisfying  $x_1 = x_2$ ,  $x_2 = x_3$ ,  $x_3 = x_4$  in three 2-D spaces and then combine them. Therefore, we start our algorithm from all  $n(n-1)/2$  unique column-pairs of the microarray data matrix. Instead of searching through all genes and conditions at the same time, we identify the defined bicluster patterns in column-pair spaces and then combine these sub-biclusters with common conditions and genes to form a complete bicluster. The computational complexity of this approach is proportional to the number of column pairs.

### **3 GEOMETRIC BI-CLUSTERING ALGORITHM**

Based on the linear structures discussed above, we propose a new biclustering algorithm. First, we identify genes of interest with linear structures in column-pair spaces and divide them into different patterns using the AMPP described below. These sub-biclusters are combined step by step to form large biclusters. Line detection in column-pair space is a crucial step in the proposed framework and we employ the HT for this task. The HT is a powerful technique widely used for line detection in digital images [2, 21, 41]. In this section, we first introduce the HT and the AMPP, and then present the geometric biclustering algorithm.

### 3.1 The Hough Transform for Line Detection

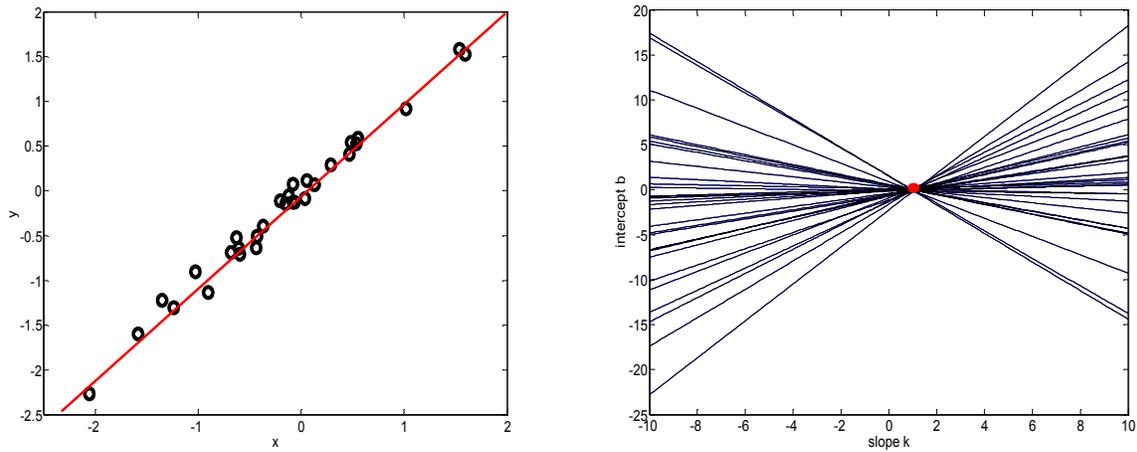


Figure 1: (a) points and regression line in data space (left); (b) corresponding lines and a point drawn in parameter space (right). Points on a line in data space produce lines that intersect at the same point in the parameter space, while random points in the data space produce random lines in the parameter space.

The HT is a methodology that detects analytic lines and curves in images through a voting process in the parameter space. A line in the x-y data space is defined by

$$y = kx + b \quad (1)$$

which corresponds to the point  $(k, b)$  in the k-b parameter space. Conversely, the line  $b = xk + y$  in the k-b space corresponds to the point  $(x, y)$  in the x-y space. The basic idea of HT can be informally described as follows. Consider  $n$  data points  $\{(x_i, y_i): i=1, \dots, n\}$  depicted in Figure 1 (a). The objective is to infer the parameters of the line that fits the data optimally. Based on the HT, the  $n$  points can be mapped to  $n$  lines  $\{(k_i, b_i): b_i = x_i k_i + y_i, i = 1, \dots, n\}$  in the Hough space, demonstrated in Figure 1(b). Thus, the co-linearity in the original points will manifest itself in a common intersection of lines in the Hough domain. To obtain the intersection of lines, the Hough domain is first quantized into cells, and each such cell maintains a count of the number of intersecting lines. The cell with the largest number of counts is the obvious estimator of the parameters of the original line [16].

Table 2. The geometrical structures of the five biclustering types in column-pair space.

Type	Equation	Pattern in data space	Pattern in parameter space $[k, b]$	Pattern in polar space $[\theta, \rho]$
Constant (C)	$x_i = x_j = a$	A point on diagonal line	A line passing $[1,0]$	A sinusoidal curve passing $[-\pi/4,0]$
Constant Rows (R)	$x_i = x_j$	Points on diagonal line	Lines passing $[1,0]$	Sinusoidal curves passing $[-\pi/4,0]$
Constant Columns (O)	$x_i = a_i, x_j = a_j$ ( $a_i \neq a_j$ )	A point off diagonal line	A line passing $[k,0]$ ( $k \neq 1$ )	A sinusoidal curve passing $[\theta,0]$ ( $\theta \neq -\pi/4$ )
Additive (A)	$x_j = x_i + b_{ij}$ ( $b_{ij} \neq 0$ )	Points off diagonal line	Lines passing $[1,b]$ ( $b \neq 0$ )	A sinusoidal curve passing $[-\pi/4, \rho]$ ( $\rho \neq 0$ )
Multiplicative (M)	$x_j = a_{ij}x_i$ ( $a_{ij} \neq 1$ )	Points on the straight line passing origin	Lines passing $[k,0]$ ( $k \neq 1$ )	Sinusoidal curves passing $[\theta,0]$ ( $\theta \neq -\pi/4$ )

Assume that  $k$  and  $b$  are each quantized into  $Q$  steps, then  $Q^2$  cells or accumulators are needed to detect lines in a 2-D space. To detect an  $n$ -D hyperplane, the number of accumulators required is  $Q^n$ , which increases exponentially as  $n$  increases. In microarray experiments, the number of conditions  $n$  can be in the order of tens to hundreds. Clearly, it is infeasible to detect biclusters with all conditions considered together. In the column-pair based approach presented in this paper, we convert the problem into simpler ones, each involving two variables only. The number of accumulators required becomes  $\underline{n}(n-1)Q^2$  in our method.

In practice, the HT is implemented in polar parameter space [2, 16, 21]. The range of quantized cells in parameter space may be very large because the dynamic range of the slope in data space is large, even infinite for vertical lines. Alternatively, the polar form can be used to describe a line:

$$\rho = x \cos \theta + y \sin \theta \quad (2)$$

where  $\rho$  is the distance of a line to the origin and  $\theta$  is the angle of the normal to the  $x$  axis. Since  $\rho$  is limited from  $-\sqrt{x^2 + y^2}$  to  $\sqrt{x^2 + y^2}$  and  $\theta$  is limited from  $-\pi/2$  to  $\pi/2$ , the ranges of the slope and intercept are compressed and a small number of quantized cells is sufficient to find all lines. Note that if the polar equation of a line is used, for each point in the  $x$ - $y$  space, a sinusoidal curve rather than a line can be drawn in the Hough space. Table 2 summarizes the five biclustering patterns in a column-pair space

and the corresponding geometrical structures in the polar parameter space.

