Project: Tracking Biological Cells in Time-Lapse Microscopy

Team: DeepVision

Deepansh Deepansh z5199370 | Ujjwal Garg z5212244 | Leonard Lee z5173917 | Son Tong z3448000 | Luke Yong z5216244

Abstract

This paper attempts at generalizing detection and tracking of biological cells in time lapse microscopy. To visualize cells and their behavior, various imaging modalities are used. These modalities introduce anomalies to images such as shading artifacts and non-uniform illumination. Other complications inherent to the domain, e.g. cell residue and cytoplasm are also present. Cell motion is erratic and early detection of cell events are crucial in cases of time sensitive diseases. Manual analysis of cell events is error prone. This paper uses morphological operations along with ellipse fitting and U-net to segment images, considers centroids and distance metric of cell motion features to track and recognize mitosis events.

Keywords: - Segmentation, Detection, Tracking, Mitosis

Introduction

To gain biological insights from cell behavior, it is often necessary to identify individual cells and follow them over time. To visualize these processes, imaging techniques like phase contrast (PhC), differential interference contrast (DIC), fluorescence microscopy are employed. The image sequences contain a large number of cells over multiple frames which makes manual annotation and event detection tiresome and error prone.

To automate cell segmentation and tracking, computer vision along with machine learning methods are taken advantage of. This entails, image preprocessing, feature extraction, tracking and cell event (e.g. mitosis) recognition. Due to various imaging modalities, the resulting image sequences are prone to anomalies such as low signal to noise ratio, lingering cytoplasm, uneven illumination and shading artifacts.

The 3 datasets considered have significantly distinct features. The cells are all of different types, shapes and sizes. The cells have different boundary structure while the image contrast and intensity fluctuate dataset to dataset. The number of mitosis events also differ hugely in each dataset. Special care has been taken during preprocessing to accommodate as generic an approach as possible.

The 3 databases have some common characteristics as well. The phenomenon of residual cytoplasm or debris is present. Cells entering and leaving in the middle of image sequences is observed. Degree of overlap of cells in all dataset are also similar.

Massive steps have been taken in the direction of tackling these challenges like active-contour, mean-shift, level-set based methods but all of them have been applied to the specific type of images or require extra heuristics to be handled. In this paper, we develop a system to automate cell segmentation and tracking while considering the challenges mentioned above. Firstly, the image is preprocessed using morphological opening followed by Otsu thresholding. The resulting contours are fitted with ellipse and bounding boxes for all identified cells are drawn. For tracking, the nuclei of identified cells are computed and compared against previous frame. Euclidean distance metric is used to correspond 2 cell nuclei in consecutive frames. Mitosis events are recognized by finding the distance between cell nuclei. Speed of a user specified cell along with total distance travelled up to that point and total displacement are also calculated.

Literature Review

Cells were detected based on either intensity, texture, or gradient and then linked in two or more frames. The graph-based technique is used to define the relationship between the cells, but this technique is computationally expensive especially when cell density is high. Segmentation errors were also not addressed in post-processing. Hence to overcome all these difficulties another author introduces a method that incorporates different filters for segmentation such as top-hat, h-maxima transformation. Topological features, motion features were used to get accurate cell trajectories along with template matching based backward tracking to recover broken paths.

Thresholding is commonly used for segmentation due to its ease of use but is error prone. Due to noise in the image caused by various microscopy techniques, more complex methods such as template matching, watershed transform, model-based segment merging are required [9].

Particle detection: First identify loci by using intensity thresholding and then to estimate their positions by computing the centroid. However, this is not as good as using Gaussian fitting which involves searching for loci with similar intensity profiles of the particle. Better methods include Laplacian of Gaussian filtering, machine learning [9].

Linking particles: Due to various cell complications, 'nearest-neighbor' may not be optimal. Suggested methods are to use spatiotemporal segmentation approach to search for optimal paths or to use cost matrix from detected particles and their similarity (between 2+ frames) to get the optimal subgraph.
Most visible characteristics are roundness with a formula to identify the circularity of the cell together with the cell centers and radii, then, brightness of cells just before going through mitosis. Secondly, when a cell is about to go through mitosis will generally have a bright halo effect on their circumference [2].
Lastly, cells rarely move when going through mitosis. A level-set based variational method of backwards and forwards tracking is used to enhance the accuracy of detection [2].
The final step is to have topology preservation of the cell, so that odd shapes like elongated cells will not differ the accuracy of the detection.
There are many tracking methods utilized for different cases. Methods such as Boosting tracking require training at runtime for positive and negative examples of an object. It needs to learn how an object looks like. However, with cells, their shape is irregular and change over time, even perform mitosis which basically form 2 new objects to track. Thus, the method would prove not be suitable for the task. Centroid tracking focuses on tracking points of an objects and measure discrepancies between the two trajectories. With our task this seemed more suitable thus we decide to adapt it for our purpose with a simpler version due to time constraints. A comprehensive effort to incorporate top hat with h-maxima transformation for nuclei segmentation, adaptive thresholding to remove noise from images , watershed to separate cells with connected boundaries, morphological opening and closing erosion and dilation to make boundaries visible, min-max filtering to remove unwanted background, ellipse fitting for cell localization used to localize nuclei in a cell which has maximum intensity, rectangle fitting for cell localization fits a rectangular box around the cell , fast radial symmetry for selecting the regions of interest and gaussian blurring to remove the noise were also made.

