University of Florida

Machine-Learning-Driven Cell Detection and Quantification in Microscopy Images

Team 10
Kellyna Voyer
Jada Brown
Daniel Osorio
Valery Fuhrer
December 3rd, 2024
BME3053C
Github: https://github.com/DanielOsorio1/Final-Project

Problem Statement:

This project aims to develop a program capable of detecting and counting cells in microscopy images with the highest possible precision. The traditional method of cell counting, which relies on a manual counter operated by the scientist while observing cells under the microscope, has serious limitations as it is error-prone, time-consuming, and lacks consistency. Furthermore, the counting process faces additional challenges due to variability in cell morphology, staining techniques, and image quality.

Literature Review:

Hemocytometer

One of the most common methods of cell counting is using a hemocytometer. This device is the most common laboratory instrument for counting cells. The hemocytometer consists of a specialized glass slide with a grid-etched chamber of known dimensions which is divided into nine large squares. Each square measures 1 mm^2, with the chamber depth precisely set at 0.1 mm, creating a known volume (0.1 μL) [1] for accurate cell density calculations. Using a microscope, cells within the grid are counted, and their concentration is calculated based on the known volume. This design allows researchers to determine cell concentrations in a sample with high precision. Despite its simplicity and effectiveness, manual counting with a hemocytometer can be time-consuming and subject to human error, particularly with dense or clumped cell samples.

Image Cytometry

Image cytometry is a method of cell counting that combines microscopy and computational image analysis to detect and count cells[2]. This approach requires taking an image of the cells under the microscope and using special software to analyze and count images based on their cellular components and features. This type of cell counting technique usually uses fluoroscopy to enhance contrast and make it easier for the software to identify different cellular components such the nuclei, membranes, or cytoskeletal structures. A common software for this type of technique is called CellProfiler [3] which is an open source software that enables researchers to identify, segment, and quantify cells in images by applying advanced algorithms for feature extraction and classification. CellProfiler is capable of handling various staining techniques and cell types, making it highly versatile for applications in cell biology, drug discovery, and clinical diagnostics.

Plate-Based Cell Counting

Another cell counting technique is the Plate-based cell counting. This is a technique that uses multi-well plate formats and signal detection systems to estimate cell numbers. [4] This approach is used where cells are cultured in microplates, such as the 96-well or 384-well. This method for counting is particularly advantageous in studies requiring rapid and automated quantification, such as drug screening, toxicity testing, and immunological assays. An example of plate-based cell counting is the ELISpot reader [5], which analyzes enzyme-linked immunospot assays. In these assays, cells are cultured in wells coated with capture antibodies specific to secreted cytokines or other proteins of interest. As cells release the target molecules, they are captured and detected through different signals. The ELISpot reader quantifies these signals, correlating their intensity to the number of cells present. While this technique allows quantifying cells it is very error-prone as it can confuse signals and is very costly.

Tissue Dissociation and cell counting

Tissue dissociation and cell counting is a technique used for isolating individual cells from solid tissues to enable accurate quantification. Tissue dissociation consists of breaking down the cell's extracellular matrix[6]. One method uses enzymes such as collagenase and trypsin to degrade the extracellular matrix and release cells. While these enzymatic methods are effective, they require careful control of enzyme concentration and incubation time to minimize cell damage and prevent cell death.

Another way to dissociate tissue is mechanical dissociation [7] which uses physical methods like mincing, grinding, or using tissue grinders to fragment the tissue. This approach is frequently combined with enzymatic digestion to ensure maximum yield while preserving cell viability. After tissue dissociation, cell counting is performed using techniques such as the hemocytometer or flow cytometry.

Automated Cell Counting

Automated cell counting technologies utilize advanced imaging, optical sensors, or impedance-based techniques to detect and count cells with minimal user intervention. Examples include automated cell counters like the Luna-II or Countess II FL, flow cytometers, and spectrophotometry-based systems. [8] These technologies are particularly valuable in high-throughput settings, enabling researchers to analyze large numbers of samples quickly and efficiently. However, one significant challenge is the issue of cell clustering, which can cause the technology to give inaccurate results. This issue is common in suspension cultures or samples with poor dissociation, where individual cells are not well separated.

Flow Cytometry

Flow cytometry is another method to analyze the physical and chemical characteristics of individual cells or particles as they flow in suspension through a laser beam. In flow cytometry, cells are labeled with fluorescent dyes or antibodies with fluorescent markers that bind to specific cellular components, such as surface proteins or DNA.[9] As the cells pass through the flow cytometer's laser, they scatter light and emit fluorescence, which is detected by specialized sensors. The data collected includes parameters such as cell size, granularity, and fluorescence intensity, allowing for detailed information on the cell.

