Abstract

At Kansas State University, a new project called BreadPheno uses novel advances in image processing and machine vision to deliver transformative mobile applications through established breeder networks. Building on the success of Field Book, a collection of user-friendly mobile applications for field-based high-throughput phenotyping (HTP) will be developed and deployed. As a part of this effort, novel image analysis algorithms are being developed to model and extract plant phenotypes. This paper describes an extension of the traditional watershed segmentation algorithm to rapidly count and characterize the seeds in an image captured with a mobile device. The new approach can be used to characterize a wide variety of crops for high-throughput phenotyping.

Keywords: Euclidean distance map, high-throughput phenotyping, mobile application development, ultimate eroded points, watershed segmentation.

1 Introduction

A new project at Kansas State University, called BreadPheno, seeks to converge novel advances in image processing and machine vision to deliver transformative mobile applications through several established breeder networks. Building on the success of Field Book (www.wheatgenetics.org/field-book) [1], a user-friendly mobile app for field-based high-throughput phenotyping (HTP) will be built and deployed. Novel image analysis algorithms will be developed to model and extract plant phenotypes. A robust development pipeline will be assisted by both real-time field testing by breeding collaborators and middle school and high school students using the app to explore plant growth and quantitative differences under genetic control. To ensure both immediate, broad deployment and functionality on a diverse set of crops, breeder networks for cassava and wheat will be engaged, providing a diverse set of target plant phenotypes, environments, breeding programs and working cultures.

Dramatic increases in the speed and ability to collect precision phenotypic data are needed to decipher plant genomes and accelerate plant breeding. Over the past decade, the availability of genomic data has exploded while the methods to collect phenotypes have made minimal advancements. This has led to a dramatic imbalance in data sets connecting genotype to phenotype and highlighting phenotyping as the remaining major bottleneck in plant breeding programs. This project seeks to advance the field of 3D graphics and modeling, data mining and deep learning through integration of simultaneous ground truth phenotypic measurements and imaging with mobile technology. By combining data from multiple research programs with ground-truth breeder knowledge, this project will lay the foundation for collecting training set data that can subsequently be used to extract and quantify complex phenotypes using deep-learning.

Food and nutritional security is a grand challenge in the coming decades. The global population will increase to over 9 billion and food demand will grow by more than 50%. To address this challenge, novel advancements to leverage genomic information and expedite the improvement of plant varieties are needed. While genomic information has become inexpensive and readily available, the complementary phenotypes needed to understand the function of plant genomes and make selections in breeding programs has remained static, with insufficient development, particularly for phenotypes collected from field trials in breeding programs.

By focusing on novel algorithms delivered through mobile apps, innovative phenotyping tools can be rapidly deployed through readily available and highly penetrant mobile technology. This approach will enable rapid dissemination and broad usability. Collectively equipping thousands of breeders around the world with tools for rapid collection, processing and analysis of complex phenotypes will provide the foundation for increasing genetic gain that will ultimately result in improved productivity, food security, nutrition and income of smallholder farmers and their families in developing countries.

Watershed segmentation algorithms are used to automatically separate particles in an image that touch. The algorithm calculates the floating point Euclidean distance map (EDM) of each pixel in the image, where Euclidean distance is defined in the usual way as the closest Euclidean
distance to any background pixel [2]. Then, it finds the ultimate eroded points (UEPs) which are the peaks or locally maximal connected components of the EDM. It then dilates each of the UEPs as far as possible, either until the edge of the particle (the background) is reached, or the edge touches a region of another growing UEP region. Watershed segmentation works best for smooth, convex objects that don’t overlap very much. In the classical algorithm, the regions are flooded in different directions on each step of the algorithm.

This paper describes several extensions that extend the traditional watershed segmentation algorithm for high-throughput phenotyping. The first step is to apply a color threshold to the image to generate a binary image, and apply the traditional watershed segmentation algorithm. The new algorithm adds a new step to compute the number of seeds based on the contour sizes and recursively split the contours that represent multiple seeds. Section 2 describes the current state of the art, and Section 3 describes our new extensions. Finally, Section 4 concludes the paper.

2 State of the Art

Rapid collection and analysis of large amounts of phenotypic data is known as high-throughput phenotyping (HTP). Image-based methods for computing phenotypic measurements provide a promising solution for high-throughput systems in plants. There have been many successful applications of image analysis to plant phenomics including seed measurement [3], time series growth measurements of roots [4], plant growth and biomass in relation to abiotic stresses [5, 6], plant morphology [7], and disease severity [8, 9]. However, these applications of image-based HTP have been largely concentrated on laboratory or controlled environment facilities. While providing highly accurate phenotypes, transferability of experiments from controlled environment faculties to field conditions remains limited. Moreover, the limited capacity and substantial infrastructure needed to develop controlled environment HTP platforms makes this approach unpromising for breeding programs that currently manage population sizes much larger than even the world’s largest facilities.