Since the success of AlexNet [2] that created a paradigm shift towards the use of deep learning in computer vision, convolutional neural network (CNN), a specific structure of deep learning that involves convolutional layers, has quickly found its way into medical imaging analysis and become a methodology of choice [3]. It has not come as a surprising development, as segmentation is an extension to the classification framework using CNNs and is also a common task found in the analysis of both natural and medical images.

A feature commonly found in a CNN infrastructure is (max) pooling, which acts to reduce the spatial size when representing an input image, in order to reduce the number of parameters, computation burdens, and overfitting [12]. However, these benefits do not come without cost: a loss of effective spatial resolution becomes an issue for dense problems like semantic segmentation as high-frequency details are washed out, leading to blurry object boundaries [7]. One of the most successful methods to prevent this decrease in resolution is a method called 'Shift-and-stitch' [5], where a fully CNN (fCNN) is applied to shifted versions of the input image, and the results are stitched together. The idea of fCNN was taken a step further by [11] with the U-net architecture, comprising a 'regular' fCNN followed by 'up'-convolutions that are used to increase the image resolution. While learned upsampling layers have been proposed before [5], the novelty of U-net involves combining opposing convolution and deconvolution layers with so-called skip connections. This allows an entire image to be processed by U-net in one forward pass, which helps consider the full context of the image [3]. U-Net was specifically developed for single-cell analysis, and since its introduction, it has become one of the most widely used CNN architectures in medical image analysis. Other popular deep learning methods that have been successfully applied to a variety of data types include Deep Cell and Mask R-CNN, while promising results have been recently obtained by generative and vector embedding approaches [10].

Methods

For the cell tracking challenge, the dataset given has three different type of cells that would require specific processing because of the variation in intensities. The difference between background and foreground is indistinguishable in some cases and the presence of cytoplasm and noise is unavoidable. There may be some discrepancy due to the image capturing technique or the duration over which the images were taken. Apart from the noise and unavoidable faults another problem encountered is the motion of cells as there are no laws governing cell motion. To overcome all these challenges, a system that is made of three parts, pre-processing module, detection module and tracking module is introduced.

Pre-processing Module:
For dataset 1, our cell segmentation method combines deep learning with watershed segmentation as proposed by [6]. This approach involves training a single U-Net infrastructure first, and outputs produced by this trained network are then processed by watershed transformation to

obtain the final image segmentation. The entire process is consisted of three stages: 1. Preprocessing: i) Normalization of each input image by Contrast limited adaptive histogram equalization (CLAHE). ii) Transformation of each input image into markers, cell mask, and a weight map, the last of which is used to define the relative importance of each pixel, resulting in a weighted loss function. 2. Predicting: Unlike the original U-Net infrastructure proposed by [11], the U-Net model here produces two different predictions: cell markers and pixel-wise binary classification of the input image into foreground versus background. 3. Postprocessing: a predicted cell marker image obtained from step 2 above is thresholded to form a binary image, upon which a morphological opening operation is applied. Connected components are determined based on this new binary image, and the former are then combined with predicted cell mask (also obtained from step 2) using a marker-controlled watershed transformation to produce the final segmentation.

For task 1 of this project, instead of training this model again from scratch, we decide to use a pretrained model provided by [6] to obtain the segmentation masks for sequence 3 and 4, since this model was trained using the same dataset (sequence 1 and 2) provided by the Cell Tracking Challenge.

Dataset 2 and 3 have significantly distinct characteristics. Number of cells are greater, as a result mitosis events and overlap scenarios are more. Traditional methods are best suited to process and segment images from dataset 2 and 3. According to the proposed method, each image from an image sequence in the dataset are read and converted to grayscale. Small cell residue or noise can be observed in the background. Empirically, a 3x3 structuring element is used to erode the grayscale image to cater this challenge. Even this structuring element for erosion resulted in reducing the size of some cells to a single dot. Hence, we dilate the image to increase the size and therefore make cell body distinguishable. The structuring element used was a 5x5 filter. The size of the filter was taken after considering of pixel intensities and helps in brightening the cell interior. The dilated image has variation in cell intensities. Hence, thresholding is done regionally resulting in different threshold values for different regions. Empirically, the region size taken is 3 pixels. The resulting image distinguishes cell boundary. To complete edges and solidify intensity for cell interior, the dilated image is subtracted from the resulting threshold-ed image. Since all cells are darker than the background, Otsu thresholding is employed to consider the binodal nature of image pixel intensities. We aim to increase separation in intensities between background and foreground. If cells were brighter than the background, the resulting image would display foreground in white and background in black. In the proposed method, since the cells observed at the last step were darker than the background, the cells appear black after Otsu thresholding which can be corrected by inverting the image.