Flow cytometry is very advantageous for cell counting as it can count cells extremely fast and can also identify cells. However, this technique requires specialized equipment, skilled operators, and careful sample preparation.[10]

Proposed Solution: Our Approach

The goal of this project was to develop a machine-learning model capable of accurately identifying and counting cells from grayscale microscopy images. To achieve this, we implemented a Convolutional Neural Network (CNN) that extracts meaningful features from the images and performs regression to predict the cell count. This document provides an overview of the approach, from data preprocessing to model evaluation.

We utilized a Convolutional Neural Network (CNN) due to its ability to effectively process image data and identify spatial patterns. The model's architecture was designed to balance computational efficiency with predictive accuracy. Convolutional layers were employed to extract local features such as edges and textures. Pooling layers reduced the spatial dimensions, focusing on the most relevant features while minimizing computational overhead. Fully connected layers combined the extracted features to make final predictions, with the output layer providing a single value for the predicted cell count. Dropout layers were added to prevent overfitting and improve generalization. The final model used the Adam optimizer for efficient weight updates and minimized the Mean Squared Error (MSE) loss, ensuring accurate regression.

Data preparation followed a systematic preprocessing pipeline. First, ground truth images were loaded from a directory where filenames encoded the actual cell counts. The images were read in grayscale format to ensure uniformity and reduce computational complexity. Pixel values were

normalized to the [0, 1] range by dividing by 255, which sped up model convergence and improved numerical stability. To ensure compatibility with the CNN, a channel dimension was added to the images, resulting in a shape of (height, width, 1). The dataset was then split into 80% training and 20% testing sets, with an additional 20% of the training data reserved for validation during training.

While CNNs inherently perform feature extraction during training, we applied additional techniques to enhance the learning process. Images were resized to a uniform size to ensure consistency across the dataset and reduce computational demands. Future work may incorporate data augmentation techniques, such as rotations, flips, and brightness adjustments, to artificially expand the dataset and improve model robustness.

The input images were passed to the CNN with a fixed shape of (height, width, 1). Mean Squared Error (MSE) was chosen as the loss function to penalize large differences between predicted and actual cell counts. Mean Absolute Error (MAE) was tracked as an evaluation metric to provide an interpretable measure of prediction accuracy. The model was trained over 20 epochs with a batch size of 32, using 20% of the training data for validation. This allowed us to monitor the model's performance on unseen examples during training.

The model's performance was evaluated on the test set, where predictions were compared to the actual cell counts. Two visualizations were created to better understand the results. A bar graph compared the true cell counts to the predicted values for test samples, visually showcasing the model's accuracy. Additionally, a line graph demonstrated the reduction in Mean Squared Error over the training period, highlighting the model's learning process. Together, these evaluations provided insights into the effectiveness of the CNN in detecting and counting cells from microscopy images.

Results and Evaluation:

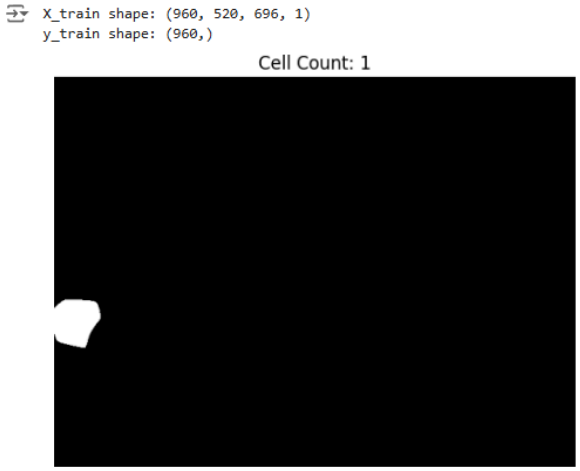

Image 1: Example image of cell count correctly being parsed from the filename.

To ensure the images were correctly loaded into Google Colab, an example image (image 1) was shown with the cell count listed above from the filename. The goal of the CNN is to:

Extract Features: Identify patterns or features (e.g., shapes, edges) that help determine the number of cells in an image.

Perform Regression: Use these features to output a continuous value representing the predicted cell count for the image.