Field-based phenotyping remains the major bottleneck in most breeding and genetics programs, including cassava and wheat. As recognized by the breeders, this bottleneck can be overcome by adopting and/or developing innovative, high-throughput, and accurate hand-held HTP tools that would greatly increase the ability of breeders to generate useful data more cheaply and rapidly. As such, there is considerable willingness and motivation in the breeding community to test and adopt new technologies that can make the breeding program more efficient.

We have developed and deployed several user-friendly mobile apps that have been adopted by hundreds of breeding programs around the world. They are available online at www.wheatgenetics.org/apps. Our apps have been designed to be standalone, simple, intuitive, and effective. They are delivering a significant increase in breeding productivity and are the foundation for the current project. These tools focus on simplifying and digitizing data collection by focusing on procedures that are common to all breeding programs. Current development is complete and a beta version has been released for ‘1KK’, an app that can give accurate size and shape parameters for seeds and tubers/roots. With the namesake of 1KK for capturing thousand (1K) kernel (K) traits, the app extends to any size and scale of ‘seeds’ (Figure 1). The app combines the image analysis library OpenCV [10, 11] with a USB scale to capture both an image of a collection of seeds and a weight measurement. Seeds or roots are spread on a colored mat that includes reference circles of a known size to convert pixels to absolute size in mm.

Integrated image processing scripts identify individual seeds/tubers/roots to calculate the length and width while attempting to count the total number. Average weight (thousand kernel weight) is calculated by dividing the total weight determined by the scale. Raw data are stored within an internal database and parameter averages are also calculated and stored on a per sample basis. Data are exported as a flat file and can be uploaded to a database.

The most common inquiry about developing field-based software is in regards to the hardware and how well it performs in field conditions. While we currently rely on Android tablets as the main data collection device, we have started working with alternative devices with eInk screens that are easily visible in the sun. Moreover, these devices are less expensive than standard tablets and have a long battery life that makes it possible to collect data for days at remote locations without the need to recharge the tablet. We have tested several devices but these have proved to be inadequate. Finding a device that solves the outdoor visibility issue and modifying Field Book to take full advantage of its features will encourage many breeding programs to adopt field-based data collection software. During this project, we will continue testing eInk, existing smartphones such as the Nokia Lumia running Windows, and alternative mobile devices for the most suitable consumer grade hardware available.

Figure 1: Wheat and cassava roots in 1KK app.
Figure 1 shows an example of the current 1KK app screen after capturing and image of wheat on the left or cassava roots on the right.

### 3 Watershed Algorithm Extensions

In addition to size, it is also important to accurately count the number of wheat seeds in a single image. In this section, we describe a novel extension to a well-known algorithm to accurately count the number of wheat seeds.

The first step is to capture an image with a known background and reference circles of a fixed size as shown in Figure 1. For example, consider the image shown below in Figure 2.

![Figure 2: Captured wheat image.](image)

The first step is to separate the three regions of interest: the green background, the blue reference circles, and the golden grain seeds. This can be completed using the ImageJ algorithm for Image + Adjust + Color Threshold [12], or a few lines of code in OpenCV [9, 10]. Note that by using the color space HSB, we can easily separate out the wheat seeds by selecting a hue range of 0-50, and leaving saturation and brightness as 0-255. Most of the pixels are green and denote the background. A few are golden or orange, denoting the wheat seeds, and a few are blue, denoting the reference circles. Once the image is filtered, the resulting mask is shown in Figure 3.

![Figure 3: Binary threshold image and resulting list of particles from traditional algorithm.](image)

The mask can be converted to binary in ImageJ using Process + Binary + Make Binary. Then, the original binary watershed algorithm based on the Euclidean Distance Map can be invoked using Process + Binary + Watershed. This is the classic flooding watershed segmentation algorithm by Soille and Vincent [13]. Other algorithms could be used as well. This algorithm can also be invoked by rewriting the code for Android platforms in OpenCV. The user can select Analyze + Particles to see all of the particles or regions remaining after flooding. In this case, there are a total of 266 particles, as shown below in Figure 3.

