J Sign Process Syst (2009) 55:1–13
DOI 10.1007/s11265-008-0205-7

Automated Protein Distribution Detection in High-Throughput Image-Based siRNA Library Screens

Yan Nei Law · Stephen Ogg · John Common · David Tan · E. Birgitte Lane · Andy M. Yip · Hwee Kuan Lee

Received: 21 January 2008 / Revised: 19 March 2008 / Accepted: 4 April 2008 / Published online: 20 May 2008

Abstract The availability of RNA interference (RNAi) libraries, automated microscopy and computational methods enables millions of biochemical assays to be carried out simultaneously. This allows systematic, data driven high-throughput experiments to generate biological hypotheses that can then be verified with other techniques. Such high-throughput screening holds great potential for new discoveries and is especially useful in drug screening. In this study, we present a computational framework for an automatic detection of changes in images of in vitro cultured keratinocytes when phosphatase genes are silenced using RNAi technology. In these high-throughput assays, the change in pattern only happens in 1–2% of the cells and fewer than one in ten genes that are silenced cause phenotypic changes in the keratin intermediate filament network, with small keratin aggregates appearing in cells in addition to the normal reticular network seen in untreated cells. By taking advantage of incorporating prior biological knowledge about phenotypic changes into our algorithm, it can successfully filter out positive 'hits' in this assay which is shown in our experiments. We have taken a stepwise approach to the problem, combining different analyses, each of which is well-designed to solve a portion of the problem. These include, aggregate enhancement, edge detection, circular object detection, aggregate clustering, prior to final classification. This strategy has been instrumental in our ability to successfully detect cells containing protein aggregates.

Keywords Keratin proteins · RNA interference · Mutant detection · Fluorescence microscopy · Image analysis

Y. N. Law · H. K. Lee (✉)
Imaging Informatics Group, Bioinformatics Institute, A*STAR,
30 Biopolis Street, #07-01 Matrix,
Singapore 138671, Singapore
e-mail: leehk@bii.a-star.edu.sg

S. Ogg · J. Common · D. Tan · E. B. Lane
Institute of Medical Biology, A*STAR,
8 Biomedical Grove, #06-06 Immunos,
Singapore 138665, Singapore

A. M. Yip
Department of Mathematics, National University of Singapore,
2 Science Drive 2,
Singapore 117543, Singapore

1 Introduction

Technological and biological advances have enabled affordable access to large RNAi libraries to study gene function in cell culture models. Additionally, the use of automated microscopy increases our ability to acquire data related to the effect of gene silencing. The speed at which this data can be generated creates a data bottleneck where post-acquisition analysis is the limiting factor when studying images for multiple morphological criteria. Moreover, special care must be taken when handling these data such that objectivity and reproducibility should be preserved. To achieve this, computer vision algorithms are needed to extract quantitative information. We developed a computational framework to treat images as quantitative data with systematic and objective processes. We wish to identify the changes in protein distribution in a high-throughput siRNA screen on in vitro cultured keratinocytes when a phosphatase gene is silenced. However, in each image of these assays, only about 1–2% of the cells exhibit a protein aggregation phenotype, the detection pattern used for positive hits. In this case, a general machine learning technique [1, 2] would fail to filter possible genes that are

related to such phenotypic changes. Therefore, a more sophisticated algorithm incorporating specific biological knowledge is needed.

1.1 Biological Background

Keratins form the largest subfamily of intermediate filament cytoskeletal proteins and are expressed in epithelial cells where type I keratins and type II keratins form obligate hetero-oligomers in a cell type specific manner [3]. For example, keratin number 5 and keratin number 14 expression is restricted to the basal proliferative compartment of the epidermis, whereas keratins number 1 and 10 are expressed in differentiating epidermis [4]. Keratins are a major stabilizing component of the cytoskeleton in epithelial cells, forming both inter and intracellular reticular networks. The importance of the normal functioning of these networks is underscored by the numerous genodermatoses caused by keratin gene mutations [5, 6]. All intermediate filament proteins (including keratins) have three characteristic structural domains: a central alpha-helical rod domain, flanked by N and C terminal unstructured "head" and "tail" domains. It is thought that these unstructured domains are sites of post-translational modification [7]. Formation of intermediate filaments proceeds spontaneously in vitro requiring neither energy nor accessory proteins. Despite this propensity to assemble in vitro, the intermediate filament network must be in equilibrium in vivo, to allow cells to remodel their intermediate filament network as required [8]. Regulation of their assembly is likely through either post-translational modifications such as phosphorylation, or association with accessory proteins [9]. Epidermolysis bullosa simplex (EBS) is an epithelial fragility disorder that is characterized by sensitivity of the skin to mild stress, resulting in blistering from intracellular cell lysis. In this disorder, mutations in either K5 or K14 genes, whose protein products form the intermediate filament network in the basal cells of stratified epithelia, result in the disease phenotype [10]. These mutations thought to destabilize the intermediate filament network, thus causing the epidermis to be less able to withstand the mechanical stresses to which they are normally subjected. Immortalized skin cells derived from patients with EBS display a reduced cell spreading capability, and abnormal intermediate filament assembly, with a tendency of the intermediate filament to disassemble into small aggregates, suggesting a compromised intermediate filament network. Network disassembly into small aggregates can also be induced by treatment with okadaic acid, sodium orthovanadate or calyculin A [11, 12]. This drug treatment affects keratin phosphorylation status, likely through inhibition of protein phosphatases. It is unknown at present which specific phosphatase is involved with keratin dynamics or whether multiple phosphatases could be involved with this process. Using an siRNA phosphatase library we have attempted at identify specific phosphatase genes that result in a change in the keratin number 14's intermediate filament network.

RNA interference is a powerful tool to study gene function in cultured cells. Together with high-throughput image screening techniques [13, 14], it offers us an enormous chance to understand the complex relationships between genes, proteins, cellular components and physiological systems [15]. At the same time, it presents a challenge for quantitative analysis, which requires efficient techniques to evaluate this unprecedented amount of data. For instance, some classification algorithms of subcellular patterns based on cell morphology have been proposed [16, 17]. Chen and Murphy [18] developed an automated protein partitioning algorithm based on location patterns. Others [19] have developed quantitative morphological profiling methods to systematically investigate the role of individual genes in the regulation of cell morphology, or methods for automatic image cytometry [20], demonstrating the benefits of using a large number of individual cell measurements when exploring data from high-throughput screens (multiparametric analysis, reviewed in [21]).

1.2 Our Contribution

In this paper, we present a framework for applying computer vision on image data sets generated by high-throughput screening of keratinocytes assays. Our approach is different from a general machine learning approach [2], in which we emphasise incorporating biological knowledge into our algorithm. The biological knowledge we utilize in our computational framework are:

1. Patterns of Biological Significance: Small keratin aggregates manifest as spots with specific size in the images.
2. Criteria for Detection: Due to the low transfection efficiency, detection of the existence of mutants is more important than the number of mutants. For example, we are both interested in images with one mutant and images with many mutants.
3. Phosphatases Gene Candidates: It is more important to find the top few phosphatases gene candidates that most likely cause the phenotypic changes rather than finding many phosphatases gene candidates that may have some effect.

Based on the above, we rank all images according to the chance of existence of mutants in descending order. The biologist would only look at a few (<1%) of the top ranked images and perform extensive and independent validation assays on not more than a handful of the top few phosphatases gene candidates.

