Published in final edited form as:
Pattern Recognit. 2009 April ; 42(4): 509–522. doi:10.1016/j.patcog.2008.09.032.

Online Phenotype Discovery based on Minimum Classification Error Model

Zheng Yin[1,2], **Xiaobo Zhou**[2,*], **Youxian Sun**[1], and **Stephen T.C. Wong**[2]

[1]State Key Laboratory of Industrial Control Technology, Zhejiang University, 38 Zheda Road, Hangzhou, Zhejiang Province 310027, China

[2]Center for Bioinformatics, The Methodist Hospital Research Institute and Weill Cornell College of Medicine, 6565 Fannin Street, Houston, TX, 77030, USA

Abstract

Identifying and validating novel phenotypes from images inputting online is a major challenge against high-content RNA interference (RNAi) screening. Newly discovered phenotypes should be visually distinct from existing ones and make biological sense. An online phenotype discovery method featuring adaptive phenotype modeling and iterative cluster merging using improved gap statistics is proposed. Clustering results based on compactness criteria and Gaussian mixture models (GMM) for existing phenotypes iteratively modify each other by multiple hypothesis test and model optimization based on minimum classification error (MCE). The method works well on discovering new phenotypes adaptively when applied to both of synthetic datasets and RNAi high content screen (HCS) images with ground truth labels.

Keywords

online phenotype discovery; RNA interference; high content screen; gap statistics; minimum classification error

1. Introduction

Cells have the ability to dramatically alter their shape in response to specific environmental cues, models on signaling networks regulating cell shape are needed to design effective therapeutics for treatment of cancer, neuron-generative diseases, and pathogen infection [1], and Drosophila genome-wide high-content RNA interference (RNAi) screen (HCS, high-content screen) is employed to construct such models. RNAi HCS visualizes various morphological phenotypes caused by knocking down different genes [2,3], and we must identify and quantify such phenotypes to assign a unique role for each gene in the regulation of cell shape. Expert ground truth labeling of phenotypes performed well in small scale screens [4] but it is infeasible to manually analyze HCS data. Several methods have been proposed to automatically classify cells into previously well-defined phenotypes [5], and the definition of such phenotypes involves some well trained biologists going over the whole dataset to gain a

*The corresponding author of this paper is addressed to: Xiaobo Zhou, Tel: 1-(713)-441-8692, Fax: 1-(713)-441-8696, E-mail address: XZhou@tmhs.org.

Publisher's Disclaimer: This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final citable form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

crude assessment of phenotype variance. However such supervised analysis is difficult to cover all the meaningful phenotypes in screens where millions of images are acquired and we have to automatically identify novel phenotypes online using crude assessment on existing phenotypes.

Ideally, online phenotype discovery methods should be designed to utilize dataset from existing phenotypes and solve four challenging tasks, i.e. 1)restoring those existing and biologically meaningful phenotypes, 2) differentiating novel phenotypes from existing ones, 3) clarifying novel phenotypes from each other and 4) updating the database and models for existing phenotypes along with online image input. Task 3) and 4) make online phenotype discovery distinct from simply identifying outliers, because it is necessary to differentiate multiple novel phenotypes from each other, rather than generally labeling them as “outliers”, what’s more, once a novel phenotypes is detected online, it is no longer “outliers” and should be modeled and involved into the database of existing phenotypes. Such transformation from “novel” to “known” makes it critical to update the phenotype models online while the continuously increase of sample size makes such model updating difficult.

In this paper, we present an online phenotype discovery method consisting of two key components: adaptive phenotype modeling and iterative cluster merging. After constructing cell data base (with thousands of cells) for important phenotypes through expert ground truth labeling, we build Gaussian Mixture Model (GMM) for each existing phenotype following [6], such model depicts the properties of each existing phenotype and make it possible to judge the effect and confidence level for restoring an existing phenotype or defining a novel one. When a new image (with tens of cells in each image) comes, GMM of each existing phenotype is sampled and combined with the new image one by one. For each combined dataset, we estimate cluster number using gap statistics [7]. Gap statistic is an established framework which can make significant estimation on cluster number and is compatible to various clustering method, here it is used to predict how many groups of cells in the new image can be considered as different from the existing phenotype. By combine a same new image with the existing phenotypes one by one rather than together, we circumvent the difficulties in aforementioned task 1) and focus on differentiating novel phenotypes from known ones in this stage. Considering the distinct properties between different phenotypes, we modified the strategy of taking reference dataset for gap statistics, that is, null distributions for existing phenotypes and the new image are estimated separately to make better use of the expert ground truth involved in defining existing phenotypes and avoid generating sparse sample dataset. The modified strategy helps improve the overall performance on restoring biologically meaningful phenotypes. After cluster number estimation, we perform clustering on the combined dataset. Due to the necessity of frequently running the clustering, a method using compactness criterion, Partitioning Around Medoids (PAM) [8], is selected for its efficiency and effectiveness. Part of cells from new image are merged by existing phenotype according to clustering results and cells remaining in new image are iteratively combined with other existing phenotypes, such iterative merging and updating steps restore the existing phenotypes one by one from the new image. Cell clusters never merged by any existing phenotypes are considered as candidates of new phenotype, and the partitions given by clustering results are indications for clarifying novel phenotypes from each other (aforementioned task 3).

We construct a bridge between clustering results given by PAM and GMM estimated using Bayesian parameter inference. The merging operation is validated using multiple hypothesis tests where the merging decision made by clustering can be rejected by *p*-values (with Bonferroni correction considering multiple existing phenotypes) generated from GMM and the influence from the outlier of model sampling are repressed. This step confirms the confidence of restoring existing phenotypes. On the other hand, the continuous adding of new images extends our knowledge on the existing phenotypes and urges an efficient updating of phenotype models along with the online image input (aforementioned task 4). After all the existing phenotypes merged their counterpart in one new image, the GMM of all phenotypes are optimized using minimum classification error (MCE) model as proposed in [6], MCE method updates GMM according to the classification result from a small sample set and avoids heavy computational burden. Here partition from PAM is used to define classification errors. Thus clustering results and GMM iteratively modify each other, the inconsistency between clustering and GMM is reduced iteratively and finally more reliable phenotype models are acquired.

Figure 1 illustrates the tasks of online phenotype discovery, the general workflow of our method, with key steps for fitting in the online scenario highlighted in red bounding boxes.

Our method is tested on one synthetic image dataset defined according to real cell images as well as two published dataset related to high content RNAi screen. Experimental results show that the proposed method is robust and efficient for online phenotype discovery in not only the context of the RNAi screen described in this work, but to diverse image-based screens.

In Section 2 of this paper, each step of online phenotype discovery method are discussed in detail, and section 3 presents experiment results on phenotype merging and discovery. Discussion and conclusion remarks are given in Section 4.

2. Methods

2.1 Problem Formulation

Suppose we have identified K_0 non-overlapping cellular phenotypes, the *i*-th cell in the *m*-th existing phenotype is denoted by vector $\mathbf{s}_i^{(m)} \triangleq [s_{i,1}^{(m)}, s_{i,2}^{(m)} \dots s_{i,p}^{(m)}]$, with each cell described by *P* morphological features. Then, let $S_m \triangleq \{\mathbf{s}_i^{(m)}\}_{i=1}^{u_m}$ denote the set of all cells in the *m*-th phenotype, with u_m indicating the number of cells in S_m. Thus, the set of all the available cells *S* is defined as:

$$\begin{gathered} S = \bigcup_{m=1}^{K_0} S_m \\ \text{s.t.} \forall m, n \in \{1, 2 \dots K_0\}, S_m \cap S_n = \varnothing \end{gathered} \tag{1}$$

and the total number of existing cells is $u=\sum_{m=1}^{K_0} u_m$. When a new image *E* comes, its *i*-th cell is denoted by $\mathbf{e}_i \triangleq [e_{i1}, e_{i2} \dots e_{iP}]$, and $E=\{\mathbf{e}_i\}_{i=1}^{v}$, where *v* is the number of cells in *E*. We usually have tens of cells in each *E*, while thousands cells exist in each S_m, i.e. $v<<u_m<u$.

Given *E*, we must determine number of new phenotypes K_{new}, based on, K_0, *S* and *E*. The issue of $v<<u$ makes it unfeasible to involve every single cell in *S* into cluster discovery, because the large scale of *S* could bias cluster analysis towards clusters with larger sample number, and also add computation burden. On the other hand, “new cluster” identified only according to *E* is vulnerable to outliers. Thus an efficient method of utilizing *S* is necessary.

2.2 Outline of the Approach

We propose to discover new phenotypes online through iterative cluster merging. Dataset of each existing phenotype S_m is first fit to a GMM and this model is sampled to form a dataset

S'_m with feasible sample number. Each S'_m, $m \in \{1,2\ldots K_0\}$ is combined with the new image E one by one to detect possible new phenotypes. The method runs as listed below.

1. Modeling each existing phenotype. A GMM is fit to each existing phenotype.
2. Sampling existing phenotype and combining sample set with new image. We sample from the GMM of one existing phenotype, say S_m, $m \in \{1, 2\ldots K_0\}$, get the sample set S'_m, and combine it with new image E to get a combined set F, i.e.

$$F = S'_m \cup E \tag{2}$$

   We empirically select sample number of S'_m as v to $5v$.
3. Estimating the cluster number in F using improved gap statistics [7].

   (3a) Sampling reference datasets separately for S'_m and E, using GMM as reference distribution for samples from S'_m and uniform distribution for samples from E, the support of two reference distributions are defined by binding box along the values of S'_m and E, respectively.

   (3c) Combining the above two reference dataset and estimating the cluster number in F based on the combined reference dataset using gap statistics [7].
4. Defining clusters on F . F is clustered into the estimated cluster number in step 3, and the clustering approach of PAM [8] is utilized.
5. Merging cells into existing phenotypes.