### 3.2 Additive and Multiplicative Pattern Plot (AMPP)

In Table 2, it is obvious that constant row (R) and column (O) biclusters R are a special cases of additive (A) and multiplicative (M) ones and constant (C) biclusters are special cases of all other types, R, O, A and M. Additive and multiplicative patterns are two general types, so the separation of the two patterns is of concern in biclustering algorithms.

Considering the collinear points detected by the HT in a column-pair data space, we need to classify them into additive or multiplicative patterns. In our algorithm we develop a visualization tool, the AMPP for this task. The AMPP is implemented as follows.

Given  $\{(x_i, y_i): I = 1, \dots, k\}$ , we assume that there are  $k$  points on a line detected using the HT in a column-pair space. Now we try to separate the points into two types of biclusters. We employ  $d_i = x_i - y_i$  and  $r_i = \arctan(x_i/y_i)$  to show the difference between the additive and multiplicative patterns in the AMPP as demonstrated in Fig. 2. The horizontal axis of the AMPP represents the change of additive patterns  $d_i$ , and the vertical axis the multiplicative patterns  $r_i$ . In AMPP, we use  $r_i = \arctan(x_i/y_i)$  instead of the direct ratio  $x_i/y_i$  to reduce the dynamic range of the ratio.

Based on the AMPP, the boxplot is used to separate the points. The boxplot was first proposed by J. Tukey, as a simple graphical summary of the distribution of variables [6]. In a boxplot, the middle line represents the median, and the lower and upper sides of the rectangle show the medians of the lower and upper halves of the data. Along the horizontal boxplot, the points in the box are considered to be shifted with their median in an A pattern and the points in box of the vertical boxplot are considered to be multiplied by their median in an M pattern. The points in their intersect set are considered as the overlapped points in the two patterns, that is, a C pattern. The

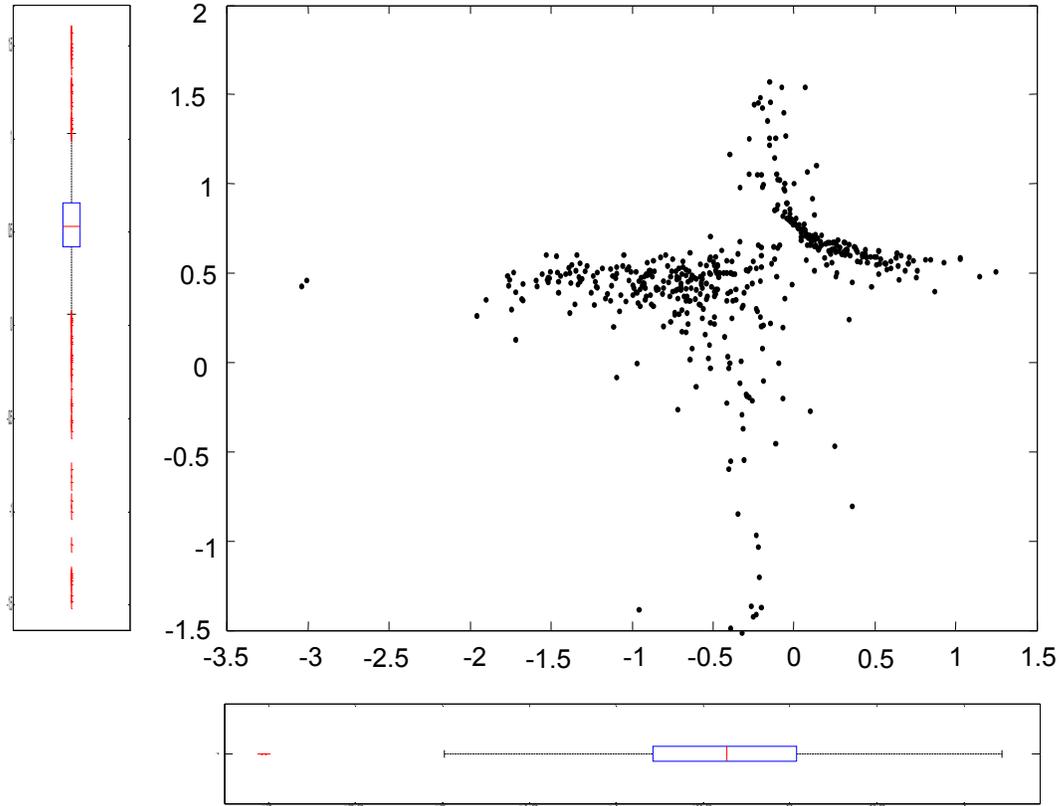


Figure 2: The additive and multiplicative pattern plot (AMPP) of microarray data.  $d_i = x_i - y_i$  and  $r_i = \arctan(x_i / y_i)$  are scaled as horizontal and vertical axes respectively. The corresponding points in the boxes of the boxplots are divided into two models.

horizontal and vertical boxplots are also shown in Fig. 2. The visualization plot can classify the points detected using the HT in a column-pair space.

### 3.3 Geometrical Biclustering Algorithm (GBC)

Given the expression data matrix  $\mathbf{D}_{N \times n}$  with  $N$  genes and  $n$  experimental conditions, we denote the index of rows (genes) as  $G = \{g_1, \dots, g_N\}$  and the index of columns (conditions) as  $C = \{c_1, \dots, c_N\}$ . An  $s \times t$  sub-matrix can be denoted as  $B = (I, J)$ , where  $I = \{g_{i_1}, \dots, g_{i_s}\}$  is a subset of  $G$  and  $J = \{c_{j_1}, \dots, c_{j_t}\}$  is a subset of  $C$ . Based on the HT and AMPP applied to the column-pair spaces, we propose the following algorithm to identify a set of maximal biclusters, where a sub-bicluster  $B = (I, J)$  is defined as a maximal one if and only if no  $\mathbf{T}'$  exists such that  $\mathbf{T} \subset \mathbf{T}'$  (that is  $\mathbf{I} \subset \mathbf{I}'$  and  $\mathbf{J} \subset \mathbf{J}'$ ) [27].

