

Cell Nuclei Identification from Confocal Microscope Images

Pinyuan Feng
scypf1@nottingham.ac.uk

Introduction

As the saying goes, “A picture is worth a thousand words.” Microscope image plays a significant role in the area of biology, which provides biologists with quantitative data and valuable information. To further analyse images, the visualization of samples cannot be done without image processing techniques. However, each image processing method has its strengths and weaknesses, and choosing an appropriate approach is quite important. In this report, an image processing pipeline (*Figure 1*) is introduced to identify the cell nuclei from a set of confocal laser microscope images of plant roots. According to experimental results, the evaluations of the corresponding methods are presented as well.

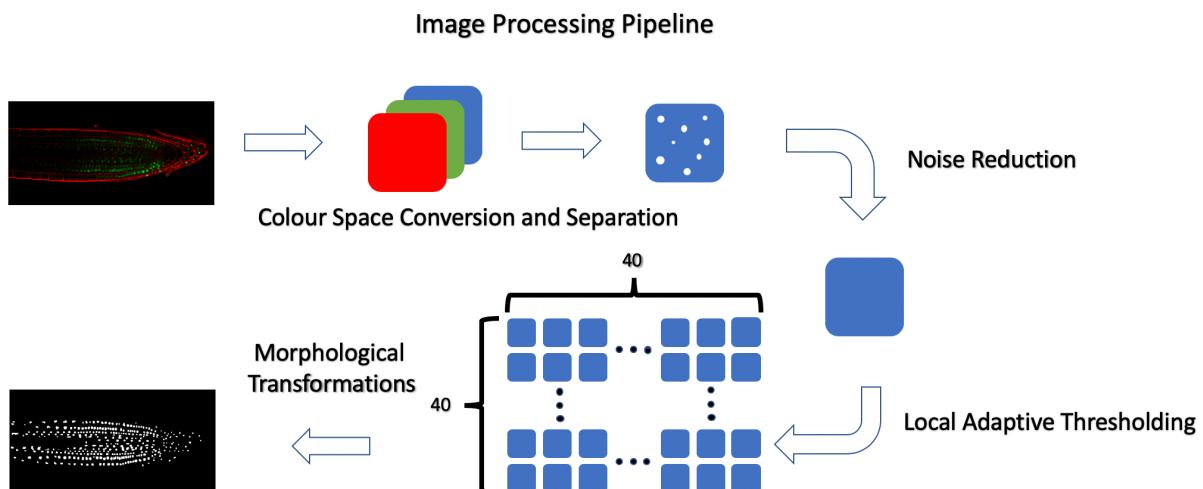


Figure 1

Experiment

Colour Space Conversion

The colour space in image processing aims to facilitate the specifications of colours in some standard way. Different types of colour space have their own advantages under specific situations. Hence, the effect of image processing may benefit from changing one type of colour-encoded signal into another.

Given three different confocal laser microscope images of plant roots, I converted the original colour space RGB into other types including HSV, YCbCr, YUV and CMYK. The processed result is shown in *Figure 2*. It is clear that RGB colour space performs the best among these candidates. Specifically, HSV colour space makes the original image less clear, where cell walls and nuclei cannot be easily distinguished. To some extent, the first picture with YCbCr and YUV colour space performs better than that with HSV colour space, but the details are trivial in the second and the third picture, whose features are difficult to catch. Although images with CMYK seem much better than the other three, cell nuclei appear different colours. Some are blue and others are much close to red and pink. This may not facilitate the identification

of cell nuclei. Therefore, RGB colour space is a better choice relatively, because the noticeable colour contrast between cell walls and cell nuclei makes image processing more feasible.

