SOFTWARE METAPAPER

CHLOE: A Software Tool for Automatic Novelty Detection in Microscopy Image Datasets

Saundra Manning[1] and Lior Shamir[1]

[1] Department of Computer Science, Lawrence Technological University, USA

The recent advancements in automated microscopy and information systems allow the acquisition and storage of massive datasets of microscopy images. Here we describe CHLOE, a software tool for automatic extraction of novelty in microscopy image datasets. CHLOE is based on a comprehensive set of numerical image content descriptors reflecting image morphology, and can be used in combination with ROI detection and segmentation tools such as ITK. The rich feature set allows automatic detection of repetitive outlier images that are visually different from the common images in the dataset. One of CHLOE's primary advantages is its versatility, allowing it to detect outlier images in image databases of different subjects and different types of microscopy without modification to the software. The code and software are publicly available for free download at http://vfacstaff.ltu.edu/lshamir/downloads/chloe.

Keywords: Image analysis; Microscopy; Outlier detection

(1) Overview

Introduction

In the past decade, automated microscopy and high-throughput/high-content biological imaging pipelines have developed rapidly, and with the increasing availability of computing resources and storage devices led to the acquisition of very large datasets of biological images. These high-content screens have become an important tool in drug discovery, but applications also include small molecule screens [1, 2], sub-cellular localization [3], gene functionality [4, 5], and more.

Automatic microscopy is also used in the medical domain such as histopathology, producing large databases of microscopy images in digital format. However, while effective autonomous imaging devices and computing and storage resources have been significantly improving in the past decade, the bottleneck for optimal use of these automated methods remains the imperfection of the machine vision and pattern recognition algorithms [6, 7].

One of the challenges of experimentalists who work with large datasets of microscopy images is outlier detection - detecting repetitive phenotypes that are visually different from the common phenotypes. That is, if a certain gene, for instance, is expressed in just 1% of the cells, the experimentalist who analyses the microscopy images manually might not easily notice that and will therefore not be able to use that information to comprehensively study the functionality of the gene. Also, a certain treatment might lead to rare but consistent phenotypes, and an experimentalist working with these image data might find it difficult to detect these rare phenotypes by manual observation, especially in cases where the resulting phenotypes are not very different from each other visually.

An outlier is a data point or points that are markedly different from other data points in the same sample set. Outlier detection is the automatic identification of these data points. Many different algorithms have been proposed for performing outlier detection based on statistic [8], distance [9, 10, 11, 12], density [13, 14], clustering [15, 16, 17, 18], and deviation [19, 20, 21].

While numerous outlier detection algorithms have been proposed, the existing literature provides far less information about experiments of outlier detection with image data, or tools that can detect novelty in large datasets of images. In particular, little work has yet been reported on methods and tools for outlier detection in microscopy images. Additionally, the outlier detection methods mentioned above aim at identifying single outliers in a dataset, a task that is not suitable for the field of microscopy since an outlier is of interest only if that phenotype is consistently detected in more than one instance and in a replicable fashion. Therefore, a tool for outlier detection in microscopy images will be of better use if it can detect phenotypes that are rare, but have more than one instance in the dataset. For instance, in the example above of a gene expressed in just 1% of the cells, an outlier detection algorithm that detects single outliers will return a set of images of peculiar cells that happen to exist in the screen, and the cells with the expressed gene might be inside that set, making it difficult for the microscopist to detect them. If the outlier detection algorithm

detects only outlier images that appear numerous times in the screen, the experimentalist will receive a set of outlier cells that only include the cells in which the gene is expressed. Automatic analysis also has the advantage of being reproducible and more objective compared to manual analysis [22].

Here we describe Compound Hierarchical Learners for Outlier Extraction (CHLOE), a software tool that can be used by experimentalists for outlier detection in a broad range of biological experiments. The method adjusts itself automatically to the data being analysed, and therefore can be applied to different subjects and different types of microscopy without changing the code. The user is not required to be familiar with pattern recognition methods in order to apply the method to their own unique data.

As described above, applications of the software tool include high-content screening, but the versatile nature of the method makes it effective for detecting outlier images in many other types of experiments. For instance, in histopathology it can be used to detect microscopy images that are different from the "typical" images in that scan, and therefore can assist in utilizing robotic microscopy to collect very many images of the same patient and analyse them automatically to detect anomalies and optimize the diagnostics power.

The method is also not limited to microscopy images, but can be applied to other types of medical imaging such as radiology, and detect outlier cases of different body parts imaged as part of population studies or in radiology image databases. Example experiments described in the paper include brightfield and fluorescence microscopy, as well as plain radiology, but applications can also include electron microscopy, histopathology, FTIR (Fourier Transform Infrared Spectroscopy), and more.

Implementation and architecture
The method is based on the CHARM (Compound Hierarchical Algorithms Representing Morphology) feature set [23], which is a comprehensive set of numerical image content descriptors that reflect very many aspects of the visual content such as texture, shape, colour, edges, fractals, polynomial decomposition of the image, and statistical distribution of the pixel intensities [23][24]. To obtain better signal, these descriptors are extracted not just from the raw image, but also from image transforms and multi-order image transforms as thoroughly discussed in [23, 24]. This large feature set provides a comprehensive analysis, and therefore can be applied to various experiments that involve different types of microscopy, magnifications, and organisms [25].

WNDCHRM is an independent image classifier and works without CHLOE. Once the numerical image content descriptors are computed by WNDCHRM, CHLOE is applied to the dataset of numerical image content descriptors to detect outliers. Therefore, CHLOE uses just the feature extraction capabilities of WNDCHRM, but does not use its analysis and pattern recognition algorithms. In any case, CHLOE is executed only after WNDCHRM is executed

and produces the values of the numerical image content descriptors. **Figure 1** visualizes the CHLOE process.

