Biomedical Cell Detection and Tracking in Time-Lapse Microscopy

Azhar Abdykadyrova z5212235, Gary Wang z5318369
Wenqing Yi z5306579, Ye Lin z3326191, Yuechuan Li z5289723,

Department of Computer Science and Engineering, University of New South Wales

Abstract

Analysing time-lapse biomedical cell images is an effective way to study the characteristic of cells. In this study, we explored various combination methodologies from three streams, traditional pixel implementation, machine learning algorithms (ML) and deep neural networks (DNNs), and eventually selected one each from tradition and DNN stream, and evaluated their performance on segmentation. Self-designed algorithms are utilized & evaluated in cell matching, motion analysis and mitosis detection. U-Net model from DNN stream is more sensitive (97.8% vs 83.8%) but less precise (95.8% vs 100%) than traditional method in cell segmentation. Also the design ideas and challenges for cell matching and motion analysis are explored and analysed. The accuracy for mitosis detection is around 50% which is expected to improve to 90+% with optimal parameter tunning.

1. Introduction

Understanding the anatomic and dynamic behaviors and properties of live cells is of great importance in living organism's development, regeneration and immune response [20]. With conscious understanding of cellular behaviour, particularly abnormalities in cell migration, division and differentiation, many diseases may be explained and taken relevant measures to prevent further development [19]. To make this come true, it needs a great amount of the time-lapse image data to conduct analysis to the cells. However, with increasing data as well as capacity of technologies it became challenging, time-consuming and error-prone for manual analysis of cell tracking [4]. In contrast, automated cell tracking improves the efficiency and accuracy of the studies processing big sequences of data in a short time and does not require sophisticated tools [4].

In this paper, we implemented automated image segmentation, cell matching, cell tracking, cell motion analysis and mitosis detection based on microscopy dataset. We firstly researched and analysed various possible techniques. For image segmentation, we did research on three streams of approaches, naive image processing (i.e. various combinations of image processing techniques), traditional machine learning (ML) algorithms, such as random forest and Deep Neural Networks (DNN), such as U-Net, J-Net and R-CNN. For cell matching, we browsed traditional machine learning techniques and path-searching algorithms, such as Euclidean distance-based clustering, greedy searching, bayesian based algorithm and self-designed matching techniques. For the rest of the work, we analysed the features of all cells and made several self-designed algorithms. After this, we have chosen naive image processing and U-Net to segment the image, for their easy implementation and superior performance, and customized criteria and algorithms to conduct matching, tracking, motion information extraction and mitosis detection.

Our original dataset and its ground truth (GT) is provided with project specification, which consists of four separate raw image sequences from time-lapse microscopy recording. Since the provided GT data is not giving a complete distribution of cells, we went to the data source, CellTrackingChallenge, to find additional raw data and its completed truth mask. In this way, we are able to make comparable evaluations and firmly reflect the performance of each technique. A 2D dataset, Fluo-N2DL-HeLa [14] is found to be the original sources of our provided dataset. Its training sets contain the same images as in Sequence 1 and 2, and it has a complete version of mask for the same sequences.

Furthermore, we pointed out some characteristics of raw input data that might challenge cell detection and tracking, such as low contrast between foreground objects and background, varying number of cells due to division, and due to long time steps (30 mins), some cells vanish from our vision, most of which could be handled by our algorithms. But there are also original defects of raw data, such as poor quality of image causing partial loss of vision and changing light conditions raising noise. These pre-existing problems limit the performance.

2. Literature Review

2.1. Naive Image Processing

2.1.1 Thresholding

One of the most popular and simple approaches is thresholding. With a given threshold value each pixel of the image is labeled either as 'foreground' or 'background', resulting in a binary image. It might work well with the image where cells are well contrasted and there is a strong difference between foreground and background in terms of intensity [4].

The Simple threshold approach, where the parameter is set to a global value, has shown better results for cell segmentation over automatic approaches like Poisson distribution and Otsu's method. The reason is that there is a constant background over the sequences, while for automatic approach, histograms cannot be fitted with Gaussian and Poisson distribution for background values closer to 0 [10].