   (5a) Defining the candidates for merging. Such candidates include samples from E while assigned to a same cluster with at least 95% samples from S'_m.

   (5b) Validate the merging operation through a statistical test with Bonferroni correction.

   (5c) When the operation of merging one cell into phenotype S_m is validated, adding it to S_m while removing it from E.
6. Going back to step 2, sampling from the GMM of another existing phenotype and starting a new merging loop with modified E (previously merged cells have been removed), looping until all existing phenotypes have been combined with E.
7. Updating phenotype models.

   (7a) Defining clusters left in E as new phenotypes and estimate GMM for them.

   (7b) Summarizing decision rules from the conditional distribution models given by GMM for each phenotype, classifying each cell in the original E using these rules.

   (7c) Figuring out the “classification errors”, i.e. the inconsistency between classification result in (7b) and clustering result after (6).

   (7d) Optimizing all GMM using MCE method and awaiting new image.

Through modeling and re-sampling, S becomes more flexible and re-useable, and we can cover the properties of each phenotypes more completely. The information from existing phenotypes is combined with new image one by one in step 3 to 5. Thus in each single loop, the task of estimating cluster number is simplified to identifying difference between new image E and only one existing phenotype. After each S_m merges its counterpart in E, clusters left in E are identified as new phenotypes.

2.3 Cluster Modeling and Sampling

Given the dataset of existing cells S, we model each phenotype S_m using a GMM:

$$S_m \sim \sum_{t=1}^{Q_m} \pi_{m,t} N(\boldsymbol{\mu}_{m,t}, \Sigma_{m,t}), m=1, 2, \ldots, K_0 \\ s.t. \sum_{t=1}^{Q_m} \pi_{m,t}=1, \pi_{m,t} \geq 0 \tag{3}$$

where N denotes Gaussian distribution. We denote the number of Gaussian terms for phenotype S_m as Q_m and define parameters for S_m as $\boldsymbol{\pi}_m=\{\pi_{m,t}\}_{t=1}^{Q_m}, \boldsymbol{\mu}_m=\{\boldsymbol{\mu}_{m,t}\}_{t=1}^{Q_m}, \sum_m=\{\sum_{m,t}\}_{t=1}^{Q_m}$. Each, $\boldsymbol{\mu}_{m,t}$ has P entries as $\boldsymbol{\mu}_{m,t}$ =[$\mu_{m,t}$ [1], $\mu_{m,t}$ [2],…$\mu_{m,t}$ [P]]; and the covariance matrix $\boldsymbol{\Sigma}_{m,t}$ are set to be diagonal initially as $\boldsymbol{\Sigma}_{m,t}$ =*diag* ($\sigma_{m,t}$ [1],$\sigma_{m,t}$ [2]…$\sigma_{m,t}$ [P]).

We use Expectation-maximization (EM) algorithm to estimate {$\boldsymbol{\pi}_m$, $\boldsymbol{\mu}_m$, $\boldsymbol{\Sigma}_m$ } from S_m. In the initialization of EM algorithm, Q_m is set four, and S_m is first partitioned into Q_m clusters using fuzzy C-means clustering method, and then initial parameters are estimated using the standard vector quantization method. For each class, the number of Gaussian terms Q_m is reduced to the minimum possible using minimum description length (MDL) technique, following [6]. For each single cell **e**, the conditional distribution of each phenotype S_m is defined as

$$S_m(\mathbf{e})=\sum_{t=1}^{Q_m} \pi_{m,t} \phi(\mathbf{e};\boldsymbol{\mu}_{m,t}, \sum\nolimits_{m,t}), m=1, 2, \ldots, K_0 \tag{4}$$

We obtain random samples from the GMM to form set S'_m having *i.i.d* S_m, S'_m, and S'_m is combined with new image E to form F. When estimating cluster number in F using gap statistics, GMM is used as reference distribution for S'_m.

2.4 Estimating Cluster Numbers using Improved Gap Statistics with GMM as Reference Distribution

To solve the problem of estimating cluster numbers, many existing methods focus on the within cluster dispersion W_k, resulting from clustering datasets (e.g. F) into k clusters, C_1, C_2,…C_k with C_r denoting the indices of samples in clusters r and $f_{i,j}$ denotes the value of j-th feature measured from i-th data point. Based on $D_r=\sum_{i,i' \in C_r} \sum_{j=1}^{p} (f_{i,j} - f_{i',j})^2$, we have $W_k=\sum_{r=1}^{k} \frac{1}{2n_r} D_r$. W_k tends to decrease monotonically as the number of clusters k increases, but from some k on, such decrease flattens markedly. Statistical folklore has it that error measure based on W_k should have an "elbow" at the desirable cluster number, thus different criterions based on W_k are defined.

Gap statistics [7] method utilizes the output of any clustering algorithm under different k, compares the change of W_k to the dispersion expected under a reference null distribution, the gap between the logarithms of these two dispersions are employed to detect cluster number. Gap statistics can detect homogenous non-clustered data against the alternative of clustered data [9]. This ability is critical when all cells in new image E belong to the same phenotype.

To estimate cluster number in *F* (defined in equation (2)), we first select a number *K* which is larger than the expected cluster number, e.g. in our case *K=10*. For each *k=1, 2…K*, *F* is divided into *k* clusters and gives a series of W_k. Then we generate *B* reference datasets from *F*, cluster them into *k* clusters and obtain W^*_{kb}, *b* =1, 2,…*B*, *k* =1, 2,…*K*. We use PAM [8] for clustering. Considering the mean dispersion $\bar{l}=\sum_b \log(W^*_{kb})/B$ across *B* (*B*=15 in our case) reference datasets, and their standard deviation $sd_k=\sqrt{\sum_b (\log(W^*_{kb})-\bar{l})^2/B}$, an item $s_k=sd_k\sqrt{1+B^{-1}}$ is taken into consideration for a better control about the rejection of null model [7], and estimated cluster number *K '* is obtained as below:

$$\begin{aligned} Gap(k) &= \tfrac{1}{B}\sum_b \log(W^*_{kb}) - \log(W_k) \\ diff(k) &= Gap(k) - (Gap(k+1) - s_{k+1}) \\ k' &= \inf_{k=1,2,\ldots K}(diff(k) \geq 0) \end{aligned} \tag{5}$$

The reference distribution is a null model of data structure. In [7], reference sets are sampled uniformly either from the range of observed values for each feature, or the range of a box aligned with the principle components of data. However it is encouraged to estimate reference distribution from existing samples rather than simply using uniform distribution. Because the binding box of the whole dataset will always include some “blank” area without any samples, and the existing cluster definition can help us focus on where the data really lies, and avoid generating a sparse reference dataset which ignores the property of original dataset. Here, S'_m we sampled S'_m from GMM of each existing phenotype. Therefore, it makes sense to use this GMM as the reference distribution for S'_m.

The problem now is to generate reference dataset from GMM, because S'_m is combined with new image *E* to form the dataset *F*, and the model of *E* is unavailable. We have to deal with S'_m and *E* separately because the distribution of *E* is unavailable. We propose to solve this problem by generating reference dataset from S'_m (GMM as reference distribution) and *E* (uniform distribution as reference distribution) separately and combine two sets together, i.e. substituting *Supp(F) = Supp* ($E \cup S'_m$) with

$$P_{-}\, Supp(F)=Supp(E) \cup Supp(S'_m) \tag{6}$$

Figure 2 illustrates the strategies used for taking reference dataset. Similar as the original gap statistics method, our method can define the support of reference distribution using different “bounding boxes”. In the following experiments, all the supports are defined as the binding box aligned to the feature axes.

GMM is an accurate model for S'_m, and using GMM as reference distribution can avoid the risk of splitting one existing phenotype. Biological properties of existing phenotypes are retained, and novel phenotypes in new images become more outstanding under uniform reference distribution. Our modification on taking reference distribution not only restores the ability of characterizing the shape of datasets (different bounding boxes), but also improve it by excluding the blank area of the dataset (handling S'_m and *E* separately).

2.5 Cluster Definition and Merging

We use Partitioning Around Medoids (PAM; also known as K-medoids) [8] to do clustering on combined set *F*. PAM provides better flexibility and robustness of choosing suitable dissimilarity measurements for different applications [10] and more efficient compared to clustering methods like Fuzzy C-means, especially in our merging loops where the clustering would be carried frequently.

After clustering, we get a non-overlapping partition on combined dataset *F*, we denote this partition as $F=\bigcup_{m=1}^{K'} F_m, \forall m, n \in \{1, 2 \dots K'\}, F_m \cap F_n=\varnothing$, where the cluster number K' is determined through gap statistics, and we adjust the cluster labels to make sure that F_1 includes the largest ratio of samples from existing phenotype S'_m. Merging operation is done according to this partition. In our merging loops, datasets $S'_m, m \in \{1, 2, \dots K_0\}$ are combined with *E* one by one, and in each merging loop, the overlapped part of S'_m and *E* (if any) are located in F_1 and is deleted from *E* and included as part of existing cluster:

$$\begin{aligned} E &\leftarrow E-(F_1 \cap E) \\ S_m &\leftarrow S_m \cup (F_1 \cap E) \end{aligned} \qquad (7)$$

after all merging loops, clusters left in *E* are defined as new phenotypes.

The above merging strategy is based on theoretical case of $F_1 \supseteq S'_m$, but in reality, when we consider random sample set S'_m from GMM of a existing phenotype, it is possible that some samples are randomly far from the centre of existing phenotype and thus assigned into different clusters with their majority counterparts in *S'm*. To prevent the influence from such outliers, two strategies are utilized to validate our merging operation.