Our detection method is based on phenotypic changes in the keratin proteins labeled with the green fluorescent protein. Upon broad-spectrum pharmacological inhibition of phosphatases, small keratin aggregates that manifest as spots in the images appear in cells in addition to the normal reticular network seen in untreated cells. Here we demonstrate that our algorithm can automatically and accurately identify the positive hits for this phenotype and can effectively rank phosphatase genes for further validation.

2 Materials and Methods

The proposed framework automatically extracts information of keratin aggregate spots for classification. A general work flow of the image analysis framework is shown in Fig. 1. Systematic parameter tuning and method selection is performed in each step. Our framework is optimized for this high throughput assay. Each step of the process is described in detail in the following sections.

2.1 Image Acquisition

Cultured cells are placed into 96-well plates and transfected with a siRNA library for the knockdown of phosphatases. There are a total of 267 targeted genes and three siRNA are used for each targeted gene. Hence, a total of 801 siRNAs are used (267 genes × 3 RNAi sequences per gene). And each set of 267 genes requires at least three 96-well plates (267/96≈3) and 89 wells of each plate are used. Hence, a total of nine plates are used. Our proposed analysis requires both high-resolution images to assess the protein aggregation phenotype pattern and a large field of view images to capture the whole well. The microscope, optimized for live cell imaging experiments, typically takes images that are 1,040 pixels by 1,392 pixels. Our microscope is based around an axiovert 200 stand. Images are acquired using a 20×, 0.8NA lens, resulting in a pixel size of 0.400 micrometers. As the pixel resolution is 400 nm/pixel, one image has a field of view of 400 μm. To cover a large field of view, 16 images are taken for each well. This makes a total of 1,424 images in each plate and approximately 13,000 images for the whole study. As a result, thousands of images with a total size of 10–20 gigabytes are generated and processed.

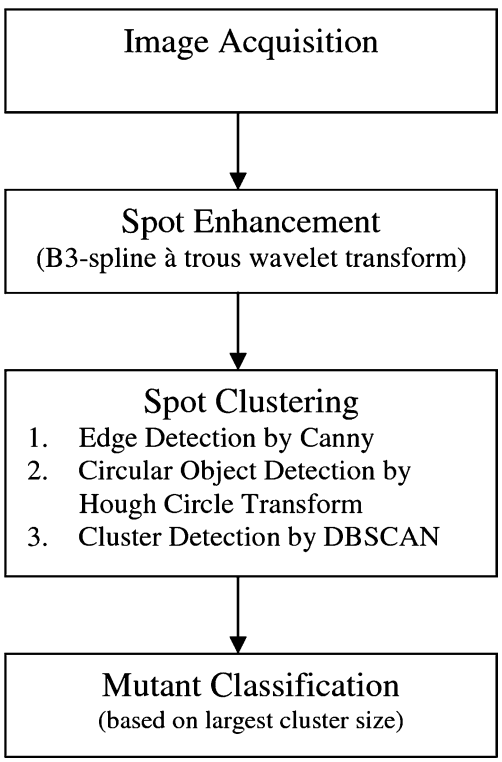


Figure 1 A general work flow of the image analysis framework.

2.2 Spot Enhancement

The keratin aggregates manifest as spots in the images. To detect these spots efficiently, a reliable spot enhancement procedure is needed. A sample image is shown in the top panel of Fig. 2. Since the performance of the clustering result greatly relies on the image quality, we first improve the images by processing them using a wavelet transform technique [22] to enhance the pattern of small spots while reducing the background and the effect of noise.

The wavelet transform is a multi-resolution analysis tool that is designed to provide different levels of local details. Among many different wavelet transform algorithms [22], we choose the B3-spline version à trous wavelet algorithms as proposed by Starck et al. [23] mainly because of its invariance under translation and the simplicity of the implementation of the direct and the inverse transform. We obtained a wavelet decomposition, which is computed by convolving the original image A_0 with a 5×5 mask $h^T h$, where h=[1/16 1/4 3/8 1/4 1/16]. At the borders of A_0, we extend it by continuity. After obtaining this smoothed image A_1, we obtain the detail coefficient W_1 from the difference $A_0 - A_1$. The same process is repeated recursively from the smoothed images A_i, $0<i\leq J$, with a filter h augmented at each scale i by inserting $2^{i-1}-1$ zeros between two nonzero entries, i.e.,

$$h = [1/16 \quad 0\ldots0 \quad 1/4 \quad 0\ldots0 \quad 3/8 \quad 0\ldots0 \quad 1/4 \quad 0\ldots0 \quad 1/16]$$

for scale i. At the end, we have the à trous wavelet representation $W_1, W_2, \ldots, A_J$ and the reconstruction formula for the original image is given by

