Biological Cell Detection and Tracking Based on Image Recognition Technology

Yin Kejian, Zhou Yuhang, Tong Zhe, Liu Haihan, Zhang Cheng

I. Introduction

Through the imaging experiment of living cells, it can be analyzed to deepen the understanding of biology. The more data the experiment produces, the better it can analyze the behavior of cell population. Therefore, in order to obtain more data and better analysis results, experiments usually generate a large number of delayed image data. In order to process a large amount of data about cell images, computer scientists have invented a variety of cell recognition and tracking technologies [1].

For cell recognition, using threshold is the simplest method, but it is also the most error prone method, especially when the contrast is low, the cells overlap or the noise is high. Therefore, it is necessary to use other more complex but more accurate methods to identify cells. One method is template matching, which requires simple cell contour and regular shape. Another method is watershed transformation, which can completely separate cells, but may produce over segmentation [2].

For cell tracking, a simple method is to first determine the centroid position of the cell, and then find the cell closest to this position in the previous frame. This method has high accuracy when the cells move slowly and there are no other cells nearby, but it cannot deal with the behavior of apoptosis or disappearance. In the case of rapid cell movement or dense distribution, this method needs to be further expanded[3]. Another method is the probability method, which uses the trend of random knowledge, such as Bayesian reasoning, which is also an effective method.

The main process is as follows: first, model the cell in order to record the data in the project. Then preprocessing the data set. In this process, the contrast stretching algorithm was mainly used to remove noise. The next step is detecting cells. OTSU method and find contours method are the key to this step. Next, carry out cell tracking. The main method used in this step is dividing the cells into four different states, and then mark, compare and record them through the analysis of the pictures. Finally, analyzing and statistics, the main method is to calculate the number of cells and the average size of cells in each picture.

II. Literature Review

A. Cell segmentation in computer vision

In recent years, various advanced methods have been proposed to improve cell segmentation. There are many traditional methods of cell segmentation, such as threshold segmentation, region growth, seed watershed segmentation and edge base segmentation. However, in recent years, computer scientists have proposed some more advanced methods for cell segmentation. Zhou et al. Proposed a segmentation method based on OTSU algorithm[4]. This method has excellent performance on the image with separated cells, but it is not suitable for the image in which the cell contour contacts or overlaps with other cells. In their paper, Sintorn et al. Pointed out that using the gradient size of object pixels and background pixels in the image to apply watershed segmentation is an effective edge detection method[5]. In general, these methods are more vulnerable to noise and over segmentation. More complex methods usually use deep learning technology to improve the accuracy of cell segmentation. For example, brox et al. Proposed a u-net architecture, which is based on the annotation data set after data enhancement processing. The image can be reconstructed using an automatic encoder to obtain neuron structure segmentation[6]. The deepcell framework proposed by Valen et al. Can segment

biological images using optimized deep convolution neural network. This framework can be implemented on different types of biological cells[7]. However, the performance of the above methods depends on the type and shape of the cell dataset we will process.

B. Cell tracking in computer vision.

In recent years, computer scientists have proposed methods to automatically track and analyze cell activities. These methods can be divided into three categories[8]. The first is to establish a tracking model to detect the path of cells. Debeir et al. Proposed a method based on mean shift algorithm to establish cell tracking path through in vitro phase contrast in video[9]. Acton et al. Automatically track cells by combining active contour and filter in the experiment[10]. However, the above method has the disadvantage that their basic structure can not be directly used for mitosis, which means that further image processing is needed to improve the tracking results. The second is based on the results of cell segmentation. Yang et al. Proposed to use the classical watershed transform algorithm for cell segmentation, and then track its path according to the cell size[11]. The tracking accuracy of this method is very dependent on the accuracy of cell segmentation. The third category is the framework based on Bayesian probability. Fieguth et al. Proposed a cell tracking method based on probability model to locate cells[12]. In conclusion, we propose to recognize cell activity based on segmentation and tracking cells.

