Image Processing Technique for Automated Viral Plaque Counting

Michael Moorman, Hood College

I. ABSTRACT

The viral plaque assay is an important procedure in virology. One step of this procedure includes the manual counting of viral plaques with the naked eye. This is often a time consuming and laborious process. A method is presented for automating viral plaque counting using image processing techniques and the OpenCV Computer Vision library. The method consumes an image of an entire platter of viral plaque samples and then outputs individual counts of each sample in the platter. The image is captured using a commercial flat bed scanner. Experimental results show that the method is up to 90 percent accurate in comparison to trained human counts. The program produces results for a single image in about one second. Compared to commercial colony counter systems, the proposed method is extremely economical; the only hardware required by the method is a personal computer and a flatbed scanner. Finally, an open source implementation of the program is provided, which includes an optional graphical user interface.

II. INTRODUCTION

The viral plaque assay is an important and commonly used procedure in biology; however, a stage of the process requires that a person manually count the instances of plaque structures . The work presented here attempts to accurately count mammalian viral plaques using image processing techniques. The image below is a typical example of an input image.

The viral plaque assay remains to be an important procedure in the field of biology. A common example of the procedure begins by preparing several dilutions of a virus stock. Petri dishes are also prepared, arranged in 4x6 platters, containing a thin monolayer of susceptible animal tissue sample. The dishes are then innoculated using the various virus dilutions. After inncoulation, the dishes are topped off with a special agar that limits viral expansion to only neighboring cells. Each sample is allowed to incubate for some amount of time, allowing the virus to attack the cells In the monolayer. Over time, the virus gradually destroys neighboring cells,
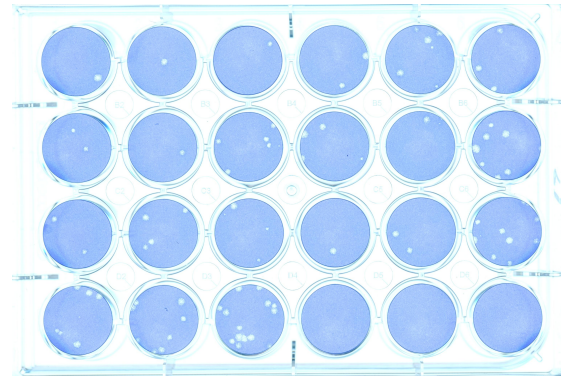

Figure I. A typical example of a viral plaque image

creating visible structures also known as viral plaques. The visibility of these plaques are sometimes enhanced with the usage of special dyes. Once the plaques have grown large enough to be seen by the naked eye, the titer of the original virus stock is discovered by observing the number of plaque-forming units (PFU) per milliliter present in each plate.

The last stage of this procedure can be particularly laborious . Counts of the number of plaque formations on each plate are manually tallied by hand. In a typical case, a sample can contain over 100 viral plaques. It is easy to see how tallying an entire platter of samples often consumes several minutes of someones time.

Automated counting of viral plaques via image processing techniques has been proposed and investigated many times before, and there are several commercial products on the market that manufacture colony counting systems. [CITE]. There is a lucrative market for automated colony counting, and several companies manufacture colony counting systems. Some of these systems can cost thousands of dollars and are often both closed source and not economically feasible. [CITE] . Other studies have approached automated viral plaque counting before and have tried a number of different approaches. Some of these include use of a Watershed Transform [CITE], Distance transform[CITE], Hough transform[CITE], parameter identification and fuzzy logic. Of these works, However, they did not feature the ability to process an

entire platter of samples.

At present time, there is not a low cost and accurate colony counting system available that enjoys wide spread adoption and that is able to accommodate an entire platter of samples. This is a primary motivation for the work presented here. In addition, the performance of some of these works was poor. In the case of [CITE], the system consumed ten seconds of wall clock computational time to analyze a single viral plaque plate.

Accurately counting plaques in these types of images presents several challenges. Perhaps the most difficult challenge that any solution must address is how to deal with the noise in the image. How does one go about discerning the difference between viral plaques and the frequent noisy particles in the agar of the plates? This is particularly problematic when dealing with small plaques, as there signal to noise ratio is very small in that case.

OpenCV is an open source computer vision library available from http://sourceforge.net/projects/opencvlibrary. The library is written in C and C++ and runs under Linux, Windows, and Mac OS X. It provides an application interface (API) that this study uses to implement the proposed method.

III. METHODOLOGY

The proposed method is implemented as a computer program written in C++ and uses the OpenCV [CITE] image processing API to perform image analysis.

A. User Supplied Parameters

The user supplies the program with an image of a platter of viral plaques, and also uses three parameters that are supplied by the user.

1) Plate radius: The radius of each plate, given in pixels.