$$A_0 = A_J + \sum_{i=1}^{J} W_i.$$

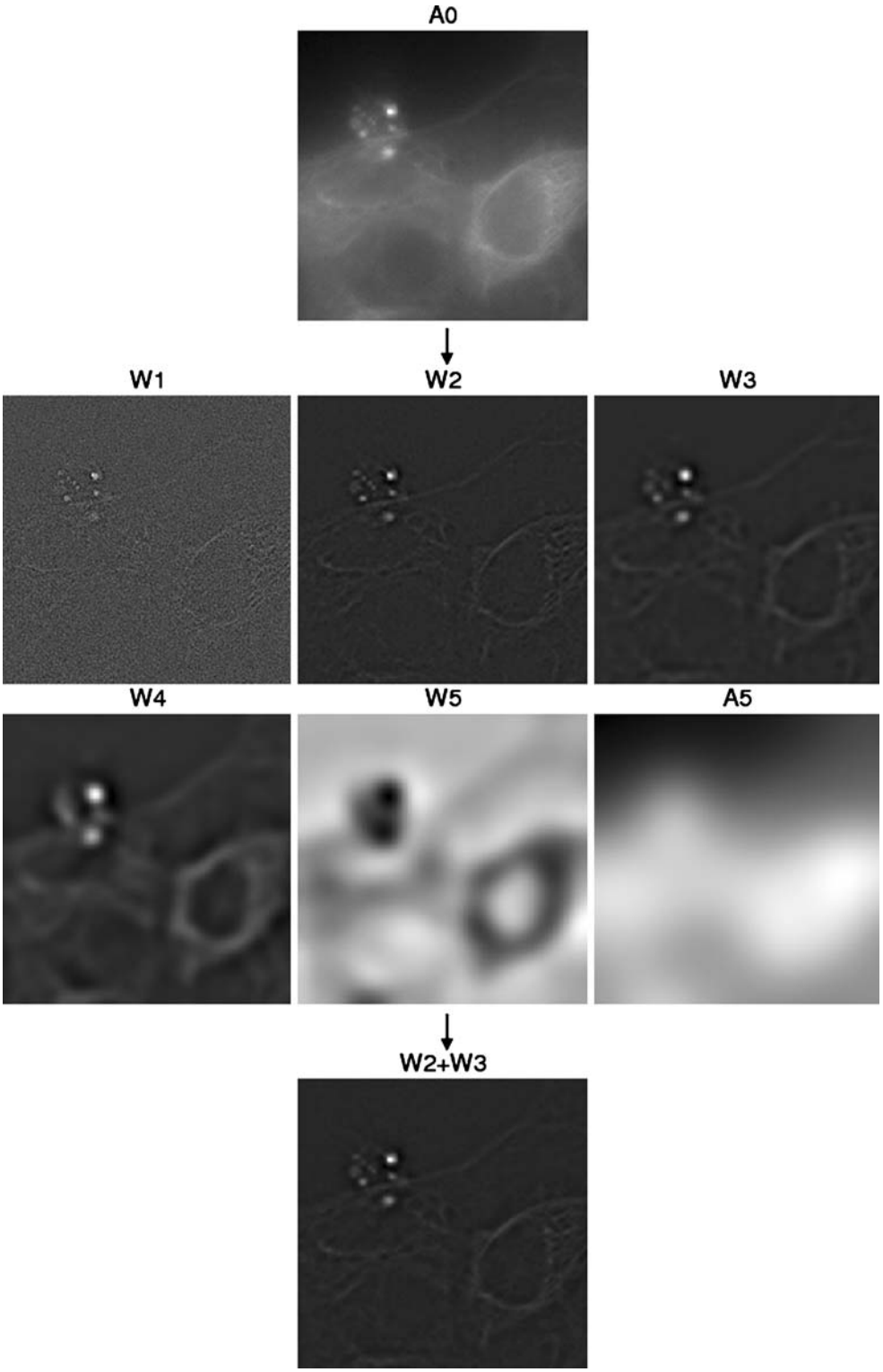


Figure 2 Wavelet preprocessing. *Top*: The original noisy image A_0; *middle*: The à trous wavelet representation of A_0 (note that W_1 contains mainly noise of the images while W_4 and W_5 contains the background of the image); *bottom*: The output image which is the recombination of detail images (note that the pattern of the small keratin aggregates in the mutated cell are enhanced while the reticular network in the wild type cell is vanished).

To recombine each decomposed images into a new output image, we can specify a set of resolution scales. In our algorithm, we propose to use $W_2 + W_3$ as our wavelet processed image. The wavelet image W_1 consists mainly of single pixel noise and is thus discarded. We also discard the higher order wavelets W_4, W_5,…, W_J and A_J, as they represent the background and coarse scale structures. Examples of the wavelet representation and the wavelet processed image are shown in Fig. 2.

To justify that wavelet pre-processing is indeed effective, we process our images without any pre-processing and with top-hat pre-processing [29]. Validation against ground truth

using the ROC curve shows that wavelet pre-processing is effective (see Fig. 8).

2.3 Spot Clustering

Once a high quality image is generated, advanced techniques are used to detect small keratin aggregates (clusters of small spot particles) in a more accurate and more robust way. We propose a spot clustering algorithm that consists of three detection steps: (1) first detect edges of the wavelet-processed image using Canny edge detector [24], (2) detect circular objects by applying Hough transform [25] on the edge image, and (3) search for small spot percolating clusters using DBSCAN [26]. The first row and the second row of Fig. 3 show the results on each detection step when a wavelet-processed image and an un-preprocessed image are used respectively. In particular, bad performance when using an un-preprocessed image verifies that a bad quality image can affect the final result. Therefore, to improve the robustness of the clustering step, it is essential to first preprocess the images.

2.3.1 Edge Detection

The first step of our spot clustering algorithm is to extract edges of the wavelet-processed image. A good edge detection result can reduce the searching space and improve the robustness of the circular object detection. In our method, we use Canny edge detector [24] implemented by Matlab. The Canny method finds edges by looking for local maxima of the gradient of the image. The gradient is calculated using the derivative of a Gaussian filter with default sigma=1. There are two thresholds, high and low, to detect strong and weak edges respectively. In our experiment, we only specific the high threshold t_{canny} and set the low threshold to be $0.4 \times t_{\text{canny}}$. The weak edges will be included only if they are connected to strong edges. Hence, the method trends to avoid creating false edges due to

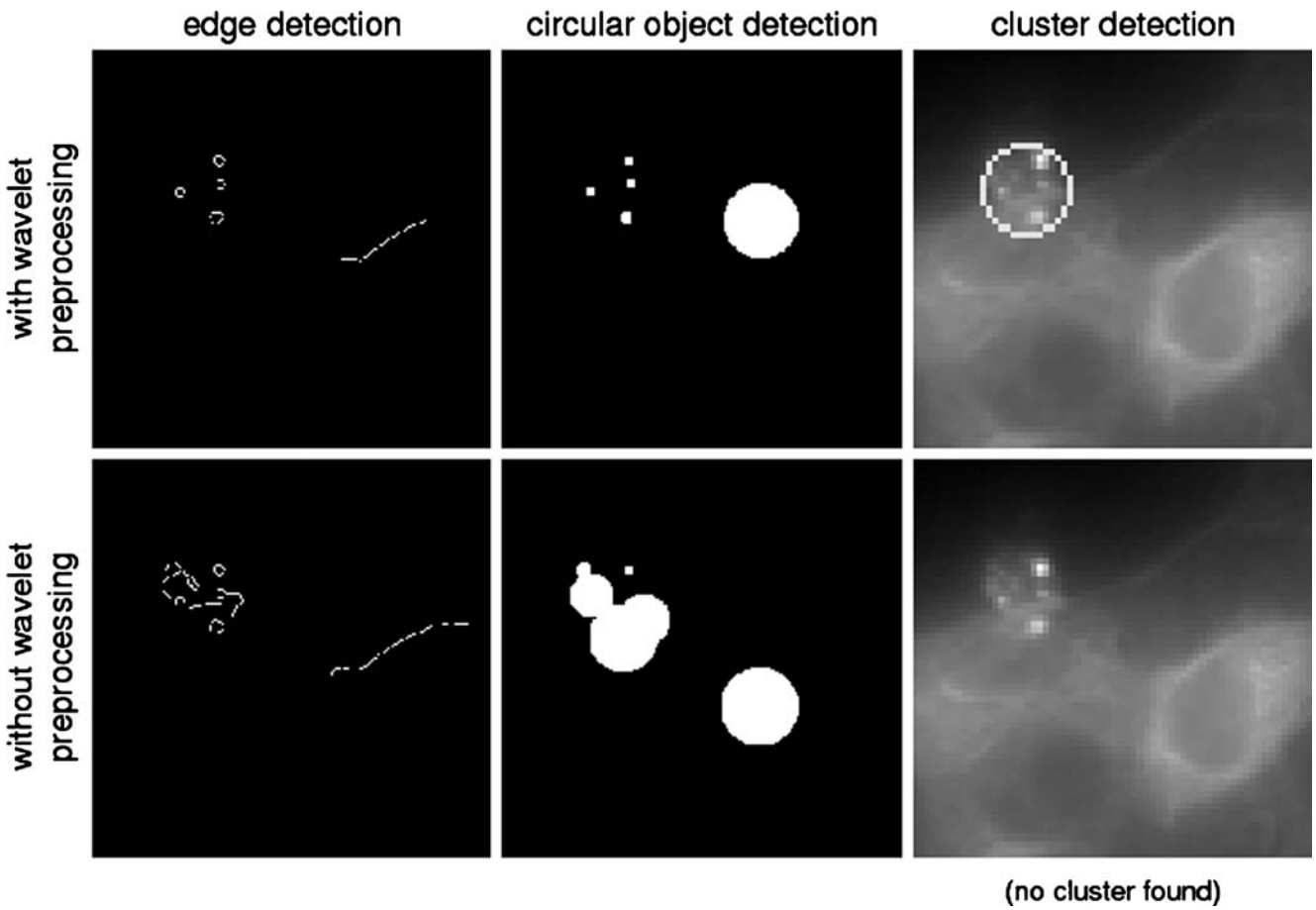


Figure 3 Intermediate results of spot cluster detection. *First column*: Canny edge detection with t_{canny}=0.18: the boundaries of the wild type cell and the spots in the mutant cell in a wavelet preprocessed image can be successfully detected, while some superfluous edges in the mutant cell are detected in the un-preprocessed image; *second column*: Hough transform circular objects detection: spots can be distinguished from the wild type cell in the wavelet preprocessed image, while many false circular objects are detected in the un-preprocessed image due to the unsatisfactory edge detection result; *3rd column*: DBSCAN cluster detection: a spot cluster can be found in the wavelet preprocessed image, while no cluster is found in the un-preprocessed image.

noise, but to detect as many true edges as possible. An example of the edge image (i.e. a binary image where the value is 1 for edge pixels and 0 elsewhere) is shown in Fig. 3, first column. With wavelet preprocessing, the boundaries of the wild type cell and the spots in the mutant cell in a wavelet preprocessed image can be successfully detected. On the other hand, many superfluous edges in the mutant cell are detected in the un-preprocessed image, which may affect the robustness of the further steps in the spot clustering algorithm (see below). This shows that spot pattern can be enhanced by our wavelet preprocessing technique.