2.1.2 Watershed Transform

Watershed transform regards the image as a topographical relief with its ridges and valleys, where the pixel value defines the altitude of the landscape [17]. The watershed approach starts to flood the basins from the markers, usually local minimal, until different basins are attributed to different markers. Watershed line separates different basins from each other.[2] This approach is widely-used for separating overlapping objects. However, if there are several local minima within the object, the algorithm might lead to over segmentation [22]. Some pre and/or post-processing methods such as noise removal and manually creating markers as starting points could be adopted to improve the result of watershed. [3]

2.2. Machine Learning Approaches

2.2.1 Random Forest Classifier

Random forest classifier consists of a large number of decision trees, where each tree in the forest will give its own prediction and all predictions come together to select the most agreeable prediction[7]. In image segmentation, random forest could be used to classify images according to its selected features, such as color, texture and amount of pixels.

2.2.2 K-means Algorithm

The K-means algorithm is another possible approach to segment images. It is setting up k initial cluster centres and transverse the surrounding pixels, assigning them into a cluster based on their distance, and after all recalculating the centre.[6] Repeating the steps until all clusters converged. Also, it may not only be based on Euclidean distance. The selection criteria could be extended into other features, such as color channel, intensity and boundary curvature.

2.3. Deep Neural Networks (DNN)

2.3.1 R-CNN & Fast R-CNN

R-CNN, Regions with Convolutional Neural Networks, generates various selected regions for each image, and leverage CNN to extract features to classify images [16]. A more concise version, Fast R-CNN established on top of R-CNN and reduced training complexity by introducing the Region of Interest (RoI) pooling layer idea [5]. The change it made on top of R-CNN allows it to train and react much faster than R-CNN, which makes it possible to be utilized in some scenarios which need prompt reaction.

2.3.2 U-Net & J-Net

The invention of U-Net was especially for biomedical image segmentation. The entire network forms a 'U' shape, in which down-sampling and up-sampling pass information context down and up to enhance features of different levels [15]. U-Net demonstrated its superior capability at segmenting objects in images, and therefore its varied networks were developed to inherit its strength, and goal-orientally enhanced, such as J-Net [1], a simplified version from U-Net, W-Net [23], a self-supervised model inspired by U-Net, M-Net [12], a network specialized in optic disc and cup segmentation, U-Net++ [24] and nnU-Net [8] and so on.

2.4. Cell Matching

The general concept for cell association is to correctly match each cell from one image to the corresponding cell in the previous images.

2.4.1 Euclidean Distance based Algorithm

One of the basic approaches is to match the current cell to the nearest cell in the previous image by taking the Euclidean distance into account. However, if there are many cells or the movement of cells from image to image is big it could lead to mismatch. Thus, in order to avoid such errors, the approach could be combined with selecting and comparing other object features such as intensity, area, volume, orientation and other features [4].

Most of the systems used for cell tracking are designed to label the images with fluorescence, and also used for the images with multiple-spot objects, which are only captured by PhC and microscopy with multidimensional fluorescence.

Moreover, X.Lou, M.Schiegg and Fred A.Hamprecht presented a combined learning strategy for cell tracking, which is a combination of structured max-margin learning and active structured learning. Max-margin structured

learning is to obtain a regularized and globally optimized model, it may minimize the overall loss across all events, however, the disadvantage of this method is its deterioration of the detection in cell divisions. Active structured perceptron is for training data retrieval, it uses structured perceptron for model update, and after the training, it may obtain reasonable runtime. After combining these two methods, the cell tracking leads to a pleasant low-error result. Whereas, there are limitations, first one is when detections are missing, it cannot recover, another is that it cannot handle overly segmented objects [11].

In the challenges done by V.Ulman et al, they observed that the Contour Evolution method [21] has a high level of detecting, but it needs overlapping cells to work properly between successive frames. And the internal texture of the cells doesn't affect the segmentation, it improves the learning capacity of the machine-learning algorithms contrarily. In summary, this method relies on the datasets with high segmentation performance, and with high temporal and spatial resolution.

2.4.2 Centroid Tracking Algorithm

Another approach presented is called Centroid Tracking Algorithm [18]. This method follows each cell in each frame, it calculates the distance between the centroids of the cell from the current frame to the following frames, to track the trajectory of this cell. This method seems to be simple and easy-achievable, but there are some limitations, some objects with fast moving speed are not capable of tracking, the difference of distances of each object may raise the errors, and the shape, texture are not considered in this method, so the accuracy may decrease.

2.4.3 Mitosis Detection

For detecting cell division A.Liu et al. compared several methods including tracking based methods, tracking free methods, hybrid methods and methods based on deep learning.