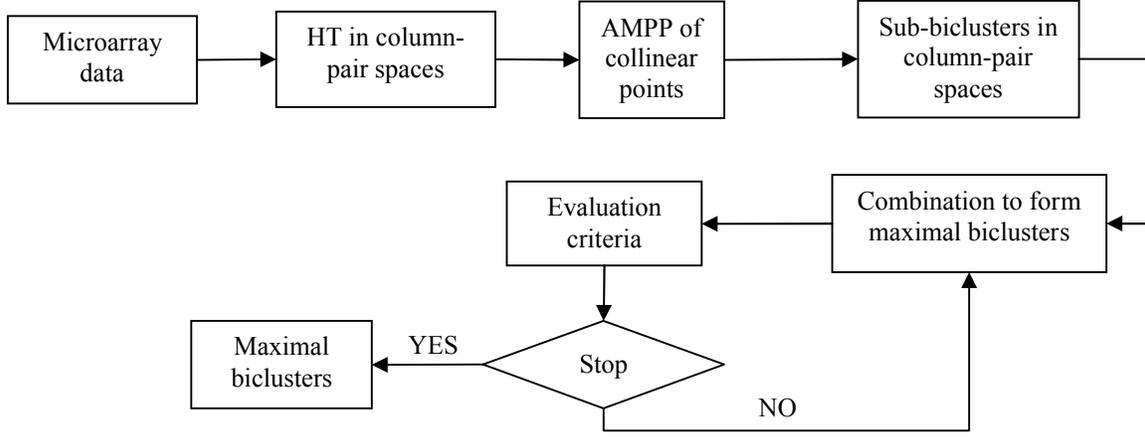


Figure 3 The overall flow diagram of the GBC algorithm.

Figure 3 shows a flowchart of the proposed method. As described in Section 3.1, the HT is used to detect the lines in all column-pair space and record  $n(n-1)/2$  sub-biclusters  $B_{ij}=(G_{ij},\{i,j\})$ . Obviously, these sub-biclusters are respectively the maximal ones in their column-pair space related to two conditions. We classify  $B_{ij}$ s into the additive or multiplicative types using the AMPP discussed above.

The next step is to combine the sub-biclusters into maximal ones. The following property of sub-biclusters provides a solution to the merging step: Let  $B_{\max}=(G_{\max},C_{\max})$  be one maximal bicluster and  $\{T_i=(I_i,J_i)\}$  be a set of maximal sub-biclusters in column-pair space. If  $J_i \subseteq C_{\max}$ , then  $G_{\max} \subseteq I_i$  [44]. So given two sub-biclusters  $B_s=(I_s,J_s)$  and  $B_t=(I_t,J_t)$ , we combine them using the operation  $\mathbf{B}_r = \mathbf{B}_s \oplus \mathbf{B}_t = (I_s \cap I_t, J_s \cup J_t)$  to form a larger one. The number of genes in the biclusters becomes smaller and smaller with the combination.

We stop the combination after all sub-biclusters are considered or if the number of genes in the merged biclusters is less than the given parameter  $\delta$ . We also filtered out biclusters whose number of conditions is less than given parameter  $\zeta$ . The overall GBC algorithm is summarized in Table 3.

---



---

Table 3: The GBC Algorithm

---

Input: Microarray data  $D(G,C)$

Input:  $q$ , quantization step size in the HT parameter space.

Input:  $\delta$ , the minimum number of genes to form a bicluster.

Input:  $\zeta$ , the minimum number of conditions to form a bicluster.

Output: biclusters

HT\_ALG: perform Hough transformation in a column-pair space

AMPP\_ALG: classify the collinear points in a column-pair space

Step 1: Perform the HT in all column-pair spaces to form sub-biclusters

for  $\forall i,j \in C$

$[G_{ij}, C_{ij}] = \text{HT\_ALG}(D(G,i), D(G,j), q)$ ;

Step 2: Use the AMPP to classify the collinear points

$[B_{ij\_Cons}, B_{ij\_Add}, B_{ij\_Mul}] = \text{AMPP\_ALG}(G_{ij}, C_{ij})$ ;

Step 3: Combine sub-biclusters

for  $\forall B^i\_Cons=(G^i\_Cons, C^i\_Cons)$  and  $B^j\_Cons=(G^j\_Cons, C^j\_Cons) \in \{B_{ij\_Cons}\}$   
 ( $\{B_{ij\_Add}\}$  or  $\{B_{ij\_Mul}\}$ )

if  $C^i\_Cons \cap C^j\_Cons \neq \emptyset$

$C^{ij}\_Cons = C^i\_Cons \cup C^j\_Cons$ ;

$G^{ij}\_Cons = G^i\_Cons \cap G^j\_Cons$ ;

Step 4: Filter biclusters

If  $\|G^{ij}\_Cons\| > \delta$  and  $\|C^{ij}\_Cons\| > \zeta$

$B^{ij}\_Cons = (G^i\_Cons, C^i\_Cons)$ ;

If  $B^{ij}\_Cons$  is completely overlapped by another bicluster

remove  $B^{ij}\_Cons$ .

---

## 4 SIMULATION STUDY

The following two important questions have been investigated in the simulation study: whether the algorithm is robust against noise and whether it has the ability to identify multiple overlapping biclusters.

In order to evaluate the performance of different biclustering methods, we define a matching score similar to the ones used in [26, 30]. Let  $B_1 = (G_1, C_1)$  and  $B_2 = (G_2, C_2)$  be two sets of biclusters. The gene matching score  $S(B_1, B_2)$  based on the Jaccard coefficient is first defined in [30]. Recently, the score is improved in [26] by adding the condition dimensions of biclusters.

The original Jaccard coefficient is symmetric. However, neither of the matching

scores satisfies the good property and usually yield different values when  $B_1$  and  $B_2$  are exchanged. So it is impossible to use the two scores to evaluate how well the true biclusters are recovered although they can reflect to what extent the generated biclusters represent the true ones [30]. Furthermore the number of the detected biclusters is much larger than that of the true ones, and thus the recovery of true biclusters is essential to the evaluation with matching scores as well as the representation in simulation studies. The two properties of matching scores should be satisfied simultaneously in the system of evaluation.

Based on the original matching scores, we define the following score

$$S(B_1, B_2) = \max_{B_1} \max_{B_2} \frac{|G_1 \cap G_2| + |C_1 \cap C_2|}{|G_1 \cup G_2| + |C_1 \cup C_2|}$$

which is symmetric about  $B_1$  and  $B_2$ . This score is more consistent with the cases in experimental data analysis because the real underlying patterns of gene expression are completely unknown in microarray experiments. Therefore, it is enough in the real analysis to select some best and meaningful biclusters from the large number of detected ones as candidates for the further biological verification.

In our following simulation, we denote  $B_{\text{opt}}$  as the set of implanted biclusters and  $B$  as the set of resulting biclusters produced by a biclustering method.  $S(B_{\text{opt}}, B)$  (or equivalently  $S(B, B_{\text{opt}})$ ) represents the best cases that each of the true biclusters is identified by the algorithm.