2.3.2 Circular Object Detection

For each edge in the edge image, it may be (part of) a wild type cell boundary, or may be (part of) a small spot boundary in a mutated cell. To distinguish between these two cell classes, we use the Hough circle transform [25] on each connected edge component. For each edge, we perform Hough circle transform to find the circle that passes through most pixels of this edge. We can then decide what kind of cell the edge belongs to, based on the size of the detected circle. If the edge forms part of a large circle, then we assign it as part of the cell body. If the edge forms part of a small circle (e.g. with radius $\leq$5 pixels), then we assign it as a small spot boundary. This threshold value (radius $\leq$5 pixels) is carefully tuned (see Fig. 5). An example of the detected circular objects is shown in Fig. 3, second column. In this example, spots can be distinguished from the wild type cell in the wavelet preprocessed image, while many false circular objects are detected in the un-preprocessed image due to the unsatisfactory edge detection result. This poor result will cause misclassification of cell type.

2.3.3 Cluster Detection

Since we are only interested in the pattern of small spots in mutant cells, there are two kinds of detected circles in the result from the previous steps we should remove before detecting clusters. First, we should remove the circles which are fully covered by another large circular object. These spots may be due to noise inside a healthy cell, and should not be considered as a candidate of a small keratin aggregate. Moreover, from the results on the training images shown in Section 3.1, we learn that the radius of small keratin particles will not exceed 5 pixels size (see Fig. 5). Therefore, we should remove those large circles with radius >5 pixel units. Once a set of uncovered small spot particles is obtained, we search for percolating clusters of aggregates. For each spot, we use its center to represent its location. We define a percolating cluster as followed: for every spot in a cluster, there must be another spot in the same cluster with distance less than a predefined radius. To search this kind of clusters, we can use a clustering technique called DBSCAN [26] by setting the number of neighbors in the searching region to be 1 and the predefined radius parameter of searching to be the average cell diameter. This quantity has been learned from a set of training sample images. An example of the detected cluster is shown in Fig. 3, 3rd column. A spot cluster (i.e. a mutant cell) can be found in the wavelet preprocessed image, while no cluster is found in the un-preprocessed image.

2.4 Mutant Classification

Once the clusters are detected, the result can be used to examine existence of mutant in the images. In particular, there are two pieces of information obtained from the detection results for mutant classification: the size (number of spots within a cluster) of every cluster and the number of clusters. As discussed in Section 1 under “Criteria for detection”, it is more important to detect existence of mutants accurately than to count the number of mutants in an image. Since images with large cluster are very likely to consist of phenotypic changes in the filament network and should be chosen as positive ‘hits’, we propose to use the size of clusters in an image as its feature/score. On the other hand, the number of clusters may not be a good feature for classification because small clusters are usually due to noise. Including this in our decision may affect the accuracy of our classifier. Since we are only interested in the existence of phenotypic changes in the images, we will not consider the number of clusters for classification. To classify the images, we first sort the size of all the clusters within an image in descending order. We then rank the images in lexical order and display them in a user-friendly platform as shown in Fig. 4. The lexical order is defined as followed: we first compare the size of the largest cluster of every image and the one within larger size will assign a higher rank. If there is a tie, then we compare the second largest and so and so forth. If there are no more clusters in one of the images for comparison, then we consider that image with a zero-spot cluster for convention. The system will then assign a lower rank for that image. Users can retrieve the results of any individual images by simply clicking on the corresponding link. Sample results can be found in Fig. 5. By defining a cutoff value of this list, we can classify all the images at the top part of the list to be mutant. As a result, the decision boundary between wild type and mutant of our classifier is determined by the cutoff value of the list.

2.5 Implementation

The framework of detection of change of protein distribution is implemented in Matlab R2006b. A computational

Rank	Thumbnail	Clusters
1	Well_D11_image_13	12 : 1
2	Well_D06_image_4	11 : 2 : 2 : 1
3	Well_D03_image_13	10 : 1 : 1 : 1 : 1 : 1 : 1
4	Well_D11_image_12	8 : 3 : 2 : 2 : 1 : 1 : 1 : 1 : 1 : 1 : 1
5	Well_D11_image_6	8 : 2 : 2 : 1 : 1 : 1 : 1 : 1 : 1 : 1
6	Well_D03_image_8	7 : 1 : 1 : 1
7	Well_D02_image_15	7 : 1
8	Well_D03_image_3	6 : 4 : 3 : 1 : 1
9	Well_D05_image_15	6 : 2 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1
10	Well_D06_image_3	6 : 2 : 1 : 1
11	Well_D07_image_7	6 : 1 : 1 : 1
12	Well_C07_image_12	6 : 1 : 1
13	Well_D04_image_14	6 : 1 : 1
14	Well_C01_image_10	6 : 1
15	Well_D07_image_13	6
16	Well_C11_image_12	5 : 5 : 2 : 1 : 1 : 1 : 1
17	Well_D02_image_10	5 : 4 : 1 : 1
18	Well_D02_image_8	5 : 3 : 1 : 1 : 1 : 1
19	Well_C08_image_4	5 : 3 : 1
20	Well_D02_image_12	5 : 2 : 1 : 1
21	Well_C02_image_10	4 : 3 : 1 : 1 : 1
22	Well_C05_image_6	4 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1 : 1
23	Well_C02_image_14	4 : 1
24	Well_C09_image_13	4 : 1
25	Well_C07_image_16	4

Figure 4 The list of the sorted images showing the top 25 ranks. *Column 2* specifies the well number and the image number, which can be used as an index of the images to find the corresponding gene. Sixteen images are taken from each well and numbered from 1 to 16. *Third column* shows the number of spots in each clusters detected using DBSCAN. For example, rank 1 image from well D11, image 13 has one cluster with 12 spots and one cluster with one spot. The images are sorted in lexical order starting from the size of their largest cluster.

intensive module, Hough transform for circular object detection, is implemented in C++ and compiled as a shared library which is executable from within Matlab. All the tests are done with a PC with an Intel Pentium D 3 GHz processor and 2 GB memory. The computer codes are available upon request from the corresponding author.