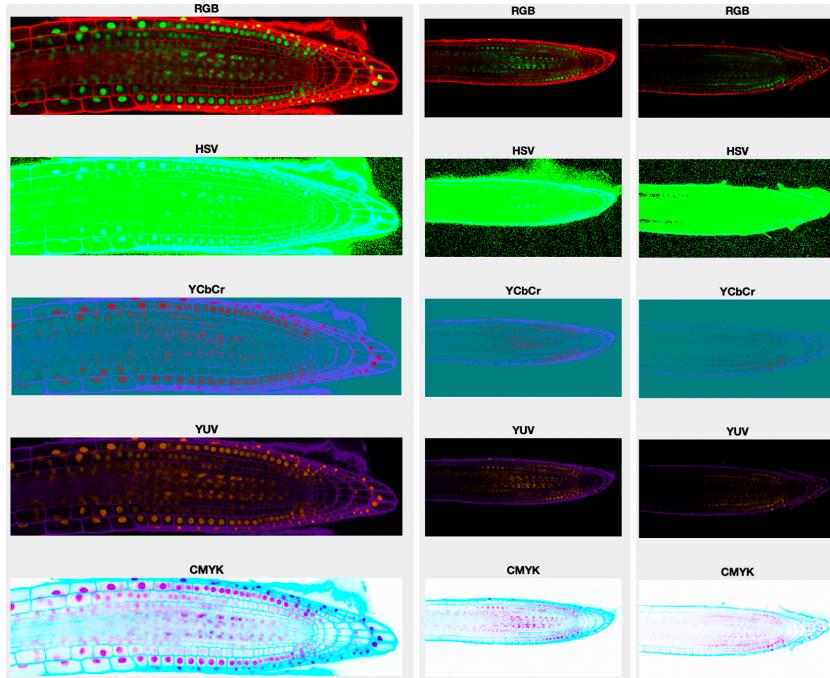


Figure 2

Image Colour Separation

After considering RGB colour space, I extracted red, green and blue channels from the original images and found that there is no information in the blue channel (*Figure 3*). Hence, I ignored the blue part and just focused on the rest of colour channels.

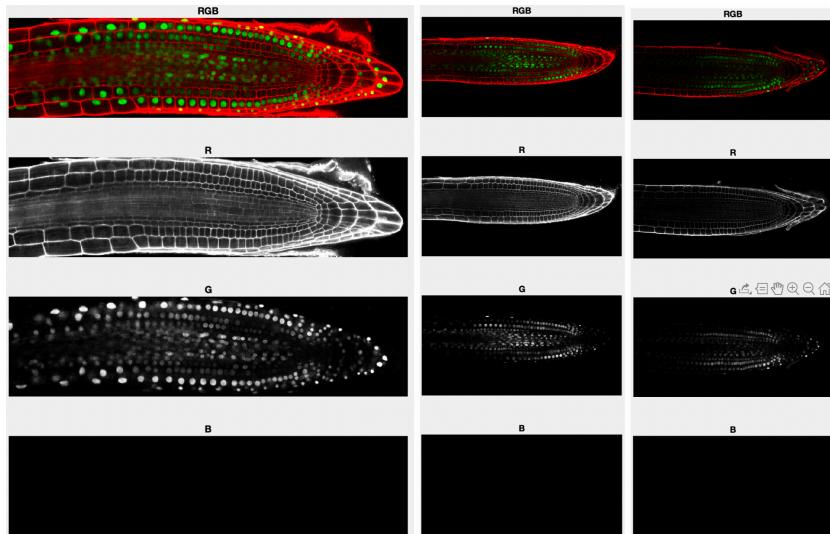


Figure 3

In this case, there are two possible ways to recognize the cell nuclei. The most direct idea is to process the green channel. And another idea is to take the red channel into consideration. The following steps are the implementation of the former idea, and the reason why I prefer the former idea over the latter one will be illustrated later.

Noise Reduction

Some noise may not be visible to human eyes, so I believe it is necessary to apply noise suppression in order to improve the image quality. Before noise reduction, a pre-processing step I find it useful is to isolate greenness from the background.

$$\text{Greenness} = G - (R + B) * 0.5$$

When removing noise, it is possible that boundaries between each cell will be blurred, especially the case where two cells are tightly close to each other. Therefore, I applied 'Anisotropic Diffusion' approach to the green channel image with MATLAB built-in function 'imdifusefilt(l)', which can preserve edges and other details. The image quality becomes higher after noise reduction, according to *Figure 4*.