The automatic outlier detection method is an expansion of a method that was previously used to detect peculiar astronomical objects [26, 27]. Once the numerical image content descriptors of the CHARM feature set are computed for each image, the values are normalized to the [0,100] interval to eliminate any numeric bias. In the next step, the mean, median, and variance of each image feature are computed. To characterize the "typical" feature values of an image in the dataset, the highest 5% and the lowest 5% of the values of each image feature are ignored when computing the mean and variance, so that extreme values that result from noise, artefacts, or outlier images will not affect the mean and variance of the "typical" images [23, 24, 28, 29, 30].

The higher the variance of the feature values, the more difficult it is to use those values to determine what a "typical" feature value is. If there is a low range of variability of a given feature, that feature is considered a potentially stronger indicator for detecting an outlier image, which might have different values of that particular feature compared to the other non-outlier images in the dataset. Since the CHARM feature set includes 2,873 different image features, it is expected that not all of them are informative for a particular image analysis experiment, and therefore many of the features represent noise [23]. In order to reduce the effect of non-informative features, by default 90% of the features with the highest standard deviations are ignored, as these are assumed to poorly represent the typical image in the dataset. The remaining 10% of the features are weighted using the standard deviation to further improve the accuracy by allowing more informative features to have a larger impact on the analysis.

Features with high kurtosis can be assigned with high feature weights. However, in that case the values of non-outlier samples are expected to be close, and therefore the total effect on the distances between samples will not be significant. If a certain non-outlier sample happens to have a substantially different value in a feature with a high kurtosis, that feature will be outweighed by the other features of the very large feature set used in the analysis.

Next, for each image sample, Euclidian distance between each pair of images u and w in the dataset is calculated using **Equation 1**.

Eq. 1
$$d = \sum_i \sigma_i \left(f_{u,i} - f_{w,i}\right)^2,$$

where d is the weighted Euclidean distance between image u and image w, $f_{u,i}$ is the value of feature i computed from the image u, $f_{w,i}$ is the value of feature i computed from image w, and σ_i is the standard deviation of the feature i across all images in the dataset. That is, the distance is the sum of the weighted square distances between the feature values of the two images, such that the weights are the standard deviation of the feature.

Once distances between all possible pairs of samples in the dataset are computed, the distances for each sample
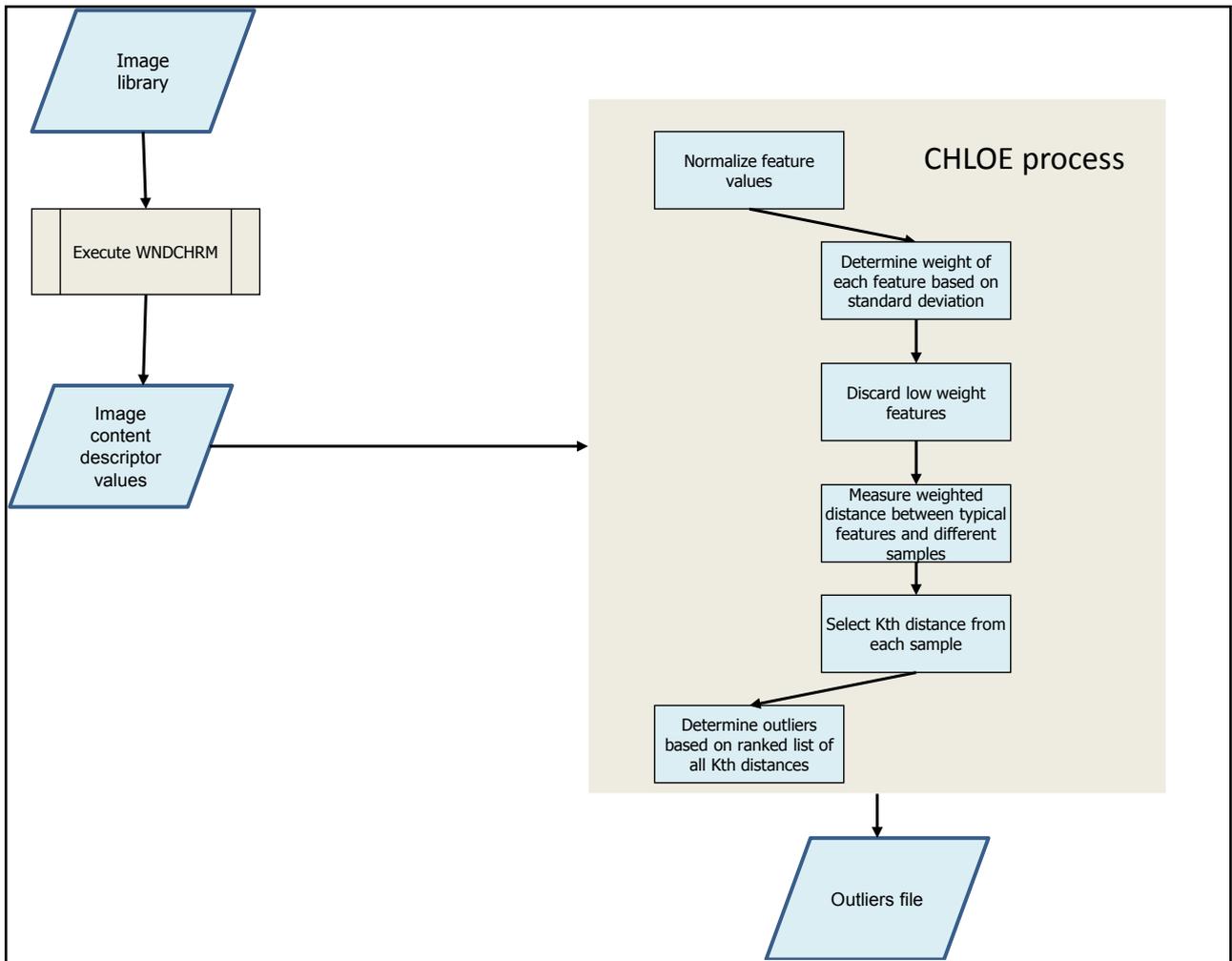
Figure 1: Steps of the CHLOE process of image analysis. First the image features are computed using WNDCHRM, and then CHLOE is applied. The CHLOE input is the feature values computed by WNDCHRM, and its output is a file that contains the most likely outliers.

are sorted in an ascending order, and the Kth shortest distance from each sample is selected. Then, the Kth distances of all samples are ordered, and the largest distances among these Kth distances are selected. The samples with the largest Kth distance are assumed to be the possible outlier samples, since they have less than K neighbours that are visually similar to it.