For tracking based methods a specialized mean-shift kernel, segmentation-driven and Kalman filter methods were discussed. However, identifying cell division through cell trajectories in a dense population over time is quite challenging. Thus, it is suggested to focus more on tracking independent approaches.

Among tracking free methods SVM, linear classification, eXclusive Independent Component Analysis(XICA), fast cascade classifier with a set of 3D Haar-like features and optical flow approaches were discussed. These tracking free methods were applied on static images and do not rely on movement of cells.

To overcome the limitation of tracking based and tracking free methods, hybrid based and deep learning based approaches were proposed. The hybrid method is defined in three main steps: 1. candidate sequence extraction 2. feature extraction 3. cell division detection with graphical model.

A. Liu et al. have applied hybrid based and deep learning based approaches on the C3H10 dataset to detect cell division. The analysis showed that deep learning based approach, particularly, Hierarchical Convolutional Neural Network (HCNN) performed the best in terms of precision, recall and F score. And the worst results were given by the Support Vector Machine (SVM) method [13].

Furthermore, for overlapping cells in large scale mitosis, Gaussian Mixture Model (GMM) would be an ideal approach. It needs to fit 2-component and 3-component GMM on each frame of the sequences, and more components in a GMM leads to lower errors. However, this approach requires identifying the first frame which contains three cells, with the computation of three features from GMM parameters, it may cause higher errors and hard to proceed the tracking part [9].

3. Methods

Although we have explored several possible approaches in each task, we are not utilizing all of them in this project for two reasons: 1. Our time frame is limited, 2. There are various approaches in each stream but only a few of them are representative.

In this project, we will only present the most superior collaboration of traditional pixel techniques, and U-Net in DNNs stream for cell segmentation. Since DNN models are less-interpretable than other methods, it is not necessary to demonstrate more in this project.

For other approaches, such as k-means, random forest, SVM and bayesian based models, they are valuable in solving some specific types of problems, but may not be rather feasible and effective for this project. For example, k-means and greedy path search may deliver inaccurate results in cell matching because permanent mismatch may occur when there are unreasonably missed cells.

3.1. Method Plan

We have divided the project into four sectors: cell segmentation, cell matching & tracking, motion analysis and mitosis detection. In cell segmentation, we will compare both traditional and U-Net approaches, with all input data be pre-processed properly, and compare their quality of output. For the rest of the tasks, we are applying self-design criteria based algorithms. The plan of work is shown in figure 1.

3.2. Cell Segmentation

In the traditional method, the work flow is shown in figure 2. The raw image input is in low contrast and 16 bits.

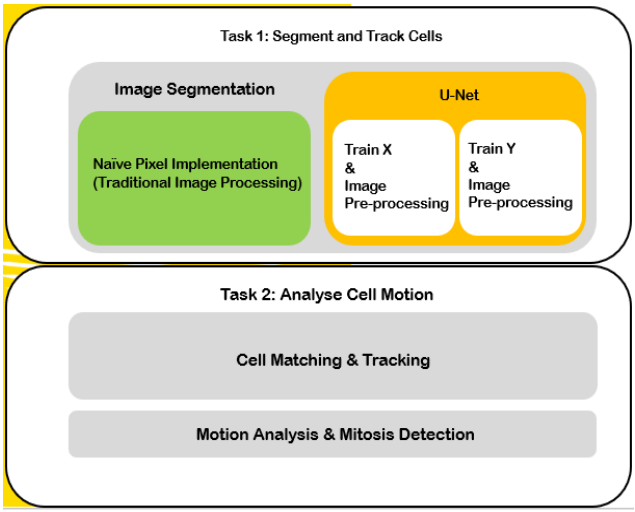

Figure 1. The work-flow plan in this project

Figure 2. Traditional methodology in segmentation

The image is converted from 16 bits to 8 bits after being applied contrast stretched. The reason that we do not do this conversely is that the pixels with low gray scale in 16 bits are likely lost in the conversion. Thresholding is the most significant part in this flow, where the threshold value is found through drawing a histogram of gray values of pixels. As the background is taking the largest number of pixels, we could observe on the histogram to see which pixel value is of the largest amount. We then could simply set the threshold value to be a few gray scales away from it. For example, if the gray scale value of the most number of pixels is 0, then we may set the threshold value to be 2 or 3, avoiding mistakenly taking noise from the background.