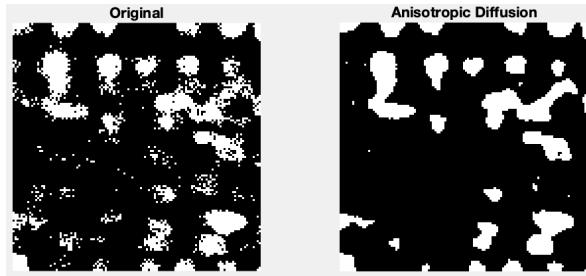


Figure 4

Thresholding and Segmentation

Thresholding is a useful step to identify nuclei region in the image. Whether to use global thresholding or local thresholding requires further implementation, so I have done the following experiments. Typically, I chose global Otsu Thresholding and global Iterated Thresholding approaches. Also, according to the histogram of the green channel image (*Figure 5*), where the shape is unimodal, I applied global Rosin Thresholding as well.

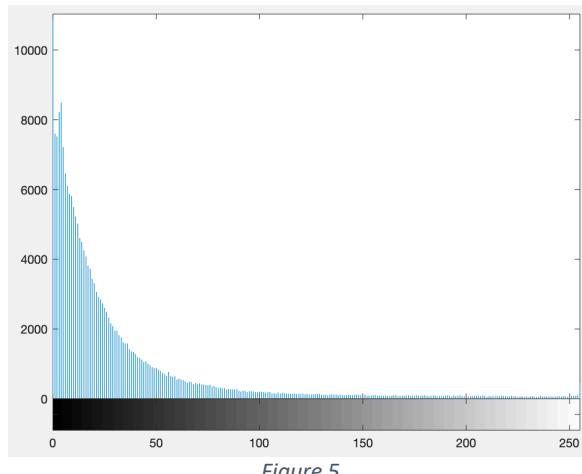


Figure 5

Observing the grey level image, we can see that there are many overlapping areas in the centre of the image, where the edges are not sharp enough. The cells on the top and bottom of the image are comparatively easier to be detected. In this case, local adaptive thresholding is worth a try, because it can generate different thresholds for different parts of the image. In this step, I use my own version of local thresholding. The whole image is divided into $40 * 40$

blocks (I have tested on different number of blocks and found that local thresholding with $40 * 40$ blocks appears to be better), and Otsu Thresholding is applied to each partition.

Morphological Transformations

Thresholding may result in some mis-classified pixels, which is unavoidable. A further step is required to clean up the image. Here, I applied image opening with a disk-shape structuring element to the binary image. The reason I did so is that it can remove small objects from an image while preserving the shape and size of larger objects in the image.

Result Representation

The ultimate image processing output is displayed in *Figure 7*. Generally speaking, the implementation achieves a good result, which shows a clear distribution of cell nuclei. As you can see, cell nuclei extracted from the original images are explicit and identifiable with less information loss, which means the core features of the cell nuclei retain. In order to obtain a quantitative result, MATLAB built-in function ‘`bwconncomp()`’ is used to label the cell nuclei and count the specific number. However, nothing is perfect. One drawback is that there are still some connected nuclei in the middle region of the images. More explanation of the result in each step will be discussed in Approach Evaluation Section of the report. In order to convince you that it is a relatively optimal result, I have developed a GUI program in MATLAB, where you can manually choose other processing methods and make a comparison.