#### 4.1 Effects of noise

We first investigate the performance of our algorithm with noisy data in comparison with other biclustering algorithms. Here we consider the following algorithms, CC [8], ISA [19, 20], OPSM [3], Bimax [30] which can be downloaded from the software toolbox BicAT [www.tik.ee.ethz.ch/sop/bimax](http://www.tik.ee.ethz.ch/sop/bimax) [5] and the newest MSBE which can be

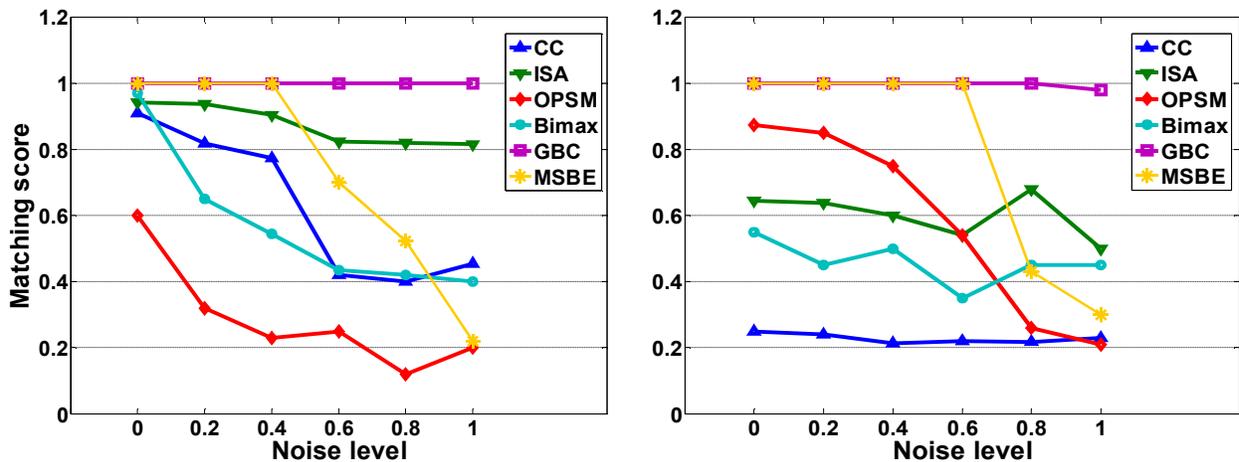


Figure 4: Results of the simulation study for non-overlapping constant biclusters (left) and additive (right) biclusters with different noise levels.

downloaded from [www.cs.cityu.edu.hk/~liuxw/msbe/help.html](http://www.cs.cityu.edu.hk/~liuxw/msbe/help.html) [26]. We use these algorithms with the default parameters set in those methods. The model used to generate synthetic gene expression data is similar to the approach described in [12]. Two types of biclusters are embedded in the microarray matrix, constant and additive patterns.

In the first case, we embed 10 non-overlapping constant biclusters of  $10 \times 10$  into a dataset of size  $100 \times 100$ . The background matrix is generated from the uniform distribution  $U(-5, 5)$ . Gaussian noise with variance from 0 to 1 is generated to degrade these constant biclusters. Similar to the constant case, we implant 10 non-overlapping additive biclusters of size  $10 \times 10$  into the dataset randomly obtained from  $U(-5, 5)$  in the second case. The additive factors of every column are randomly obtained from  $U(-5, 5)$ . Gaussian noise with variance from 0 to 1 is generated to degrade these additive patterns.

Figure 4 summarizes the performance of different biclustering methods. For constant biclusters, all algorithms have the highest scores with the noiseless data. Even all of them but OPSM can find more than 95% of the implanted biclusters. With the increasing of noise levels, ISA and GBC still keep good performance, but the matching

scores of the other methods significantly decreased, especially MSBE from 1 to 0.2. The proposed method GBC performs well as it can identify the all biclusters for all six noise levels tested and produces a perfect matching score  $S = 1$ . The HT is well-known to be robust against noise and this why GBC, which is developed based on the HT, has a superior performance. ISA has a relative stable performance and the matching score changes only slightly from 1 to 0.8 as noise level increases because the reference gene sets in this method play an important role to decrease the effect of noise [19, 20].

In contrast, the performances of CC, Bimax, MSBE, and OPSM are very sensitive to the noise in the constant biclusters. The CC algorithm computes the similarity of the selected gene expression data only and can easily be trapped at local optimal points [8]. To implement Bimax, the synthetic data should be discretized to binary values with the pre-defined threshold or percentage. In this experiment, we set the top 10% altered genes to value 1 and the rest to value 0 in the simulation study. Since noise blurs the difference between background and biclusters, the binarization process can degrade the biclustering performance [30]. The score for OPSM changes from 0.6 to 0.2. The algorithm seeks the clear trends of up- or down-regulated genes and may not work well with constant biclusters [3]. However, OPSM is more suitable to identify additive biclusters because the changes along the condition direction represent optimal order preserving sub-matrices. However, the significant decrease of the performances may be caused by the greedy algorithm in OPSM: only a single bicluster is considered for the linear ordering of the columns.

For additive biclusters, the proposed GBC algorithm also shows better performance than Bimax, CC, ISA, MSBE, and OPSM. In comparison with the scores around 1 in GBC, the scores in CC are around 0.2, in ISA around 0.6, in Bimax around 0.5. In the case of ISA, only up- and down-regulated expressions are used so that some rows and columns are missed when an additive bicluster contains elements of normal expression levels [19, 20]. Thus, ISA does not perform as well for additive biclusters as for constant

ones. When the noise is low, MSBE shows the best performance in the two cases and the scores are equal to 1. With the increase of noise, however, the scores are rapidly decreased especially when the variance of noise is set as 0.4 in the constant biclusters and 0.6 in the additive ones. The significant alteration may be caused by the random selection of reference rows and columns in the biclustering algorithm.