To clear the noise, Gaussian blur and erosion combined with dilation is utilized and it also helps make the pixels around cell borders smooth. The kernel size is not set too big, to avoid damage to the cell borders. Also we should prevent likely over-segmentation, so we applied erosion and dilation to the image, in order to reduce details within cells, and simultaneously make cell borders more clear. Iterations of erosion were carefully tested to reduce the edge touching. Three or four iterations produce a visually qualified mask. As a well-defined mask with clear foreground information is produced, watershed technique was used to find clear boundaries between cells.

The workflow of U-Net approach is shown in figure 3. We took images from sequence 03 and 04 as test set, and images from sequence 01 and 02 as training set due to its corresponding masks from CellTrackingChallenges. The masks are used as training labels. For better quality of prediction, both training set and training labels could go through extra processing, such as additional erosion to make clear borders. In this case, only training labels went through thresholding and erosion for we expect to visually compare the mask with training data.

U-Net is then well-trained by the training data and labels, and test X is fed in one by one to produce its predicted mask.

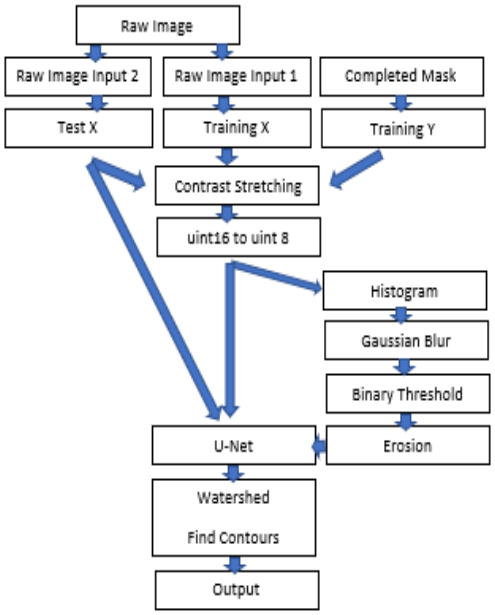

Figure 3. U-Net methodology in segmentation

3.3. Cell Matching & Tracking

Before matching & tracking, all cells in the first image will be assigned a unique ID and color of contours. Furthermore, all relevant information, such as centre coordinate, area and past displacement will be recorded.

Distance-based matching algorithms, such as k-means and greedy path search, were found to cause permanent mismatch of cells. Alternatively, we use the overlapping area

between a cell in the current image, to one of the cells in the previous image as a classifier. The cells with the largest overlapping areas are thought to be the same cells in the two images.

This algorithms are based on two assumptions:

1. None of the cells will move longer than the diameters of its circumscribed circle
2. The size of the cells will maintain approximately unchanged

From the result of the experiment, the overlapping checking is the most effective way to match cells. Other criteria such as distance-checking may be added to assist but there may not be significant improvement.

Occasionally, a few cells may unreasonably vanish or suddenly show up. We hardly believe that these cells were more tremendously vigorous than others. This phenomenon could be caused by the unstable quality of images, but we found no effective way to solve this problem. Hence, the easiest way is to dispose of the vanishing cells, and treat the appeared cells as new cells. Alternatively, new cells could also be matched from earlier images, but it will be more time costly.

Hence, all vanishing cells will be removed and all new cells be assigned an unique ID and color. Contours and its displacement are drawn in accordance to our record.

3.4. Motion Analysis & Mitosis Detection

After drawing some images with cells being labelled, we could observe features of dividing cells. We found three features of dividing cells, which may not come together:

1. become brighter one or two time steps earlier
2. start to grow larger
3. obvious change in shape is observed

We hence set three approximate criteria to allocate the changes:

a. brightness comparison,
b. size comparison,
c. perimeter comparison

For example, the current brightness of a cell is calculated from the sum of gray values of all pixels. If its current brightness is higher than 150% of average brightness of all cells, we then think the cell is dividing in a few time steps.

All dividing cells are drawn with thick red contours around them, and the contours will stay alive until it is thought to be wrong. After all, appropriate parameters are needed to deliver accurate and precise mitosis predictions.

4. Results and Evaluation

4.1. Results of Cell Detection

4.1.1 Image Segmentation

a. Traditional Method

The ground truth data provided is not a completed mask of all the cells on the image. In order to properly evaluate the results, we compare it with the complete Ground Truth data we found online pixel by pixel to obtain the True-Positive (i.e. the pixels we correctly predicted as cells), True-negative (i.e. the pixels we correctly predicted as background), False-positive (i.e. the background pixels we wrongly predicted as cells) and False-negative (i.e. the cells pixels we wrongly predicted as background).