III. METHODS

Existing cell images have been sorted by time, but the cell-background differences in each image are extremely insignificant, namely the low contrast of the picture. At the same time, a noise may exist in the image due to the equipment, lighting site, dust and other reasons. We aimed to identify the real cells from these actual photographs and compute them based on the characteristics and motor characteristics of these cells, while performing annotation manipulation of mitotic cells.

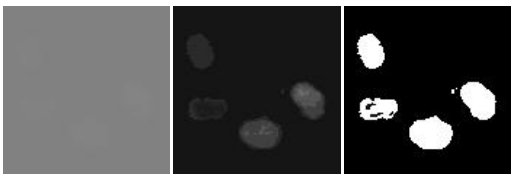
A. Cell Modelling

Although the cells are not clear enough in the picture, we must find ways to detect it accurately to facilitate the corresponding processing in subsequent calculations. Therefore, it is necessary to model the cells with their motor, mitotic characteristics.

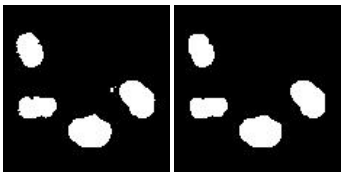
Analyzing the characteristics and objectives of the cell, we decided to record its ID information, cell center point, size, area, color, state, movement trajectory for each one.

Among them, the ID information is sorted by the order of cell discovery, from small to large. The cell size is a pair of scalar sizes, recording the values of cell length and width. For the area of the cells, computations were obtained by the contour Area method provided by the OpenCV library. Each cell does not actually have a color, but to facilitate presentation in the generated image, it is necessary to randomly assign one color to each cell. These colors will be maintained as the cell moves in each frame until it disappears, or mitosis occurs. A cell should be in one state: survival, disappearance, mitosis. When cells in survival, their trajectories can be expressed by a serial list of coordinates consisting of their central points.

B. Preprocessing Dataset



(a) (b) (c)



(d) (e)

Figure 1. Sample fragment during preprocessing

There are many challenges in identifying cells in samples. First, the image contrast was low and the cells

were almost fused with the background. The background is full of noise. The brightness depth of the cells is also inconsistent, and the boundaries of some cells are not clear enough, as shown in Figure 1 (a).

For the low contrast problem, we tried to use the contrast stretching algorithm. The results showed that the image after contrast stretching is ideal to clearly distinguish between cells and background. The processed results are shown in Figure 1 (b).

Next, we want to finalize the image. We first attempted the OTSU method, but found that the results were not ideal enough and that the OTSU threshold was too high, resulting in many cells disappearing after binarization treatment.

Looking at the picture, we can know that the area of the cells in the image is always smaller than the area of the background, that is, the largest majority of a specific gray color value in the background color. Accordingly, we can binary the image according to the peak point in the image histogram as the threshold to isolate the cells from the background. The results are shown in Figure 1 (c).

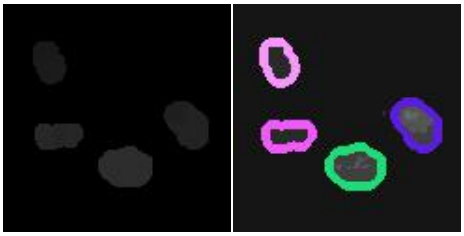
Continue looking at the image at this time, you can find that some cells have holes inside due to the uneven color distribution. To fill these holes, we decided to fill the background using the flood flooding method so that the inner cell could be partially connected together. The results are shown in Figure 1 (d).

After flood treatment, the cells are filled, but some cells have more prominent concave and convex parts. This is not ideal enough to require the optimization of the shape of these cells. To this end, we do the corrosion and expansion of the image, which can remove the noise in the image, and make the cell more round. The results are shown in Figure 1 (e).

C. Cell Detection

At this point, in the picture, the cells are completely separated from the background, so the next thing to consider is the labeled cells in the map and put into the established cell model.