Since the actual number of wheat seeds is 267, and the traditional algorithm found 266, on first blush, this appears to be a very good result. However, if we look closer, we see that some of the single seeds are displayed as two seeds, and some of the nearby seeds are displayed as a single seed. The results can be viewed in tabular form or as an image, as shown in Figure 4. The minimum particle size is 334, the maximum particle size is 4408, the median particle size is 1265, and the mean is 1378.3.

![Figure 4: Traditional algorithm contours.](image)

From this run and observation, we can conclude that the typical wheat seed has a size of roughly 1378 pixels. So, the largest particle shown in Figure 4, particle 8, actually denotes several seeds. Since particle 8’s size (4058) divided by the mean size (1378) is roughly 2.9, and divided by the median is 3.2, we can correctly conclude that that particle 8 actually represents 3 seeds. Likewise, particle 40 is size 748 and particle 42 is size 467, and they actually represent a single wheat seed together.

The problem results because the Euclidean Distance Map of particle 8, shown in Figure 5, does not indicate any separation between the wheat seeds. Consequently, only a single connected component of Ultimate Eroded Points (UEPs) is computed by the classic watershed algorithm with default thresholds [13].
Counting and Splitting

What is needed is a way to effectively count, and then split or merge particles that do not represent a single wheat seed. To accurately count the number of seeds, the extension to the algorithm is actually quite simple and efficient. Set the estimated seed size \((\text{est}_\text{size})\) to be the mean size. For particle regions whose size is greater than \(1.5 \times \text{est}_\text{size}\), compute the estimated number of seeds as:

\[
\text{est}_\text{seeds} = \text{round}(\text{particle}_\text{size/est}_\text{size})
\]

Table 1: Seed estimate (partial list, sorted by size):

<table>
<thead>
<tr>
<th>Particle</th>
<th>Area</th>
<th>X</th>
<th>Y</th>
<th>Area / Est. Size</th>
<th>Est. Seeds</th>
</tr>
</thead>
<tbody>
<tr>
<td>104</td>
<td>4408</td>
<td>504.914</td>
<td>868.929</td>
<td>3.198143</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>4058</td>
<td>926.472</td>
<td>196.346</td>
<td>2.944207</td>
<td>3</td>
</tr>
<tr>
<td>235</td>
<td>3686</td>
<td>2009.27</td>
<td>1630.978</td>
<td>2.674309</td>
<td>3</td>
</tr>
<tr>
<td>122</td>
<td>3459</td>
<td>441.198</td>
<td>959.789</td>
<td>2.509613</td>
<td>3</td>
</tr>
<tr>
<td>96</td>
<td>3429</td>
<td>481.351</td>
<td>815.862</td>
<td>2.487847</td>
<td>2</td>
</tr>
<tr>
<td>157</td>
<td>2899</td>
<td>1489.136</td>
<td>1134.785</td>
<td>2.103316</td>
<td>2</td>
</tr>
<tr>
<td>26</td>
<td>2778</td>
<td>767.602</td>
<td>345.492</td>
<td>2.015526</td>
<td>2</td>
</tr>
<tr>
<td>112</td>
<td>2638</td>
<td>805.904</td>
<td>923.841</td>
<td>1.913952</td>
<td>2</td>
</tr>
<tr>
<td>84</td>
<td>2424</td>
<td>1522.054</td>
<td>761.502</td>
<td>1.758688</td>
<td>2</td>
</tr>
<tr>
<td>131</td>
<td>2093</td>
<td>799.921</td>
<td>989.804</td>
<td>1.518537</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>1979</td>
<td>623.968</td>
<td>196.807</td>
<td>1.435827</td>
<td>1</td>
</tr>
</tbody>
</table>

For this example, we only fail on one estimate for particle 122 as shown in Table 1. So, even though the actual number of seeds is 267, we estimate a total of 268, for an error of less than 0.5%. If we were to use the median size to be the estimate, we estimate a total of 269, for an error of 2/267, still less than 1%.

Naturally, the accuracy of the estimate depends on having a relatively large number of single seeds in the image. Then, the mean size or median size is a good estimate of the actual seed size. We could also use historical info to estimate the average seed size.