The weighted distances are highly important for accurately reflecting the image morphology in the context of an image analysis problem [24, 30], and therefore the weights of the features are critical in outlier image detection when the image morphology is complex, and each image is represented by a large number of features. The standard deviation is a simple weighting policy, but is effective in providing a consistent distance between all pairs of samples, while its sensitivity to possible non-informative features with high weights is highly limited due to the use of the very large feature set.

Except for the file name where the feature values are stored, CHLOE takes three parameters. One is the K parameter described above, which is the minimum number of neighbours that a sample is required to have to be detected as an outlier. The second parameter is q, which is the number of output outliers that CHLOE returns as

output, ordered by their distance from the "typical" image. The distance reflects the dissimilarity between the sample and the "typical" image in the dataset, and therefore the likelihood of the sample to be a true outlier. The third parameter is j, which is used for the purpose of testing the performance of the method when the regular and outlier images are known. That parameter determined the number of images from the outlier class that are combined in the class of the regular images, so that the capability of the algorithm to detect them can be tested.

The output of CHLOE is a list of samples that are the most dissimilar to the "typical" image in the dataset, and there are more likely to be outliers. As mentioned above, the outliers also need to meet the criterion specified by the k parameter.

In order to perform outlier detection using CHLOE, the first required task is computing image content descriptors for all images in the dataset. These values describe the image content in a numeric fashion that can be processed by pattern recognition tools. For that purpose we use the WNDCHRM tool, which computes the CHARM comprehensive set of numerical image content descriptors, and is available for free download. WNDCHRM runs from the command line using the following command syntax:

```
> wndchrm train [options] images feature_file
```

where *feature_file* is the resulting output file that stores the image feature values, images is a path to the top folder where the images of the dataset are stored, and [*options*] are optional switches that can be specified by the user. *WNDCHRM* is thoroughly described in [23, 24]. A sample command line run in Microsoft Windows, using a library of pollen images [25], follows:

```
>wndchrm train –ml c:\path\to\pollen c:\path\to\
pollen\pollen.fit
```

The file "pollen.fit" is the file that is generated by *WNDCHRM*, and contains all image features computed for all images in the sub-folders of the "pollen" folder [23]. Since the *-m* switch is used, when the features of a certain image are computed, a .sig file (with the name of the image) will be created in the same folder of that image [23].

Computing the image features can be slow, and depending on the number and size of the images typically takes between a few hours to a few days to complete computing [23]. In the case of a system with more than one processor several instances of *WNDCHRM* to can be run to expedite the process and utilize the computing power of the system. Starting another instance of *WNDCHRM* can be done by repeating the same command line. A full description of the *WNDCHRM* command line utility and the *WNDCHRM* algorithm can be found at [23].

The second step in the process uses the CHLOE program described in this paper. CHLOE does not have an integrated graphical user interface, and all user interactions are performed using simple command-line instructions.

Once the *WNDCHRM* output file has been created, CHLOE can be used to analyse the file for detecting outliers.

```
>Chloe.exe rank  -q10 –k10 inputfile.fit
```

Where *inputfile.fit* is the output file created from the *WNDCHRM* utility, *-k* is a parameter value which specifies the minimum number of similar samples, and *-q* is the number of likely outliers the program should return. The k parameter can be entered on the command line as either an integer value (using the lowercase k) or a percentage of the number of image files in inputfile.fit (using an uppercase K). As an example, if the user enters a value of –K10, CHLOE converts it to represent 10 percent of the number of images in the input file. If the input file contains 200 images, a value of 20 will be used for k.

Once the program execution is complete, a file called "chloe_output.txt" is created and saved to the same directory, which contains the chloe.exe executable file. The chloe_output.txt file is a comma-separated text file, and can be manually viewed or used to load into any spreadsheet, database, or word processing program for analysis. It contains two items: the order the proposed outlier was returned, and the proposed outlier image file. The lines in the "chloe_output.txt" file are ordered such that the first

line is the sample that is most likely to be an outlier in the dataset.

It should be noted that CHLOE ranks the samples by their relative likelihood to be outliers, based on the weighted Euclidean distances. Therefore, the top samples detected by CHLOE are not necessarily outlier images, and require further analysis by the experimentalist. However, in a large microscopy image dataset, CHLOE can point the experimentalist to the phenotypes of interest, a task that might be labour intensive without using automation.

The −q value switch given to the program determines the number of outliers returned. If the user runs the program with a −q value of 5, the program will return the 5 most likely outlier images such as the following:

- 1,/iicbu/shamirl/images/fruitflyannotation/Tiff-Data/stage4_6/g463mask_s2_87.tiff
- 2,/iicbu/shamirl/images/fruitflyannotation/Tiff-Data/stage13_16/g463mask_s6_62.tiff
- 3,/iicbu/shamirl/images/fruitflyannotation/Tiff-Data/stage13_16/g463mask_s6_8.tiff
- 4,/iicbu/shamirl/images/fruitflyannotation/Tiff-Data/stage13_16/g463mask_s6_98.tiff
- 5,/iicbu/shamirl/images/fruitflyannotation/Tiff-Data/stage13_16/g463mask_s6_73.tiff

In the example above, the samples ranked at the sixth place or under will not be shown to the user. If the user is interested in viewing more outlier samples, she needs to set the −q value to the desired number of outliers she wishes to receive from the program.

Neither WNDCHRM nor CHLOE apply any automatic detection of ROI. Therefore, using CHLOE should be preceded by a first step of ROI detection and separation of these ROIs from the raw images. This task can be done by using some of the mature open source segmentation tools that exist such as ITK [31].