To apply connected components and finally label all cells, we erode the final image first to reduce overlap. Empirically, a 2x2 filter with 3 iteration was found to be optimal in terms of information loss and cell boundary overlap reduction. Connectivity chosen for connected components algorithm is 4 – connected objects.

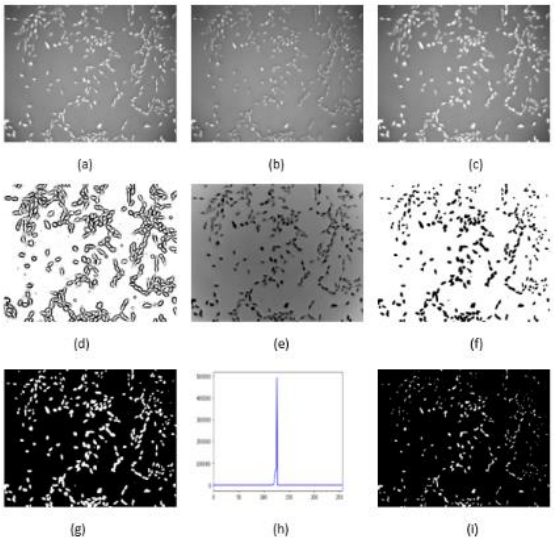


Figure 1: Illustration of illumination correction and noise removal of the input image. (a) Original frame in PhC-C2DL-PSC dataset. (b) Eroded image. (c) Dilated image. (d) Adaptive thresholded image. (e) After background removal. (f) Otsu thresholded image. (g) Inverted image. (h) pixel intensity, number of pixels (x, y) histogram for image (e). (i) Final erosion to remove any noise and overlapping.

Figure 1 illustrates the morphological and thresholding operations performed. Fig 1(a) is a frame in dataset PhC-C2DL-PSC Sequence 2. Figs 1(b) and (c) are erosion and dilation for noise removal and cell brightening. Fig 1(d) is the regionally thresholded image displaying cell boundary. To solidify cell structure, fig1(c) is subtracted from fig1(d). As evident from fig 1(e) a cell has a single intensity. To support this, fig 1(h) displays 2 peaks. The higher background peak and a smaller peak in its left neighborhood denoting the black identified cells. This binodal phenomenon is treated by Otsu thresholding as displayed in fig 1(f). Fig 1(g) inverts the previous image to identify cells as white. To detect overlapping cells, erosion was used the result of which can be seen in fig 1(i).

Detection Module:

To accurately detect cells, unique labels have been created for each cell in the pre – processing module. In detection module, due emphasis has been laid on

cell structure and mitosis events. An approximation of cell structure as an ellipse rather than fitting the exact cell structure, which can change in any frame, as a technique to make bounding boxes has been employed. Distance between cell radii have been compared to recognise mitosis events.

To define cell structure, the shape of cell over its lifetime has been approximated to ellipses. A unique label for each cell was acquired as a result of preprocessing module. The labels mark each cell with a unique number starting from 1, where 0 is for background. In the proposed method, contours are drawn for each uniquely identified label and are checked for shape. Hence the bounding boxes are created. Centre points of ellipse along with semi major axis are used for mitosis detection and tracking.

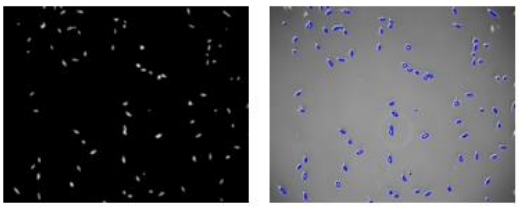


Figure 2: Illustration of cell segmentation and bounding box fitting cell structure in dataset PhC-C2DL-PSC dataset. (a) Bounding boxes on thresholded image. (b) Bounding boxes on original image

Automatic mitosis detection is crucial in the tracking process of the cells. When provided with a large dataset, it is extremely tedious and time-consuming to identify and count each instance of mitosis detection when labelling new cells. The effectiveness of automatic mitosis detection also depends very heavily on the segmentation results as the algorithm works separately and assumes that segmentation is done flawlessly. It can be foreseen that some noise from the images that are not successfully filtered out might cause inaccuracies in the detection process. It is observed from the images that mitotic cells share certain minuscule traits that other individual cells do not. From what is seen, cells that went through mitosis generally stay relatively stationary compared to their counterparts. Obviously, when mitosis takes place, another elliptical oval or circular shape can be seen budding out of the original parent cell. In this way, there will be two nuclei or rather two centroids in this cell structure.