2) Minimum Plaque Radius: The minimum radius of a viral plaque structure, given in pixels.

3) Maximum Plaque Radius: The maximum radius of a viral plaque structure, given in pixels.

The method uses these parameters and a series of procedural morphological image transformations to ultimately produce viral plaque counts.

B. Image Aquisition

Image acquisition is the first stage of the method. A flat bed scanner is used to capture an image of a platter of viral plaque samples, and they are encoded into an image file.

C. Segmentation

Segmentation in the next stage of the method. The only interesting parts of the image lay within each welll. Segmentation is an essential step that attempts to create a mask that shadows every part of the image that is not with in the barriers of a plate.

Perhaps the most difficult problem that segmentation seeks to resolve is the platter orientation problem. Image acquisition makes no guarantees as to the position of the platter with relation to the scanner surface. Platters may be butted up against the top left corner of the scanner, have a slight offset on any axis, or have any slight orientation about them. A global segmentation approach overcomes these challenges. This approach is summarized below.

1) Grayscale the image.
2) Use Otsu's thresholding method [CITE] to transform the image into a binary image. This is depicted in figure II.
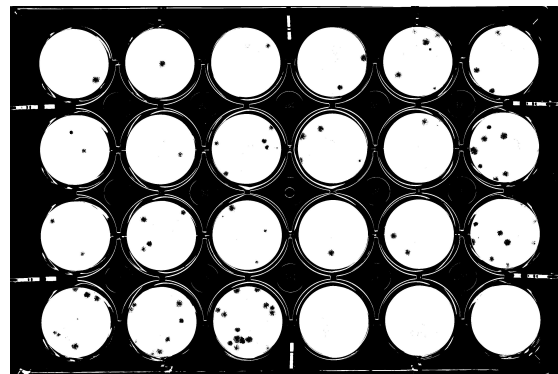

Figure II.   Otsu thresholding

3) Perform a morphological dilation on the image that is sufficient enough to close viral plaque holes. This is shown in figure III.
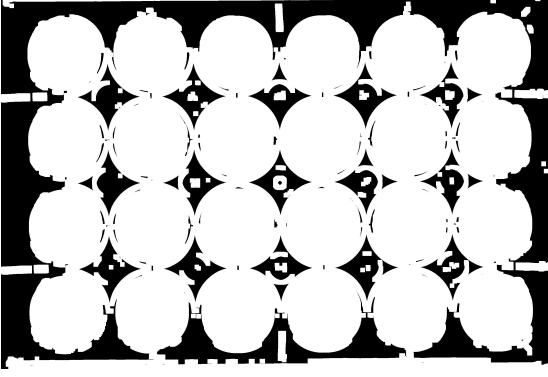
Figure III.   Use of dilation to fill holes.

Figure V.   Use of flood fill

4) Aggressively perform several iterations of morphological erosion on the image until the number of artifacts that remain on the image equal the number of wells on the platter. This is shown in figure IV.
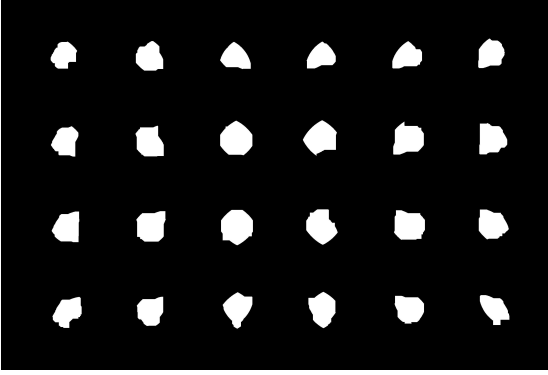
6) Recalculate the centroids of the new artifacts. Draw a circle at these origins of radius specified by the plate radius parameter. This is shown in figure VI.
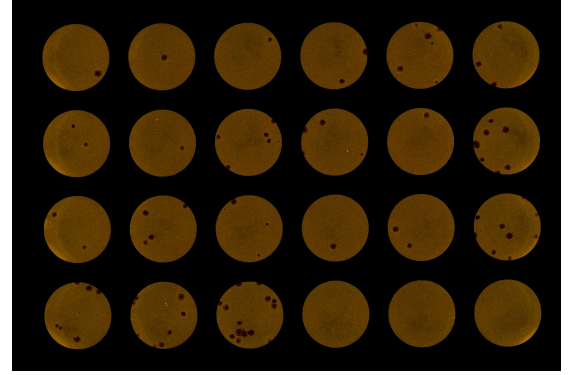

Figure VI.   Use of plate radius parameter

Figure IV.   Use of massive erosion

Segmentation produces clean regions of interest for each plate in the platter. This part of the method is very effective at identifying regions of interest. It exploits the observation that images of platters used for viral plaque studies tend have a similar overall shape. The method expects that the image artifacts that represent the agar in the plates are the most predominant features of the overall image. Thus, the image can be reliably eroded until only these features remain.