1. Merging operations only happen when some cells in *E* are assigned into F_1, and meanwhile F_1 contains more than 95% of samples in S'_m. And such samples in *E* are considered candidates for merging operation.
2. Given the existing phenotype S_m and each merging candidate $\mathbf{e}_j$, we carry out multiple hypothesis test with Bonferroni correction to validate the merging operation. A *p*-value is calculated based on the conditional distribution model $S_m(\mathbf{e}j)$ given in equation (4). This *p*-value indicates the possibility of obtaining a value at least as extreme as (if not more) this candidate under this GMM for S'_m. The corrected *p*-value for $\mathbf{e}_j$ is defined as its *p*-value with respect to S_m divided by the number of existing phenotypes K_0,. If the corrected *p*-value is lower than $0.05/K_0$, the merging operation is rejected and we keep $\mathbf{e}_j$ in *E*, or else $\mathbf{e}_j$ is merged into S_m and deleted from *E*.

Figure 3 illustrates how the cluster merging loops work on a synthetic dataset. After all the merging loops, we define the number of identified novel phenotypes as K_{new}. We estimate a GMM for each newly identified phenotype and further optimize all the existing GMM based on minimum classification error.

2.6 Online Optimization of GMM based on Minimum Classification Error

In this step, we optimize all the phenotype models so that they can restore the clustering result in future loop. Now that we have identified K_{new} novel phenotypes from image *E*, and using PAM clustering method, all the *v* cells in original *E* are assigned into $K_0 + K_{new}$ clusters, denoted as $C_1, C_2, \dots C_{k0+K_{new}}$, with C_m denoting the indices of samples in clusters *m* according to the result from PAM clustering. On the other hand, a GMM has been estimated for each phenotype and a series of classifiers can be built regarding to the conditional distribution shown in equation (4), and a series of classes $\tilde{C}_1, \tilde{C}_2, \ldots \tilde{C}_{K_0+K_{new}}$ are thus defined, with $\tilde{C}_m$ denoting the indices of samples assigned to the *m*-th class. The ultimate target of phenotype modeling is to make $\tilde{C}_m = C_m, m = 1,2 \ldots K_0+K_{new}$, i.e. the GMM should restore the operation made by the clustering (merging of cells or definition of new phenotypes), and make sure that similar cells in future images can be assigned in a same cluster. Such consistency between clustering and classification result is critical when a new cluster has just been identified, and the number of cells used to build the GMM are not so large. For such phenotypes, we must adjust the GMM to include new members indicated by clustering.

Thus, after obtaining K_0+K_{new} clusters from *E*, we continue to refine the conditional distribution models $\{S_m(\mathbf{e})\}_{m=1}^{K_0+K_{new}}$ by adjusting the parameters $\left\{\boldsymbol{\pi}_m, \boldsymbol{\mu}_m, \sum_m\right\}_{m=1}^{K_0+K_{new}}$, similar to [6] this is done by minimizing a penalty function that is related to the classification error, while in our case, the classification error is defined as the inconsistency between clustering result C_m and classification result $\tilde{C}_m$ given by GMM.

Given a vector **e** representing a single cell from the original new image *E*, we consider a set of class conditional likelihood function given by

$$g_m(\mathbf{e}) = \log(S_m(\mathbf{e})), m = 1, 2 \ldots K_0 + K_{new} \tag{8}$$

We define the parameter set related to $S_m(\mathbf{e})$ as $\boldsymbol{\lambda}_m = \{\boldsymbol{\pi}_{m,t}, \boldsymbol{\mu}_{m,t}, \sum_{m,t}\}_{t=1}^{Q_m}$, And the entire parameter set of the classifier is $\boldsymbol{\Lambda} = \{\boldsymbol{\lambda}_m\}_{t=1}^{K_0+K_{new}}$. For $\mathbf{e}_i, i = 1,2 \ldots v$ (each cell in the image *E*) and $\tilde{C}_m$ (the set of cell indices assigned to class *m*) the decision rule based on GMM is given by

$$i \in \tilde{C}_m, \text{ if } g_m(\mathbf{e}_i) = \max_{1 \le j \le K_0 + K_{new}} g_j(\mathbf{e}_i) \tag{9}$$

Define the following class misclassification measure

$$d_m(\mathbf{e}) = g_m(\mathbf{e}) + \log[\frac{1}{K_0 + K_{new} - 1} \sum_{j=1, j \ne m}^{K_0 + K_{new}} \exp[g_j(\mathbf{e})\eta]]^{1/\eta} \tag{10}$$

Where η is a positive number. Hence, d_m (**e**) >0 implies misclassification and d_m (**e**) ≤0 means correct decision. Next, we define the following empirical penalty function of the parameter set $\boldsymbol{\Lambda}$ based on the classification result for all *v* cells in image *E*

$$L(\boldsymbol{\Lambda}) = \frac{1}{v} \sum_{i=1}^{v} \sum_{m=1}^{K_0+K_{new}} l_m(\mathbf{e}_i; \boldsymbol{\Lambda}) 1_{\{i \in C_m\}} \tag{11}$$

where

$$l_m(\mathbf{e}_j;\mathbf{\Lambda}) \triangleq \frac{1}{1+\exp(-\gamma d_m(\mathbf{e}_j))} \tag{12}$$

$$1(\mathrm{X})=\begin{cases} 1, & \mathrm{X\ is\ true} \\ 0, & \mathrm{X\ is\ not\ true} \end{cases} \tag{13}$$

with γ being a constant. Notice that the logic function is defined using the clustering result C_m, thus this method add penalty to the inconsistency between $\tilde{C}_m$ and C_m, and thus update GMM to make $\tilde{C}_m$ consistent with C_m.

In order to maintain the original constraints in the GMM, we apply the following transformations on the parameters: (i)$\pi_{m,t} \rightarrow \tilde{\pi}_{m,t}$, where $\pi_{m,t}$ = exp $\{\tilde{\pi}_{m,t}\}$ / Σ_i exp $\{\tilde{\pi}_{m,t}\}$; (ii) $\mu_{m,t}$ $[p\ ]\rightarrow\tilde{\mu}_{m,t}$ $[p\ ]=\mu_{m,t}$ $[p\ ]/\sigma_{m,t}$ $[p\ ]$ and (iii) $\sigma_{m,t}$ $[p\ ] \rightarrow \sigma_{m,t}$ $[p\ ]=\log\sigma_{m,t}$ $[p\ ]$, where p= 1, 2…P.

To minimize L $(\mathbf{\Lambda})$ with respect to $\mathbf{\Lambda}$, we use the stochastic gradient algorithm. Starting from the nth cell in image E, the mixture weights are updated according to

$$\tilde{\pi}_{m,t}^{(n+1)}=\tilde{\pi}_{m,t}^{(n)} - e_1(n)\frac{\partial l_m(\mathbf{e};\mathbf{\Lambda})}{\partial \tilde{\pi}_{m,t}}\bigg|_{\mathbf{\Lambda}=\mathbf{\Lambda}_n} \tag{14}$$

with

$$\frac{\partial l_m(\mathbf{e};\mathbf{\Lambda})}{\partial \tilde{\pi}_{m,t}}=-\gamma l_m(\mathbf{e};\mathbf{\Lambda})[1-l_m(\mathbf{e};\mathbf{\Lambda})]\cdot(2\pi)^{-P/2}\left|\sum\nolimits_{m,t}\right|^{-1/2}[\,\overline{q}_m(\mathbf{e})]^{-1}\cdot\xi \tag{15}$$

where

$$\xi=\exp\{-\frac{1}{2}(\mathbf{e}-\boldsymbol{\mu}_{m,t})^T\sum\nolimits_{m,t}(\mathbf{e}-\boldsymbol{\mu}_{m,t})\} \tag{16}$$

The elements in the mean vectors are updated according to

$$\tilde{\mu}_{m,t}^{(n+1)}[\,p]=\tilde{\mu}_{m,t}^{(n)}[\,p]-e_2(n)\frac{\partial l_m(\mathbf{e};\mathbf{\Lambda})}{\partial \tilde{\mu}_{m,t}[\,p]}\bigg|_{\mathbf{\Lambda}=\mathbf{\Lambda}_n}\quad ,p=1,2\ldots P \tag{17}$$

with

$$\frac{\partial l_m(\mathbf{e};\boldsymbol{\Lambda})}{\partial \tilde{\mu}_{m,t}[p]} = \gamma l_m(\mathbf{e};\boldsymbol{\Lambda})[1 - l_m(\mathbf{e};\boldsymbol{\Lambda})] \cdot \pi_{m,t}(2\pi)^{-P/2} \left|\sum_{m,t}\right|^{-1/2} [S_m(\mathbf{e})]^{-1} \cdot \xi \cdot \left[\sum_{m,t}(\mathbf{e} - \mu_{m,t})\right]_p \tag{18}$$

where $[\mathbf{x}]_p$ denotes the p-th entry of a vector $\mathbf{x}$.

And the elements in the covariance matrices are updated following

$$\tilde{\sigma}_{m,t}^{(n+1)}[p] = \tilde{\sigma}_{m,t}^{(n)}[p] - e_3(n) \frac{\partial l_m(\mathbf{e};\boldsymbol{\Lambda})}{\partial \tilde{\sigma}_{m,t}[p]}\Big|_{\boldsymbol{\Lambda}=\boldsymbol{\Lambda}_n}, \quad p = 1, 2 \dots P \tag{19}$$

with

$$\frac{\partial l_m(\mathbf{e};\boldsymbol{\Lambda})}{\partial \tilde{\sigma}_{m,t}[p]} = -\gamma l_m(\mathbf{e};\boldsymbol{\Lambda})[1 - l_m(\mathbf{e};\boldsymbol{\Lambda})] \cdot \pi_{m,t}(2\pi)^{-P/2} \left|\sum_{m,t}\right|^{-1/2} [\overline{q}_m(\mathbf{e})]^{-1} \cdot \xi \cdot [\mathbf{e} - \mu_{m,t})^T \sum_{m,t}^{-1}(\mathbf{e} - \mu_{m,t}) - 1 \tag{20}$$

In our experiments, we choose the following parameters: $\eta = 1, \gamma = 0.4$, and the step sizes $e_1(0) = 0.01$, $e_1(n) = e_2(n) = e_1(0)n^{0.8}$ and $e_3(n) = 0.2\, e_1(n)$.