We calculate the precision, sensitivity, Jaccard similarity coefficient (JSC) and Dice similarity coefficient (DSC) of each image and get the average for the whole sequence. The results are shown in Table 1. The formulas each of the metrics are:

$$Precision = \frac{|TP|}{|TP| + |FP|} \tag{1}$$

$$Sensitivity = \frac{|TP|}{|TP| + |FN|} \tag{2}$$

$$JSC = \frac{|TP|}{|FP| + |TP| + |FN|} \tag{3}$$

$$DSC = \frac{2|TP|}{|FP| + 2|TP| + |FN|} \tag{4}$$

Data	Precision (%)	Sensitivity (%)	JSC	DSC
Sequence 01	96.61	71.71	69.93	82.22
Sequence 02	96.01	76.12	73.77	84.89

Table 1. Evaluation of the traditional method

We can see that the precision of the traditional method is quite high but it is not sensitive enough.

b. U-Net

The U-net is trained using the data from sequence 01 and sequence 02. Figure 4 & 5 show the training result of the U-Net. Therefore, the same analysis conducted in evaluation of traditional methods is not suitable here. A more proper evaluation would be using the ground truth data of sequence 03 and 04 but we cannot manage to find that. Some manually checked results are shown in the later section to compare the performance of U-net and traditional methods.

c. Comparison

Due to the lack of a complete full ground truth data, we have

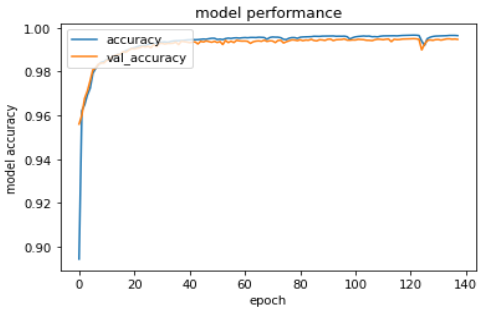

Figure 4. U-Net accuracy performance

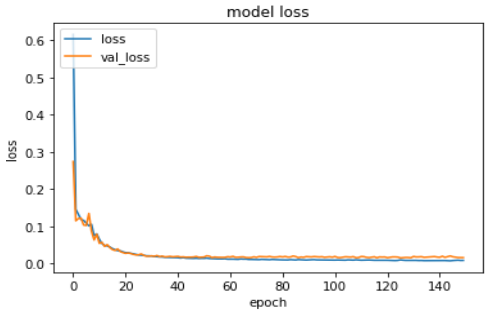

Figure 5. U-Net loss performance

to manually check the result image to compare the performance of the two methods we used. Table 2 and 3 are constructed for better visualization of the performance comparison and the image output of both the U-Net and traditional method are shown in figure 6(a)(b) & figure 7(a)(b).

Sequence 03 t019	U-Net	Traditional
Total detected cells	93	78
TP	91	78
FP	2	0
FN	2	15
Precision	95.8%	100%
Sensitivity	97.8%	83.8%

Table 2. Evaluation of the traditional method

Sequence 01 t035	U-Net	Traditional
Total detected cells	74	64
TP	74	64
FP	1	0
FN	0	9
Precision	98.6%	100%
Sensitivity	100%	87.7%

Table 3. Evaluation of the traditional method

Figure 6. Seq03 t019 (a) Traditional Method (b) U-Net

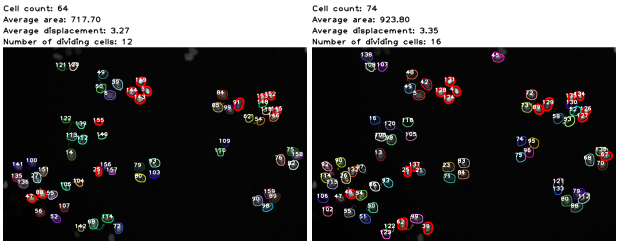

Figure 7. Seq01 t035 (a) Traditional Method (b) U-Net

4.1.2 Watershed Method

The u-net method and threshold method can only extract cells from the background but it cannot distinguish between different cells. We use the watershed method to do that since in the given data, there are lots of edge-touching cells. Generally the watershed method works pretty well but there still some over-segmentation and under-segmentation in some images.

Examples of over-segmentation and under-segmentation are displayed in figure 8. As shown in the image, cell 132 on the left should actually be segmented as three separate cells and cell 266 and 276 on the right should be merged as one cell. It is hard to find a general set of parameters that works for every image.