The method that was incorporated to detect mitotic cells is to use the Euclidean distance between the cell centroid. Although the nucleus is not detected, it can be safe to assume that the centroid of the cell has the highest probability of containing the nucleus. The cells are being eroded so that it is corresponding radiuses and contours are as tight to the centroid as possible. Then, the Euclidean distances between each cell centers are tracked and stored. Depending on the type of cells in each dataset, a pre-determined standard distance is being fixed. This is because, in different datasets, the cell sizes are different and so are their proximity within each other. If the Euclidean distance is smaller than the predetermined distance, then it can be assumed that mitosis is likely to have occurred. The mother cell can be identified as being exiting first before the child cells appear and a new cell id is given. In which case, a bounding circle shall be placed around them for easier identification.

Tracking Module:

Centres and semi major axis detected to define cell structure come in play in the tracking module. The tracking module tracks cells based on cell centroids, computes and compares Euclidean distance between cell centroids to associate same cells in consecutive frames. The proposed method checks if any cells are being currently tracked, if not, it registers the centroids resulting from pre-processing module and assigns the unique ids to cell centroids. If cells were being tracked in the previous frame and centroids are presented in the current frame, the algorithm determines Euclidean distance between the cell centroids of previous and current frame. Main assumption for tracking made is that the cells will potentially move in successive time frames, but the distance travelled by cell between frames will be smaller than distance between other objects detected. This helps in avoiding tracking of fast-moving residual cytoplasm in image sequences. If cells being tracked in the previous frame are more than the number of centroids provided by the pre-processing module for current frame, algorithm determines the disappearing cells and deregisters them if for n consecutive frames, the cell does not re-appear for which n is a pre-defined number of frames. If number of centroids in current frame are greater than cells being tracked in the previous frame, the algorithm registers the new cells found.

After associating two pairs of centroids, the centroids are used to calculate and store the speed, distances and confinement ratio of the cell. It also uses the centroids to draw the trajectory path of the cell.

First the Euclidean distance between the two points are calculated by:

$$D = ((x_{n-1}, y_{n-1})^2 + (x_n + y_n)^2)^{\frac{1}{2}}$$

n being frame number. Speed was calculated by getting the Euclidean distance D, between the two points and divided it by the time unit t (which is 1 frame). Formula is given as:

$$S = \frac{D}{t}$$

Total distance is the sum of all Euclidean distance calculated.

$$TD = sum(D_1 \ldots . D_n)$$

Since all centroids of the cell are stored, net distance was calculated using the first centroid stored in the object and current centroid.

$$ND = ((x_1, y_1)^2 + (x_n + y_n)^2)^{\frac{1}{2}}$$

Lastly Confinement ratio is calculated using the total distance and net distance.

$$CR = \frac{TD}{ND}$$

Experimental Setup

Datasets are unique and similar in a lot of aspects. The difference in dataset characteristics lead to difference processing techniques of tracking and segmentation. Although, the approach is different for all dataset, the accuracy is calculated on common evaluation metrics. Metrics used for evaluation are Precision, Recall, F-measure, Jaccard similarity and Dice similarity.

Precision is the ratio of correctly segmented cells with the total segmented cells.

$$P = \frac{TP}{TP + FP}$$

Recall is the ratio of correctly segmented cells to the actual number cells.

$$R = \frac{TP}{TP + FN}$$

F-measure is the weighted average of Precision and Recall.

$$F = 2 * \frac{R * P}{R + P}$$

Jaccard similarity is the fraction of union of number of segmented cells to true number of numbers

$$JSC = \frac{TP}{FP + TP + FN}$$

Dice similarity is the fraction of segmented cells joined with true cell set that is correctly segmented.

$$DSC = 2 * \frac{TP}{FP + 2 * TP + FN}$$

Results and Discussion

Results for segmentation are presented in table 1,2,3 and 4. Using the silver truth masks from the cell tracking challenge website, TP, FP, FN were calculated using manual counting. Three frames were considered each from dataset PhC-C2DL-PSC sequence 2 and from dataset DIC-C2DH-HeLa sequence 2 to capture the essence of the algorithm.