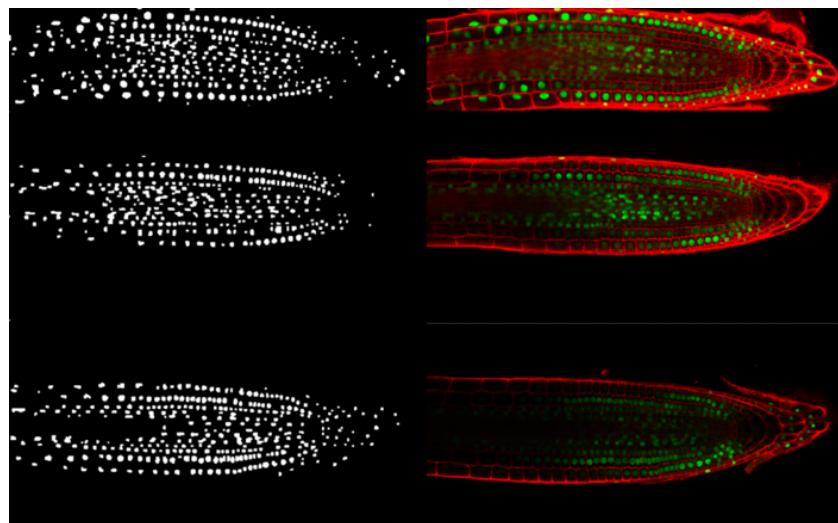


Figure 6

Approach Evaluation

Overall, my implementation leads to a satisfactory result. The image processing pipeline is automatic without user interaction and works well on all three provided confocal laser microscope images. However, there are some weaknesses of my approaches. An explicit evaluation is presented below.

In this section, the third provided image is chosen as a typical example to make the evaluation more straightforward. In the first place, colour space conversion and channel separation helped me decide to focus on green channel, a lower-dimensional space, which reduced image processing workload. However, ignoring the red channel means some possibly useful

information might be lost. Making full use of those information may have a good result or even better.

In the process of noise reduction part, I selected a local area of the image and applied 4 noise reduction methods respectively. Since the noise-reduced images are not clear enough, so binary images are generated to facilitate comparison. According to *Figure 7*, ‘Anisotropic Diffusion’ approach has a good effect of preserving edges to a higher extend and other details while making the image smooth. Other noise reduction approaches, such as mean filtering, Gaussian filtering, may blur the boundaries of objects and degrade details, especially in the middle area where overlap occurs. The weakness of the chosen approach is that some noise still remains.

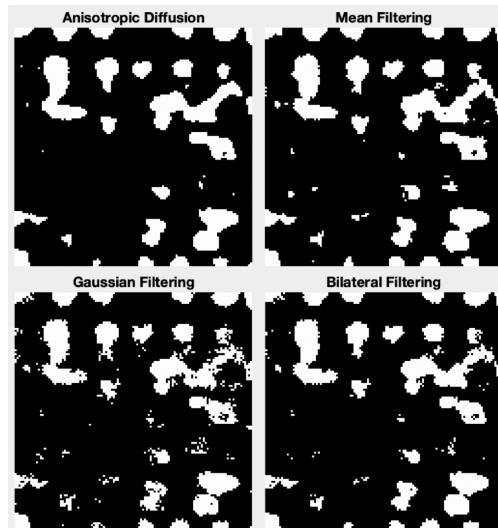


Figure 7

Next step is to apply thresholding methods. Since local areas with different brightness have an irregular distribution, applying local adaptive method appears to be an optimal choice. Overall, my version of local thresholding works better than other built-in thresholding function according to the comparison (*Figure 8*).

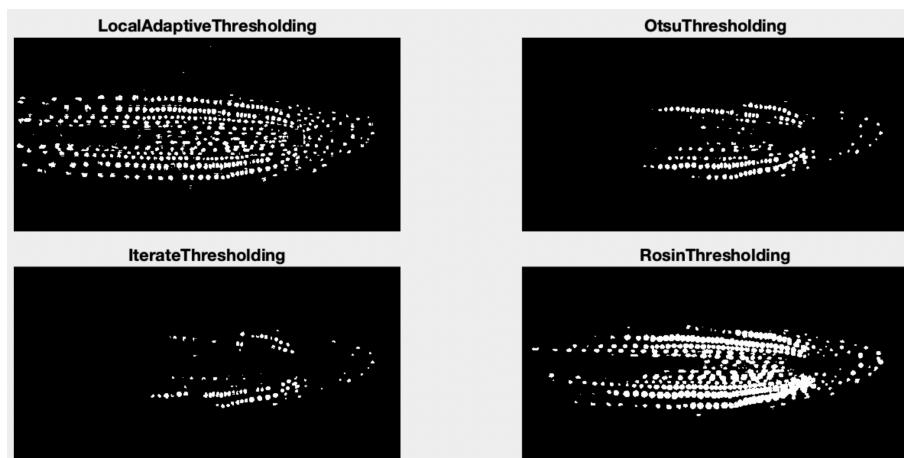


Figure 8

Specifically, local adaptive thresholding not only generates different thresholds for local areas but also keeps most information intact. For Otsu and Iterated thresholding, a high threshold

is selected resulting in loss of information; on the contrary, Rosen thresholding chooses a low threshold leading to information redundancy and making the overlap between cells apparent. However, according to *Figure 9*, the shapes of some cell nuclei are destroyed. Also, some noise or mis-classified points are visible.