Figure 8. Example of under-segmentation and over-segmentation

4.2. Results of Cell Detection

The evaluation of the cell tracking is made by manually checking the cell trajectories and cell matching between two random consecutive time frames from a random sequence.

An example of evaluation for t038 and t039 from sequence 03 is displayed in figure 9 to show the performance of cell tracking. The double arrows indicate the mismatching between two time frames. 80 cells are detected in t038 and we can see that 76 cells are tracked successfully. The same cells in two time frames share the same unique ID. Therefore, if the cell matching is correct, the trajectories drawn on the image must be correct.

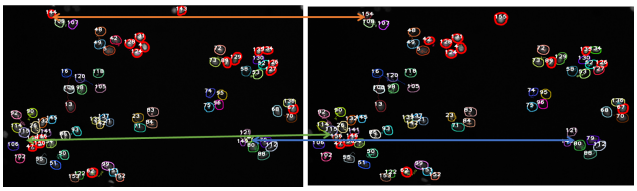

Figure 9. Seq03 t038 (a) and t039 (b) cell matching result

When using our method to track the moving cells between two frames, we made an assumption that the moving distance of cells between two time points will not exceed the diameters of its circumscribed circle. Based on the resultant image, the assumption made here is valid for most of the case. The cell movement is rather slow between two time frames. However, for some smaller cells, sometimes the cells move too fast, our algorithm might falsely recognize it as a new cell. Sometimes due to the quality of the image, the cell disappears in the segmentation process and therefore cannot find a match in the next time frames. Overall 95% of the cells are correctly tracked between two time frames.

4.3. Results of Mitosis Detection

It is hard to find out when exactly the cell will start the mitosis process since some cells may take longer to divide. In order to evaluate our method, we use 10 time frames as a standard. If the detected cell will divide within this period, we treat the detection as valid. An example is here from sequence one for better visualization. As shown in the figure 10, there are ten cells detected by our method. Table 4 shows whether the detection is valid and the time frame when the mitosis happens.

Only five of the detections are correct. The accuracy is about 50%. The precision of the method is quite low. However, the good thing is that the method does not miss any cells that will divide. Our method high sensitivity.

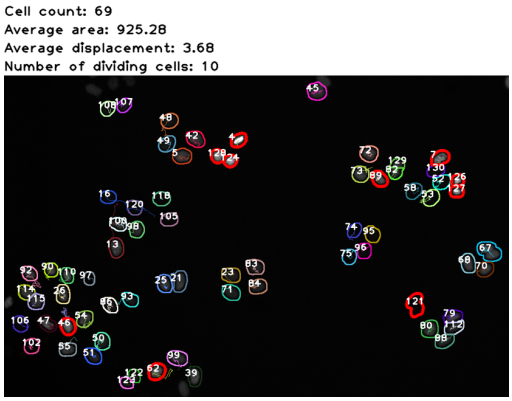

Figure 10. Seq01 t032 Mitosis Detection

Detected cell ID (%)	Validity	Dividing Time
4	Valid	t033
46	Valid	t038
62	Valid	t040
121	Valid	t033
7	Valid	t034
128	Not	N/A
124	Not	N/A
126	Not	N/A
127	Not	N/A
89	Not	N/A

Table 4. Result of mitosis detection of t032 from Seq 01

5. Discussion

5.1. Pros and Cons Analysis

From the comparable evaluation of traditional and U-Net methodologies, U-Net is showing both higher precision and sensitivity than traditional methods. However, this does not mean U-Net is naturally superior to traditional pixel implementation methods. For the traditional method, it heavily relies on manually parameter tuning. This means such a method is less-adaptive to various scenarios. For example, we spent a large amount of time tuning the size of the kernel in Gaussian blur and erosion & dilation. Each single step may only cause little difference and the significant improvement will only be achieved by having a well-collaborative parameter tuning in every step. Typically, Deep Neural Networks (DNNs) are much more adaptive to different datasets, but it needs enough training data to reach a good accuracy of its prediction. So that is to say, DNNs, U-Net in this case, is having larger potential to improve its capability both in precision and sensitivity metrics, but not having adequate training data may largely limit its performance. Therefore, traditional methods are more applicable to complete simple tasks, such as image sharpening and feature extractions, while U-Net could be used in various scenes due to its high adaptability. Furthermore, we also found the traditional method is relatively easy to interpret, i.e. we could understand its mechanism and adjust the parameters and techniques accordingly, while there are always challenges to interpret DNNs. Hence, concerns are raised to utilize DNNs in some industries with high security demand.