Quality control

Testing was conducted using four different biological image libraries: the CHO (Chinese Hamster Ovary) dataset [32], consisting of fluorescence 512x382 microscopy images of different sub-cellular compartments, the Pollen dataset [33], which is a dataset of 25x25 images of geometric features of pollen grains, the HeLa dataset [33], and a library of fruit fly microscopy images taken at different days of development. The first three datasets are available for free download as part of the IICBU-2008 benchmark suite [25] at http://ome.grc.nia.nih.gov/iicbu2008, and sample images of the pollen classes used in the experiments are shown in **Figure 2**. All experiments were done in Windows 7 operating system. It should be noted that the CHO and HeLa dataset could have a certain bias in the way the data were acquired [34], and therefore the outlier can be detected also by the batch in which each image was acquired rather than the morphology of these cells.

Larger datasets that were tested include the RNAi dataset [35], which is a screen of DAPI stained fruit fly cells such that 16 different genes are knocked down, and the cells were separated from the images to produce a dataset
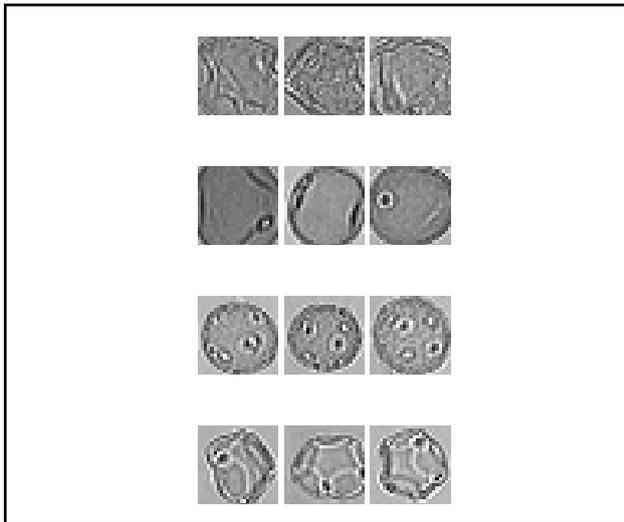
Figure 2: Sample images of class "212" (first row), "406" (second row), "198" (third row), and "216" (fourth row) taken from the pollen dataset, and class "giantin" (last row).

of 12,583 cell images of dimensionality of 60x60 [35]. Another relatively large dataset that was used is a dataset of 1,600 knee x-rays taken from the Osteoarthritis Initiative (OAI), such that 1000 knee x-rays are of women

and 600 x-rays are of knees of men. The dataset of C. elegans terminal bulb at different ages [36] is also different from the other datasets in the sense that the subjects are tissues and not cells. The images were acquired using differential inference contrast (DIC) microscopy, and imaged the terminal bulb of C. elegans at different ages, from 0 to 12 days.

The image datasets used in this study, and how they are compared to each other in the experiments, are listed in **Table 1**.

After all of the numerical image content descriptors were obtained by running WNDCHRM, a Perl script was run to create a test directory of particular content descriptors to analyse. For instance, 90 samples of pollen grains from a "typical" class and 1-10 samples from an "outlier" class were combined into a single feature file. The Perl script then executed CHLOE against the image content file. The Perl script communicates what the expected outlier file is, based on the test directory contents, to enable the ability to measure the performance against the expected outcome.

The performance of CHLOE was evaluated as the number of times the method correctly detected one or more of the outlier images divided by the number of times the program was executed. For example, when using a library of 95 images containing 90 base class images and

Dataset	Typical class	Outlier class	Images in typical class
Pollen	198	212	90
Pollen	212	198	90
Pollen	216	406	90
Pollen	406	216	90
CHO	hoechst	giantin	69
CHO	gianti	hoechst	69
Hela	actin	dna	98
Hela	dna	Actin	87
Hela	golgpp	Er	86
Hela	er	golgpp	85
Fruit Fly	stage 4 to 6	stage 13 to 16	90
Fruit Fly	stage 13 to 16	stage 4 to 6	90
C. elegans TB	Day 0	Day 6	112
C. elegans TB	Day 0	Day 10	112
C. elegans TB	Day 0	Day 12	112
RNAi	Untreated	CG7825	1500
RNAi	Untreated	CG8114	1500
RNAi	Untreated	CG8711	1500
OAI	Women	Men	1000
OAI	Men	Women	600

Table 1: Image datasets used for the experiments. Each experiment includes one set of typical images, and one set of outlier images. The method is used to detect a single outlier image in the set of outlier images.
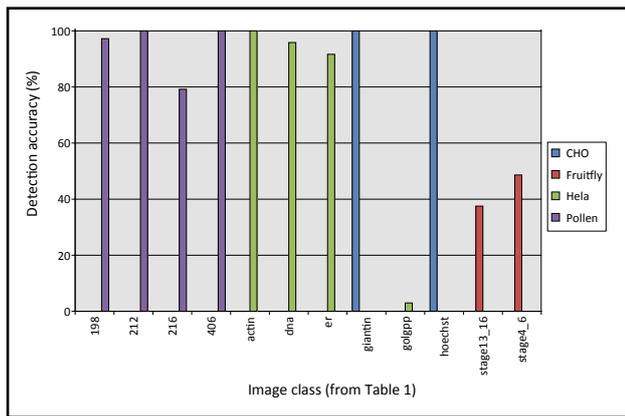
Figure 3: The average detection accuracy of the outlier images for all image classes in Table 1 (where $q = 5$ and $j = 5$). In each experiment one class of images was used as the collection of outlier images, and another class was used as the typical images as specified in Table 1. The detection accuracy reflects the ability of the method to correctly detect an outlier image in the set of typical images.

Rank	Image file	Image
1	obj_406_5	
2	obj_212_9	
3	obj_406_7	
4	obj_406_6	
5	obj_406_9	
6	obj_406_1	
7	obj_406_10	
8	Obj_406_14	

Figure 4: The eight top images detected by CHLOE when using the pollen experiment. The outlier is ranked second among the 91 images used in the experiment.

5 outlier class images, if the program found any of the outlier images during the single execution that execution was considered successful. **Figure 3** charts the average detection accuracy for all image classes tested.