## 4.2 Effects of overlapping biclusters

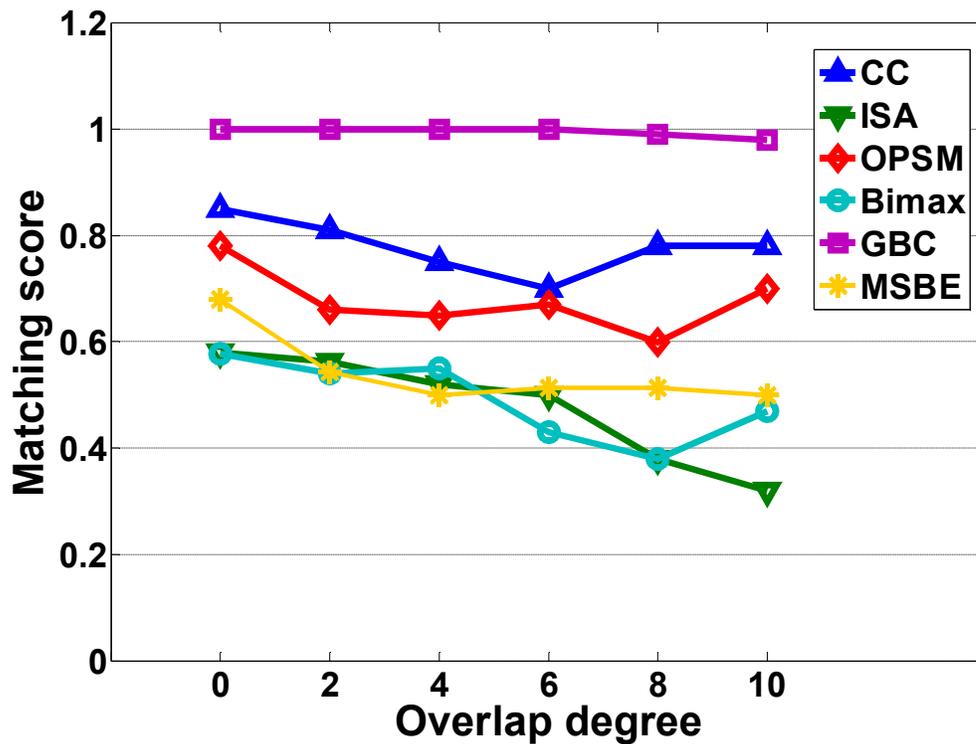


Figure 5: Results of the simulation study for overlapping additive biclusters with different overlap degrees.

A gene may activate in more than one function, and accordingly the gene may belong to several functional groups. This means that there can be overlapping biclusters in microarray data. This is one of the significant advantages of biclustering over the traditional clustering methods.

To test the capabilities of different algorithms for resolving overlapping biclusters, we first generate two  $20 \times 20$  additive biclusters with  $v$  overlapped rows and columns,

where  $v$  is called the overlap degree. Also we implant the  $v$  overlapped biclusters into a  $100 \times 100$  randomly generated matrix. The noise level is simulated with Gaussian distribution  $N(0, 0.4)$ . We test the performance of the five methods on the overlapped biclusters whose overlap degree  $v$  ranges from 0 to 10.

The results are shown in Figure 5, where the x-axis is the overlap degree and y-axis is the matching scores. The matching scores of CC are higher than those of the other three existing methods, but are still lower than the ones from our proposed GBC algorithm. In CC, the discovered biclusters have to be marked with random values in order to find more than one bicluster in a given matrix [8]. ISA appears to be sensitive to the increased overlap degree. The first normalization step in ISA may cause this problem. Because the range of expression values after normalization becomes narrower with increased overlapping, the differences between normal and significant expression values blur and are more difficult to separate. Bimax also employs a similar normalization step and thus the corresponding matching scores are also low. However, the increasing trend is shown at  $v=10$  as shown in Figure 5. As to CC and OPSM, the performance is not significantly affected by the overlap degree. GBC is nearly not affected by the overlap degree of the implanted biclusters. In the combination steps, all specific sub-matrices satisfying the conditions are identified so that the details of sub-biclusters can be tracked and used to detect all overlapping biclusters.

## **5 EXPERIMENTS ON REAL GENE EXPRESSION DATA**

In this section, we compare the performance of GBC with that of other prominent biclustering algorithm using benchmark datasets. To study the biological relevance of extracted biclusters, we make use of the gene ontology (GO) annotations and the Kyoto Encyclopedia of Genes and Genomes (KEGG) pathways. GO is a well accepted standard for gene function categorization [36]. It provides a controlled vocabulary for various genomic databases of diverse species to show the essential features shared by

all the organisms. We use the web tool FuncAssociate (<http://llama.med.harvard.edu/cgi/func/funcassociate>) to evaluate the discovered biclusters. FuncAssociate first uses Fisher's Exact Test to compute the hypergeometric function score of a gene set, and then a procedure for multiple hypothesis testing [4]. While GO is organized into hierarchical annotations, the KEGG database organizes the genes (gene products) into pathway reaction maps. The database records networks of molecular interactions in the cells, and variants of them specific to particular organisms [22]. The experiments are performed using a web tool GENECODIS (<http://genecodis.dacya.ucm.es/>).

## 5.1 Application to *S. Cerevisiae* data

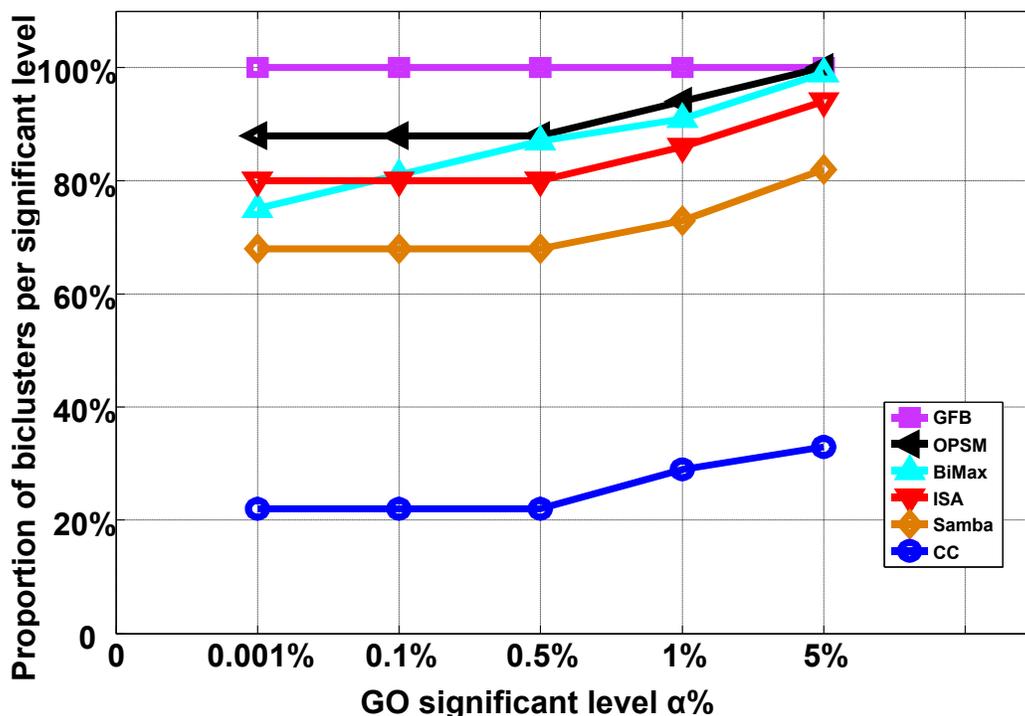


Figure 6: Proportion of biclusters significantly enriched by any GO biological process category (*S. cerevisiae*).