3 Results

Cells are cultured in 96-well plates as described in Section 2.1. Usually due to experimental fault, a small portion of images will be excluded from the analysis. Therefore, the total number of images in each plate may be varied and the amount we used is about 1,350 images. We test our algorithm using images from Plate-1, Plate-2 and Plate-3 of the second set of 267 genes. First, we work on ten selected training images. These images are manually selected by the biologist to represent the typical pattern we aim to detect. There are 20 mutants altogether in these images, since the pattern we want to search for is very specific, 20 mutants is sufficient for tuning the parameters.. Then we work on a larger set of images (about 200–300 images) and test our algorithm under different situations to show the effect on (1) threshold parameter used in edge detection, (2) the presence of the preprocessing step, and (3) class distribution, i.e. the fraction of positive samples and the fraction of negative samples. To validate our algorithm, we manually classify the images and report the ROC curves of different studies. Finally, we work on the whole set of images from Plate-3 and report the accuracy with different cutoff values.

There are three parameters in our algorithm: (a) sensitivity threshold for Canny edge detection t_{canny}, (b) maximum radius of small keratin spots r_{spot}, and (c) maximum distance of paired spots within each cluster d_{cluster}. For r_{spot}, we can learn from a set of training images as prior biological knowledge tell us that K14 aggregates are of almost identical sizes. For d_{cluster}, the K14 aggregates must be contained within the cell membrane, therefore, we can set d_{cluster} to be the average cell diameter. To learn r_{spot}, we first manually find out all the spots in every training image and report the spot size count in Fig. 5. From the histogram of the keratin particles detected from the training images, we observe that almost 90% of the particles' radii do not exceed 5 pixels. Therefore, we fixed r_{spot} to be 5 pixel units. To learn d_{cluster}, we sample the cells of every training image and obtain their radius manually. Since the

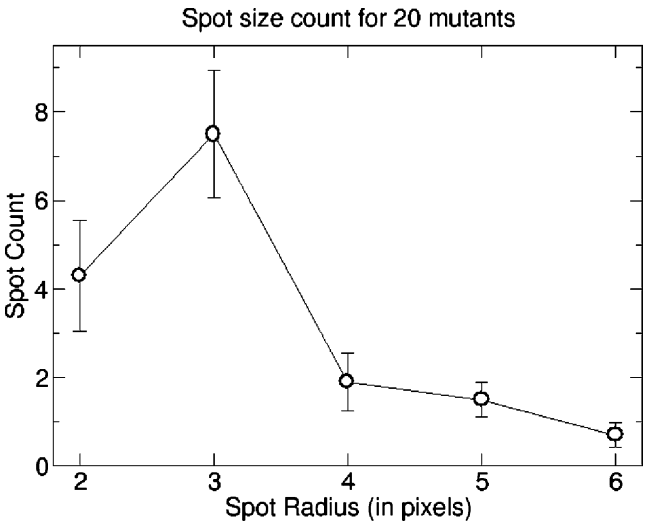


Figure 5 Average number of spots with different radii over 10 training images. Almost 90% of the spot radii do not exceed 5 pixels.

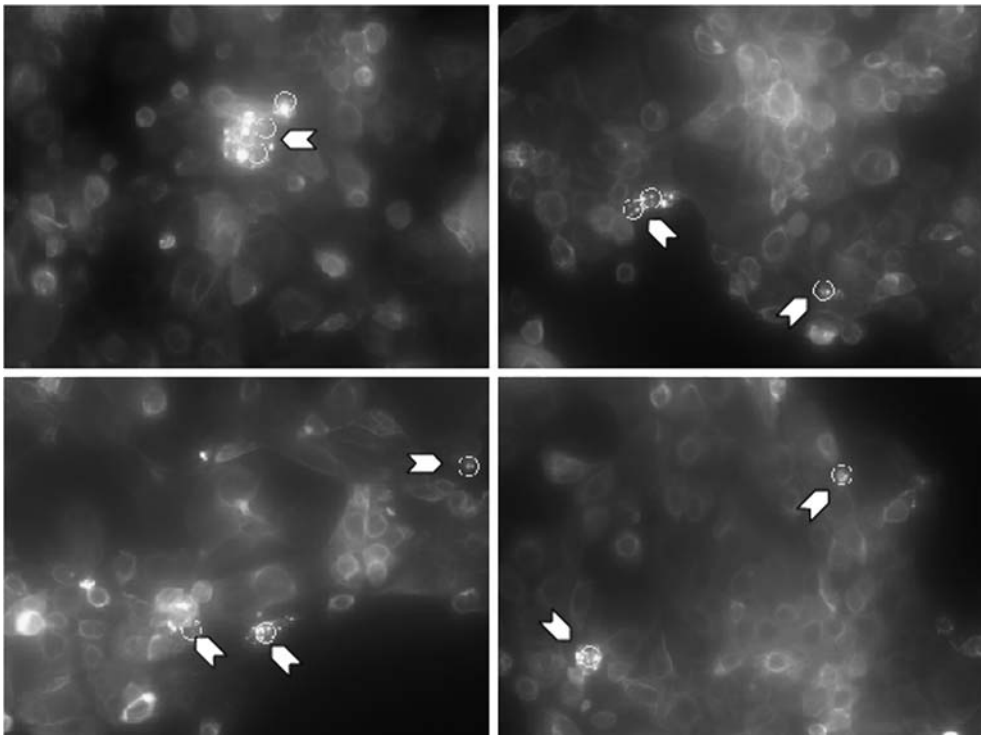


Figure 6 The cluster detection results of 4 training images. Results show that our algorithm is able to detect and locate the keratin aggregates in the images, even when they are located in a sea of untreated cells.

average radius of every image is within 23–30 pixels, we fixed d_{cluster} to be 60 pixel units. For tuning the Canny parameter t_{canny}, we run the algorithm with different values of t_{canny} for a larger set of images (about 200–300 images). Then we compare the results against ground truth to get the ROC curves (see Fig. 7) in order to get the optimal value for t_{canny}.

3.1 Results on Training Images

We first apply our image analysis method to a set of training images, which are some images with mutated cells. There are two aims for this experiment: first, we try to examine the resulting images in order to verify the effectiveness of our method. Next, we can obtain the best values for the parameters r_{spot} and d_{cluster} from the training images. The training images, chosen by the biologists, are less blurred and less noisy. Hence, the patterns of the particles of interest are more significant than that in other images, which is perfect for learning from them the suitable parameter values for the whole set of images. Figure 6 shows that the training images marked with their clusters. We can conclude from the results that our algorithm is able to extract the small particles and locate the keratin aggregates if they exist in the images.

3.2 Results on a Subset of Images

In the first part of this experiment, we choose 288 images prepared in a same experimental run. To show the accuracy of the classification results, we manually obtain the true labels of the images. The class distribution of this set of images is quite uneven: 47 positive and 241 negative. First we study the effect on different Canny threshold values. Next we investigate the difference in the classification results with or without the presence of preprocessing step. We report the true positive rate and the false positive rate

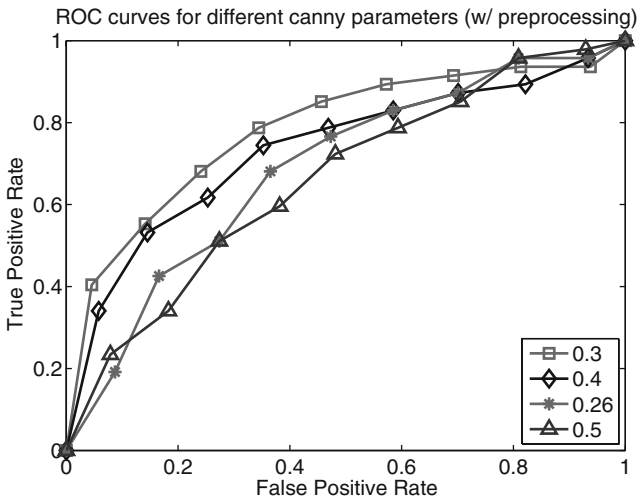


Figure 7 Effect on threshold for edge detection. ROC curves are used to record the different true positive rate and the false positive rate by setting different cutoff values (i.e. 30, 60, 90, etc.) of the list. ROC curves for the scheme with wavelet preprocessing: the scheme with t_{canny}=0.3 outperforms the scheme with other t_{canny} settings almost everywhere.

