**High-throughput method for computing maize ear, cob, and kernel attributes automatically from images**

Introduction

A well developed ear of maize may have a thousand kernels formed from as many flowers tightly clustered along and around the cob, the structural core of the female inflorescence (Kiesselbach, 1949).

A high-throughput, automated method for measuring the characteristics of ears and kernels would advance our understanding of the genetic mechanisms controlling these fundamental yield components, on the basic side, or our ability to evaluate yield potential in a breeding program on the practical side.

Yield measured as kernel mass per unit land area is the resultant of interdependent features of ears, cobs, and kernels that can be counted, measured, or weighed. These features, or traits, include kernel size, weight, and number per ear.

While modern harvesting equipment automatically measures grain yield on a plot level, yield component traits such as ear and kernel dimensions are typically measured by hand.

Digital image analysis provides a potentially automatable means for measuring seed traits that are yield components.

However, maize kernels are **more variable**. Some may be as wide as they are tall. Some are triangular, while others are nearly circular.

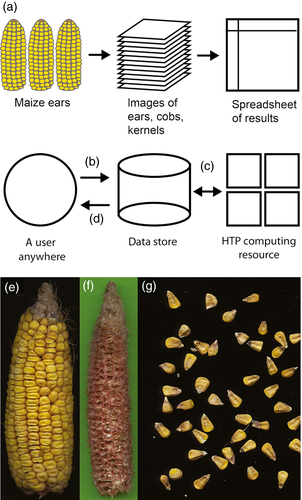
Therefore, different algorithms are needed to measure the standard features of maize kernels. If maize ears, cobs, and kernels could be automatically measured with greater objectivity and precision, more could be learned about the genetic bases of yield components and how to improve them using current and future maize genetic resources.

**Workflow**

The overall goal of the project reported here is to enable a researcher to extract useful quantitative information from large numbers of digital images of maize ears, cobs, and kernels and return this information in a suitable format to the researcher (Figure 1a).

To accomplish this goal, a high-throughput computational workflow that leverages community cyberinfrastructure was created. Currently, geographically distributed users upload their image files (Figure 1b) to a resource that functions as a drop box.

Successful image upload to this data store triggers analysis of each image as a separate computational job. Because each image is a separate job, the overall task is well matched with parallel computing.



**Figure 1**

Automated workflow enabling members of the community regardless of location to analyze maize ear, cob, and kernel attributes from images using the custom algorithms described in this report.

(a) The overall goal is to convert many images of many ears, cobs and kernels into a table of useful quantitative results.

(b) Users upload images to their data store accounts.

(c) Each image pulled from the data store is analyzed as an independent job on large-scale distributed computing resources. The processed results are pushed back to the user's account in the data store.

(d) Users retrieve spreadsheets containing processed results from their data store accounts.

(e) Example ear image.

(f) Example cob image against green background, which assists image processing.

(g) Subsection of an example image containing many kernels.

As an input, the analysis algorithms expect an eight-bit color **tag image format file** (TIFF) with a resolution of 1200 dots per inch.

The same arrangement is expected for cobs; the background should be green to ensure proper processing of the darkest red individuals. All of the data used in this report were obtained with flatbed document scanners though, in principal, other imaging devices that produce equivalent images should suffice.

[**Kernel length**](https://onlinelibrary.wiley.com/doi/full/10.1111/tpj.13320)

The space a single kernel occupies along the major axis of the ear, hereafter kernel length (KL), may correlate with interrelated yield components such as kernel weight, number, size, and shape.

Manual methods for estimating this trait include measuring the length of a segment completely overlying ten kernels near the midpoint of the ear axis and measuring directly the dimensions of an individual kernel with **calipers**.

The first step in our automated method for measuring this trait is to threshold the value channel from a hue-saturation-value (HSV) representation of the image and then apply standard binary morphology operations in order to mask the background, dust, debris, and scratches. The series of kernel boundaries along the ear axis was treated as a source of a periodic signal in a gray-scale representation of the ear thus masked.