Data	TP	FP	FN	P(%)	R(%)
Frame - 150	108	29	15	78.8	87.8
Frame - 200	155	32	26	82.8	85.6
Frame - 250	214	46	35	82.3	85.9

Table 2: Precision and Recall measures on dataset PhC-C2DL-PSC sequence 2

Data	F - measure (%)	Jaccard(%)	Dice(%)
Frame -150	83.1	71.1	83.1
Frame -200	84.2	72.7	84.2
Frame -250	84.1	72.5	84.1

Table 3: F -measure, Jaccard Similarity and Dice Similarity measures on dataset PhC-C2DL-PSC sequence 2

Data	TP	FP	FN	P (%)	R (%)
Frame - 015	7	3	0	70	100
Frame - 045	9	2	0	82	100
Frame - 075	12	3	0	80	100

Table 3: Precision and Recall measures on dataset DIC-C2DH-HeLa sequence 2

Data	F- measure (%)	Jaccard (%)	Dice (%)
Frame - 015	82	70	82
Frame - 045	90	82	90
Frame - 075	89	80	89

Table 4: F -measure, Jaccard Similarity and Dice Similarity measures on dataset DIC-C2DH-HeLa sequence 2

However, this result should be taken as a reference only, since sequence 1 was used as training set and

sequence 2 as validation set for the U-Net model. The accuracy of a deep learning model should only be evaluated on an unseen test set to avoid the issue of an overfitted model that has limited generalization power. With the lack of ground truths for sequences 3 and 4, we use a few sample outputs produced by U-Net below to provide a visual illustration of the model accuracy.

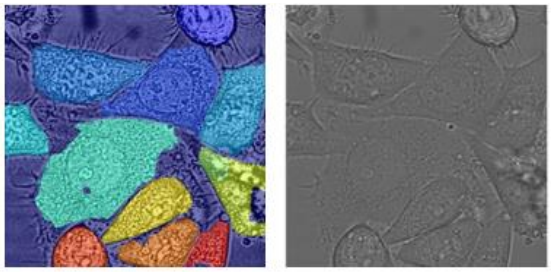


Figure 3: Illustration of cell segmentation in dataset DIC-C2DH-HeLa *(a) Correctly segmented cells (b) Original frame*

Due to the complexity of the DIC dataset, occasionally the outputs contain some segmentation errors, as shown by fig 4(a) and 4(b), from sequence 3. Despite these occasional errors, overall, the model seems to consistently produce satisfactory results.

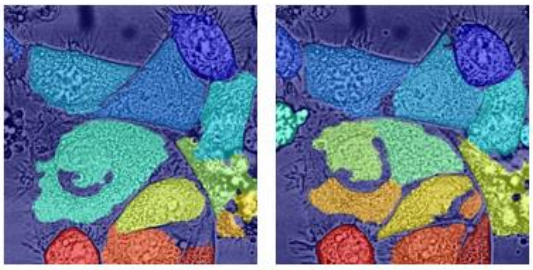


Figure 4: Illustration of cell segmentation error in dataset DIC-C2DH-HeLa *(a) frame 96 in dataset (b) frame 97 in dataset*

As depicted from figure 5(b) and 5(c), ground truth masks fail to segment a few cells where the proposed algorithm's segmentation depicts cell present. This also introduces variation in the result metrics.

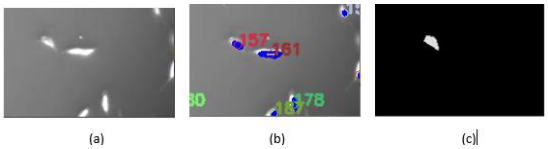


Figure 5: Illustration of cell segmentation in comparison with ground truth mask (a) Original frame in dataset PhC-C2DL-PSC *(b) Segmentation and Tracking from proposed method (c) Cells detected by ground truth mask.*

It is noted that a simple implementation of cell mitosis detection will not be sufficient in getting a near perfect result. In the cell tracking benchmark from the cell tracking challenge, an average of roughly 80% accuracy was being attained. Those examples have used very extensive methods of mitotic detection like forwards and backwards frame propagation, topology preservation or special biasness for the distinction of different cell states.
From the results we have obtained, it can be roughly estimated that majority of the mitosis process are being detected at a rate of about 65% based on manual counting of the mitotic process in a selected sequence. Most of which are very similar to the example shown in Figure 6. As we can see from Fig6(a), the cell in the bounded box of interest is entering its mitotic process and in the subsequent frames, the mitosis process are being identified by the red bounded boxes. After which the child has spitted from its parent cell, a new ID was given to it and the detection process continues from there in the background.