3. Experiments and results

3.1 Synthetic dataset

3.1.1 Dataset Description—After going over fluorescent microscopy cell images generated for Drosophila genome wide RNAi screen (Bakal *et al*, unpublished), we picked up seven types of cells (RGB colour image) and simulated them using seven types of polygons in 8-bit gray level images (256 level gray scale with intensity 0 indicating black and 255 for pure white). The information of these cells and polygons are shown in Figure 4.

The key geometry and gray level parameters of each polygon were sampled from random variables defined according to the information from real cells. For instance, gray level filled into an ellipse was sampled from uniform distribution in the range of [0, 50], while the rectangles were filled with gray levels sampled uniformly from the range of [120, 180]. Each polygon was described using six features, including mean and standard deviation of gray levels, and geometry features like length of longest axis, length of shortest axis, perimeter and area of the polygons.

We generated 2000 polygons from each of seven types to serve as training dataset, i.e. the set of existing phenotypes, and we also had another 2000 polygons from each type to form the test dataset, i.e. cells in a series of synthetic images used to carry out experiments. In each experiment, we started from a certain set of existing phenotypes and built GMM from training samples. On the other hand, we iteratively chose two of seven polygon types and selected 100 polygons apiece out of the testing dataset to form a synthetic test image, altogether 70 images can be formed for one experiment. Using the model estimated from training set, we can identify existing and novel type of polygons from these synthetic images, and observe the performance of our method under different number of novel phenotypes and order of image input.

3.1.2 Performance under different sets of existing phenotypes—

Figure 5 shows the general performance of our method under different sets of existing phenotypes. Figure 5 *left* shows the performance of the method with improved gap statistics and MCE model optimization, and these results are compared with the performance of the method with improved gap statistics but no model optimization, which are listed in Figure 5 *right*.

We changed the number and composition of existing phenotypes, (composition means which types are chosen as "existing"), and for each set of existing phenotypes, we shuffled the order of image input 50 times. Whatever set of existing phenotypes we used, the synthetic images always contained all seven phenotypes and we can always define accuracy for each polygon type as "the proportion of test samples restored into its original cluster". If one phenotype is used as existing phenotype in one experiment, the accuracy on this phenotype is defined as the proportion of testing cells (in this phenotype) merged into the that existing cluster; while if the phenotype is novel, the accuracy of this phenotype equals the proportion of testing cells left along in a separate cluster after all the merging loops. Thus, we can average the accuracy across all different order of image input and composition of existing phenotypes to report the general performance of our method under different experiment conditions.

We can see from Figure 5 that our method have consistent performance for different polygon types under different conditions, with all the accuracy values over 85%, and the best performance comes when the number of existing phenotypes are 3 and 4. What's more, the method with GMM optimization based on MCE continuously outperforms the method without model optimization step. The model optimization procedure dramatically improves the performance on most polygon types when the number of existing phenotypes is lower than 3. This is consistent with our expectation of adding the model updating steps: modifying the models for novel phenotypes according to the clustering result so that it can characterize the new phenotype better with relatively low number of training samples. When the number of existing phenotype is 6, both method suffered from false negative (especially for rectangle and trapezoid phenotypes) as novel samples were merged into existing phenotypes, and this indicates the importance of cluster validation and the necessity of more refined multiple hypothesis tests.

3.1.3 Modifications on gap statistics and model updating improve the performance—

Specifically, we pick up two sets of experiments both with six existing phenotypes, and compare the performance of three online phenotype discovery methods. One series of experiments are carried out with ellipses serve as novel phenotype, and the other series have 16-point star as novel phenotype. Both types of experiments have 70 test images with 200 polygons each, the order of image input are shuffled 50 times and the average accuracy for each polygon type is shown in Figure 6.

All of the three involved methods are based on iterative cluster merging, but one directly uses gap statistics to estimate the cluster numbers, one uses improved gap statistics with reference dataset sampled separately, and the other one is the method proposed here, with improved gap statistics as well as phenotype model optimization using MCE. We can see from Figure 6 that the method here outperforms the other two in at least five of six phenotypes.

The method using original gap statistics suffers from the complicated composition of existing phenotype, especially when distinct phenotypes of rectangles and 16-point stars are both used as existing phenotypes (Figure 6 *left*). Taking reference dataset separately from each existing phenotype dramatically improves the performance, because the reference dataset focus on where the samples really lies, and thus each reference dataset is less sparse and more representative than taken from a single distribution with support defined by the whole dataset.

In both type of experiments, the accuracy on the novel phenotype is improved by the involvement of model optimization based on MCE. After a novel phenotype is identified, the GMM is estimated based on a small amount of samples, and following samples in this phenotype are prone to be mistakenly merged into existing phenotypes. With the phenotype models continuously updated according to each new image, such false merging is effectively avoided, and the accuracy related to novel phenotypes are improved.

3.2 Performance validation on published genetic screening dataset aiming at defining local signalling networks regulating cell morphology

3.2.1 Dataset Description—We download quantified data related to [1] from http://arep.med.harvard.edu/QMS and use it to validate our method's ability of restoring biological meaningful clusters. In this dataset, the morphological change of 12601 cell segments from 249 different treatment conditions (TCs) are described by twelve morphological indicators. A quantitative morphological signature (QMS) is defined for each TC based on the morphological change it causes. All TCs are clustered based on their QMS, and seven out of 41 resulted clusters are highlighted in [7]. We selected altogether 6782 cell segments from six of these biological meaningful clusters and use their quantified morphology indicators to carry out our experiments. The information of those six clusters is summarized in Table 1.

The work in [1] represents the recent attempt of using fluorescent microscopy images based genetic screen to define local network regulating cell morphology. Gene over-expression and double-stranded RNA (dsRNA) induced RNAi are utilized individually or in combination to form altogether 249 TCs. A QMS is defined for each TC according to the observed morphological change on cultured *Drosophila* BG-2 Cells. Through cell culturing, stochastic labeling and segmentation, altogether 12601 individual cells are obtained for analysis..

Three level of quantification are employed to generate the QMS for each TC: from individual cell image to 145 morphological features for each cell; from 145 features to a group of twelve morphological indicators for each cell (constructing twelve neural networks (NN) based on twelve group of training cells with typical morphological changes and measuring the similarity between each cell and each of the training sets); and from morphological features of each cell to QMS for each TC (a normalized Z-score across all the cells for a same TC). Seven scores are used to form final QMS vector for each TC.

3.2.2 Restoring biological meaningful clusters—In this case we try to validate the ability of restoring biological meaningful "pheno-clusters" for our method. Six phenol-clusters highlighted in [1], namely Cluster 6, 8, 18, 33, 1 and 27 are used. Quantified morphological indicators for 2,800 cell segments are divided into a flow of image input consisting of 28 synthetic images with 100 cells in each image. Two series of experiments with different sets of existing pheno-clusters are carried out and the performances of three online phenotype discovery methods are compared. Both kinds of experiments are repeated 100 times with different order of inputting test images. Figure 7 shows the result of these experiments.

We get similar trends as in the synthetic datasets. The method proposed in this paper outperforms the other two methods in at least four of six phenol-clusters, and also improve the accuracy on novel phenotypes. Specifically, adding cluster 33 into the group of existing phenotypes helps improve the results. Cluster 18 is the largest phenol-cluster in the dataset, featuring large, flat cells typically with extensive lamellipodia, and cluster 27 includes three TCs featuring cells displaying an aberrant number of long protrusions [1], by adding cluster 33 (featuring cells subject to dsRNA TCs targeting at Rho1 family) to existing phenotypes, we can identify cells with spiky and polarity structure better, thus reduce the possibility of merging cells in cluster 27 into cluster 18.

This case shows our method's ability of restoring biological meaningful clusters/phenotypes in the online scenario, and supplies the confidence for applying our method when trying to extend genetic screen in [1] to genome wide scale.

3.3 Identifying novel phenotype from high-content RNAi genome-wide screening

3.3.1 Dataset Description—We utilized the fluorescence images mentioned in [5]. A genome wide RNAi screen was carried out to implicate genes involved in Rac (a subfamily of the Rho family GTPases) signaling using Drosophila Kc167 cell line. The corresponding cellular phenotypes were analyzed for changes, and the authors present a framework of analysis with the core step of classifying the cells into three pre-defined phenotypes, namely normal, ruffle and spiky.

Image Acquisition and Segmentation: All the fluorescent images are obtained following the protocol in [5], among three channels in the original color images, cytological profiling was based on analyzing cell shapes in the gray level image of actin channel. A two-step segmentation method is involved in cell segmentation. It includes nuclei segmentation on DNA channel, and cell body segmentation on the combined images of Actin, Tubulin and DNA channels [11,12,13]. In the nuclei segmentation, the nuclei are first detected by a background correction based adaptive threshold method and the gradient vector field. Then the clustered nuclei are segmented using the marker watershed algorithm. Segmentation results of nuclei are used as the 'seed' images for marker watershed based cell body segmentation. To accurately delineate the boundaries of cell bodies, an automated feedback system is used to reduce the over-segmentation of cells following [12]. Detailed shape and boundary information of nuclei and cell bodies is obtained after the segmentation procedure.