The purpose of the Input Layer Purpose is to define the shape of the input images. Here, the images are grayscale (1 channel) with a fixed height and width. The CNN processes 2D-pixel data and learns spatial relationships.

The purpose of the convolutional layer is to apply filters (small sliding windows) to the image to extract features such as edges, corners, or textures.

The purpose of the Pooling Layers is to reduce the spatial dimensions of the feature maps, retaining the most important information while reducing computational complexity.

- Reasoning: Prevents overfitting by simplifying the data. It makes the model robust to small shifts or distortions in the image.

Flatten Layer

- Purpose: Transforms the 2D feature maps (from convolution and pooling layers) into a 1D vector.
- Reasoning: This vector representation is fed into the dense layers for prediction.

Fully Connected (Dense) Layers

- Purpose: The first dense layer learns high-level representations of the image features.The final dense layer outputs a single continuous value representing the cell count.

Dropout Layer

- Purpose: Prevents overfitting by randomly "dropping out" 50% of the neurons during training.
- Reasoning: Ensures the model doesn't rely too heavily on specific neurons and generalizes better to unseen data.

Loss Function

- Purpose: Measures the average squared difference between predicted and actual cell counts.
- Reasoning: In regression tasks, minimizing MSE leads to predictions that are closer to the ground truth.

Optimizer

- Purpose: Updates the weights of the CNN during training to minimize the loss function.
- Adam Optimizer: Combines the benefits of two other optimizers, AdaGrad and RMSProp, for efficient and effective training.

Metrics

- Purpose: Tracks the mean absolute error (MAE) during training, providing an interpretable measure of prediction accuracy.

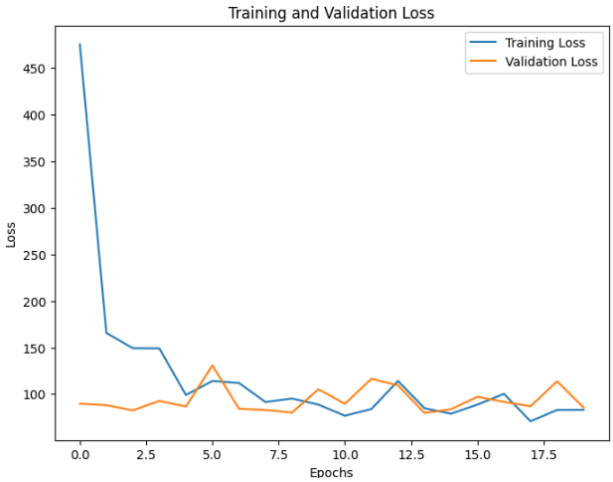

Figure 2: Training Loss and Validation Loss over 20 Epochs

Twenty different training sessions, or Epochs were used to train the CNN.

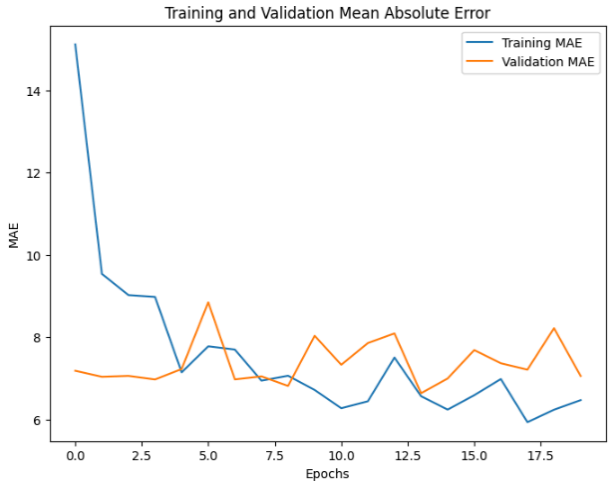

Figure 3: Training Mean Absolute Error and Validation Mean Absolute Error over 20 epochs.