We decided to use the watershed algorithm to split the individual cells in the image and perform annotation processing.



(a) (b)

Figure 2. Example fragments during cell detection

Afterwards, we used the findContours method provided by OpenCV to find the profile of the cells. The results are shown in Figure 2 (a).

At this point, the recognition and processing section of the image is over, and the next work is to incorporate the cells identified in the image into the cell library.

We assigned a new id and color to each emerging cell and saved it to the cell library. In this way, if the new image identifies the new cells, it can compare with the existing cells in the cell library, and then make the next judgment.

D. Cell Tracking

Before starting this part of the task, we considered tracking using established methods. But for the statistical and custom considerations, we finally decided to develop the algorithm to track the cells in the image.

For the cells that were already distinguished, we observed that some cells suddenly appeared, or suddenly disappeared, in the next image. The shape and location of each cell will also constantly change over time. Meanwhile, some cells also undergo mitosis.

In view of the above situation, we divided the cells in each image into several states, including survival, disappearance, and mitosis. Labeling the cells in the cell library according to the actual situation helps to make the comparison and update when the next image appears.

In fact, it can be said that each cell showed more or less changes in the next image. The first thing we have to do is to distinguish which cell in the new image is moved from the original cell.

Considering the many properties of a cell, including location, size, area, shape, etc. We consider determining cell identity based on the location and area of each cell.

For the moving distance between two cells, we used the Euclidean distance to calculate:

$$d = \sqrt[2]{(y_2 - y_1)^2 + (x_2 - x_1)^2}$$

Where, d is the European distance, x_i for the cell center point in X coordinates, y_i for the cell center point in Y coordinates.

Meanwhile, we designed the similarity algorithm. The proposed algorithm has the formula of:

$$S = ((1 - \eta) \times max(0, 1 - (\frac{d}{d_{max}})^{1.2}) + \eta \times (\frac{a^1 + bias}{a^2 + bias})) \times 100\%$$

Where, d is the Euclidean distance of the two cells, d_{max} is the maximum range of a cell movement between two images, actually it was set to 50. a^1And a^2 for the cell's area, we always assume that the area of a^1 is less than a^2. The bias was designed to prevent the cell too small to make the ratio too large, and actually it was set with 25. η is the proportion of the area value and the distance value in the statistical similarity.

The final values calculated by the similarity algorithm are all between 0 and 100. The larger the value, the higher the similarity of the two cells.

With this algorithm, we can traverse the surviving cells in the cell library and compare them one-to-one with the existing cells. When the similarity reaches a certain threshold, the information and similarity of the two cells are added to a list. After the full cell traversal is complete, this list is sorted by the size of the similarity, and the cells in the cell library are established corresponding to the cells in the image based on this list.

E. Cell Changes

As described above, the cell changes can be divided into the following four categories: addition, disappear, reappear, and mitosis.

Whether a cell is new or not can be determined by whether the cell is in the cell library. That is, all the cells in the cell library could not match a cell in the image, and the cell is considered a new cell at this time.

For new cells, in addition to the sudden cells in the image, two other possibilities may correspond to the emergence of new cells: recapitulation and mitosis.

Reappear refers to some cells in the cell library, due to problems in camera equipment, appeared in the pre-sequence image, lost in a certain image, and again from the next image. For this case, we compare the new cells with the images just labeled as disappearing in the cell library. If a cell is determined as reproducible case, the cell is again labeled as alive from the cell library, and the cells in the cell library are associated with the cell.

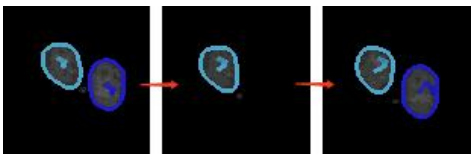


Figure 3. Example of reappear of a cell

Mitosis is a relatively complex scenario in which new cells appear in the image that can find its "sibling" cell in the current image, and their corresponding mother cell in the cell library. Depending on the mitotic properties, two daughter cells must be smaller than the blast area, which sum roughly the same as the blast area. Based on these characteristics, it is determined whether a newly emerging cell is the mother cell. Once confirmed, the mother cell were labeled as the dividing state in the cell library, while both cells were set to the new state.