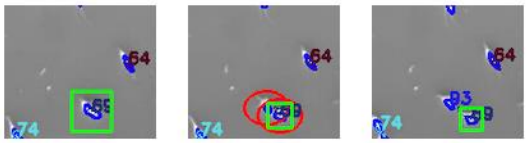
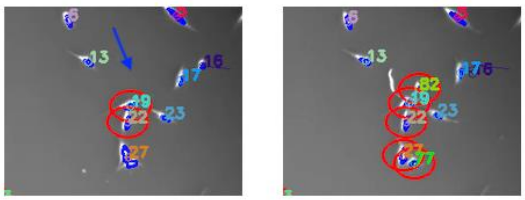


Figure 6: Illustration of cell mitosis event (a) Mitosis process in progress (b) New cell has split from its parent and the bounded box identifies the process (c) Contains parent and child cell clearly labelled with new ID.

Due to the simplicity of the mitosis detection algorithm, there are a few cases where the detection has made errors. These errors are challenging to solve and requires additional features and complexity to be added to the existing algorithm. One such example is when the cell mitosis process did not take place, but the algorithm has interpreted it as an instance of the mitotic process. This is known as a False-positive as the algorithm made an error thinking that there is in fact a cell that has split but in fact it is not the case. This False-positive has been observed to be cause by two different scenarios. One of which is when random noise, or microbial matter passes through the cell in very close proximity and happens to have similar shape and size of the cell. Noise that appear in the images is the result of some minor faults of incomplete segmentation as there will still be edge cases in the process. This leads to incorrect detection as seen in Figure 6 whereby random matter in Figure 6b is seen to be taking up the void of space indicated

by a blue arrow in Figure 7a. Another more straightforward False-positive scenario is when two cells that do not participate in mitosis moves very close to each other almost overlapping, this causes the algorithm to assume that mitosis has just been complete but in fact is not the case.

Another flaw about the algorithm can be depicted with a false negative example. This means that the cell mitotic process did in fact take place, but the algorithm failed to detect it. Such cases are in fact much rarer than other scenarios mentioned because the pre-determined standard distances have been already trailed and for each dataset and the best value is being set. A very good example can be found in Figure 8a and 8b, whereby the cell that is undergoing mitosis is being stretched in a slightly abnormal way. The child cell will then break off from its parent counterpart (cell in attention bounded by a green box) in a wider distance compared to the majority cases of mitosis. As it can be noted that majority of the cells that undergo mitosis are usually stationary and not stretched out like this, there is no way the algorithm is able to account for such a phenomenon. The algorithm can be further improved in order to prevent this from happening for example tracking new cell IDs by using forwards and backwards tracking. This way, new cells that appear can then be taken inconsideration and be focused on to find where the parent cell is.

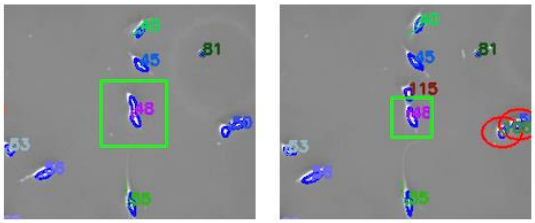


(a) (b)

Figure 7: False Positive event of cell mitosis (a) Mitosis process in progress (b) New cell has split from its parent and the bounded box identifies the process

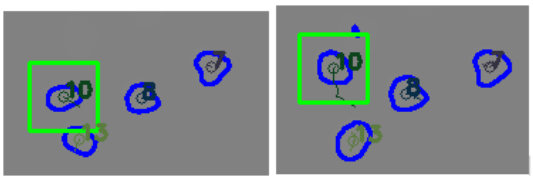
Finally, there are some cases whereby the mitosis detection has failed due to external events that is not within the scope of the tracking procedure. Some cells that undergo mitosis can be seen with new cell IDs was not captured at all by the detection algorithm due to the jump in fames in the given dataset. When the images are not taken frequently enough, some cell mitotic process are being missed and all is left in the images are just new cells that seemingly appear out of nowhere. In such case, those mitotic scenarios are not taken into consideration when evaluating the results of the detection algorithm.

The results and accuracy of 65% was attained by manually policing the mitotic process in random sequences in the dataset. By inspecting a few windows of subsequent frames in succession we can identity the mitosis process, for example, by taking frames 30,31,32 and 33, we count the events of mitosis and compare it with how many has the algorithm correctly or incorrectly detected as a mitotic event.



(a) (b)

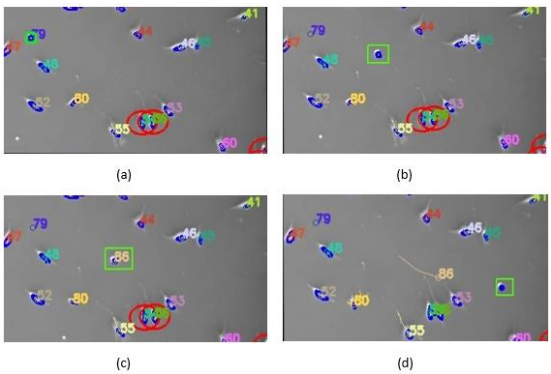
Figure 8: Illustration of cell mitosis event along radial axis (a) Mitosis process in progress (b) New cell has split from its parent but no mitosis detected.