To identify the edges between seeds, we look for points where the contour is no longer convex. For each consecutive pair of edges, compute the cross product of the vectors defined by the edges adjacent to a given point \((x[i], y[i])\) by:

\[
\text{cross product} = ((x[i]-x[i-1])*(y[i+1]-y[i])-y[i-1]+y[i-1] = (x[i+1]-x[i])
\]

If the cross products are all positive or all negative, then the particle contour is convex. To minimize the effects of local noise in the contour points, it is helpful to select points that are farther apart and parameterize the cross product definition with an index offset parameter \((\Delta)\):

\[
\text{cross product}(\Delta) = ((x[i]-x[i-\Delta])*(y[i+\Delta]-y[i])-y[i-\Delta]+y[i-\Delta] = (x[i+\Delta]-x[i])
\]

where \(\Delta \geq 1\). All arithmetic is performed in a modular fashion. Empirically, we found that a parameter value of 4 works well for the images under study, with images of size 2520x1886. Conceptually, the cross product determines if the difference in slopes between adjacent lines connected to a given contour point changes in sign, then the point is an inflection point.

If we detect some inflection points, they generally denote the border points between seeds, shown in green in Figure 5 and in Table 2 which is sorted by cross product. The remaining contour points all have negative cross products.

Table 2: Inflection points in contour 8:

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>Cross Product</th>
</tr>
</thead>
<tbody>
<tr>
<td>902</td>
<td>183</td>
<td>106</td>
</tr>
<tr>
<td>949</td>
<td>183</td>
<td>100</td>
</tr>
<tr>
<td>951</td>
<td>207</td>
<td>52</td>
</tr>
<tr>
<td>914</td>
<td>210</td>
<td>43</td>
</tr>
<tr>
<td>916</td>
<td>157</td>
<td>12</td>
</tr>
<tr>
<td>949</td>
<td>213</td>
<td>8</td>
</tr>
</tbody>
</table>

It is important to select the correct points – points that are adjacent to the same seeds. Once identified, one solution to the problem is add lines connecting these points as background points in the binary image, and then just re-run the algorithm. In this example, the first two points, (902, 183) and (949, 183), and the middle two points, (951, 207) and (914, 210), should be used, and we obtain the result shown below in Figure 6. Note that the single particle 8 is now split into 3 particles.

In general, we can’t just use the two contour points with the largest cross products to denote a line. Instead we project the line from the largest cross product at (902, 183) across the seed using the slope of the line between the given point and the point which is generated by averaging the adjacent contour points. For this example, the slope is estimated as 0.167, and the line is computed as \(y = 0.167 \times x + 33.333\). Then, the remaining inflection point closest to the line is selected. For the second point (949, 183), the
distance to the line is 7.83; for the third point (951, 207), the distance to the line is 15.833, and it gets worse for the others. Thus, a line is drawn in the background color between the first two points, and the remaining points are considered in the same way, looking next at (951, 207).

Likewise, a merge can be accomplished by removing UEPs that caused a single seed to be split into two regions. For this case, we just augment the threshold algorithm to distinguish edges that are added due to hitting the background from edges that are added due to running into another small particle.

To verify the effectiveness of the algorithm across a broad range of images, we have performed an extensive experiment on a large set of images gathered using the 1KK app from different crops in different fields. We randomly selected images from those available, and achieved an error rate of less than 1% on all of the hundreds of tests performed to date.

The next step will be to conduct a performance evaluation under field conditions. We have also extended the algorithm to additional crops, such as soybeans, canola (as shown in Figure 8), cassava, potato, and silphium (as shown in Figure 10). Different hue frequencies are used based on the type of crop. Also, we plan to extend the algorithms to process images with different lighting conditions and automatically adjust the images based on camera angle using the known points to compute camera angles and project the images onto the plane. Some of the crops, such as silphium, will be more challenging due to their irregular shapes, as shown below in Figure 10.
4 Conclusions

In this paper, we described a new research project currently underway at Kansas State University, called BreadPheno. This project requires the use of novel advances in image processing and machine vision to deliver transformative mobile applications through established breeder networks. Building on the success of Field Book a collection of user-friendly mobile apps for field-based high-throughput phenotyping (HTP) are being developed and deployed. As a part of this effort, novel image analysis algorithms are being developed to model and extract plant phenotypes. This paper described a simple, new approach to extend the traditional binary watershed segmentation algorithm with merging and splitting steps to more accurately count seeds. By identifying conglomerates of several seeds and removing redundant data, we guide the algorithm to find a more accurate count and a more accurate image contour of the seeds. The new approach simply adds steps to recursively merge and split partial or conglomerate seeds found using the traditional algorithm. All of the code is implemented to run in both the desktop and mobile application environments. This will facilitate wide use in the K-12 classroom as we have shown in [14].

Acknowledgements

We greatly acknowledge the staff in the Department of Plant Pathology for providing the images used in this research.

This material is based on work supported by the National Science Foundation under NSF-BREAD Grant No. 1543958. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References