Figure 4 shows the images that were detected by CHLOE as outliers in the pollen dataset, also available in CHLOE download page. The images are ranked by their distance from the "typical" image, so the most likely outlier images are ranked higher. The outlier (212_9) is ranked second among the 91 images used in the experiment.

Several variables and their effect on the performance were evaluated. The number of likely outliers (q), the order of the nearest neighbour to select for each sample to evaluate its relative distance to other samples in the dataset (k), and the number of outliers to place in the test directory (j) were changed during different program executions. As expected, the higher the q value, the greater likelihood that the correct outlier was detected and the run was considered successful. **Figure 5** displays the
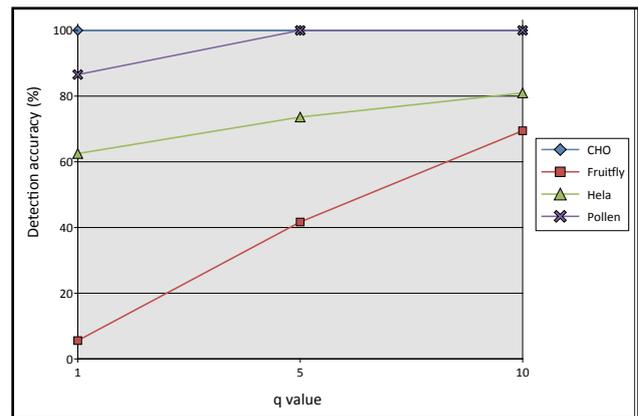
Figure 5: The detection accuracy of the outlier image as a function of the q value (where k = 10). Clearly, the higher the number of likely outlier gets larger, the higher probability that the actual outlier will be among the detected outliers. The downside of increasing the value of q is that the program will make more detections, and therefore will require more human labour to analyse the output manually. For instance, when q is equal to 10 it means that the experimentalist will need to examine 10 output samples manually.
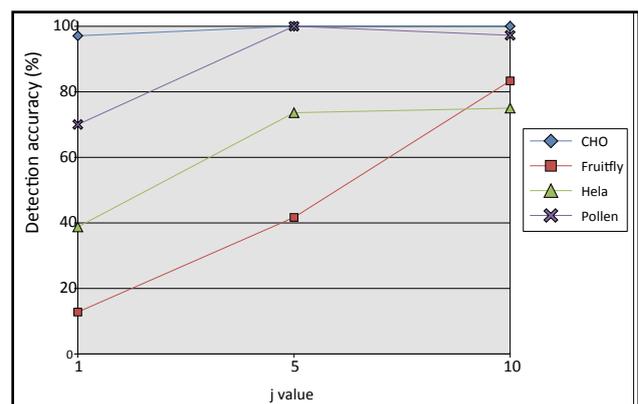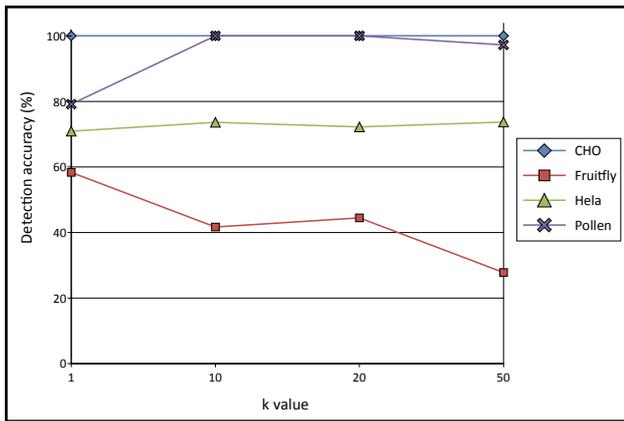

Figure 6: The detection accuracy of the outlier image as a function of the j value (where $q = 5$ and $k = 10$). When more outliers are placed among the typical images, the method has a higher chance of detecting one of them as the actual outlier.

detection accuracy as a function of the value of q, where k is equal to 10.

The number of outliers placed in the test directory (j) was changed in order to test whether more outlier images would make it more difficult or easier for the method to detect the outlier. **Figure 6** shows the detection accuracy as a function of the value of j, where q is equal to 5 and k is equal to 10. A detection attempt is marked as successful if even a single outlier is detected; therefore the detection accuracy increases as more outliers are placed in the test directory.

Different values of k were used to determine whether an outlier in a directory with other members of the same outlier class could still be detected. In other words, an outlier image can have neighbours and still be an outlier if it has less than K neighbours. Therefore, the k value allowed

Figure 7: The detection accuracy of the outlier image as a function of the k value (where $q = 5$ and $j = 5$). A higher k value reduces the chance of detecting a single outlier that has no similar samples in the dataset.
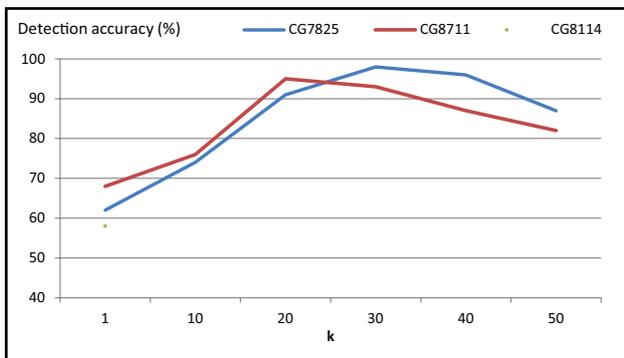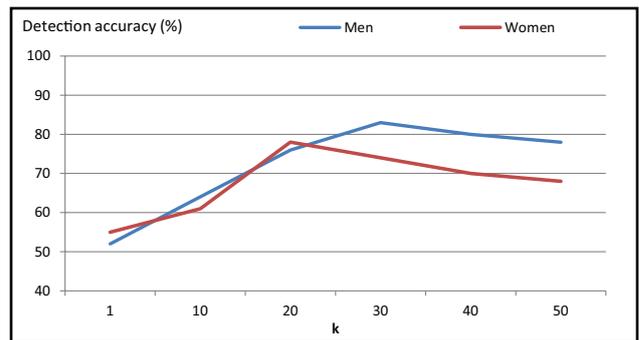