In this experiment, we first apply our algorithm to the gene expression data of *saccharomyces cerevisiae* which contains 2993 genes and 173 conditions [13]. The original microarray gene expression data can be obtained from the website

<http://www.tik.ee.ethz.ch/sop/bimax>. The database was analyzed by many biclustering algorithms [26, 30]. We perform our algorithm with the parameters  $r = 2^{15}$ ,  $\delta = 5$ , and  $\zeta = 2$  to identify the significant biclusters.

Then we try to investigate whether the set of genes discovered by biclustering methods shows significant enrichment with respect to a specific GO annotation provided by the Gene Ontology Consortium <http://www.geneontology.org> [36]. It provides a systematic tool to study the functional and biological significance of genes and gene products in our results. The p-values of every gene set are calculated using the hypergeometric probability model and the values are adjusted for multiple testing. The top 100 significant biclusters are filtered to compare with the outputs of OPSM, Bimax, ISA, Samba and CC obtained from [5]. The results are summarized in Figure 6 where x-axis is the preset significant level for the multiple testing and y-axis is the percent of 100 biclusters annotated by GO. Obviously, all of the biclusters filtered by GBC are statistically significant. In comparison with other methods, the gene sets discovered by GBC are highly enriched with the GO biological process category.

## 5.2 Application to Multiple Human Organs

We apply our algorithm to analyze the gene expression dataset of multiple human organs [30]. The dataset captured 18,927 unique genes for 19 different organs from 158 normal human tissues of 30 donors. The data can be downloaded from the web site at <http://www.genome.org> and <http://home.ccr.cancer.gov/ontology/oncogenomics/> [31]. We perform two procedures, with the whole and mean expression matrix respectively, to explore the expression patterns in human organ.

First, we directly perform the GBC algorithm with the entire expression matrix. In [31], t-testing of the mean expression matrix is the main analysis tool. Obviously, the information among the replicated samples of the same organ is ignored. We perform our biclustering analysis with  $18927 \times 158$  expression matrix and list the typical

biclusters of the organs in Table 4. The gene expression patterns of each organ are characterized by the listed biclusters whose rows are the detected genes and columns are the replicated samples of each organ.

Table 4. The typical biclusters of the 19 organs detected by the GBC algorithm and their corresponding categories in terms of GO and KEGG pathways.

Organ	# Samples	# Genes in [30]	# Genes in GBC	GO term	KEGG pathways
Adre.	9	2	36	GO: 0015247	--
Blad.	9	104	165	GO: 0005604	--
Cerebe.	6	4	41	Brain development (GO: 0007420)	Arginine and proline metabolism (00330)
Cerebr.	7	5	38	GO: 0030594	--
Colon	8	--	57	GO: 0045078	--
Heart	7	5	38	Regulation of heart contraction rate (GO: 0008016)	Calcium signaling pathway (04020)
Ileum	10	--	32	GO: 0006629	--
Kidn.	10	11	59	Ion transport (GO: 0006811)	Calcium signaling pathway (04020)
Liver	10	54	118	GO: 0016491	--
Lung	9	17	64	GO: 0006955	--
Ovary	5	--	25	GO: 0007338	--
Panc.	6	6	25	Digestion (GO: 0007586)	Neuroactive ligand-receptor interaction (04080)
Pros.	8	3	22	GO: 0006334	--
S. mu.	9	10	61	GO: 0008307	--
Sple.	10	7	26	GO: 0001584	--
Stom.	10	--	34	GO: 0042894	Bile acid biosynthesis (00120)
Test.	7	25	43	GO: 0019953	--
Uret.	8	4	37	GO: 0006366	--
Uter.	10	--	16	Development (GO: 0007275)	Wnt signaling pathway (04310)

These typical biclusters are conducted the enrichment analysis of biological function with GO and KEGG pathways and we compare the results with that of [31] in Table 4. The 2nd column is the number of the replicated samples of each organ, and the 3rd and

4th columns are respectively the number of genes given in [31] and in our biclusters with respect to the organ-specific GO. Obviously the number of genes in our bicluster is much larger than that in [31]. We are also able to extract biclusters in colon, ileum, ovary, stomach and uterus, which were not detected in [31]. The 5th and 6th columns are the significantly annotated terms of GO and KEGG pathways to explore the biological function of our biclusters.

The typical expression patterns of the organs are explored above. Now we try to discover the relations among the 19 organs with the microarray experiment. In [31], the mean values of the gene expressions of the different organs are calculated in the analysis. So we also perform GBC with the mean expression matrix. We first filter the genes and obtain a  $5298 \times 19$  mean expression matrix.

Some biological information among the organs is implied in the steps of the GBC algorithm. First, we have discovered that the detected sub-biclusters in column-pair spaces are consistent with the known function of organs. In fact,  $19 \times 18 / 2 = 171$  sub-biclusters are obtained by the HT in GBC. In all sub-biclusters, the number of columns is always two and that of rows is the peak count of accumulator arrays after the HT in the corresponding parameter space. We show the heat map of all sub-biclusters in Fig. 7 to explore the relations among the 19 organs. The indices of rows and columns in Fig. 7 are 19 different organs, and the values of the cross points are the number of genes in the corresponding sub-bicluster in the column-pair space. We set the diagonal value to zero. Obviously, the square matrix is symmetric. We use different gray scales to represent different count values. The darker the intensity is, the larger the value is. The largest value of the square matrix is 468 in the sub-bicluster composed of colon and ileum, that is, their gene expression patterns are very similar, which are in logical agreement with the known functions of the organs.

In the following steps of GBC, we merge the sub-biclusters in the column-pair spaces into the maximal biclusters. We discover that the procedures of combination