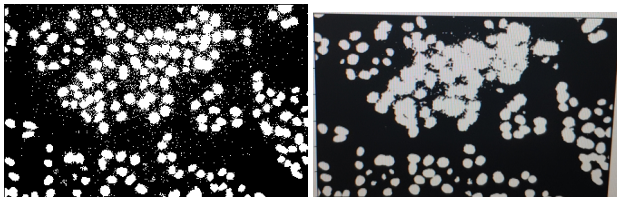

Figure 11. (a) Noise example 1 (b) Noise example 2

5.2. Challenges Analysis

First of all, the largest bottleneck is from the unstable quality of images. For example, we observed sudden light condition change in some images, and it makes our chosen global threshold value inappropriate for such images, as shown in figure .

Also, the unstable quality of images could be another reason to explain unreasonable vanished and appeared cells. Non-uniform intensity inside a cell could make it partially invisible, so it may be waived since the area of visible part is smaller than area threshold.

The capability of both methodologies to accurate foreground and background information does not raise our concern, due to the little difference in performance. We have seen there are challenges for both the methodologies. For traditional methods, various combinations of pixel implementation techniques were tested and we found no certain optimal solutions.

High time & space complexity is also a problem of our algorithms. In the training stage of U-Net, it costed a large amount of time for the model to be well-trained, due to the size of images (700 * 1100). When doing motion analysis, all types of information, such as centre coordinates, contours, and areas for all cells are remarkably time-costing. The more cells in the image, the more time it will take to compute. For example, images in sequence 02 have the largest number of cells and it usually takes more than 10 mins to finalize such an image, while other sequences only took 20 to 30 seconds for one image. Furthermore, we still demand cross-images comparison of cells to match and track them. This makes the algorithm even less efficient. We have not devoted enough time to optimize complexity of algorithms because we consider this project to be goal-oriented. If the project needs prompt response from our model, the architecture would then be re-designed.

5.3. Future Work & Improvement

First of all, the time & complexity of our algorithms shall be revised and redesigned, as we expect it to run both fast and accurately. Another point is to take more approaches, such as Euclidean distance and features based k-means, R-CNN and deepwater models, and make comparisons on performance among them. Eventually, the parameter tuning for mitosis detection will be optimised. We just could not conclude valid features of mitosis without printing out all images so parameter tuning is largely limited by time complexity of the algorithm.

6. Conclusion

Based on the experiment, we concluded that U-Net is more superior than traditional methodology in industry use, due to its potential, better adaptability and generalization. But we should also point out that the traditional method is more efficient and effective than DNNs in conducting simple tasks since it relies on expertise and does not need much resources to grow, i.e. training data. Given their strength and weakness, globally superior techniques do not exist, and all techniques should be placed in the right scenario and a mixture of those may be successful approaches in practice.

7. Contribution of Group Members

A. Azhar Abdykadyrova

Wrote introduction for the report & collaborated in literature review.

B. Gary Wang

Major contributor in designing, composing & testing project code. Collaborated in literature review and wrote Results and Evaluation in the report.

C. Wenqing Yi

Collaborated in literature review. Completed project demo for presentation.

D. Ye Lin

Major contributor in designing, composing & testing project code. Collaborated in literature review and wrote Methods and Discussion & Conclusion in the report.

E. Yuechuan Li

Wrote FindContours and Watershed function in project code.

References

[1] J-net: Multiresolution neural network for semantic segmentation. 2

[2] E. Bengtsson and J. Lindblad. Robust cell image segmentation methods. *Pattern Recognit Image Anal*, 14, 01 2004. 2

[3] W. Bieniecki. Oversegmentation avoidance in watershed-based algorithms for color images. pages 169 – 172, 03 2004. 2

[4] N. Emami, P. S. Pakchin, and R. Ferdousi. Computational predictive approaches for interaction and structure of aptamers. *Journal of theoretical biology*, page 110268, 2020. 1, 2

[5] R. Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. 2

[6] J. A. Hartigan and M. A. Wong. Algorithm as 136: A k-means clustering algorithm. *Journal of the royal statistical society. series c (applied statistics)*, 28(1):100–108, 1979. 2

[7] T. K. Ho. Random decision forests. In *Proceedings of 3rd International Conference on Document Analysis and Recognition*, volume 1, pages 278–282 vol.1, 1995. 2