Figure 8: Detection accuracy of cells treated by knockdown of different genes among untreated cells. The detection accuracy increases with k as it leads to the rejection of single outliers that are not related to the different gene, but it starts to decrease when the method is not able to detect more than k self-similar outliers.

the algorithm to consider a varying number of neighbour outliers and still detect the images as outliers compared to the base class. Since the user does not know the value of k before the experiment (and cannot even assume the existence of a single outlier in the database), an estimation needs to be made based on the criterion of the user for what she considers an outlier. If the user considers every image that is different than the rest of the images as an outlier, the value of k should be set to 1. However, a single image that is different from the other images is often not considered an outlier unless images similar to it also appear in the database. To search for repetitive outliers that appear several times in the database the user needs to specify k values higher than 1, based on the number of repetitions the user considers an outlier that justifies attention and manual observation. If the number of repetitive outliers in the database is lower than k, these outliers will not be reported to the user, as they do not meet the criteria that the user specified. **Figure 7** displays the results of measuring the detection accuracy with different values of k.

To test the system with a larger sample, we used the RNAi dataset [35], fluorescence microscopy images of
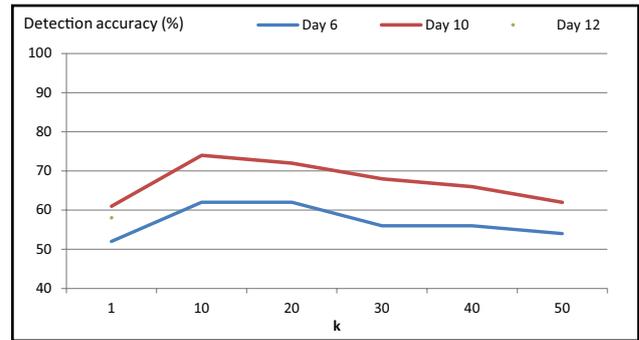
Figure 9: Detection accuracy of C. elegans terminal bulb microscopy images at an older age detected in a set of images taken at a younger age.

Figure 10: Detection accuracy of men knee x-rays among a dataset of women knee x-rays, and women knee x-rays among a dataset of men knee x-rays.

drosophila where the different classes were created by knockdown of different genes [35]. Part of the dataset is also available for free download [25]. Three genes were used in the experiment: CG7825, CG8114, CG8711, each produces a different phenotype [35], as well as images of untreated cells. In each experiment, 1500 60x60 images of untreated cells were used for the "typical" class, and 30 cells of each of the treated cells were used as the outlier class in each run, simulating an experiment in which 2% of the cells are outlier phenotypes. The parameter q was set to 20, which is clearly practical for manual inspection of the outlier images. Each run was repeated 100 times such that the outlier cell images are selected randomly. **Figure 8** shows the outlier detection rate when using different values of k.

As the figure shows, when the actual outlier rate is as low as 2%, in most cases CHLOE can correctly detect the outliers in a set of 20 suggested outliers, showing that the relatively quick task of applying CHLOE to large datasets of microscopy images can potentially lead to detection of anomalies. Clearly, applying CHLOE is far less labour intensive compared to manual inspection of a dataset of thousands of cell images, a task that is normally not practical.

To demonstrate that CHLOE can also process microscopy images of subjects that are not cells, the C. elegans terminal bulb dataset was also tested. The q parameter was set to 5, and the detection accuracy is displayed in **Figure 9**, showing that CHLOE can informatively detect outliers also in a dataset of tissue images. As the figure shows, the

detection of a terminal bulb of a 6-day old worm in a set of terminal bulbs of newborn worms is less accurate than the detection of the terminal bulb of older worms, which is expected due to the gradual morphological change in c. elegans terminal bulb tissues as the animals age [36].

Another test with a larger dataset was with the knee x-rays taken from the OAI dataset. The dataset is not of microscopy images, and can therefore also demonstrate the breadth of the outlier image detection method. As was shown in previous experiments, computers are able to differentiate between the knee of men and women by analysing the x-ray of the knee [37]. Twenty x-rays of men knees and 30 x-rays of women knees were used as the outlier groups to the datasets of women and men knee x-rays, respectively. As before, for each K the experiment was run 100 times, q was set to 10, and the accuracy is the average detection rate of these runs. **Figure 10** shows the detection accuracy.

(2) Availability
Operating system
Windows XP, Windows 7, Windows, 8.1.

Programming language
C++

Additional system requirements
Minimum hardware requirements for WNDCHRM and CHLOE are a PC with 512MB of RAM and a 1Ghz Intel Pentium 4 processor. However, due to the computational complexity of WNDCHRM, a faster machine with multiple cores will significantly shorten the response time of the system. For example, features can be extracted from one 256×256 image in ~100 seconds using a system with a 2.6 GHZ AMD Opteron and 2 GB of RAM [23].

WNDCHRM can run on Linux or Windows, and CHLOE has binaries for Windows, but the source code is open and can be compiled for other platforms by advanced computer users.

Both WNDCHRM and CHLOE are written in C++. CHLOE was tested using a Sony laptop computer with 4 gigabytes of RAM, and a 1.30-gigahertz processor, running Windows 7 Professional with Service Pack 1. It was also tested on an IBM Lenovo T2400 laptop with 3 gigabytes of RAM and a 1.83 GHz processor, running Windows XP Professional Version 2002 with Service Pack 3, and HP Z-Book with 16 gigabytes of RAM and Intel core-i7 4800 processor running Windows 8.1.

Dependencies
WNDCHRM (http://vfacstaff.ltu.edu/lshamir/downloads/ImageClassifier)

Installation
CHLOE is a single executable file "chloe.exe", so no installation procedure is required. Users should download the file to their hard drive, start the Command Prompt utility, and change the working folder to the folder where chloe.exe is located. The execute "chloe.exe" according to the instructions provided in the "implementation and architecture" section of this paper.