Segmentation also exploits the fact that petri dishes used in viral plaque studies are machined to be perfectly round[CITE]. This knowledge combined with the plate radius parameter that the user provides allows this phase of the method to produce clean regions of interest, which is vastly important to the success of later processing stages.

5) Calculate the center points of each remaining artifact. Use these points as seeds for a flood fill [CITE] operation that grows the region outward from each of these points on an RGB thresholded rendering of the original image. Aggressively perform several iterations of morphological erosion on the image until the number of artifacts that remain on the image equal the number of wells on the platter. This is shown in figure V.

D. Plaque Structure Counting

Next remains the task of counting each viral plaque in each plate. Given the often noisy nature of the agar and

viral plaques in the plate, its difficult to devise a method that is accurate for all varieties of input. Viral plaques can be very small, very large, malformed, or even overlapped on top of one other. Nevertheless, the plaque counting method attempts to be simple and accurate for the most commonly occurring instances of viral plaques. It is summarized below.

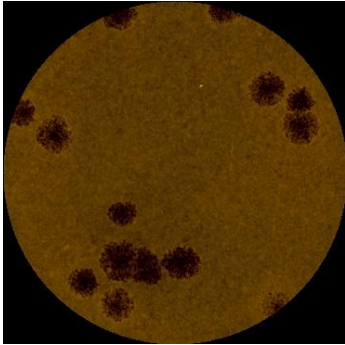1) Let a masked region of interest be image A. This is a color image containing a single plate.

Figure VII.   Image of a single well, with uninteresting regions masked out

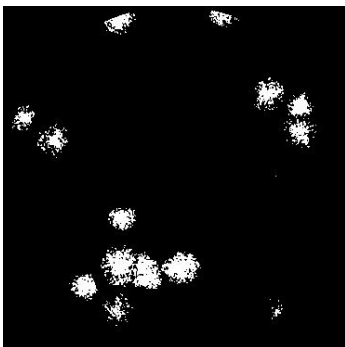2) Grayscale and then apply Otsu thresholding upon image A.

Figure VIII.   Otsu thresholding applied to image A

3) Let images B,C, D, and E be zeroed copies of image A.
4) On image B, render an eroded image.

   a) Render a morphological open [CITE] operation on image B using a 2x2 rectangular shaped kernel. Choose the number of iterations based on a constant derived from the maximum plaque radius parameter.
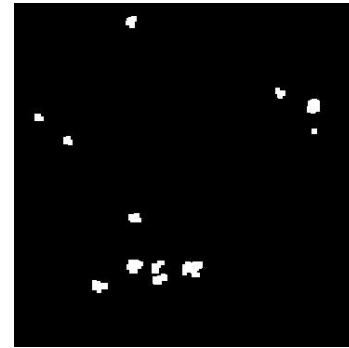

Figure IX.   Morphologically eroded rendering of image B.

5) On Image C, render a dilated image.

   a) Find every contour in A. If the contour area is smaller than a constant derived from the minimum plaque area, remove it from the image.
   b) Dilate C using a 2x2 rectangular shaped kernel. Choose the number of iterations based on a constant derived from the minimum plaque radius parameter.
   c) Perform an operation similar to the MATLAB imfill() [CITE] on image A to close any internal holes in the contours.

Figure X.   Morphologically dilated rendering of image C.

6) Merge images B and C into a new image A using a binary image merge of the form
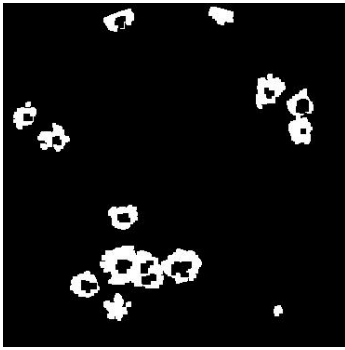
```
A = ~B & C
```

Figure XI.   Images B and C merged together

7) Obtain the hierarchy of contours in A. Let the total number of contours be determined by the following pseudocode:

```
minP = min plaque area
maxP = max plaque area
for contour C in outerContours:
  if( area(C) < minP):
    continue;

  if( area(C) > maxP:
    totalContours += 1;

  totalContours +=
    max(1, innerContourCount)
```

The plaque counting method attempts to find a balance between finding feint and sparse plaques VS. properly dividing and detecting dark, prominent plaques. Feint plaques are typically discovered through analysis of the dilated image (Image C). Dilation effectively joins plaque clusters and forms a cohesive, filled areas. The erosion rendition of the plate is focused on discerning prominent plaque structures. Without an erosion step, a cluster of viral plaque structures are at risk of being inaccurately tallied as a single plaque. Combining the dilated and eroded versions of a plate image takes advantages of both morphological transforma ions. Small, feint plaques are emphasized and large clustered plaques are separated. Both transformations scale with the user supplied parameters maxPlaqueRadius and minPlaqueRadius.