Morphological Feature Extraction and Feature Selection: To capture the geometric and appearance properties, we extract 211 cellular morphology features belonging to five categories following [5], including 85 wavelet features, 10 Zernike moments features, 48 Haralick features, 14 geometric region property features, and 54 phenotype shape descriptor features. To describe the dataset more informatively, an unsupervised feature selection method is adopted. It is based on iterative feature elimination using k nearest neighbor (kNN) features, following [14]. We use correlation coefficients as the similarity measure for each pair of features, iteratively pick up the feature having biggest correlation coefficient with its kNN features, discard these k neighbor features and update parameter k until k is reduced to 1. A set of 15 features is finally selected.

Training and testing dataset: We use the three phenotype defined in [5] as the existing phenotypes. To assure the reliability of models for existing phenotypes, we pick up an extended training dataset from the combination of the original 1 000-cell-patch database in [5] and the cells processed from the whole genome wide screen. Using the EM algorithm, a GMM with two Gaussian items are fit to normal phenotypes, while ruffle and spiky phenotype end up with four Gaussian items, respectively. Another 1 000 cells are selected as testing dataset. The information of our training and testing dataset is listed in Table 2.

3.3.2 Box and whisker plot showing the performance on restoring existing phenotypes—Using the 1000 testing cells from three phenotypes, we form a flow of ten synthetic images; each image contains 100 cells from two of three phenotypes. Using our method of online phenotype discovery with improved gap statistics and model optimization based on MCE, we try to merge those testing cells into their original phenotypes using the GMM estimated in previous step. The order of image input is shuffled 100 times and the accuracy for each phenotype is calculated. Across the 100 repeated experiments, the average accuracy for normal phenotype is 91.49%, with standard deviation of 1.11%; some cells in ruffle phenotype are not merged due to the various texture property caused by actin accumulation, and the average accuracy still reach 87.10%, with standard deviation of 1.44%; and the spiky phenotype has an average accuracy of 90.87% and standard deviation of 1.35%.

We rank the accuracy from each experiment, and illustrate the performance using box and whisker plot. The accuracy for each phenotype in each experiment is sorted in descending order and plotted along the Y-axis, the two horizontal edges of boxes indicate upper and lower quartile of accuracy values while the red line in the middle shows the median value. The whiskers and lines extending from the end of boxes show the extent of the rest data, and red crosses (+) are outliers with accuracy values beyond 1.5 times of inter quartile range. The box plots for all three phenotypes are shown in Figure 8. With all the inter quartile range smaller than 5%, we can confirm the robustness of our method against different order of image input.

3.3.3 Discovering novel phenotypes from the genome wide RNAi screen—Finally we combine our method with the whole dataset of genome wide RNAi screens. Starting from the three existing phenotypes, we identified another three novel phenotypes, and the information about these phenotypes is shown in Table 3. These phenotypes are different from the original ones in level of actin accumulation (uniform, punctuate or highly accumulated), distribution of actin (along the boundary or across cell body), size and stage of polar formation, etc. The biological mechanism behind these phenotypes remains to be explored, but by doubling the number of morphological phenotypes, our method does show its ability to supply a better insight into the high content RNAi screen, and based on the method in [5] as well as our discovery, a more versatile scoring system can be built to better reveal the detailed role of genes involved in the screen.

At present stage, all functions are developed in Matlab 7.0 and run in PC with Intel® Core™ 2 T7200 2.00GHz CPU and 2.00GB of RAM. Starting from three existing phenotypes, the average running time for the method based on improved gap statistics and model optimization using MCE is 2.2 seconds on a group of 100 segmented cells, 13.2 seconds on a group of 600 cells and 25.4 seconds on a group of 1000 cells. When we change the PAM clustering method into fuzzy C-means method from the fuzzy logic toolbox of Matlab, the average running time on a group of 100 cells is larger than 30 seconds, which is intractable for online application. On the other hand, k-means method is as efficiency as PAM method, our method using PAM outperforms k-means based method on the accuracy of restoring existing phenotypes by 2-5%. Considering the fact that cell number in each image is seldom over 300 in the reported high content screen, our method recruit proper technologies to solve all the major challenge against online phenotype discovery and the present framework is efficiency and robust for online application.

4. Discussion and Conclusions

Identifying novel morphological phenotypes online is a major challenge in specific high-throughput image-based screens. Manual phenotype labeling of high-throughput image-based data is a laborious and inordinately time-consuming process, while available automatic identification methods usually classify cells into a limited set of predefined phenotypes which have been determined through biased means and won't be updated according to the online image input. As millions of images are now generated during the course of a comprehensive genome-scale screen new methods are needed to effectively identify novel phenotypes in such massive databases. Here we report the development of an online phenotype discovery method which models existing phenotypes, compares cells in new images with existing phenotype models through cluster analysis, assigns some new cells to existing phenotypes, and finally identifies and validates novel phenotypes online.

Instead of hanging with single Gaussian model, we use Gaussian mixture model to describe each biologically meaningful cluster. Theoretically GMM can approximate closely any continuous density function for a sufficient number of mixtures and appropriate model parameters [15]. Gap statistics plays a key role in cluster analysis and merging. Using GMM as the reference distribution for the existing phenotypes in gap statistics, we can cover the complete properties of phenotypes more efficiently. Furthermore, we iteratively combine the new image with the sample set from each existing phenotypes and update the content of new image. Thus the task in each loop is simplified to compare part of new image with only one existing phenotype.

Originally, clustering based on compactness criteria and methods based on GMM are of different types. We set a bridge between them through two ways, one is multiple hypothesis test used for cluster validation and another is model optimization based on MCE methods. The former step validates the merging operation based on the GMM to avoid the influence of outliers while the latter one modifies the GMM to get more robust results.

For analysis of high content screen data, many researchers choose to summarize the information of single cells or objects to supply a normalized signature for higher level concepts (e.g. treatment conditions, genes, complexes, etc). Thus it is critical to identify different phenotypes related to a same treatment condition. When applied to the analysis of high content screens, our method can be involved in identifying multiple phenotypes related to single well, help the definition of refined phenotype signature and supply more detailed insight into related questions. Under the frame of iterative cluster merging, our future goals are to build more reliable phenotype models, construct complete pipelines of cluster analysis with validation procedures and finally obtain more reliable definition of phenotypes.

Acknowledgments

We would like to thank colleagues of Perrimon Lab in the Department of Genetics at Harvard Medical School, for their excellent collaboration. The authors would also like to thank research members of The Methodist Hospital Research Institute, Radiology Department. The research is funded by The Methodist Hospital Research Institute (Stephen T.C. Wong).

References

1. Bakal C, Aach J, Church G, Perrimon N. Quantitative morphological signatures define local signaling networks regulating cell morphology. Science 2007;316:1753–1756. [PubMed: 17588932]
2. Friedman A, Perrimon N. A functional genomic RNAi screen for novel regulators of RTK/ERK signaling. Nature 2006;444:230–234. [PubMed: 17086199]
3. Friedman A, Perrimon N. Genetic screening for signal transduction in the era of network biology. Cell 2007;128:225–231. [PubMed: 17254958]
4. Kiger AA, Baum B, Jones S, Jones MR, Coulson A, Echeverri C, Perrimon N. A functional genomic analysis of cell morphology using RNA interference. Journal of Biology 2003;2(27):271–285.
5. Wang J, Zhou X, Bradley PL, Perrimon N, Wong STC. Phenotype recognition for high-content RNAi genome-wide screening. Journal of Molecular Screening 2008;13:29–39.
6. Zhou X, Wang X. Optimisation of Gaussian mixture model for satellite image classification. IEE Proc-Vision, Image and Signal Process 2006;153(3):349–356.
7. Tibshirani R, Walther G, Hastie T. Estimating the number of clusters in a dataset via the gap statistic. J Royal Stat Soc B 2001;32(2):411–423.
8. Kaufman, L.; Rousseeuw, P. Finding groups in Data: An introduction to Cluster Analysis. Wiley; New York: 1990.
9. Yan M, Ye K. Determining the Number of Clusters Using the Weighted Gap Statistics. Biometrics 2007;63:1031–1037. [PubMed: 17425640]

10. Thalamuth A, Mukhopadhyay I, Zheng X, Tseng G. Evaluation and comparison of gene clustering methods in microarray analysis. Bioinformatics 2006;22(19):2405–2412. [PubMed: 16882653]

11. Zhou, X.; Liu, KY.; Bradley, PL.; Perrimon, N.; Wong, STC. Towards automated cellular image segmentation for RNAi genome-wide screening. Lecture Notes in Computer Science; MICCAI 2005; p. 885-892.

12. Yan P, Zhou X, Shah M, Wong STC. Automatic Segmentation of RNAi Fluorescent Cellular Images with Interaction Model. IEEE Trans on Info Tech in Biomedicine 2008;12(1):109–117.

13. Li FH, Zhou X, Wong STC. An automated feedback system with the hybrid model of scoring and classification for solving over-segmentation problems in RNAi high content screening. Journal of Microscopy 2007;226(2):121–132. [PubMed: 17444941]

14. Mitra P, Murthy CA, Pal S. Unsupervised Feature Selection Using Feature Similarity. IEEE Trans PAMI 2002;24(3):301–312.

15. Zhao Y, Zhuang X, Ting S. Gaussian mixture density modeling of non-Gaussian source for autoregressive process. IEEE Trans Signal Processing 1995;43(4):894–903.