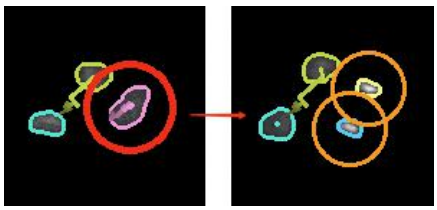


Figure 4. Example of a mitosis cell

Cell disappearance means that some of the cells cannot match any of the cells in the new image after the traversal of the cell library. This means that this cell has disappeared from the surveillance. In addition to the "reappear" situation, the cell will forever mark the extinction state and no longer participate in the computation of the cells in subsequent images.

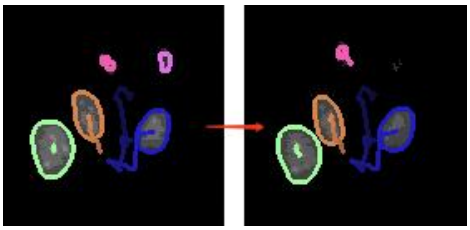


Figure 5. Example of disappear of a cell

F. analytic statistics

For each image, we were concerned about the number of cells contained, and the average cell size. The average cell size was calculated by the following formula:

$$area_{avg} = \frac{\sum_{i=1}^{n} area_i}{n}$$

Where, $area_{avg}$ for the final average area, $area_i$ for the area of each cell, n is the number of cells.

To avoid distortion situations, we ignored those cells at the edge of the image when calculating.

We similarly focused on the average displacement of the cells. The value can be calculated by the following formula:

$$displacement_{avg} = \frac{\sum_{i=1}^{n} |c_i^1 - c_i^2|}{n}$$

Where, $displacement_{avg}$ for the mean cell displacement, c_i^1 and c_i^2 for the center point location of the cells in the current and next image, and n is the number of cells.

We also focused on the number of mitosis and the number of newly generated cells. When analyzing whether cells undergo mitosis, we can record the number of times of this division through variables, while adding to the statistics at new cell generation.

Other information we focus on, such as the number of cells per image, the number of separations, the number of generated cells, will also statistically show in each image.

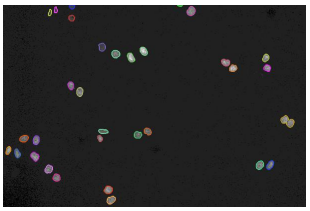
IV. RESULTS

This article uses a combination of qualitative analysis and quantitative analysis to evaluate the results of the project.

A. Results of Cell Detection

For the identification of cells, we can see from Figure 6. According to the algorithm we give, the contour of each cell is covered by lines of different colors, and the color of the cell remains unchanged over time. Obviously, our method is stable and reliable for cell recognition.

(a)



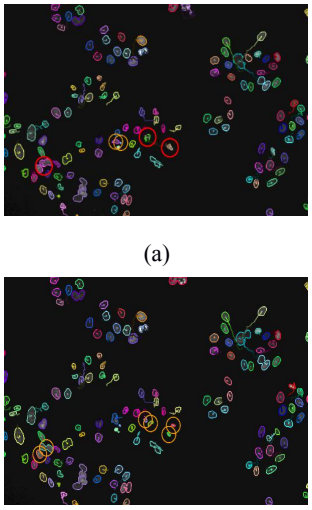
(b)

Figure 6. Original image, Recognized image.

However, due to the limitations of the method, some overlapping cells will still be mistakenly identified as one cell. For example, in Figure 6(b), the two cells marked with a yellow outline on the far right are identified as one cell.