The final step of the per plate counting routine considers the max and minPlaqueRadius parameters when tallying total plaque counts.

IV. EXPERIMENTS

For these experiments, a viral plaque images are logically grouped as a study. In a given study, platters are prepared with similar parameters. All of the viral plaque platters in a study have been allowed to incubate for about the same amount of time. Each study consists of platters that use the same or very similar innoculation techniques. The also use similar agar, tissue mediums, and viruses for experiments. This organization scheme allows for a pairing of one set of program parameters to be attached to all the images for a given study. Likewise, parameters are configured for the program on a per study basis.

Truth data for each viral plaque platter was obtained by manually counts. These results were compared with data produced by the computer program. Accuracy was measured by finding the sum of errant counts for each plate and then dividing that sum by the total truth data count for the platter. Errant counts for each plate were determined by finding the absolute difference between a count produced by the program and the truth data for a given plate.

Each image was aqcuired using an Epson Perfection 799 flat bed scanner, configured to capture at 24 bit color mode and 600 dots per inch. In each experiment, the plates were prepared using a .5% Methyl Cellulose overlay in lieu of agar. A .4% solution of Crystal Violet was used as a dye as well. The Plates used in each study were Costar/Corning 24 well Tissue culture plates.

A. Study One

In the first study, African Green Monkey Kidney Cells Cell strain Vero E6 were infected with a Vaccinia Virus Western Reserve strain. Seven sample images were captured and sent to the program for processing. Each plate contained 24 wells, and a total of 168 data points were compared with corresponding truth data points.
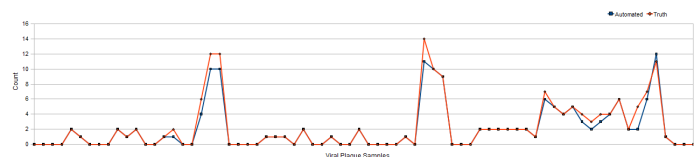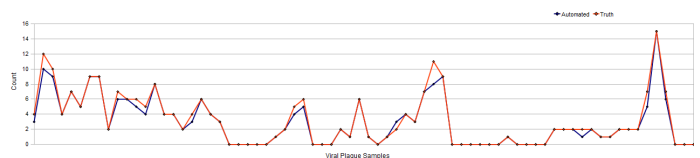

Figure XII.   Study one results

Figure XIII.   Study one results cont.

Total error for study one : .09

B. Study Two

In the second study, African Green Monkey Kidney Cells Cell strain Vero E6 were again infected with Vaccinia Virus Western Reserve strain. Six sample images were captured and sent to the program for processing. Each plate contained 24 wells, and a total of 144 data points were compared with corresponding truth data points.
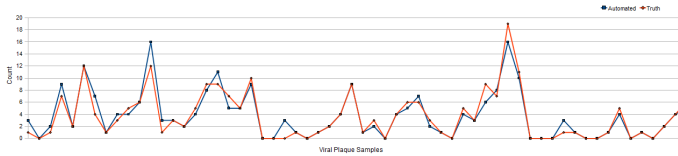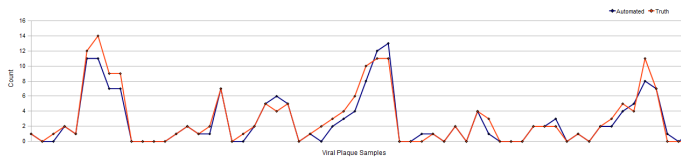

Figure XIV.   Study 2 results

Figure XV.   Study 2 results cont.

Total error for study two : .20

C. Study Three

In the second study, African Green Monkey Kidney Cells Cell strain Vero E6 were again infected with Monkey Pox Zaire strain. Three sample images were captured and sent to the program for processing. Each plate contained 18 wells, and a total of 52 data points were compared with corresponding truth data points.
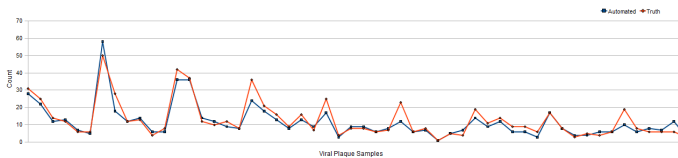

Figure XVI.   Study three results

Total error for study three: .20

V. DISCUSSION