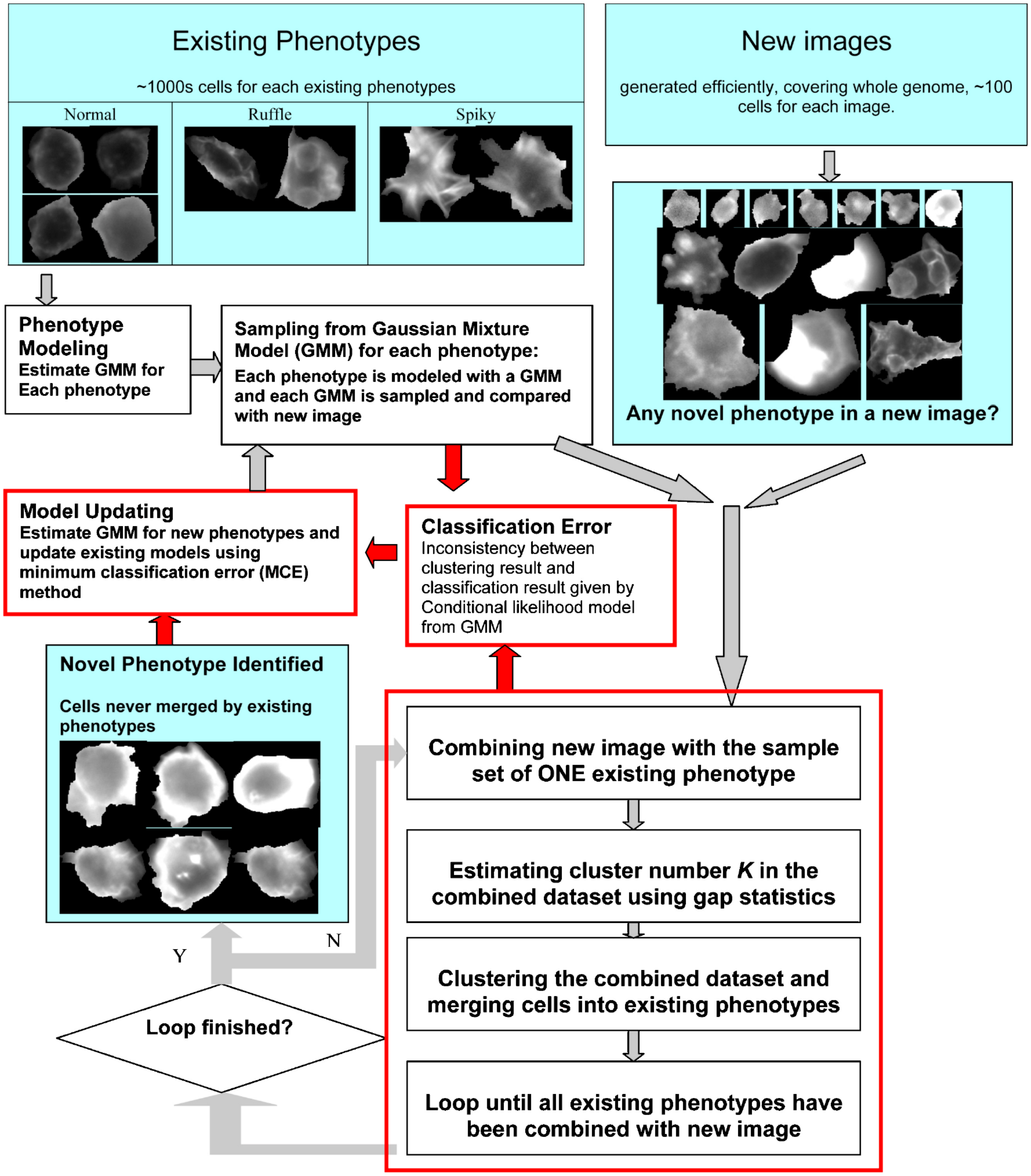

Existing Phenotypes
~1000s cells for each existing phenotypes
Normal
Ruffle
Spiky
New images
generated efficiently, covering whole genome, ~100 cells for each image.
Any novel phenotype in a new image?
Phenotype Modeling
Estimate GMM for Each phenotype
Sampling from Gaussian Mixture Model (GMM) for each phenotype:
Each phenotype is modeled with a GMM and each GMM is sampled and compared with new image
Model Updating
Estimate GMM for new phenotypes and update existing models using minimum classification error (MCE) method
Classification Error
Inconsistency between clustering result and classification result given by Conditional likelihood model from GMM
Novel Phenotype Identified
Cells never merged by existing phenotypes
Combining new image with the sample set of ONE existing phenotype
Estimating cluster number K in the combined dataset using gap statistics
Clustering the combined dataset and merging cells into existing phenotypes
Loop until all existing phenotypes have been combined with new image
Loop finished?
Y
N

Figure 1. Tasks of online phenotype discovery for high-content RNAi screens

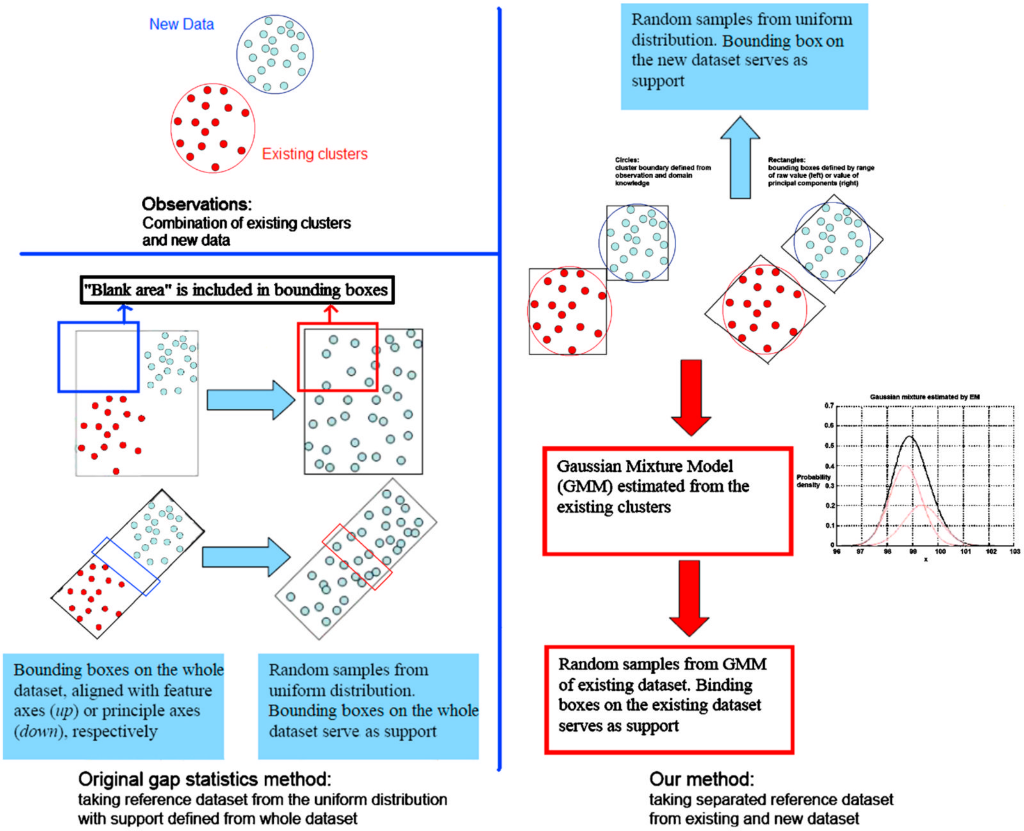
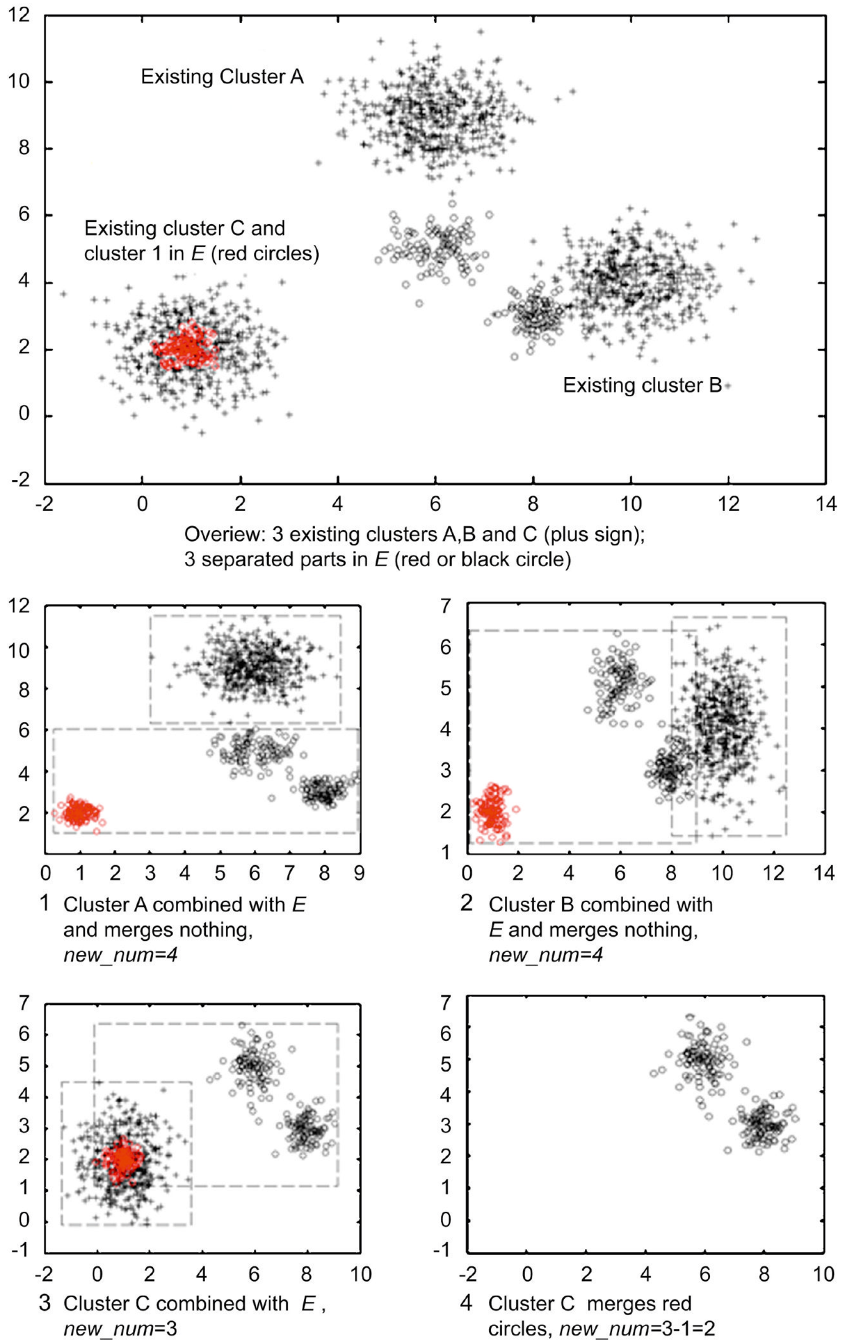