[8] F. Isensee, J. Petersen, A. Klein, D. Zimmerer, P. F. Jaeger, S. Kohl, J. Wasserthal, G. Koehler, T. Norajitra, S. Wirkert, et al. nnu-net: Self-adapting framework for u-net-based medical image segmentation. *arXiv preprint arXiv:1809.10486*, 2018. 2

[9] Y. Li, F. Rose, F. Pietro, X. Morin, and A. Genovesio. Detection and tracking of overlapping cell nuclei for large scale mitosis analyses. *BMC Bioinformatics*, 17, 04 2016. 3

[10] A. Liu, Y. Lu, M. Chen, and Y. Su. Mitosis detection in phase contrast microscopy image sequences of stem cell populations: A critical review. *IEEE Transactions on Big Data*, 3:443–457, 2017. 2

[11] X. Lou, M. Schiegg, and F. Hamprecht. Active structured learning for cell tracking: Algorithm, framework, and usability. *Medical Imaging, IEEE Transactions on*, 33:849–860, 04 2014. 3

[12] R. Mehta and J. Sivaswamy. M-net: A convolutional neural network for deep brain structure segmentation. In *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*, pages 437–440. IEEE, 2017. 2

[13] E. Meijering, O. Dzyubachyk, I. Smal, and W. Cappellen. Tracking in cell and developmental biology. *Seminars in cell developmental biology*, 20:894–902, 09 2009. 3

[14] B. Neumann, T. Walter, J.-K. Hériché, J. Bulkescher, H. Erfle, C. Conrad, P. Rogers, I. Poser, M. Held, U. Liebel, et al. Phenotypic profiling of the human genome by time-lapse microscopy reveals cell division genes. *Nature*, 464(7289):721–727, 2010. 1

[15] P. F. O. Ronneberger and T. Brox. U-net: Convolutional networks for biomedical image segmentation. *Springer, Cham*, page 234 – 241, 2015. 2

[16] T. D. R. Girshick, J. Donahue and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *IEEE Conference on Computer Vision and Pattern Recognition*, page 580 – 587, 2014. 2

[17] J. B. Roerdink and A. Meijster. The watershed transform: Definitions, algorithms and parallelization strategies. *Fundamenta informaticae*, 41(1, 2):187–228, 2000. 2

[18] A. Rosebrock. Simple object tracking with opencv - pyimagesearch, 2021. 3

[19] A. Sacan, H. Ferhatosmanoglu, and H. Coskun. CellTrack: an open-source software for cell tracking and motility analysis. *Bioinformatics*, 24(14):1647–1649, 05 2008. 1

[20] C. Svensson, A. Medyukhina, I. Belyaev, N. Al-Zaben, and M. Figge. Untangling cell tracks: Quantifying cell migration by time lapse image data analysis: Untangling cell tracks. *Cytometry Part A*, 93, 10 2017. 1

[21] V. Ulman, M. Ma*vs*.ka, K. E. G. Magnusson, O. Ronneberger, C. Haubold, N. Harder, P. Matula, P. Matula, D. Svoboda, M. Radojević, I. Smal, K. Rohr, J. Jaldén, H. M. Blau, O. Dzyubachyk, B. P. F. Lelieveldt, P. Xiao, Y. Li, S.-Y. Cho, A. C. Dufour, J.-C. Olivo-Marin, C. C. Reyes-Aldasoro, J. A. Solís-Lemus, R. Bensch, T. Brox, J. Stegmaier, R. Mikut, S. Wolf, F. A. Hamprecht, T. Esteves, P. Quelhas, Ö. Demirel, L. Malmström, F. Jug, P. Tomançak, E. H. W. Meijering, A. Muñoz-Barrutia, M. Kozubek, and C. O. de Solórzano. An objective comparison of cell tracking algorithms. *Nature methods*, 14:1141 – 1152, 2017. 3

[22] C. Vachier and F. Meyer. The viscous watershed transform. *Journal of Mathematical Imaging and Vision*, 22(2):251–267, 2005. 2

[23] X. Xia and B. Kulis. W-net: A deep model for fully unsupervised image segmentation. *arXiv preprint arXiv:1711.08506*, 2017. 2

[24] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang. Unet++: A nested u-net architecture for medical image segmentation. In *Deep learning in medical image analysis and multimodal learning for clinical decision support*, pages 3–11. Springer, 2018. 2