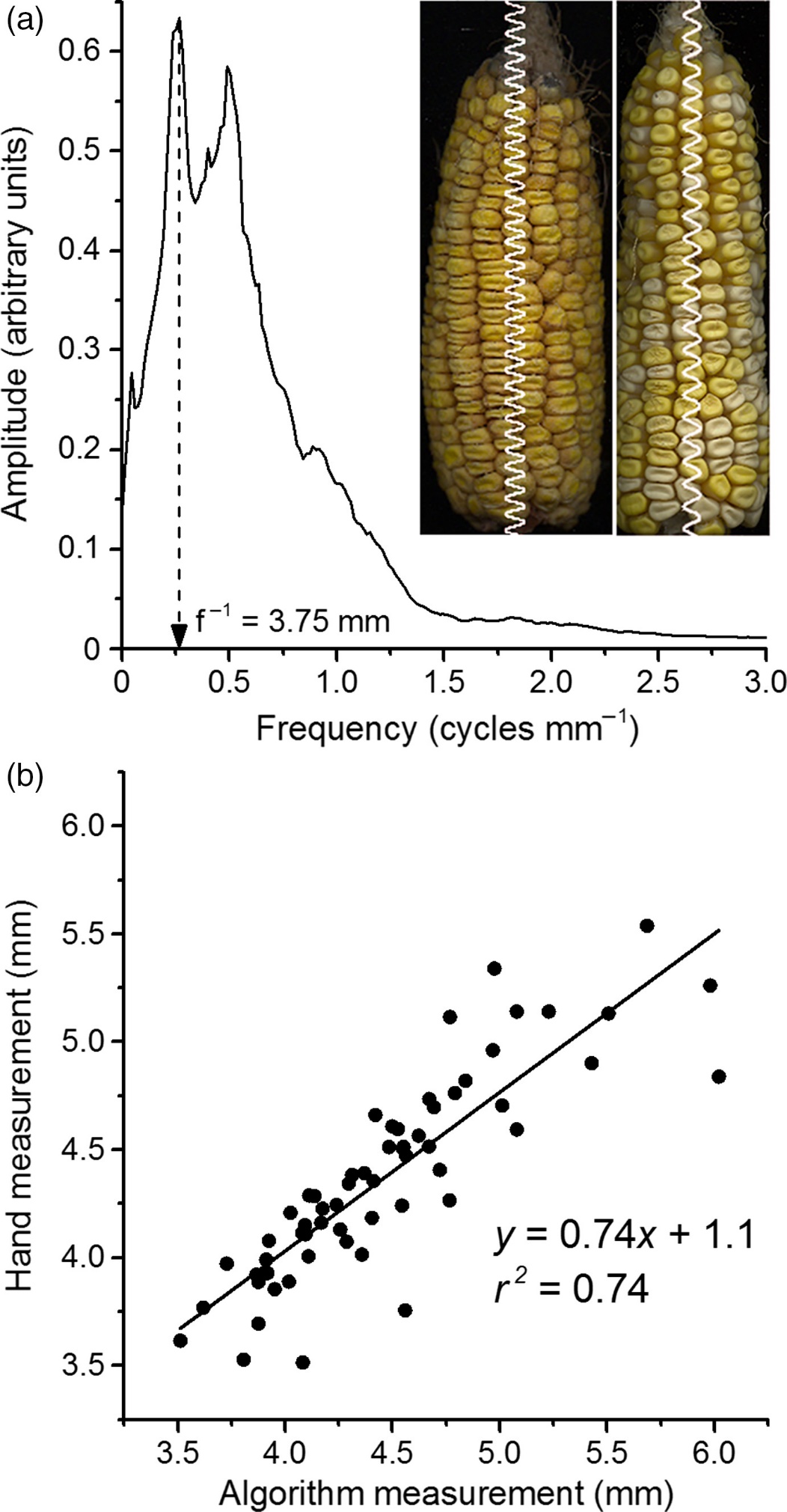
The characteristic frequency of the signal was determined by **Fast Fourier Transform** (FFT) analysis of a 3-pixel-wide rectangular window placed at every tenth pixel within the mask, then extended from **1200 to 1600** pixels in 25-pixel increments.

Figure 2(a) shows a result for one typical window. The lowest frequency major peak inversely relates to KL because the longest regularly-repeating pattern along an ear axis (the lowest frequency signal in the image) is formed by the inter-kernel spaces.

**Figure 2**

Fast Fourier Transform (FFT) analysis determines average kernel length from ear images.

(a) The first peak in the frequency histogram is the reciprocal of the average kernel length. The inset shows two example ears overlaid with a sine wave having a period equal to the computed kernel length to give a visual representation of the agreement.



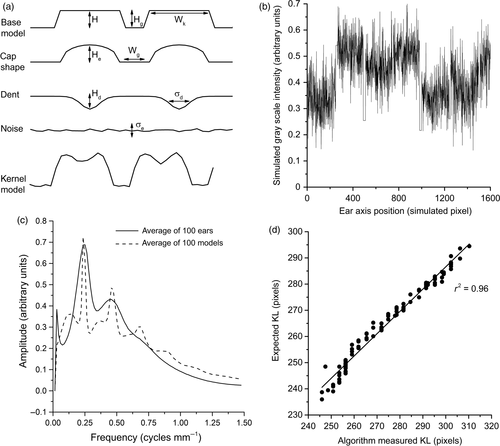
(b) Average kernel lengths measured manually and automatically by the FFT method on 60 ears agreed well.