(a) (b)

Figure 9: Illustration of tracking module in dataset Fluo-N2DL-HeLa (a) Frame tracking cell id 10 (b) tracking of cell id 10 in successive frames.

As seen in fig 9, the tracking module can maintain a unique ID for a given cell and track its trajectory across different frames. The path of the cell is recorded in the output as shown. Since the module uses a centroid tracking algorithm that uses minimum distance to associate cells, it can track cell moving at the relatively slow pace across frames.



(a) (b) (c) (d)

Figure 10: Illustration of cell tracking in event of fast movement (a) shows cell with ID 79 on the top right of the image, boxed in green initial position. (b) shows the same cell, boxed in green, losing its label after 2 frames (c) shows the same cell being labelled as cell 86 after another 2 frames. (d) shows the cell, boxed in green but once again losing its label after 2 frames.

However, as seen in Fig.10, the tracking module cannot handle cells travelling at fast paces and fails to associate the cell properly. Here, the cell moves a fair amount of distance in 8 frames. The image in Fig.10a to Fig.10b shows the module losing track of the cell and leaving the ID of the cell in its original position even after the cell has moved as seen in Fig.10b. In Fig.10c, the tracking module labels the unlabeled cell in Fig.10b with a new label 86, thinking of it as a new cell that appears on the screen. In Fig.10d, the tracking module once again loses track of the cell, it does however, show it was able to track cell 86 up to a certain point as seen by the path left behind with label 86.

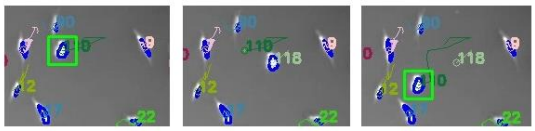


Figure 11: Illustration of misidentification in tracking module (a) Tracking of cell 110 (b) ID being assigned even though segmentation error observed. (c) Reappearance of cell being tracked.

There are also cases where cells are mistakenly identified and thus swapped. In Fig 11a, we see cell 110 where it originally was. In Fig.11b shows the same cell moved slightly to its right, but due to a segmentation error, the new centroid was created and given the ID 118. In Fig.11c the same cell moves down and slightly left. However, the IDs, 110 and 118 are swapped leaving ID 118 to with no cell. Although the error was due to a segmentation error, the frame in Fig.11c proves that the algorithm is susceptible to these edge cases and overall has not shown the correct trajectory path of cell 110.

Conclusion

The fact that all the top 3 results for the challenging DIC-C2DH-HeLa dataset are produced by a deep learning framework has reaffirmed the success of deep learning in biomedical applications, particularly for complex datasets. It is also worth noting that the work by [6], which has produced a top 2 result for this dataset, uses a combination of both computer vision and deep learning in a novel approach. Their work has led to an interesting avenue for future research: rather than being viewed and used as two separate tools, these two methods can be working in tandem and complementing each other to produce exceptional results.

The algorithm for the mitosis detection is capable of detecting mitosis in most cases but there is much room for improvement. By only relying on the Euclidean distance as its main detection factor is not enough to minimize detection errors during the varying types of events that was stated. Because standard distances must be pre-determined and fixed, it will be much more ideal if the algorithm does it automatically. By adopting circularity measure, the cell's area and their circumference can be worked out and suitable adjustments to erode the cell on different datasets can be achieved in a more effective way. This makes the task more flexible to different changes and for different datasets. One very useful feature that can improve the mitosis detection dramatically is to include backwards and forwards tracking. In this way the frames can then be linked, and cell locations and movements can be easily measured and follow up on to determine if the cell did in fact undergo mitosis or is just a random biomolecular matter that is interfering with the images. By using this technique, the cells that is undergoing apoptosis can be also measured and tracked so it will not just be counted as another artifact of noise or random particulate. Advanced implementations of forwards and backwards tracking also includes cell topological differentiation and monitorization. This is done so that it will be possible to identify apoptotic cells and mitotic cells to achieve a far greater accuracy in the results. Given enough time, the mentioned advanced techniques can be stacked on top of the current algorithm to increase the percentage accuracy of cell mitosis detection.