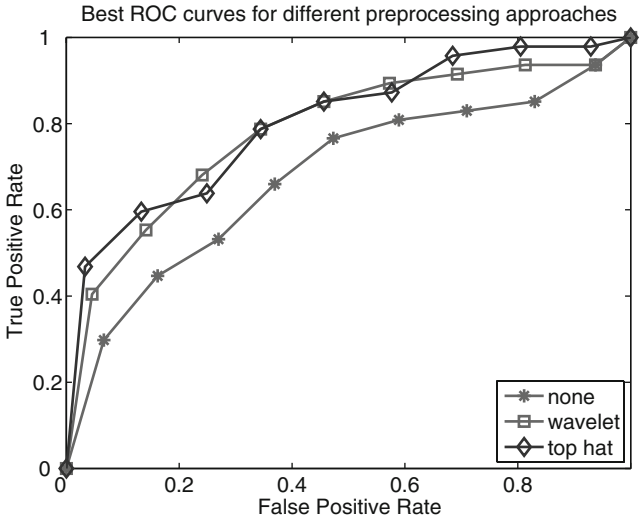


Figure 8 Effect on preprocessing step. The comparison of the best ROC curves shows that our method with wavelet preprocessing always outperforms the one without wavelet preprocessing, while the method with wavelet or with top-hat filter performs comparably. This clearly demonstrates that the wavelet preprocessing step is able to improve the quality of the image for further image analysis task.

after examining every 30 images (i.e. 30, 60, 90, etc.) from the top of the list and the detailed descriptions are as follows:

- Effect on threshold for edge detection. We first preprocess the images using the wavelet filtering technique and apply the Canny edge detector using different Canny parameters (0.26, 0.3, 0.4, 0.5) on the preprocessed images (see Fig. 7). To avoid losing some true edges, one may select a small Canny parameter to make the edge detector more sensitive. However, we observe from the ROC curves that using small parameter (i.e. 0.26) on edge detection usually leads to a poor performance of the classification result. It is caused by the increasing number of false detected edges found by an insensitive edge detector. On the other hand, the poorer performance when using $t_{\text{canny}} > 0.4$ also verifies the fact that using too large parameter for edge detection tends to lose some true edges, which will affect the robustness of the further classification steps. Hence, this result suggests that a suitable parameter (i.e. 0.3 or 0.4) should be chosen to avoid too many false detected edges, while still able to detect useful edges. In fact, the result in Fig. 7 shows that the setting $t_{\text{canny}}=0.3$ gives the best performance. In conclusion, the parameter should be chosen in a range of [0.1, 0.4] in order to obtain a reasonable result.
- Effect on preprocessing step. We first omit the preprocessing step and apply the Canny edge detection algorithm using different Canny parameters (0.14, 0.18, 0.22, 0.26) on the unprocessed images. For the unprocessed images, we use a smaller threshold value since the pattern of the particles of interest is not as significant as in the preprocessed images. We observed that the difference between the results (not shown) using different Canny parameters are quite small. This indicates that there is not much observable improvement by parameter selection. Moreover, the overall performance is poor. We only show the best result in Fig. 8 for comparison. In Fig. 8, we can see that the performance of the classification algorithm with the presence of image preprocessing step is much better. For instance, 80% of true positive images are contained

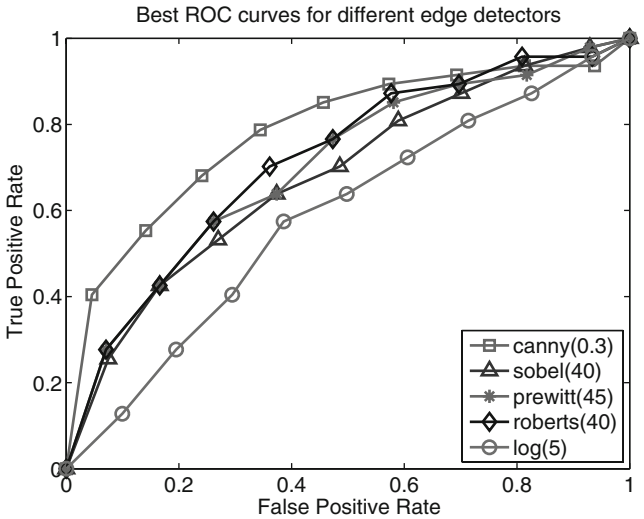


Figure 9 Effect on edge detector. We test the performance of our framework when using different edge detection methods. The sensitivity threshold with the best performance is chosen and its value is shown in the legend. From the result, we notice that Canny outperforms other edge detectors.

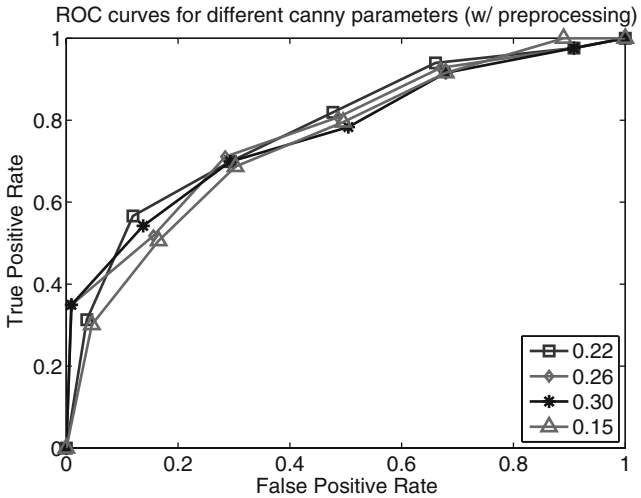


Figure 10 Effect on class distribution. A similar classification results (using different Canny parameters 0.15, 0.22, 0.26, 0.30) on an image set with even class distribution to the result on uneven distributed data (cf. Fig. 6, second column) shows that the performance of our algorithm will not be affected by class distribution of the data.

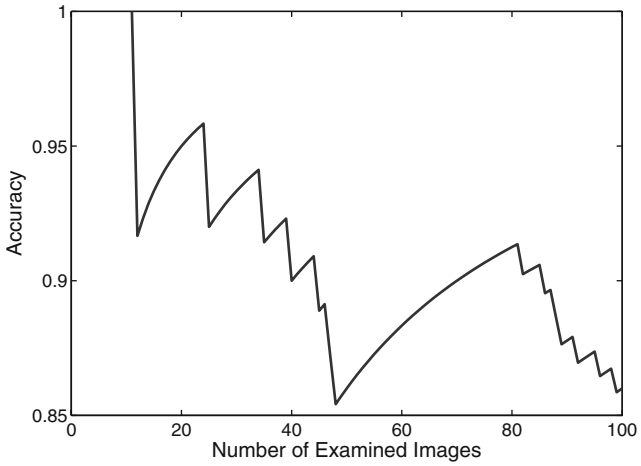


Figure 11 Accuracy of our classifier. A large amount (85 out of the top 100) of images with mutant cells can be found from 7.5% of the whole set of images.

at the top 40% of the list obtained by our method with wavelet preprocessing. However, when un-preprocessed images are used, we need to examine those images at the top 60% of the list to get the same amount of positive images. This can clearly demonstrate that the preprocessing step is able to improve the quality of the image for further image analysis task. Next, we apply another preprocessing method for comparison. In particular, we use a morphological operator called top-hat filter, which is commonly used for detecting and characterizing spots in an automatic manner [27, 28], and keep the same settings in other steps. Since we learnt from the training images that the radii of spots are around 2–6 pixels. Therefore, we apply top hat filter using a disk with 5 pixel radius as our structuring element parameter. From Fig. 7, we can see that the performances of the classifier using wavelet and using top-hat are comparable. This shows again that the preprocessing step is very important. However, one of the disadvantages of the top hat filter is that we need to know in advanced about the size limit of the spots [29], which may be varied in different applications.

