School of Engineering and Applied Science (SEAS), Ahmedabad University

ECE501: Digital Image Processing

Group Name: NetraByte

Project: 11. Automated Object Counting (Coins, Cells, etc.)

Team Members:

- Aashi Shah (AU2340041)

- Bansi Mahkana (AU2340191)

- Diya Patel (AU2340184)

- Nirjara Jain (AU2340010)

I. What has been done so far (Progress)

A. Dataset Selection

- For this initial phase, we selected the BV7 cell subset containing 456 grayscale images from the LIVECell dataset.

- The reason for choosing this cell subset is the shape of the cells; they are round, which makes the processing easier.

B. Methodology Implemented

We have developed two approaches to compare segmentation performance on BV7 cell images.

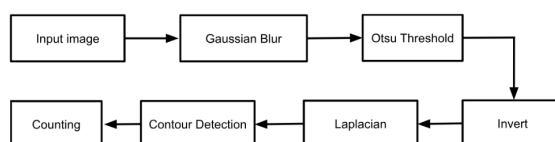### B..1 Otsu Thresholding and Contour-based Segmentation

Figure 1: Flowchart of Approach 1

1. Pre-processing Converted all the images into grayscale images if it was not. Applied a Gaussian Blur with a 5x5 kernel to reduce the image noise.

2. Segmentation Apply Otsu's thresholding for creating a binary image. Inverted the binary mask to make the cells (foreground) white. Used a Laplacian filter to sharpen the edges.

3. Post-processing Removed very small regions based on area to eliminate noise.

4. Counting and Visualization Detected the object using contour detection. Drew green contours around every cell and counted the overall contours that met area constraints.

B..2 Adaptive Thresholding and Connected Component Analysis
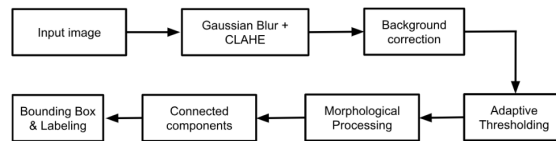

Figure 2: Flowchart of Approach 2

1. Pre-processing Applied Gaussian Blur to remove noise. Applied CLAHE (Contrast Limited Adaptive Histogram Equalization) to increase the contrast. Corrected the background by large-kernel Gaussian blur and subtraction.

2. Segmentation Used Adaptive thresholding for better handling of uneven illumination.

3. Morphological Processing Applied a Laplacian filter for edge enhancement.

4. Object Detection and Labeling Identified objects using connected component analysis. Drew bounding boxes and assigned unique labels to detected cells. i.e, cell number.

The contour-based approach works well on high-contrast, isolated cells, while the adapted thresholding approach performs better on uneven or clustered cell images.
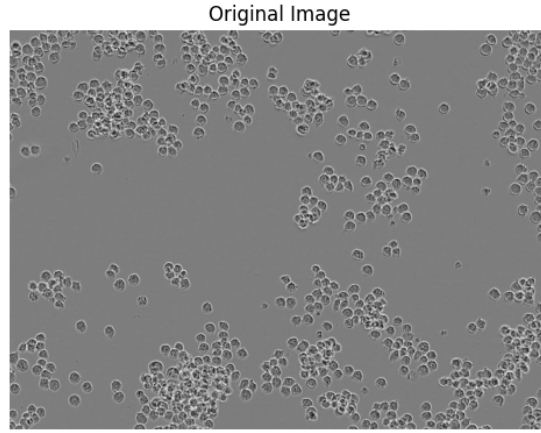
C.   Results and Observations
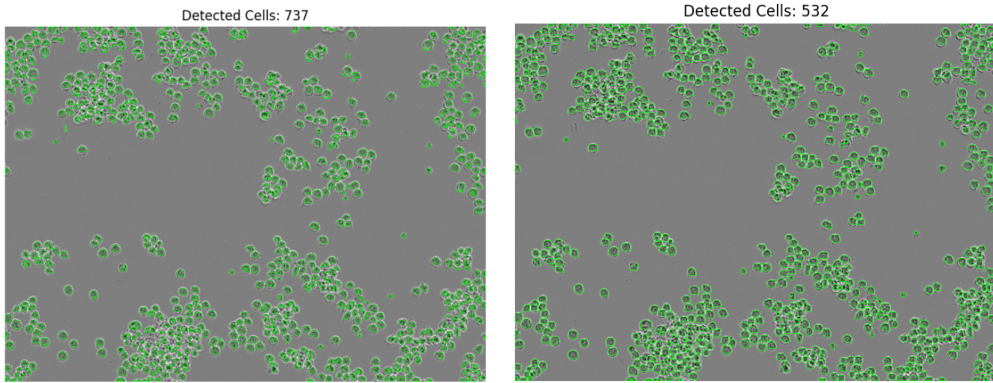

Figure 3: Original Image

Figure 4: Solution Circuit

C..1   Otsu Thresholding

Otsu's algorithm finds a global threshold that separates the whole picture based on a maximum variance between background and foreground pixels. Though straightforward and efficient when there are uniformly lit pictures, it overcounts for noisy pictures, pictures that have non-uniform lighting, or pictures that have cells that are tightly packed. Small fluctuations in a cell or highlights in the background can be confused as individual cells. Otsu's thresholding in our picture counted 737 cells, which shows over-segmentation compared to adaptive thresholding.

C..2   Adaptive Thresholding

Adaptive thresholding calculates the threshold locally for different regions of the image, accounting for variations in illumination and contrast. This makes it effective in segmenting cells in areas where brightness or background intensity varies. As a result, it provides a more

reliable estimate of the true number of cells by reducing false positives caused by background noise or uneven lighting. In our image, adaptive thresholding counted 532 cells, which likely represents a more accurate cell count.

II. What is planned for next week

1. Currently, we are working on only 1 type of cell (out of 8 types). We plan to create a model that will be trained and evaluated on the entire dataset.

2. The ground truth/annotations of our dataset are present in JSON file format. We have yet to develop a pipeline for converting JSON to images.

3. In the future, we will compare our segmented output to the provided ground truth/annotations for calculating the accuracy.

References

[1] C. Edlund, T. R. Jackson, N. Khalid, N. Bevan, T. Dale, A. Dengel, S. Ahmed, J. Trygg, and R. Sjögren, "LIVECell—A large-scale dataset for label-free live cell segmentation," *Nature Methods*, vol. 18, no. 9, pp. 1038–1045, 2021. [Online]. Available: `https://www.nature.com/articles/s41592-021-01249-6`

[2] Nitya123-Github, "Automation of Counting Objects Using Image Processing Techniques," 2023. [Online]. Available: `https://github.com/nitya123-github/Automation-of-counting-objects-using-Image-processing`

[3] Majumder M, Wilmot C. Automated Vehicle Counting from Pre-Recorded Video Using You Only Look Once (YOLO) Object Detection Model. Journal of Imaging. 2023; 9(7):131. `https://doi.org/10.3390/jimaging9070131`