The accuracy of this approach for measuring KL was tested in two ways. First, KL was manually measured at five locations along an ear in an image with a pixel counting software tool. The same ear was analyzed by the automatic FFT method. This comparison was repeated on 60 ears. Figure 2(b) shows the good agreement obtained between these pairs of manual and automatic measurements of KL.

A model of a gray-scale intensity ear profile was created by superimposing four submodels. The baseline model is a rectangular pulse train with pulse spacing of Wg, pulse height of H, and pulse length of Wk.

The length of the pulse represents KL and the pulse height models the gray-scale intensity. The second sub-model simulates the rounded profile of some kernels by adding an ellipse of height He. The third sub-model simulates the potential for dent corn via an inverted height normalized Gaussian distribution with width σd and height Hd.

The fourth and final sub-model is Gaussian noise with zero mean and a standard deviation of σs. Figure 3(a) depicts each of the submodels and their superposition. Figure 3(b) shows a simulated gray-scale intensity profile over a length of 1600 units equivalent to pixels in an image.



**Figure 3**

Validating the kernel length algorithm with synthetic data.

(a) Data model construction. The kernel model is the superposition of submodels that add kernel-like shape and noise to a base rectangular pulse signal.

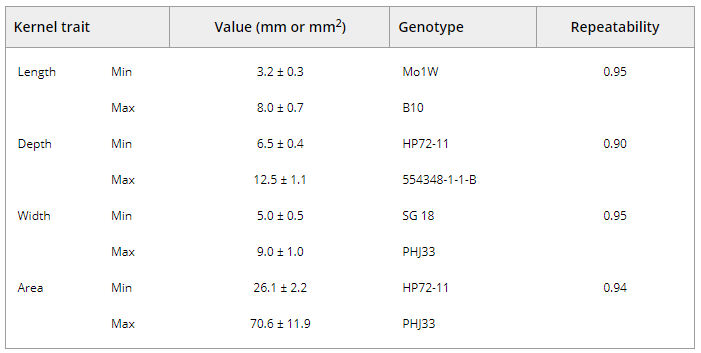
(b) Synthetic data produced by one instance of the model.

(c) Average FFT spectrum of 100 varying instances of the model superimposed on the average FFT spectrum from 100 real ears.

(d) Plot of measured kernel length versus the kernel length parameter used to create the synthetic ear profile.

We used this method to measure KL from ear images representing 445 inbred genotypes within the Wisconsin Diverse Association Panel.

**Table 1.** Range of kernel widths, heights, and areas observed in a population of 445 diverse inbreds and the genotypes that expressed them



**Kernel depth and width**

An RGB image containing approximately 100 kernels mechanically removed from the ear and randomly scattered against a black background was converted to HSV color space and the value channel was binarized. Measuring the depth (major axis) and width (minor axis) of each kernel in the image (Figure 4) depends on correct identification of the kernel tip.

The statistical method we developed for this purpose begins by creating a set of points defining the contour of each single (not clustered) kernel in the binary image. We determine the signed curvature at each contour location for each single kernel.

The point of maximum curvature is used to set an axis passing through the center of mass to define the major axis of the kernel. However, the point of maximum contour curvature is not always the true anatomical tip. Therefore, additional measurements based on this first-assigned tip position are made for each kernel in the image to create a multivariate distribution that is enriched for features associated with anatomical tip points.