B. Results of Cell Tracking

For the movement trajectory of the cell, the movement trajectory of the cell is consistent with the contour color of the cell, and the result is shown in Figure 7. All the trajectories from the appearance of a cell to the current position of the cell. When a cell undergoes apoptosis and disappears from the image, the trajectory also disappears; when mitosis generates new cells, a new trajectory appears. It is obvious from Figure 2 that the trajectory of the cells is clear and reasonable.



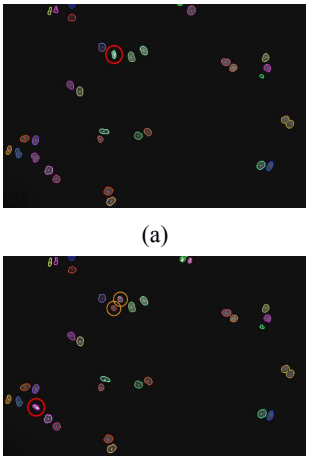
(a)

(b)

Figure 7. Cell movement trajectory image.

C. Results of Detection of Cell Division (Mitosis)

For the mitosis of the cell, as shown in Figure 8. We use red circles to mark the cells that will undergo mitosis. At the same position in the next frame, two new cells will be produced, which are marked with two yellow circles.



(a)

(b)

Figure 8. Mitosis image.

D. Results of Cell Motion Analysis

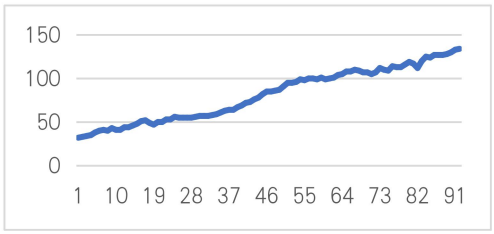
In the cell movement part, we recorded the following information: the cell count; the average size (in pixels) of all the cells; the average displacement (in pixels) of all the cells; the number of cells that are in the process of dividing. Taking the 01-folder data as an example, we get Table 1:

It can be found from the data in Figure 9(a) that the number of cells shows an increasing trend; from the data in Figure 9(b)(c), it can be found that the average size and average displacement of the cells are stable within a certain range. These results are in line with our expected reasonable conjecture, and also verify the reliability of our algorithm from the side.

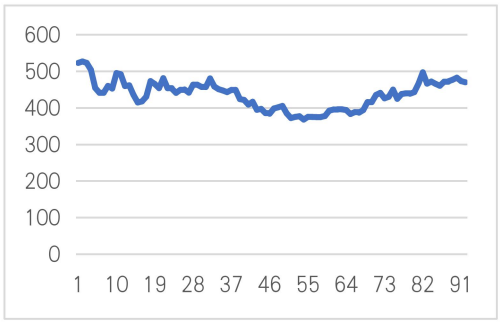
We also compared the number of cells according to the given ground truth data, and finally concluded that on the data provided by the number 01 folder, the accuracy of each picture is shown in Figure 9(c), and the average accuracy is 85.03%; on the data provided in the No. 02 folder, the accuracy of each picture is shown in Figure 9(d), and the average accuracy is 90.14%. Other data are not referenced, so no assessment can be given here. In general, our algorithm can achieve satisfactory results in identifying the number of cells.

Table 1. part of statistics for 01 folder

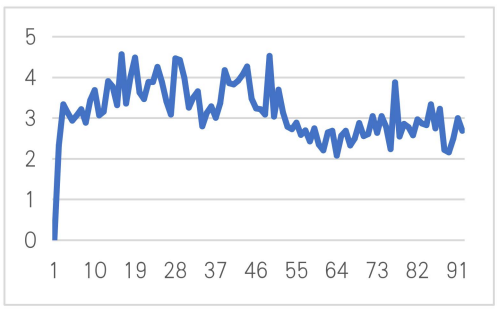
frame_name	cell_count	area_average	displacement_average	in_mitosis
t000.tif	32	523.17	0	0
t001.tif	33	527.07	2.33	1
t002.tif	34	523.19	3.34	1
t003.tif	35	504.2	3.13	4
t004.tif	38	454.75	2.94	0
t005.tif	40	441.43	3.07	1
t006.tif	41	441.85	3.22	1
t007.tif	40	460.29	2.89	1
t008.tif	43	453.13	3.44	0
t009.tif	41	495.47	3.69	0
t010.tif	41	492.24	3.07	3