CHLOE requires WNDCHRM to compute the numerical image content descriptors before CHLOE is applied. WNDCHRM should be downloaded with the Dynamic Link Libraries (DLLs) specified in the WNDCHRM download page (http://vfacstaff.ltu.edu/lshamir/downloads/ImageClassifier). The DLLs should be placed in the same folder as the "wndchrm.exe" file.

List of contributors
· Saundra Manning
· Lior Shamir

Archive
Name
Figshare

Persistent identifier
http://dx.doi.org/10.6084/m9.figshare.994254

License
LGPL

Publisher
Lior Shamir

Date published
10/04/14

Language
English

Support
Support is provided on a best effort basis by contacting the authors on lshamir@mtu.edu.

(3) Reuse potential
The software allows for a library of images to be analysed automatically, with no previous intervention or knowledge by the experimentalist, and the outlier images are automatically detected so that further analysis can be performed by the experimentalist. This unsupervised method can be particularly useful where the number of images in the dataset is too large to be analysed manually. The software can be used by experimentalists to perform outlier detection in any microscopy image library.

The paper provided example applications to brightfield and fluorescence microscopy, as well as radiology images. Application of the method can be to a broad range of high-content screening experiment, in which rare but repetitive phenotypes are of high interest, but are difficult to detect due to the high amount of data. The method can also be applied to radiology images in population studies or large radiology databases to identify physiological anomalies that are visible through radiographs. Another possible application is pathogen detection in food quality control, where the detection of new unknown pathogens is critical to the prevention of potential outbreaks.

Due to its versatility, CHLOE can also be applied to other types of microscopy such as electron microscopy and FTIR (Fourier Transform Infrared Spectroscopy), where it can identify the outlier cells from which FTIR spectrum is measured. In the medical domain it can be applied to histopathology, where CHLOE can be used to detect anomalies in large sets of samples, improving the diagnostics power.

References

1. **Tanaka, M, Bateman, R, Rauh, D, Vaisberg, E, Ramachandani, S, Zhang, C, Hansen, K C, Burlingame, A L, Trautman, J K, Shokat, K M** and **Adams, C L** 2005 An Unbiased Cell Morphology-based Screen for New, Biologically Active Small Molecules. *PLoS Biology,* 3(5): e128. DOI: http://dx.doi.org/10.1371/journal.pbio.0030128

2. **Peng, H** 2008 Bioimage Informatics: A New Area of Engineering Biology. *Bioinformatics,* 24: 1827-1836. DOI: http://dx.doi.org/10.1093/bioinformatics/btn346

3. **Boland, M V** and **Murphy, R F** 2001 A Neural Network Classifier Capable of Recognizing the Patterns of All Major Subcellular Structures in Fluorescence Microscope Images of HeLa Cells. *Bioinformatics,* 17: 1213-1223. DOI: http://dx.doi.org/10.1002/cyto.a.20280

4. **Lamprecht, M R, Sabatini, D M** and **Carpenter, A E** 2007 CellProfiler: Free, Versatile Software for Automated Biological Image Analysis. *BioTechniques,* 42: 71-75. DOI: http://dx.doi.org/10.2144/000112257

5. **Loo, L H, Wu, L F** and **Altschuler, S J** 2007 Image-based Multivariate Profiling of Drug Responses from Single Cells. *Nature Methods*, 4:445-453. DOI: http://dx.doi.org/10.1038/nmeth1032

6. **Ljosa, V** and **Carpenter, A E** 2009 Introduction to the Quantitative Analysis of Two-dimensional Fluorescence Microscopy Images for Cell-based Screening. *PLoS Computational Biology*, 5:12. DOI: http://dx.doi.org/10.1371/journal.pcbi.1000603

7. **Shamir, L, Delaney, J D, Orlov, N, Eckley, D M** and **Goldberg, I G** 2010 Pattern Recognition Software and Techniques for Biological Image Analysis. *PLoS Computational Biology*, 6:11. DOI: http://dx.doi.org/10.1371/journal.pcbi.1000974

8. **Barnett, V** and **Lewis, T** 1984 *Outliers in Statistical Data*. John Wiley & Sons, Chi-chester – New York – Brisbane – Toronto – Singapore.

9. **Knorr, E M** and **Ng, R T** 1997 A Unified Notion of Outliers: Properties and Computation In: *Third International Conference on Knowledge Discovery and Data Mining Proceedings*, 219-222.

10. **Knorr, E M** and **Ng, R T** 1998 Algorithms for Mining Distance-based Outliers in Large Datasets In: *Proceedings of the 24th International Conference on Very Large Data Bases*, 24-27.

11. **Knorr, E M** and **Ng, R T** 1999 Finding Intentional Knowledge of Distance-based Outliers In: P*roceedings of the 25th International Conference on Very Large Data Bases,* 211-222.

12. **Ramaswamy, S, Rastogi, R** and **Shim, K** 2000 Efficient Algorithms for Mining Outliers from Large Datasets In: *Proceedings of ACM SIGMOD International Conference on Management of Data.* DOI: http://dx.doi.org/10.1145/335191.335437

13. **Breunig, M M, Kriegel, H P, Ng, R T** and **Sander, J** 2000 LOF: Identifying Density-based Local Outliers In: *Proceedings ACM SIGMOD International Conference on Management of Data*, 93-104. DOI: http://dx.doi.org/10.1145/10.1145/342009.335388

14. **Papadimitriou, S, Kitagawa, H, Gibbons, P B** and **Faloutsos, C** 2003 LOCI: Fast Outlier Detection Using the Local Correlation Integral In: *19th International Conference on Data Engineering*, 315-326. DOI: http://dx.doi.org/10.1109/ICDE.2003.1260802

15. **Jain, A k, Murty, M N** and **Flynn, P J** 1999 Data Clustering: A Review. *ACM Comp. Surveys*, 31(3):264-323. DOI: http://dx.doi.org/10.1145/331499.331504

16. **Jiang, M F, Tseng, S S** and **Su, C M** 2001 Two-phase Clustering Process for Outliers Detection. *Pattern Recognition Letters*, 22(6):691-700. DOI: http://dx.doi.org/10.1016/S0167-8655(00)00131-8

17. **He, Z, Xu, X** and **Deng, S** 2003 Discovering Cluster based Local Outliers. *Pattern Recognition Letters,* 24(9):1641-1650. DOI: http://dx.doi.org/10.1016/S0167-8655(03)00003-5

18. **He, Z, Xu, X** and **Deng, S** 2002 Squeezer: An Efficient Algorithm for Clustering Categorical Data. *Journal of Computer Science and Technology*, 17(5):611-624. DOI: http://dx.doi.org/10.1007/BF02948829

19. **Arning, A, Agrawal, R** and **Raghavan, P** 1996 A Linear Method for Deviation Detection in Large Databases In: *Second International Conference on Knowledge Discovery and Data Mining Proceedings*, 164-169.