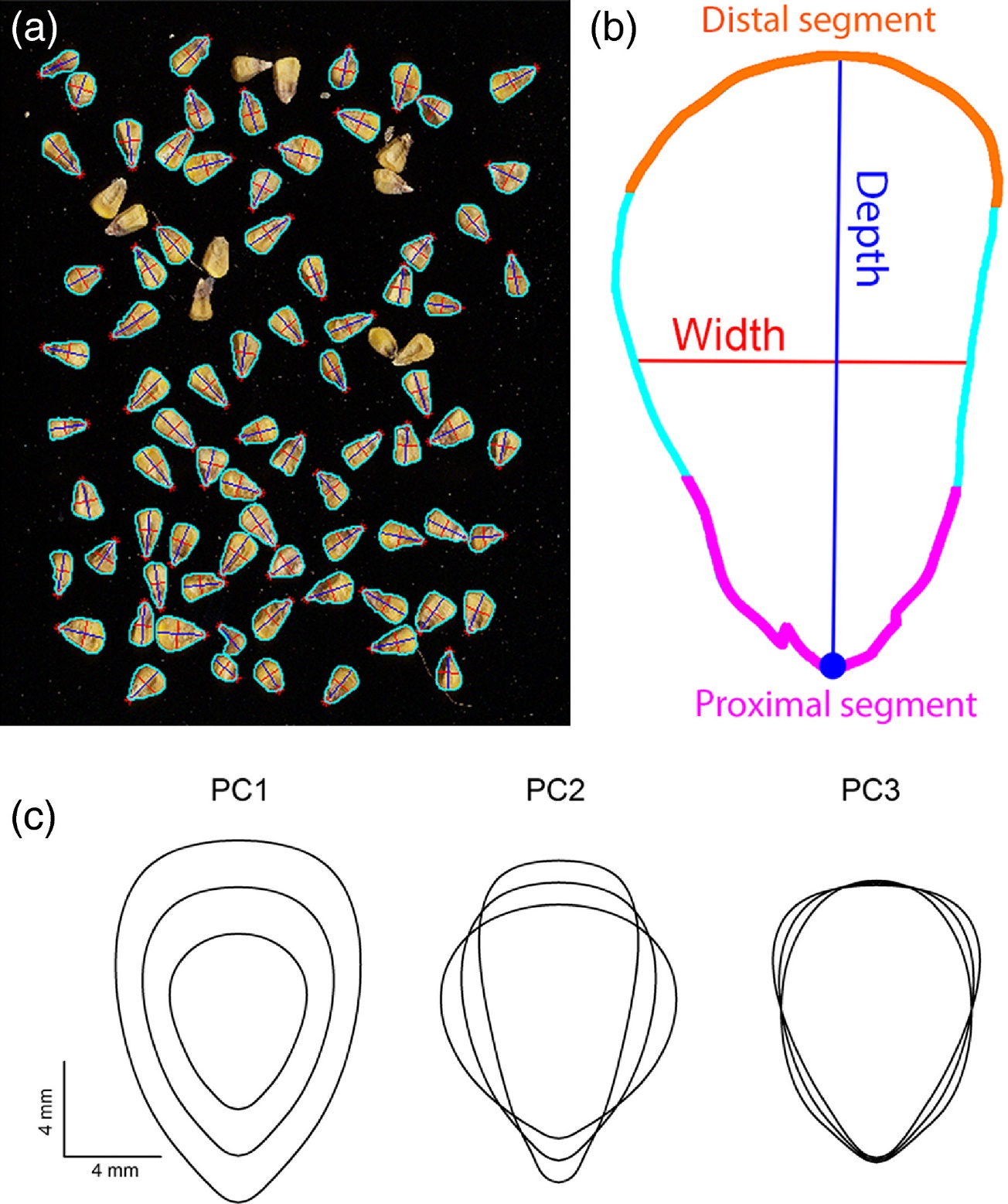
The additional measurements include kernel depth, width, depth to width ratio, and symmetry about the major axis. To these five geometric descriptors we add information about curvature within contour segments near the putative tip (proximal) and opposite it (distal). Specifically, principal components analysis of curvature values within proximal and distal contour segments (Figure 4b) generates two metrics of contour shape at putative tip and cap regions.

Figure 4

(a) Example of raw image data with computed contour, kernel depth (major axis), and kernel width (minor axis) overlaid.

(b) Features used in the statistical modeling approach to finding the tip of each kernel, which produced the automated measurements of kernel area, length, and width.

(c) Principal components analysis of the kernel contour data sets enabled their shapes to be described by three principal components PC1, PC2, and PC3. Changing PC1 causes the kernel to change size similarly in all directions. Changing PC2 changes eccentricity, or the ratio of the minor and major axes, while changing PC3 changes angularity of the contour.



**Kernel contour analysis**

We rotated each kernel to align their major axes, then translated each to place the center of mass at the origin of a coordinate system used to parameterize one half of the contour from tip to mid-cap with 500 equidistant x, y points. The 500 x values and 500 y values were stored in a 1000-dimensional vector for each kernel.

Principal components analysis decomposed this high-dimensional vector space to produce a three-dimensional representation of the data that captured 98% of the variation.

**Kernel counting**

The images of kernels from which depth and width were calculated were used to develop a counting algorithm. Most of the objects in these images consisted of single kernels though many were present in clusters of two or more kernels. The first step in the counting algorithm determines the areas of the n binary objects in the image and places the results in a vector A. Next, we divide the area of each object Ai by the area of each object Aj where i and j index the n objects in A. We round the Ai/Aj quotients to produce the square matrix of whole numbers M.