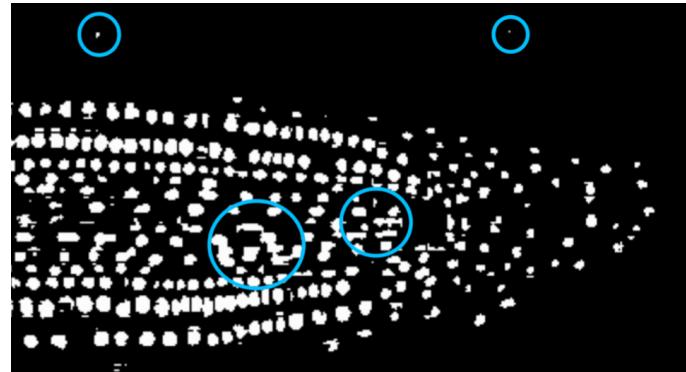


Figure 9

In order to remove those uncategorized points, some morphological transformations are required to clean up the whole image. There is a trade-off here. If morphological transformations are applied repeatedly or the radius of the structuring element is large, many small but useful points were removed as well, which reduces the accuracy of cell nuclei counting. If smaller structuring element is chosen, more information will be preserved as well as misclassified noise.

After a series of testing, the shape of structuring element is chosen to be ‘disk’ and radius is set to 3 in order to retain the shape of nuclei and useful information. With image opening applied, small noises are eliminated but the shape and size of larger objects in the image remain almost un-changed. One defect of this method is that the problem with linked points still cannot be perfectly solved (*Figure 10*).

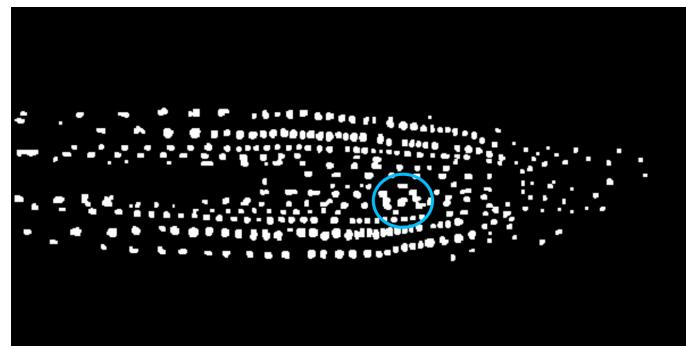


Figure 10

Additional Attempt & Analysis

There are some areas where cell nuclei are extremely close. Applying noise suppression may further blur the edges of cell nuclei and thus causes overlap. We can use the red channel image to identify the cell regions and separate cell nuclei, because ideally one cell contains exactly one nucleus and cell wall serves as a boundary between two adjacent cells. The general implementation process is as follows (*Figure 11*).