New Data
Existing clusters
Observations:
Combination of existing clusters and new data
"Blank area" is included in bounding boxes
Bounding boxes on the whole dataset, aligned with feature axes (*up*) or principle axes (*down*), respectively
Random samples from uniform distribution. Bounding boxes on the whole dataset serve as support
Original gap statistics method:
taking reference dataset from the uniform distribution with support defined from whole dataset
Random samples from uniform distribution. Bounding box on the new dataset serves as support
Circles:
cluster boundary defined from observation and domain knowledge
Rectangles:
bounding boxes defined by range of raw value (left) or value of principal components (right)
Gaussian Mixture Model (GMM) estimated from the existing clusters
Gaussian mixture estimated by EM
Probability density
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
96
97
98
99
100
101
102
103
x
Random samples from GMM of existing dataset. Binding boxes on the existing dataset serves as support
Our method:
taking separated reference dataset from existing and new dataset

Figure 2. Take reference data set separately for existing clusters (red circles) and new dataset (blue circles)



Overiew: 3 existing clusters A,B and C (plus sign);
3 separated parts in *E* (red or black circle)

1 Cluster A combined with *E*
and merges nothing,
new_num=4

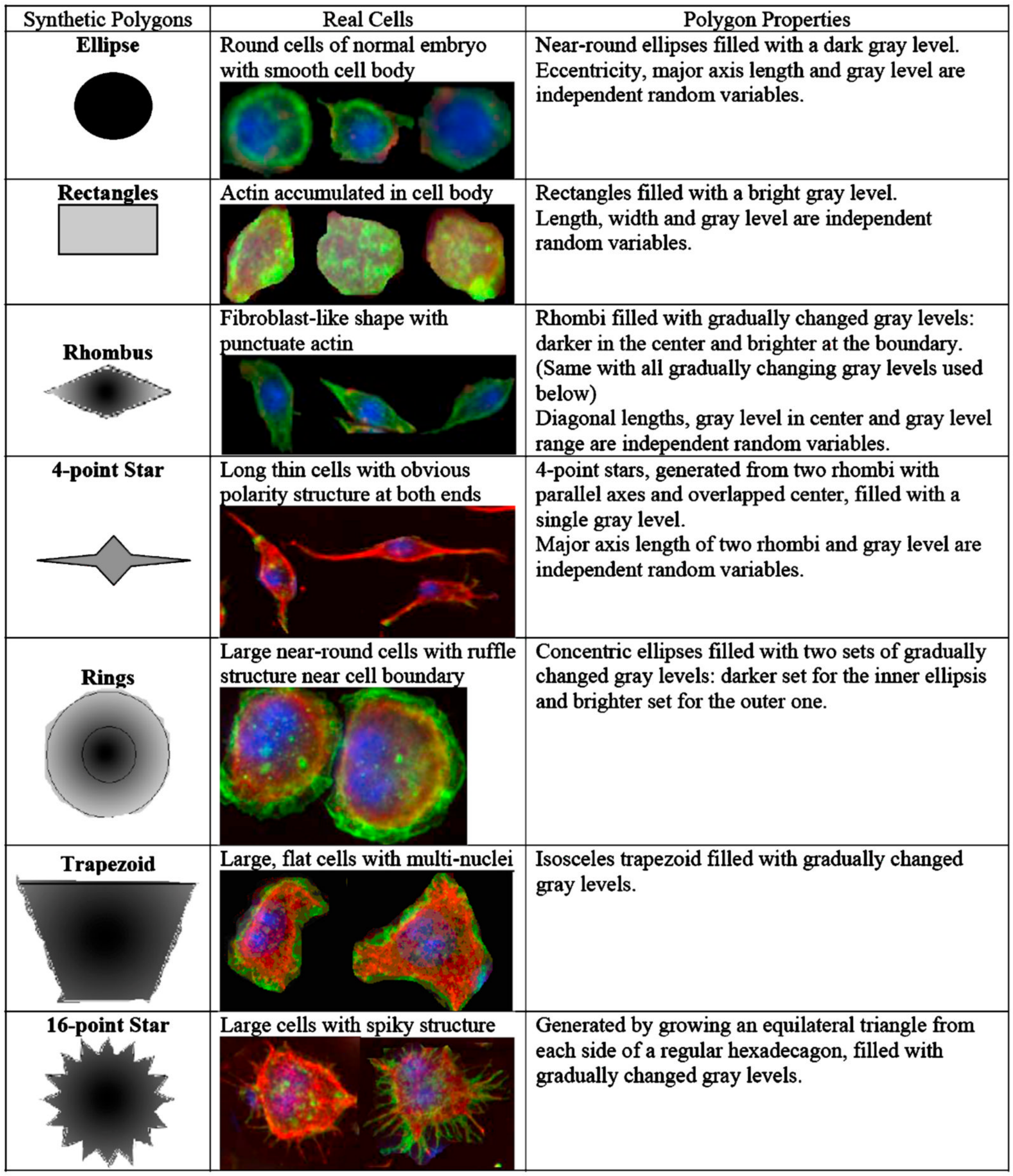
2 Cluster B combined with
E and merges nothing,
new_num=4

3 Cluster C combined with *E* ,
new_num=3

4 Cluster C merges red
circles, *new_num*=3-1=2

Figure 3. Illustration of merging loops on a synthetic data set

The largest subplot shows the overview of a synthetic dataset, and four smaller subplots labeled as 1-4 shows three merging loop (1-3) and the final result of carrying out iterative cluster merging on this dataset



Synthetic Polygons	Real Cells	Polygon Properties
Ellipse	Round cells of normal embryo with smooth cell body	Near-round ellipses filled with a dark gray level. Eccentricity, major axis length and gray level are independent random variables.
Rectangles	Actin accumulated in cell body	Rectangles filled with a bright gray level. Length, width and gray level are independent random variables.
Rhombus	Fibroblast-like shape with punctuate actin	Rhombi filled with gradually changed gray levels: darker in the center and brighter at the boundary. (Same with all gradually changing gray levels used below) Diagonal lengths, gray level in center and gray level range are independent random variables.
4-point Star	Long thin cells with obvious polarity structure at both ends	4-point stars, generated from two rhombi with parallel axes and overlapped center, filled with a single gray level. Major axis length of two rhombi and gray level are independent random variables.
Rings	Large near-round cells with ruffle structure near cell boundary	Concentric ellipses filled with two sets of gradually changed gray levels: darker set for the inner ellipsis and brighter set for the outer one.
Trapezoid	Large, flat cells with multi-nuclei	Isosceles trapezoid filled with gradually changed gray levels.
16-point Star	Large cells with spiky structure	Generated by growing an equilateral triangle from each side of a regular hexadecagon, filled with gradually changed gray levels.

Figure 4. Information of seven types of polygons used as synthetic dataset

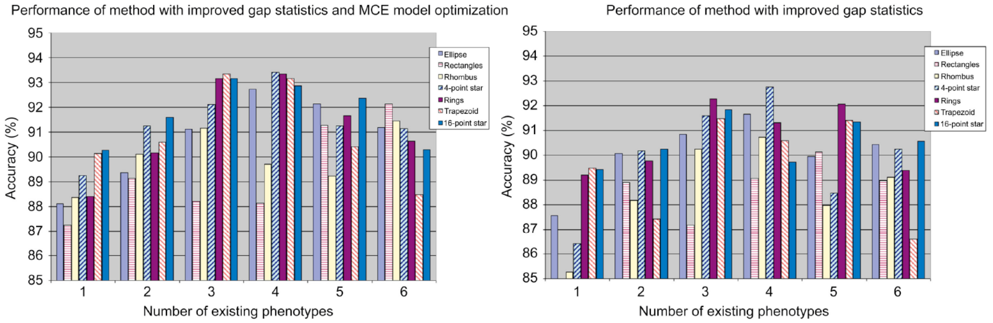


Figure 5. Performance of two online phenotype discovery methods on the synthetic dataset

left) The method with improved gap statistics and MCE model optimization; *right)* The method with improved gap statistics but no model updating. The number along X axis shows the number of phenotypes used as existing phenotypes. All the accuracy values for each phenotype are averaged across different composition of existing phenotypes (when number of existing phenotypes varies from 1 to 6, the number of possible compositions are 7, 21, 35, 35, 21 and 7, respectively) and 50 different order of image input for each composition.