- Effect on edge detector. The aim of this experiment is to verify if our choice of Canny edge detector is the best among several well-known edge detectors including Sobel method, Prewitt method, Roberts method and Laplacian of Gaussian method. We fix the settings in other steps; while only change the choice of edge detector. For each method, we use the function implemented by Matlab and only specify the sensitivity threshold. We then try different threshold values and report the one with the best performance in Fig. 9. From the result, we noticed that Canny outperforms other edge detectors. This demonstrates that our choice for edge detection works well in our images.

Note that the class distribution of the first set of images is quite uneven. In the second part of the experiment, we are interested to test our algorithm on a set of images from Plate-1 with even class distribution. We selected 192 images with 83 positive images and 109 negative images.

- Effect on class distribution. We apply our algorithm on this set of images using different Canny parameters and

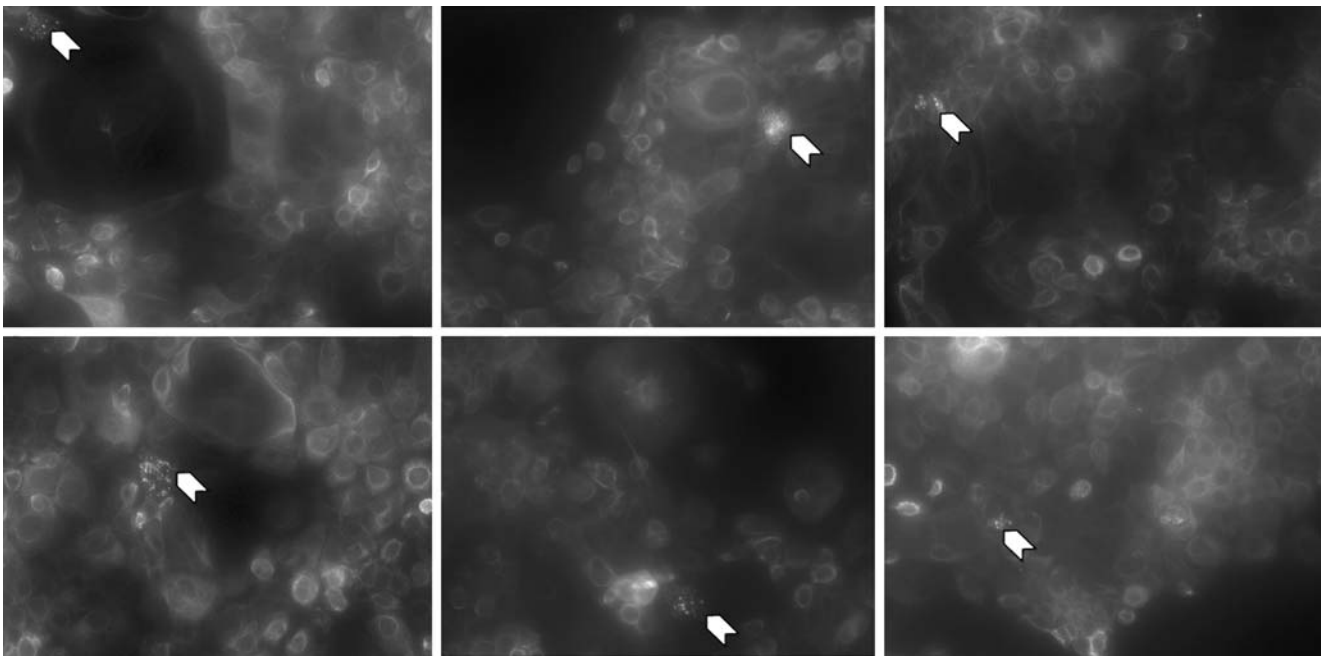


Figure 12 The top 6 images in the list obtained by our algorithm are shown. All of them consist of mutants.

the result is shown in Fig. 10. A similar result to the result on uneven class distributed data shows that the performance of our algorithm will not be affected by class distribution of the data.

3.3 Results on the Whole Set of Images

In this subsection, we demonstrate how to use our method in practice. For this experiment, instead of using only the ranking result obtained by one single value of t_{canny}, we combined the results obtained by different t_{canny} as followed: we first sort a list of ranks of each image in ascending order and then rank the images by their list in lexical order. Suppose we have a set of 1,344 images (i.e. Plate-3). The images will be processed in the following way:

- Apply the image analysis framework using different t_{canny} to the images to extract their cluster information.
- Sort the images according to their maximum cluster size to obtain a ranking list for each t_{canny}.
- Obtain a combined ranking result by combining the ranks with different t_{canny} of every image as described above.
- Examine the images from the top of the list.

In this experiment, we set t_{canny}=0.15, 0.26, 0.3, 0.4, 0.5. Moreover, as mentioned in Section 1, there is less than 10% of the silenced genes cause phenotypic changes. Therefore, we set the cutoff to be 100 (~7.5%), which is a reasonable amount of images for manually examining by the biologist. Then the top 100 images is examined and labeled manually as ground truth. The accuracy of the classification results is reported in Fig. 11. Note that there are 85 positive images out of the top 100 images, which shows that our algorithm can successfully filter out many wild type images therefore making inference of gene knockdown much easier. In particular, more than ten images at the top of the list are all containing mutant cells and the top 6 images are shown in Fig. 12.

4 Discussion

High-throughput imaging techniques become very popular since they can help scientists to study biological events in a dynamic way. However, due to the size of such huge data set, automatic image analysis tools are essential. In this paper, we proposed an image analysis framework which is based on advanced image processing techniques to extract spot patterns and classify mutants among a large set of images. Tests performed on the set of skin cell images demonstrate that our method can successfully filter out many wild type images therefore making inference of gene knockdown much easier. Moreover, we observe that our framework can be easily generalized to other high-throughput image analysis problems. For instance, we can combine different wavelet coefficients to obtain different interest patterns. Therefore, a more generic image processing framework will be considered as one of our future directions.

Acknowledgement This work was supported (in part) by the Biomedical Research Council of A*STAR (Agency for Science, Technology and Research), Singapore.