20. **Sarawagi, S, Agrawal, R** and **Megiddo, N** 1998 Discovery-driven Exploration of OLAP Data Cubes In: *Proc. of the Sixth Int'l Conference on Extending Database Technology (EDBT), Valencia, Spain.*

21. **Ren D** and **Zhang, R** 2006 A Probabilistic Model with Multi-modal Reinforcement for Outlier Detection in Image Database In: *Proceedings of the 7th International Workshop on multimedia Data Mining.*

22. **Perner, P** 2009 Flexible High-content Image Analysis System for the Automatic Image Analysis and Interpretation of Cell Images: Computerized Methods in System Biology. *Transactions on Mass-data Analysis of images and Signals*, 1(2): 113-131.

23. **Shamir, L, Orlov, N, Eckley, D M, Macura, T, Johnston, J** and **Goldberg, I G** 2008 Wndchrm: An Open Source Utility for Biological Image Analysis. *Source Code for Biology and Medicine*, 3:13. DOI: http://dx.doi.org/10.1186/1751-0473-3-13

24. **Orlov, N, Shamir, L, Macura, T, Johnston, J, Eckley, D M** and **Goldberg, I G** 2008 WND-CHARM: Multi Purpose Classification Using Compound Image Transforms. *Pattern Recognition Letters*, 29(11):1684-1693. DOI: http://dx.doi.org/10.1016/j.patrec.2008.04.013

25. **Shamir, L, Orlov, N, Eckley, D M, Macura, T** and **Goldberg, I** 2008 IICBU-2008: A Proposed Benchmark Suite for Biological Image Analysis. *Medical & Biological Engineering & Computing*, 46(9):943-947. DOI: http://dx.doi.org/10.1007/s11517-008-0380-5

26. **Shamir, L** 2013 Unsupervised Detection of Outlier Images Using Multi-order Image Transforms. *Theory and Applications of Mathematics & Computer Science*, 3(1): 13-31.

27. **Shamir, L** 2012 Automatic Detection of Peculiar Galaxies in Large Datasets of Galaxy Images. *Journal of Computational Science*, 3(3): 181-189. DOI: http://dx.doi.org/10.1016/j.jocs.2012.03.004

28. **Shamir, L** 2008 Evaluation of Face Datasets as Tools for Assessing the Performance of Face Recognition Methods. *International Journal of Computer Vision*, 79(3): 225-230. DOI: http://dx.doi.org/10.1007/s11263-008-0143-7

29. **Shamir, L, Ling, S M, Scott, W, Boss, A, Orlov, N, Macura, T, Eckley, D M, Ferrucci, L** and **Goldberg, I G** 2009 Knee X-ray Image Analysis Method for Automated Detection of Osteoarthritis. *IEEE Transactions on Biomedical Engineering*, 56(2): 407-415. DOI: http://dx.doi.org/10.1109/TBME.2008.2006025

30. **Shamir, L, Macura, T, Orlov, N, Eckley, D M, Goldberg, I G** 2010 Impressionism, Expressionism, Surrealism: Automated Recognition of Painters and Schools of Art. *ACM Transactions on Applied Perception*, 7(2):8. DOI: http://dx.doi.org/10.1145/1670671.1670672

31. **Martin, K, Ibáñez, L, Avila, L, Barré, S** and **Kaspersen, J** 2005 Integrating Segmentation Methods from the Insight Toolkit into a Visualization Application. *Medical Image Analysis*, 9(6):579-593. DOI: http://dx.doi.org/10.1016/j.media.2005.04.009

32. **Boland, M V** and **Murphy, R F** 2001 A Neural Network Classifier Capable of Recognizing the Patterns of All Major Subcellular Structures in Fluorescence Microscope Images of HeLa Cells. *Bioinformatics*, 17: 1213-1223. DOI: http://dx.doi.org/10.1093/bioinformatics/17.12.1213

33. **Duller, A W G, Duller, G A T, France, I** and **Lamb, H F** 1999 A Pollen Image Database for Evaluation of Automated Identification Systems. *Quaternary Newsletter*, 89: 4-9.

34. **Shamir, L** 2011 Assessing the Efficacy of Low-level Image Content Descriptors for Computer-based Fluorescence Microscopy Image Analysis. *Journal of Microscopy*, 243: 284-292. DOI: http://dx.doi.org/10.1111/j.1365-2818.2011.03502.x

35. **Shamir, L, Eckley, D M, Delaney, J, Orlov, N** and **Goldberg, I G** 2009 An Image Informatics Method for Automated Quantitative Analysis of Phenotype Visual Similarities. *IEEE Life Science Systems and Applications Workshop*, 96-100. DOI: http://dx.doi.org/10.1109/LISSA.2009.4906718

36. **Shamir, L, Wolkow, C** and **Goldberg, I** 2009 Quantitative Measurement of Aging Using Image Texture Entropy. *Bioinformatics*, 25:3060-3063. DOI: http://dx.doi.org/10.1093/bioinformatics/btp571

37. **Shamir, L** 2011 A Computer Analysis Method for Correlating Knee X-rays with Continuous Indicators. *International Journal of Computer Assisted Radiology and Surgery*, 6(5):699-704. DOI: http://dx.doi.org/10.1007/s11548-011-0550-z