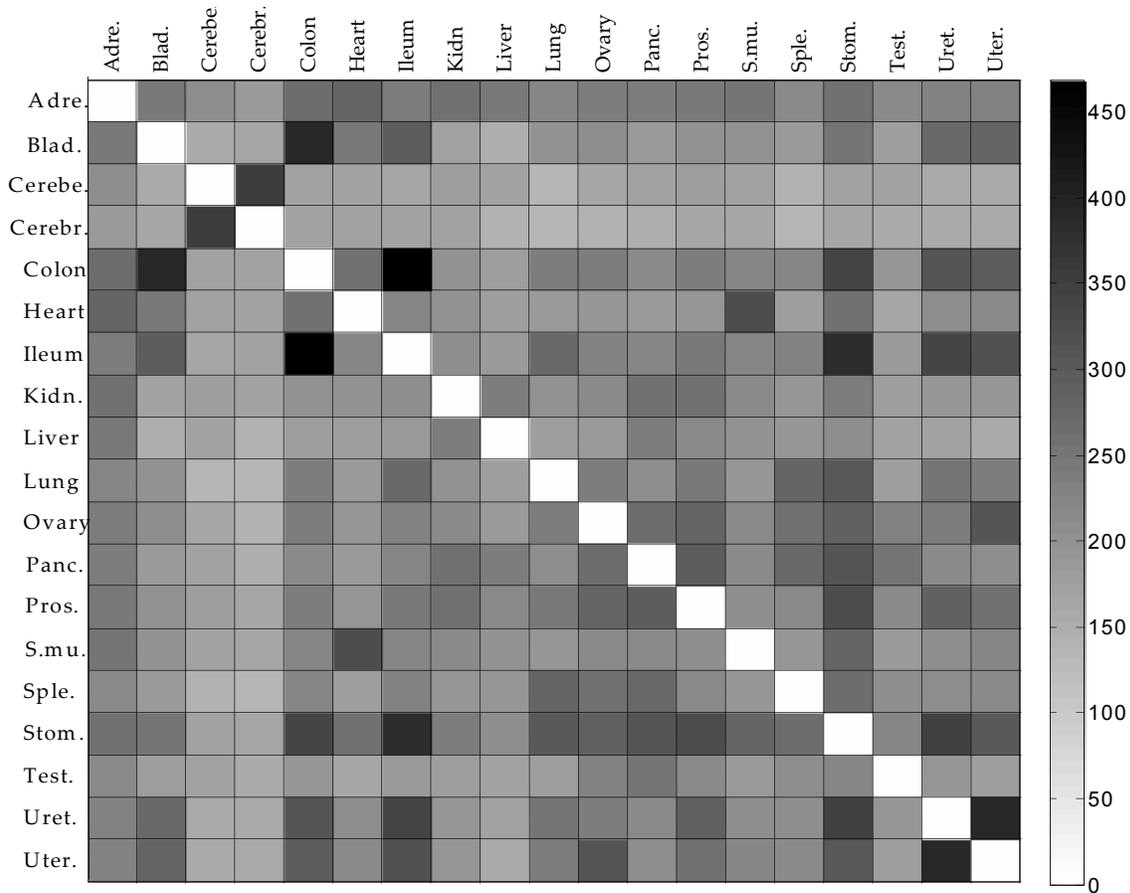


Figure 7: Heat map of the symmetric square matrix of highest count in the column-pair space. The rows and columns represent 19 organs. The cross values are the highest count number of the accumulator array after the HT. The diagonal values are set to zero.

nearly coincide with their corresponding organ functions. For example, we have merged colon, ileum, bladder and stomach into one significant block with the largest number of common genes after first iteration of the algorithm.

## 6 CONCLUSION

We have proposed a new geometric biclustering algorithm for analysis of large-scale microarray data. In comparison with the original geometric method, the HT is only employed in column-pair spaces in the new method, so the computational complexity is reduced significantly. The proposed algorithm also improves the flexibility of identifying different types of biclusters, and makes it easy to analyze overlapped

patterns using the AMPP, with which we can divide the data points into additive, multiplicative and overlapping patterns. Based on the results in column-pair spaces, the maximal biclusters are obtained after combination of sub-biclusters. Our algorithm has a superior performance compared with existing ones. The biclusters obtained using our method show useful biological meanings, which can be analyzed using the GO.

## ACKNOWLEDGMENT

This work is supported by the Hong Kong Research Grant Council (Projects CityU 122506).

## REFERENCES

- [1] Alizadeh, A.A., Eisen, M.B., Davis, R.E., Ma, C., Lossos, I.S., Rosenwald, A., Boldrick, J.C., Sabet, H., T., Tran, Yu, X., Powell, J.I., Yang, L., Marti, G.E., Moore, T., Hudson, J., Lu, L., Lewis, D.B., Tibshirani, R., Sherlock, G., Chan, W.C., Greiner, T.C., Weisenburger, D.D., Armitage, J.O., Warnke, R., Levy, R., Wilson, W., Grever, M.R., Byrd, J.C., Botstein, D., Brown, P.O., Staudt, L.M., 2000. Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. *Nature* 403, 503-511.
- [2] Ballard, D.H., Brown, C.M., 1982. *Computer vision*, Prentice-Hall, Englewood Cliffs, N.J..
- [3] Ben-Dor, A., Chor, B., Karp, R., Yakhini, Z., 2002. Discovering local structure in gene expression data: the order-preserving sub-matrix problem. In: Myers, G., Hannenhalli, S, Sankoff, D., Istrail, S. Pevzner, P. Waterman, M. (Eds.), *Annual Conference on Research in Computational Molecular Biology, Proceedings of the 6th Annual International Conference on Computational Biology*, ACM Press, New York, NY, USA, pp. 49–57.
- [4] Berriz, G.F., King, O.D., Bryant, B., Sander, C., Roth, F.P., 2003. Characterizing gene sets with FuncAssociate. *Bioinformatics* 19, 2502-2504.
- [5] Barkow, S., Bleuler, S. Prelic, A., Zimmermann, P., Zitzler, E., 2006. BicAT: a biclustering analysis toolbox. *Bioinformatics* 22, 1282-1283.
- [6] Busygin, S., Jacobsen, G., Kramer, E., 2002. Double conjugated clustering applied to leukemia microarray data. *Proc. Second SIAM ICDM, Workshop Clustering High Dimensional Data*.

- [7] Celveland W.S., 1993. Visualizing data, At & T Bell Laboratories, Murray Hill, N.J..
- [8] Cheng, Y., Church, G.M., 2000. Biclustering of expression data. Proc. Eighth Int'l Conf. Intelligent Systems for Molecular Biology (ISMB '00), 93-103.
- [9] Desper, R., Khan, J., and Schaffer, A.A., 2004. Tumor classification using phylogenetic methods on expression data. *J. Theor. Biol.* 228, 477–496.
- [10] Dudoit, S., Fridlyand, J., Speed, T.P., 2002. Comparison of discrimination methods for the classification of tumors using gene expression data. *J. Am. Stat. Assoc.* 97, 77–87.
- [11] Eisen, M.B., Spellman, P.T., Brown, P.O., Botstein, D., 1998. Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci. USA* 95, 14863–14868.
- [12] Gan, X., Liew, A.W.C., Yan, H., 2005. Biclustering Gene expression data based on high dimensional geometric method. Proc. Int'l Conf. Machine Learning and Cybernetics, IEEE SMC Society 6, 3388-3393.
- [13] Gasch, A.P., Spellman, P.T., Kao, C.M., Carmel-Harel, O., Eisen, M.B., Storz, G., Botstein, D., Brown, P.O., 2000. Genomic expression programs in the response of yeast cells to environmental changes. *Mol Biol Cell.* 11, 4241–4257.
- [14] Getz, G., Levine, E., Domany, E., 2000. Coupled two-way clustering analysis of gene microarray data. Proc. Natural Academy of Sciences US, 12079-12084.
- [15] Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P, Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A., Bloomfield, C.D., Lander, E.S., 1999. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science* 286, 531–537.
- [16] Goldenshluger, A., Zeevi, A., 2004. The hough transform estimator. *Ann. Stat.* 32, 1908-1932.
- [17] Hartigan, J.A., 1972. Direct clustering of a data matrix. *J. Am. Statistical Assoc.* 67, 123-129.
- [18] Hastie, T., Levine, E., Domany, E., 2000. 'Gene shaving' as a method for identifying distinct set of genes with similar expression patterns. *Genome Biology* 1, 0003.1-0003.21.
- [19] Ihmels, J., Friedlander, G., Bergmann, S., Sarig, O., Ziv, Y., Barkai, N., 2002. Revealing modular organization in the yeast transcriptional network. *Nat. Genet.* 31, 370–377.
- [20] Ihmels, J., Bergmann, S., Barkai, N., 2004. Defining transcription modules using large-scale gene expression data. *Bioinformatics* 20, 1993–2003.
- [21] Illingworth, J., Kittler, J., 1988. A survey of the hough transform. *Computer Vision, Graphics, and Image*