References

1. Haralick, R. M., Shanmugam, K., & Dinstein, I. (1973). Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, *3*(6), 610–621.
2. Duda, R. O., Hart, P. E., & Stork, D. G. (2000). *Pattern classification* (2nd ed., p. 680). New York: Wiley-Interscience.
3. Moll, R., Franke, W. W., Schiller, D. L., Geiger, B., & Krepler, R. (1982). The catalog of human cytokeratins: Patterns of expression in normal epithelia, tumors and cultured cells. *Cell*, *31*, 11–24.
4. Fuchs, E., & Weber, K. (1994). Intermediate filaments: Structure, dynamics, function, and disease. *Annual Reviews of Biochemical*, *63*, 345–382.
5. Irvine, A. D., & McLean, W. H. (1999). Human keratin diseases: The increasing spectrum of disease and subtlety of the phenotype-genotype correlation. *British Journal of Dermatology*, *140*, 815–828.
6. Szeverenyi, I., Cassidy, A. J., Chung, C. W., Lee, B. T., Common, J. E., Ogg, S. C., et al. (2007). The human intermediate filament database: Comprehensive information on a gene family involved in many human diseases. *Human Mutation*, *29*, 351–360.
7. Ku, N. O., Liao, J., Chou, C. F., & Omary, M. B. (1996). Implications of intermediate filament protein phosphorylation. *Cancer Metastasis Reviews*, *15*, 429–444.
8. Herrmann, H., Hesse, M., Reichenzeller, M., Aebi, U., & Magin, T. M. (2003). Functional complexity of intermediate filament cytoskeletons: From structure to assembly to gene ablation. *International Review of Cytology*, *223*, 83–175.
9. Coulombe, P. A., & Omary, M. B. (2002). 'Hard' and 'soft' principles defining the structure, function and regulation of keratin intermediate filaments. *Current Opinion in Cell Biology*, *14*, 110–122.
10. Coulombe, P. A., Hutton, M. E., Letai, A., Hebert, A., Paller, A. S., & Fuchs, E. (1991). Point mutations in human keratin 14 genes of epidermolysis bullosa simplex patients: Genetic and functional analyses. *Cell*, *66*, 301–311.
11. Toivola, D. M., Zhou, Q., English, L. S., & Omary, M. B. (2002). Type II keratins are phosphorylated on a unique motif during stress and mitosis in tissues and cultured cells. *Molecular Biology of the Cell*, *13*, 1857–1870.
12. Windoffer, R., & Leube, R. E. (2004). Imaging of keratin dynamics during the cell cycle and in response to phosphatase inhibition. *Methods in Cell Biology*, *78*, 321–352.
13. Neumann, B., Held, M., Liebel, U., Erfle, H., Rogers, P., Pepperkok, R., et al. (2006). High-throughput RNAi screening by time-lapse imaging of live human cells. *Nature Methods*, *3*(5), 385–390.
14. Yarrow, J. C., Perlman, Z. E., Kirchhausen, T., & Mitchison, T. J. (2003). Phenotypic screening of small molecule libraries by high throughput cell imaging. *Combinatorial Chemistry & High Throughput Screening*, *6*(4), 279–286.
15. Kneller, A. (2006). The new age of bioimaging. Paradigm, Fall, pp. 18–25.

16. Chen, X., Velliste, M., & Murphy, R. F. (2006). Automated interpretation of subcellular patterns in fluorescence microscope images for location proteomics. *Cytometry A*, *69A*(7), 631–640.
17. Conrad, C., Erfle, H., Warnat, P., Daigle, N., Lörch, T., Ellenberg, J., et al. (2004). Automatic identification of subcellular phenotypes on human cell arrays. *Genome Research*, *14*, 1130–1136.
18. Chen, X., & Murphy, R. F. (2005). Objective clustering of proteins based on subcellular location patterns. *Journal of Biomedicine and Biotechnology*, *2005*(2), 87–95.
19. Bakal, C., Aach, J., Church, G., & Perrimon, N. (2007). Quantitative morphological signatures define local signaling networks regulating cell morphology. *Science*, *316*, 1753–1756.
20. Jones, T. R., Carpenter, A. E., Sabatini, D. M., & Golland, P. (2006). *Methods for high-content, high-throughput image-based cell screening*. Proceedings of the Workshop on Microscopic Image Analysis with Applications in Biology, pp. 65–72.
21. Glory, E., & Murphy, R. F. (2007). Automated subcellular location determination and high-throughput microscopy. *Developmental Cell*, *12*, 7–16.
22. Burrus, C. S., & Copinath, R. A. (1997). *Introduction to wavelets and wavelet transforms* (p. 268). NJ: Prentice Hall.
23. Starck, J. L., Murtagh, F., & Bijaoui, A. (1995). Multiresolution support applied to image filtering and restoration. *Graphical Models and Image Processing*, *57*(5), 420–431.
24. Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *8*, 679–714.
25. Jain, A. K. (1988). *Fundamentals of digital image processing* (p. 592). NJ: Prentice Hall.
26. Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining, pp. 226–231.
27. Bright, X., & Steel, E. B. (1987). Two-dimensional top hat filter for extracting spots and spheres from digital images. *Journal of Microscopy*, *146*(2), 191–200.
28. Breen, X., Joss, G. H., & Williams, K. L. (1991). Locating objects of interest within biological objects: the top hat box filter. *Journal of Computer-Assisted Microscopy*, *3*(2), 97–102.
29. Olivo-Marin, J. C. (2002). Extraction of spots in biological images using multiscale products. *Pattern Recognition*, *35*, 1989–1996.

Yan Nei Law received her PhD degree in Computer Science from the University of California, Los Angeles, in 2006. She is a post doctoral research fellow in Bioinformatics Institute, Singapore. Her research interests include issues related to bioimaging with emphasis on developing image processing and data mining techniques for high throughput images.

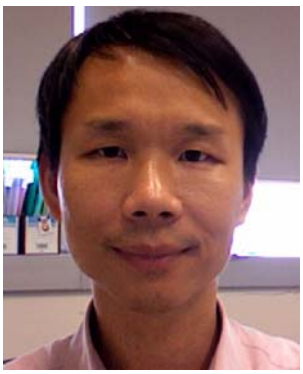
Stephen Ogg is a Senior Scientist in the Institute of Medical Biology and head of the institute's microscopy unit. His interest lies in understanding the dynamics of intermediate filaments in keratinocytes & exploring unique methods for extracting biological information from micrographs.

John Common completed his Ph.D. at the Centre for Cutaneous Research, Queen Mary, University of London under the supervision of Professor David Kelsell. He is now working with Professor Birgit Lane in the Epithelial Biology Programme, Institute of Medical Biology, Singapore his interests are to understand disease mechanisms in the skin and other epithelial tissues.

David Tan is a PhD student in Fiona Watt's laboratory in the Wellcome Trust Centre for Stem Cell Research, University of Cambridge, UK. He is funded by the Agency for Science, Technology and Research (A*STAR), Singapore.

Birgitte Lane is Coordinator of the Epithelial Biology program and Executive Director of the Institute of Medical Biology, Biomedical Sciences Institutes, A*STAR, Singapore. Her research focuses on elucidating the molecular details underlying inherited skin fragility disorders caused by mutations in intermediate filament genes.

Andy M. Yip received the B.Sc. degree in mathematics from the Chinese University of Hong Kong in 1998, the M.Phil. degree in mathematics from the University of Hong Kong in 2000, and the Ph.D. degree in mathematics from the University of California, Los Angeles, in 2005. He joined the Department of Mathematics at the National University of Singapore in July 2005 as an Assistant Professor. His research interests include variational and PDE methods in image processing and data mining algorithms.



Lee Hwee Kuan is a group leader in the Imaging Informatics division in Bioinformatics Institute. After obtaining his Ph.D. in Theoretical Physics from Carnegie Mellon University, he held a joint postdoctoral position with Oak Ridge National Laboratory (USA) and University of Georgia. In 2003, was awarded a fellowship from the Japan Society for Promotion of Science, to work in Tokyo Metropolitan University. In 2006, he joined Bioinformatics Institute as a Principle Investigator in the Imaging Informatics Division.