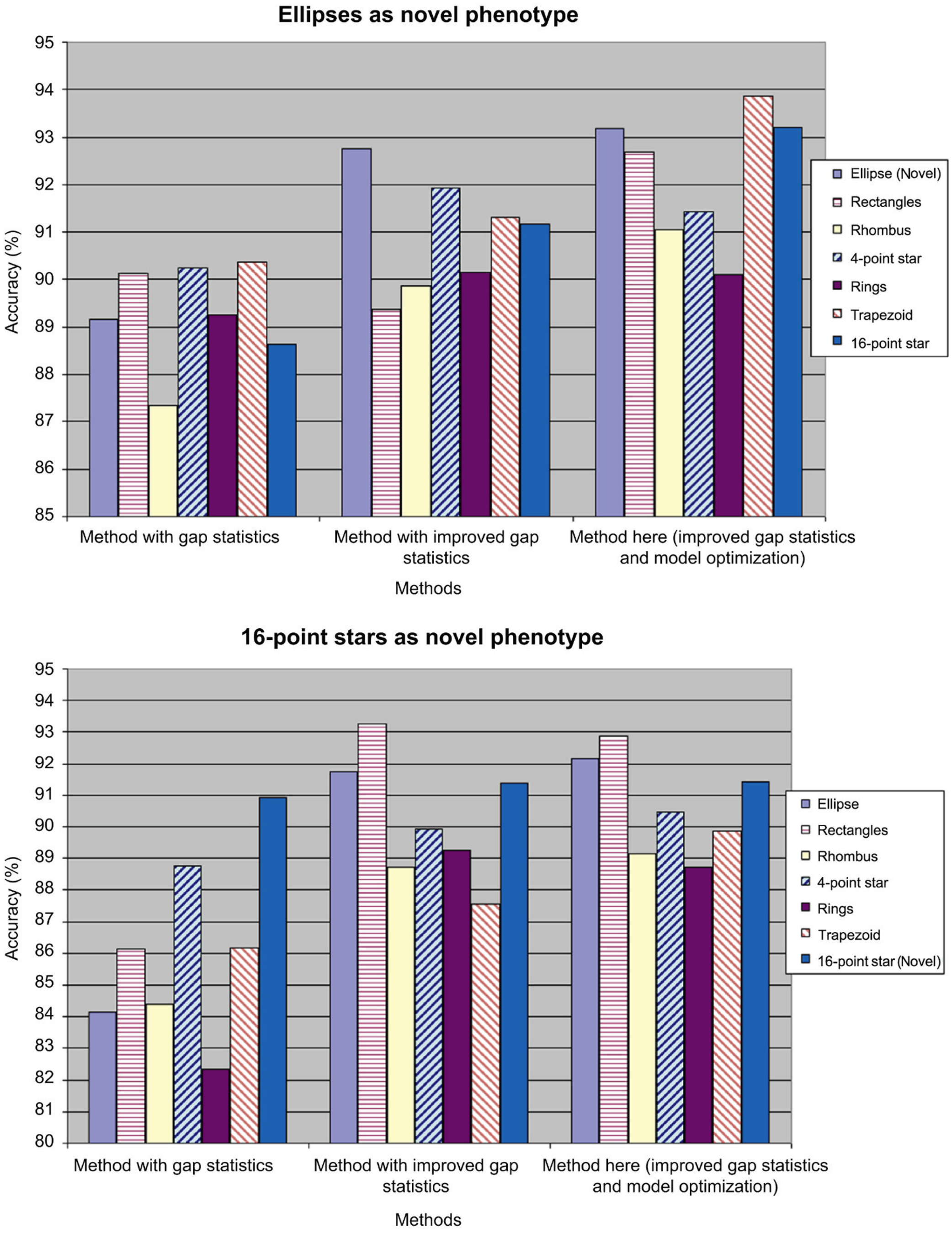


Figure 6. Performance of three online phenotype discovery methods on two specific experiments on synthetic dataset

left) Average performance across 50 experiments with ellipses serve as novel phenotype; *right)* Average performance across 50 experiments with 16-point stars serve as novel phenotype. The accuracy on an existing phenotype is defined as the ratio of testing cells in this phenotype and merged to its original phenotypes; and the accuracy for a novel phenotype is calculated as the proportion of test cells in this phenotype, while left alone in a separate cluster after all the merging loops.

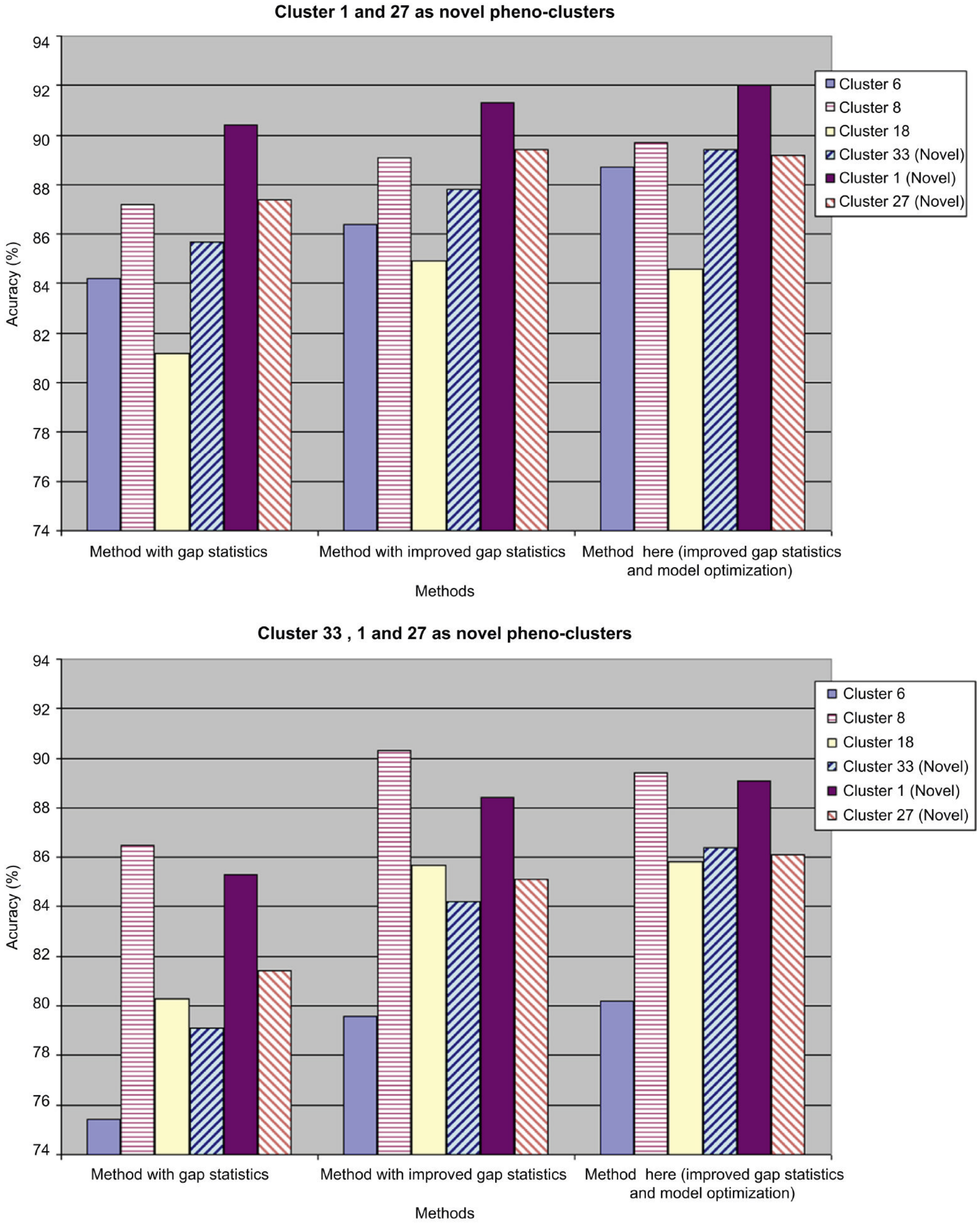


Figure 7. Performance comparisons among three methods on restoring biological meaningful cluster from published high throughput screen data

left) Average performance across 100 experiments with cluster 1 and 27 serve as novel phenotype; *right)* Average performance across 100 experiments with cluster 23, 1 and 27 serve as novel phenotype.

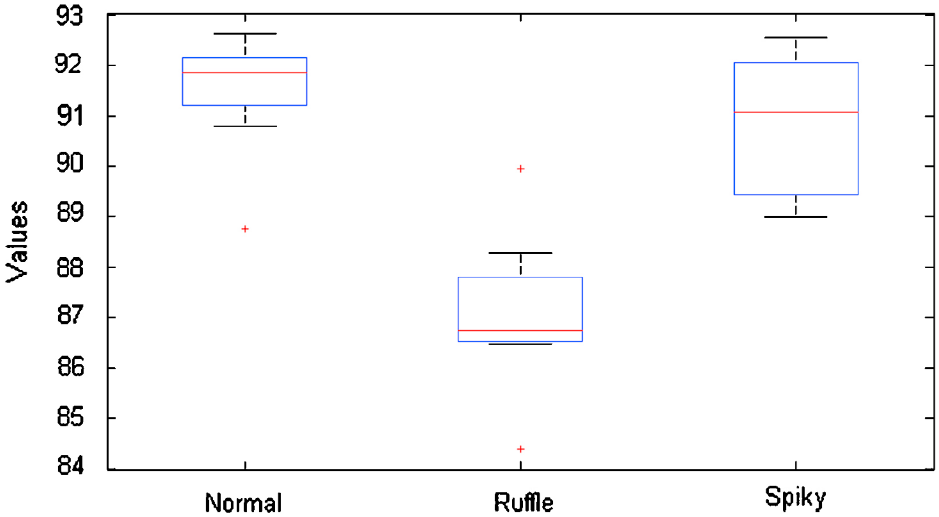


Figure 8. Performance on restoring existing phenotypes from testing dataset