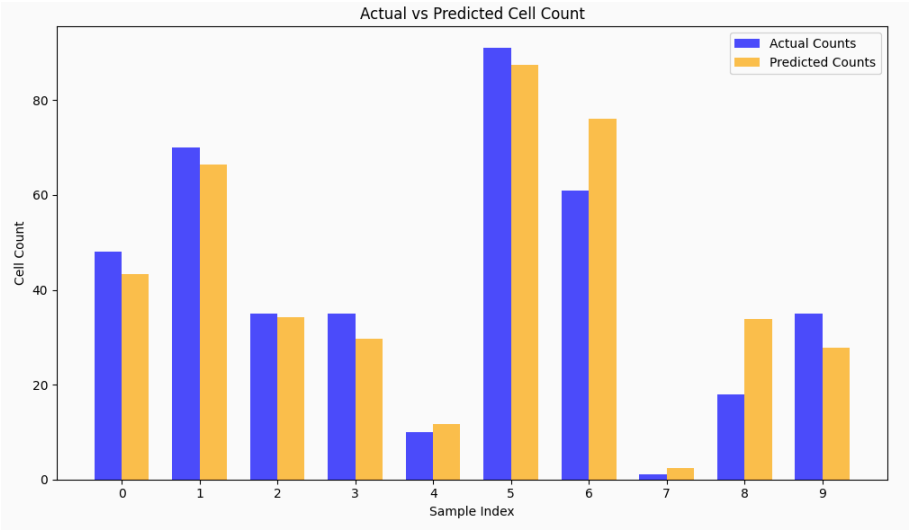

Figure 4: Nine sample guesses versus actual cell count

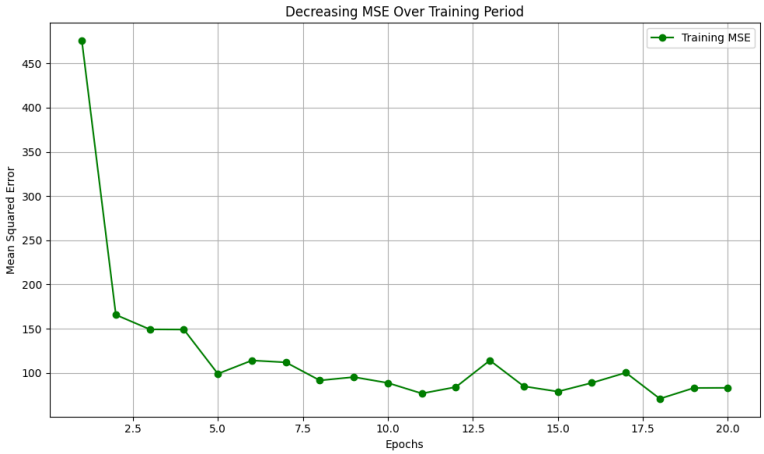

Figure 5: Mean standard error over 20 epochs

Practical Application:

This proposed project offers a solution for biomedical research, particularly in applications requiring precise and efficient cell counting, such as cancer and stem cell studies. By automating the cell counting process, this model significantly reduces the workload for scientists, making the process faster, less labor-intensive, and more consistent. Its scalability allows for the efficient processing of large datasets, which is important in drug development and clinical trials. This efficiency not only streamlines workflows but also makes the research process smoother, faster, and more cost-effective.

Additionally, because this model does not harm the cells, it enables the non-invasive study of cell proliferation dynamics by tracking cell counts at various stages of growth. This capability is particularly useful in drug development, where it can help researchers observe how cells respond to treatments, such as changes in their proliferation rates when compared to control groups.

References:

[1] W. Strober, "Trypan blue exclusion test of cell viability," *Current Protocols in Immunology*, 2001.

[2] A. E. Carpenter, et al., "CellProfiler: Image analysis software for identifying and quantifying cell phenotypes," *Genome Biology*, 2006.

[3] "CellProfiler: About," [Online]. Available: https://cellprofiler.org/about. [Accessed: Dec. 8, 2024].

[4] L. G. Luna, "Automated cell counting applications," *Manual of Histologic Staining Methods of the Armed Forces Institute of Pathology*, 1968.

[5] "ImmunoSpot Analyzers," [Online]. Available: https://immunospot.com/products/analyzers. [Accessed: Dec. 8, 2024].

[6] M. Dominici, et al., "Dissociation protocols for stem cell counting in tissues," *Stem Cells*, 2006.

[7] "Cell Dissociation Blog," [Online]. Available: https://www.akadeum.com/blog/cell-dissociation/?utm_source=chatgpt.com. [Accessed: Dec. 8, 2024].

[8] "Validation of three viable-cell counting methods: Manual, semi-automated, and automated," [Online]. Available: https://www.researchgate.net/publication/275104128_Validation_of_three_viable-cell_counting_methods_Manual_semi-automated_and_automated. [Accessed: Dec. 8, 2024].

[9] H. M. Shapiro, *Practical Flow Cytometry*. Wiley-Liss, 2003.

[10] L. A. Herzenberg, et al., "Flow cytometry and cell sorting," *Science*, 2002.