- Processing 44, 87-116.
- [22] Kanehisa, M., Goto, S., Hattori, M., Aoki-Kinoshita, K.F., Itoh, M., Kawashima, S., Katayama, T., Araki, M., Hirakawa, M., 2006. From genomics to chemical genomics: new developments in KEGG. *Nucleic Acids Res.* 34, D354-357.
- [23] Klugar, Y., Basri, R., Chang, J.T., Gerstein, M., 2003. Spectral biclustering of microarray data: coclustering genes and conditions. *Genome Research* 13, 703-716.
- [24] Lazzeroni, L., Owen, A., 2002. Plaid Models for Gene Expression Data. *Statistica Sinica* 12, 61-86.
- [25] Liu, W., Li, R., Sun, J.Z., Wang, J., Tsai, J., Wen, W., Kohlmann, A., Williams, P.M., 2006. PQN and DQN: algorithms for expression microarrays. *J. Theor. Biol.* 243, 273-278.
- [26] Liu, X., Wang, L., 2007. Computing the maximum similarity bi-clusters of gene expression data. *Bioinformatics* 23, 50-56.
- [27] Madeira, S.C., Oliveira, A.L., 2004. Biclustering algorithms for biological data analysis: a survey. *IEEE/ACM Trans. Computational Biology Bioinformatics* 1, 24-45.
- [28] Murali, T.M., Kasif, S., 2003. Extracting conserved gene expression motif from gene expression data. In *Proceedings of the 8th Pacific Symposium on Biocomputing Lihue, Hawaii*, 77-88.
- [29] Nahar, J., Chen, Y.P.P., Ali, S., 2007. Kernel based naive bayes classifier for breast cancer prediction. *J. Biol. Syst.* 15, 17-25.
- [30] Prelic, A., Bleuler, S., Zimmermann, P., Wille, A., Buhlmann, P., Gruissem, W., Hennig, L., Thiele, L., Zitzler, E., 2006. A systematic comparison and evaluation of biclustering methods for gene expression data. *Bioinformatics* 22, 1122-1129.
- [31] Son, C., Bilke, S., Davis, S., Greer, B., Wei, J., Whiteford, C., Chen, Q., Cenacchi, N., Khan, J., 2005. Database of mRNA gene expression profiles of multiple human organs. *Genome Research* 15, 443-450.
- [32] Stoughton, R.B., 2005. Applications of DNA microarrays in biology. *Annu. Rev. Biochem.* 74, 53-82.
- [33] Tanay, A. Sharan, R., Shamir, R., 2002. Discovering statistically significant biclusters in gene expression data. *Bioinformatics* 18, 136-144.
- [34] Tanay, A. Sharan, R., Shamir, R., 2006. Biclustering algorithms: a survey. In: Aluru, S. (Eds), *Handbook of computational molecular biology*. Chapman & Hall/CRC, Boca Raton, FL.
- [35] Tang, C., Zhang, L., Zhang, A., Ramanathan, M., 2001. Interrelated two-way clustering: an unsupervised

- approach for gene expression data analysis. Proc. Second Ann. IEE Int'l Symp. Bioinformatics and Bioeng. (BIBE 2001), 41-48.
- [36] The Gene Ontology Consortium, 2000. Gene Ontology Tool for the Unification of Biology. Nat. Genet. 25, 25-29.
- [37] Turner, H.L., Bailey, T.C., Krzanowski, W.J., Hemingway, C.A., 2005. Biclustering models for structure microarray data. IEEE/ACM transactions on computational biology and bioinformatics 2, 316-329.
- [38] Wang, H., Wang, W., Yang, J., Yu, P.S., 2002. Clustering by pattern similarity in large data set. In Proc. ACM SIGMOD Conference, 394-405.
- [39] Wu, C., Kasif, S., 2005. GEMS: a web server for biclustering analysis of expression data. Nucleic Acids Research 33, W596-W599.
- [40] Wu, S., Liew, A.W.C., Yan, H., Yang, H., 2004. Cluster analysis of gene expression data based on self-splitting and merging. IEEE Trans. Information Technology Biomedicine 8, 5-15.
- [41] Xu, L., Oja, E., 1993. Randomized hough transform (RHT): basic mechanisms, algorithms, and computational complexities. CVGIP: Image Understanding 57, 131-154.
- [42] Yang, J., Wang, W., Wang, H., Yu, P., 2002.  $\delta$ -cluster: capturing subspace correlation in large data set. Proc. 18th IEEE Int'l Conf. Data Eng., 517-528.
- [43] Yang, J., Wang, W., Wang, H., Yu, P., 2003. Enhanced biclustering on expression data. Proc. Third IEEE Conf. Bioinformatics and Bioeng., 321-327.
- [44] Yoon, S., Nardini, C., Benini L, De Micheli G., 2005. Discovering coherent biclusters from gene expression data using zero-suppressed binary decision diagrams. IEEE/ACM Trans. Computational Biology Bioinformatics 2, 339-354.
- [45] Zhao, H., Liew, A.W.C., Yan, H., 2007. A New strategy of geometrical biclustering for microarray data analysis. In: Sankoff, D., Wang, L, Chin, F. (Eds.) Proceedings of the 5th Asia-Pacific Bioinformatics Conference, 47-56.
- [46] Zhao, H. and Yan, H., HoughFeature, a novel method for assessing drug effects in three-color cDNA microarray experiments. BMC Bioinformatics, 8:256, 2007.