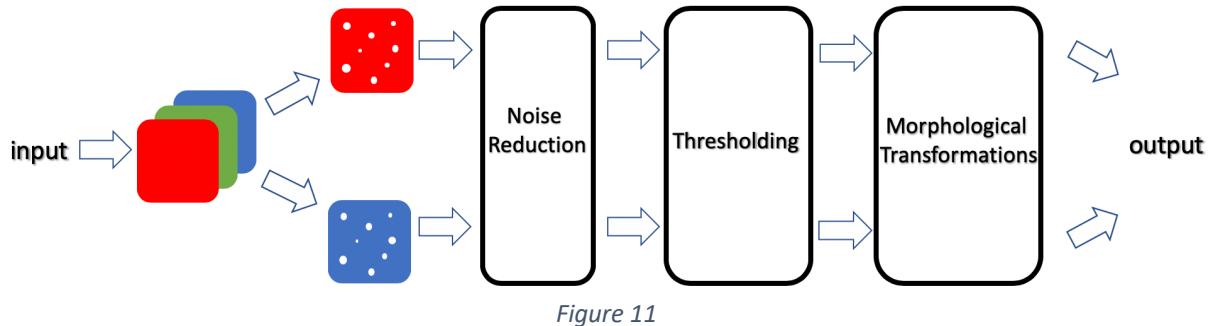


Figure 11

Following this idea, I extracted red channel and green channel from the image, and processed them with different suitable methods respectively. After that, I combined them together by subtracting cell wall binary image matrix from cell nuclei binary image matrix, hoping to eliminate a number of connected cell nuclei. In this way, the difference of pixel values between two light areas (overlapping areas in the former image and cell walls in the latter image) will be zero, which represents black in binary image. The rest part of the image remains unchanged as positive pixel value is still one representing white colour and negative pixel value is considered to be zero by default. The *Figure 12* below shows the principle behind in a straightforward fashion.

$$\begin{array}{c} \text{cell nuclei} \quad \text{cell wall} \quad \text{cell nuclei} \\ \text{---} \\ \text{---} \end{array}$$

Figure 12

However, the result (*Figure 13*) is not as good as previous implementation. Although the number of connected cell nuclei is reduced, much information gets lost. In other words, the distribution of cell nuclei is sparser compared to the result of previous idea. Also, the shape of cell nuclei is not perfect.



Figure 13

From the above result, we can conclude that the idea may not be appropriate under this situation. There are some possible factors that influence the result of image processing. Firstly, the binary image of cell walls has a low performance (*Figure 14*). As you can see, some cell walls are disconnected and cannot serve as boundaries to separate the connected cell nuclei. Secondly, the middle part of the cell wall image is not as clear as expected and thus the connected nuclei still exist in that area.

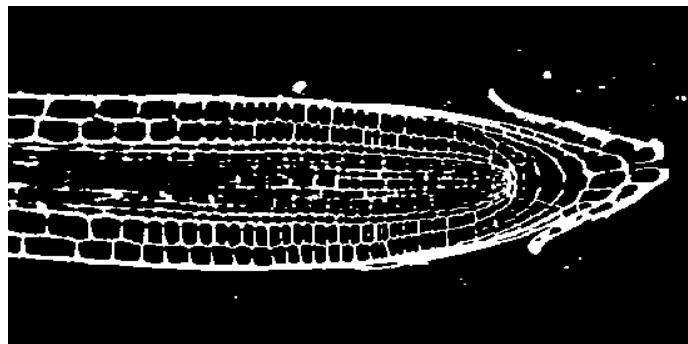


Figure 14

Lastly, due to multiple layers in the original cell sample, when projecting 3D-cell sample into 2D-image, cell walls in deeper layers are covered and become less visible. Thus, in some areas of the image (*Figure 15*), two or more nucleus are displayed in one cell and they are partially overlapping. Also, some high-layer cell walls ‘separate’ some low-layer cell nuclei apart. After a set of image processing operations, those cell nuclei become incomplete and are possibly regarded as noise to be removed. This means the number of cell nuclei will be further underestimated.

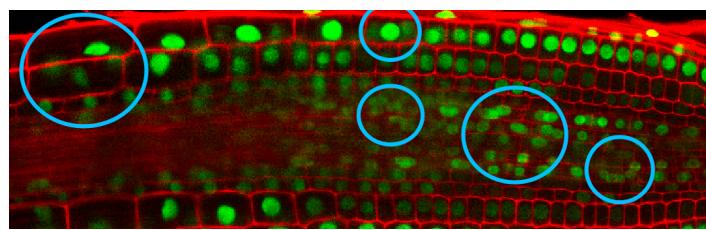


Figure 15

Conclusion

In conclusion, my implementation pipeline, integrated with feasible image processing approaches, gives a great performance in cell nuclei identification. The pipeline uses the RGB image as an input and primarily focus on the green channel image. Then ‘Anisotropic Diffusion’ makes it possible to smooth the image while retaining much useful information. Next, Local adaptive segmentation selects different thresholding for local areas. Finally, with morphological transformations, the binary image is able to show the main features of cell nuclei. Although the processing pipeline may not be able to solve the connected nuclei problem completely, it works well in most local areas of the image. Also, the solution is automatic in that it allows cell nuclei identification from three provided images without changing user-supplied parameters.