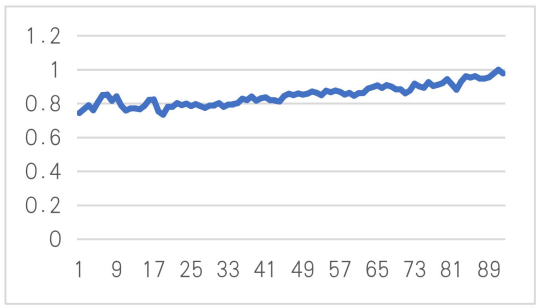
(a) The line graph of the number of cells.



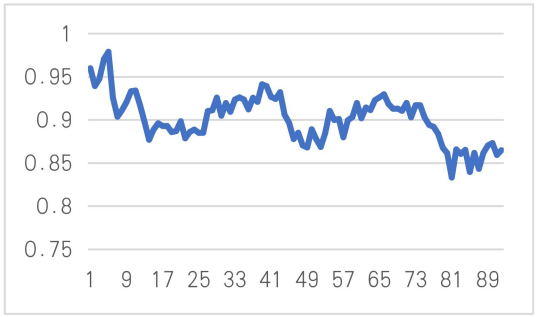
(b) The line graph of average cell area.



(c) The line graph of average cell displacement.



(d) The accuracy of 01 data set.



(e) The accuracy of 02 data set.

(cont) Figure 9 . Visualization of results.

V. Discussion

After using the above methods, we found three main problems through observation of the results. Firstly, we have some problems with the image preprocessing. For example, we compared the original image and the processed image. In the processed image, some of the cells on the original image have disappeared, and some of the cells have changed in

shape after Gaussian blur. We originally planned to reduce the Gaussian kernel value but we found that if the Gaussian kernel value is reduced, a black area will appear in the lower left corner of the image, and some messy cells will appear in the processed image. The optimal threshold obtained by the OTUS algorithm is not accurate. So we changed the image pretreatment method, looking for the best threshold.

Secondly, in some cases, cells cannot be correctly identified and counted. For instance, in the images, some cells have divided, and one cell will divide into two cells, but in some image s the edges of the two new cells that have split are connected together, then our method will recognize the two connected cells as one cell and draw the outline of the two cells in the same color. Our current method cannot determine whether this cell is in a divided state or in another state because the contours of these two cells are connected.

The last one problem is the area of some of the cells may not be calculated correctly, in our method, at the beginning, we treat each cell as a rectangle to calculate its area, then we use a formula to calculate the inscribed circle of this rectangle, and find that the value obtained in this way may be more realistic. But this is not very accurate. Finally, by searching the information online, we decided to use the contour Area function to calculate the area of the cells. We will ignore some cells that are too small because we think these may be noise。

VI. Conclusion

Because it is hard to track the cells in microscopy by hand. We present an algorithm for tracking the cells and 2-D segmentation in the microscopy images. To make sure the accuracy, in the pre-processing, Flood Fill and Eroding and Dilating have been used to get the clear outlines. Watershed also been used to definite threshold. In processing, firstly use the first image as target layer and then get the cell's bank through pretreatment. Then ergodic the gallery of cell images, by comparing the cells obtained in each image with the cell bank and constantly updating the cell bank, record the cell's 'state'. After that, comparing the elements to compute the only result, according to the cell's 'state', including the location and size of each cell, and finally get the cell segmentation and its trajectory.