With the tracking module, we employ a simple method that relies on a critical assumption of cell behavior, using spatial information (proxied by the Euclidean distance) between cells from one frame to the next. While it can yield results of a certain accuracy level, we acknowledge the drawback of this approach in situations where cells have rapid movements or live in a dense population [9]. The method was mostly sufficient with the datasets given. Movement of the cells in the datasets were mostly slow and the difference in position of most cells between frames were small, thus the algorithm was able to successfully associate most cells correctly. As mentioned, the tracking method would fail if a cells position was too different between frames. Fortunately, most of the cells with this behavior were mostly abnormalities or noise and thus did not need to be tracked. However valid cells with this behavior were still missed and thus the method is not a 100% accurate. It was also observed we could get false negatives through the cell ID swapping, showing that the algorithm is not very robust and is susceptible to certain edge cases.

In the future, we may introduce other metrics into the calculation to associate centroids. These metrics include the Characteristics of the cell such as intensity, volume, orientation, etc. to calculate the 'nearest distance', it should be noted that the characteristics of cells are quite dynamic across time and also diverse in different datasets, making it quite challenging to impose certain universal assumptions

to solve the tracking problem. Adding these considerations to the association process of the algorithm may improve its accuracy and mostly limit the number of false positives occurring in the sequence

Since time-lapse microscopy images brings about the added temporal information of cells, their division and movements, several traditional computational methods have taken advantage of this extra information. An alternative framework to make use of the combined spatial-temporal information is deep learning methods, which can learn intrinsic, discriminative features of mitotic cell images from the input. As a result, a CNN model combined with the recurrent neural network (RNN) or graphical model is a potential candidate to explore for further performance enhancement [4].

Contribution of Group Members

Deepansh Deepansh (z5199370)
Worked on pre-processing using traditional steps. Segmentation done using ndilabels and watershed. Fitted convex hull to draw boxes. Tracking done by using Euclidean distance metric. Collaborated with Ujjwal for coding as well as corresponding report sections.

Ujjwal Garg (z5212244)
Worked on finding traditional techniques to work on. Pre-processed the image using adaptive thresholding and opening-closing operations. Tracked the cells once segmentation was done using Euclidean distance metric. Work done in conjugation with Deepansh on code as well as relating report sections.

Leonard Lee (z5173917)
Paper reviews related on task 1 and 2. More emphasis on task2's implementation.
Attempted UVA-NL's approach of using Siamese tracking with forwards and backwards tracking but was not successful.
Attempted close mimicking of FR-Fa-GE's version of tracking.
Did task 2's implementation and evaluation of the results.

Son Tong (z3448000)
Did paper reviews for the report write up. Implemented U nets for dataset 1 and wrote corresponding sections in the report.

Luke Yong (z5216244)
Implementation of path tracking, calculations of task 3 and user select feature for task 3.
Attempted Unet approach to segmentation but ended up dropping it due to its completion by another member.

References

[1] Grah, J. (2014). Methods for Automatic Mitosis Detection and Tracking in Phase. Münster: Master's Thesis, University of Münster.

[2] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet Classification with Deep Convolutional Neural Networks. Communications of the ACM, 84-90.

[3] Litjens et. al. (2017). A survey on deep learning in medical image analysis. Medical Image Analysis, 60-88.

[4] Liu, A.-A., Lu, Y., Chen, M., & Su, Y.-T. (2017). Mitosis Detection in Phase Contrast Microscopy Image Sequences of Stem Cell Populations: A Critical Review. IEEE Transactions on Big Data, 443-457.

[5] Long, J., Shelhamer, E., & Darrell, T. (2015). Fully Convolutional Networks for Semantic Segmentation. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE.

[6] Lux, F., & Matula, P. (2019). Dic image segmentation of dense cell populations by combining deep learning and watershed. 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019) (pp. 236-239). IEEE.

[7] Marmanis, D., Schindler, K., Wegner, J. D., Galliani, S., Datcu, M., & Stilla, U. (2018). Classification with an edge: improving semantic image segmentation with boundary detection. ISPRS Journal of Photogrammetry and Remote Sensing, 158-172.

[8] Meijering, E., Dzyubachyk, O., & Smal, I. (2012). Chapter nine - Methods for Cell and Particle Tracking. Methods in Enzymology, 183-200.

[9] Meijering, E., Dzyubachyk, O., Smal, I., & Cappellen, W. A. (2009). Tracking in cell and developmental biology. Seminars in Cell & Developmental Biology, 894-902.

[10] Moen, E., Bannon, D., Kudo, T., Graf, W., Covert, M., & Valen, D. V. (2019). Deep Learning for cellular image analysis. Nature Methods, 1233-1246.

[11] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: convolutional networks for Biomedical Image Segmentation. Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015 (pp. 234-241). Springer, Cham.

[12] Stanford. (2020, August 4). CS231n: Convolutional Neural Networks for Visual Recognition. Retrieved from https://cs231n.github.io/convolutional-networks/

[13] Ulman et. al. (197). An objective comparison of cell-tracking algorithms. Nature Methods, 1141-1158.