Admittedly, there are also some problems need to be improved. Such as sometimes it's hard to judge is there are a cell segmentation or just two cells separate when they are too close. So, pre-processing or observed device should be improved to get a clearer processed image which can fix this problem easily. Finally, after series testing, this algorithm can totally satisfy the requirement of Task1 and Task2. It can mark the cell segmentation accurately, and it can also mark the tracking, location and size of each cell and the number of cells in images.

VII. Contribution of group members

- **Yin Kejian (z5281025):**

Read the reference document. The main coding of Task 2. Writing the method of report, clipping videos, and preparing for the demo.

- **Zhou Yuhang (z5292084):**

Read the reference document. The main coding of Task 1. Testing of the whole code. Writing the experimental results of report and preparing for the demo.

- **Liu Haihan (z5327216):**

Read the reference document. Taking part in the coding of Task 1 and 2. Testing of the whole code. Writing the introduction and the literature review and references and proofreading the format of the report and preparing for the demo.

- **Zhang Cheng(z5342963):**

Read the reference document. Taking part in the coding of Task 1 and 2. Testing of the whole code. Writing the discussion of report and the PowerPoint of the demo and preparing for the demo.

- **Tong Zhe (z5298319):**

Read the reference document. Taking part in the coding of Task 1 and 2. Improving the pre-process. Writing the conclusion of report, and preparing for the demo.

VIII. Reference

[1] E. Meijering, O. Dzyubachyk and I. Smal, “Methods for cell and particle tracking,” Methods in Enzymology, vol. 504, no. 9, pp. 183- 200, February 2012.

[2] N.N. Kachouie, P. Fieguth, J. Ramunas, E. Jervis, “Probabilistic modelbased cell tracking,” International Journal of Biomedical Imaging, pp. 1–10, 2006.

[3] E. Meijering, O. Dzyubachyk, I. Smal, W. A. van Cappellen, “Tracking in cell and developmental biology,” Seminars in Cell and Developmental Biology, vol. 20, no. 8, pp. 894-902, October 2009.

[4] X. W. Chen, X. B. Zhou and S. T. C. Wong, “Automated segmentation, classification, and tracking of cancer cell nuclei in time-lapse microscopy,” IEEE Transactions on Biomedical Engineering, vol. 53, no. 4, pp. 762-766, April 2006.

[5] C. Wahlby, I.M. Sintorn, F. Erlandsson, G. Borgefors & E. Bengtsson, ¨ “Combining intensity, edge and shape information for 2D and 3D segmentation of cell nuclei in tissue sections,” Journal Of Microscopy, vol. 215, pp. 67–76, 2004.

[6] O. Ronneberger, P. Fischer, T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” Med Image Comput Comput Assist Interv, pp. 234–241, 2015.

[7] DA. Van Valen, T. Kudo, K.M. Lane, D.N. Macklin, N.T. Quach, M.M. DeFelice, I. Maayan, Y. Tanouchi, E.A. Ashley, M.W. Covert, “Deep learning automates the quantitative analysis of individual cells in live-cell imaging experiments,” PLoS Comput Biol, vol. 12, no. 11, p.e1005177, 2016.

[8] M. A. A. Dewan, M. O. Ahmad, M. N. S. Swamy, “Tracking biological cells in time-lapse microscopy: An adaptive technique combining motion and topological features,” IEEE transactions on biomedical engineering, vol. 58, no. 6, pp.1637-1647, 2011.

[9] O. Debeir, P. Van Ham, R. Kiss, et al, “Tracking of migrating cells under phase-contrast video microscopy with combined mean-shift processes,” IEEE transactions on medical imaging, vol. 24, no. 6, pp. 697-711, 2005.

[10] N. Ray, S. T. Acton, K. Ley, “Tracking leukocytes in vivo with shape and size constrained active contours,” IEEE transactions on medical imaging, vol. 21, no. 10, pp. 1222-1235, 2002.

[11] J. Yan, X. Zhou, Q. Yang, et al, “An effective system for optical microscopy cell image segmentation, tracking and cell phase identification,” 2006 International Conference on Image Processing. IEEE, pp. 1917-1920, 2006.