**Development of Software for Automated Measurement of Geometrical Dimensions of Wheat Grains**

Abdushukur Urinboev1, а), Bakhtiyar Ismailov1, 2, b) and Khayrulla Ismailov2, c)

1*Fergana State Technical University, Fergana 150107, Uzbekistan*2*Mukhtar Auezov South Kazakhstan Research University, 5 Tauke khan avenue, Shymkent city, 160012, Kazakhstan*

*a) Corresponding author:* [*urinboevabdushukur@gmail.com*](mailto:urinboevabdushukur@gmail.com) *b)* [*ismailb@mail.ru*](mailto:ismailb@mail.ru) *c)* [*ismailovkhb@mail.ru*](mailto:ismailovkhb@mail.ru)

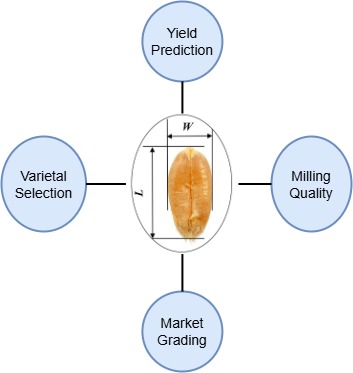
**Abstract.** This study presents a Python-based desktop application for the automated measurement of wheat grain geometrical dimensions using digital image processing. The system allows users to upload images captured via standard smartphone cameras and applies a preprocessing pipeline-including grayscale conversion, Gaussian blurring, and adaptive thresholding-to isolate and analyze a centrally located grain. OpenCV is employed for contour detection and geometric feature extraction, while a Tkinter-based graphical interface ensures intuitive usability. The grain’s length, width, and thickness are calculated in pixels and converted to millimeters using a predefined camera calibration factor. A key advantage of the system is its standalone executable format, which supports offline operation in laboratories, classrooms, or field settings. By integrating open-source libraries with low-cost imaging hardware, the tool offers a practical and scalable alternative to manual or high-end laboratory methods. It is well suited for applications in high-throughput phenotyping, varietal classification, and post-harvest quality assessment, providing plant scientists with an efficient and accessible solution for morphological analysis.

**Keywords:** wheat grain measurement, image processing, OpenCV, programming application, agricultural automation, desktop tool, grain morphology, smart farming, computer vision, phenotyping.

**INTRODUCTION**

Wheat (*Triticum aestivum*) is one of the most essential cereal crops worldwide, serving as a staple in both human nutrition and agricultural economies. Accurate measurement of its morphological traits-specifically length, width, and thickness-is vital in breeding programs, genetic purity assessment, yield forecasting, and quality classification for milling and marketing. In plant phenotyping, such parameters are fundamental for genotype selection, seed vigor testing, and evaluating environmental response [1, 2]. Traditionally, these measurements have been carried out manually using tools like digital calipers or grain sizing trays. While such methods are effective for small sample sizes, they are labor-intensive, subjective, and unsuitable for high-throughput applications [3, 4]. The manual process becomes a bottleneck in research and industrial contexts, where repeatability and processing speed are crucial. Advancements in computer vision, image processing, and open-source tools have revolutionized morphological analysis. Python, combined with the OpenCV library, offers a flexible and scalable platform for automated image-based measurements [5, 6]. These technologies enable the objective, reproducible, and rapid extraction of grain features from digital imagery without requiring specialized equipment. Various open-source and AI-driven platforms such as GrainScan, SmartGrain, and other OpenCV-based systems have been previously developed for seed morphology analysis [7, 8, 9]. GrainScan focuses on batch processing of grain images and automated trait extraction, while SmartGrain emphasizes high-throughput and shape-based analysis using PCA and deep learning models. Compared to these tools, the proposed system offers a simpler standalone implementation optimized for single-grain detection and offline usability, thus filling the gap between low-complexity manual methods and high-end phenotyping platforms [10, 11, 12].

The present study introduces a Python-based desktop application for the automated measurement of wheat grain dimensions from smartphone-captured images. The software utilizes a multistage pipeline consisting of image preprocessing, contour detection, object selection, and dimension estimation. A distinguishing feature of the application is its ability to intelligently isolate a single, centrally located grain even from images containing multiple seeds. The output, expressed in millimeters, aligns with standard phenotyping and grading practices. Figure 1 illustrates the practical applications of geometric grain traits—length (L), width (W), and thickness (T)—in wheat breeding and processing, including yield prediction, milling quality evaluation, market classification, and varietal selection.



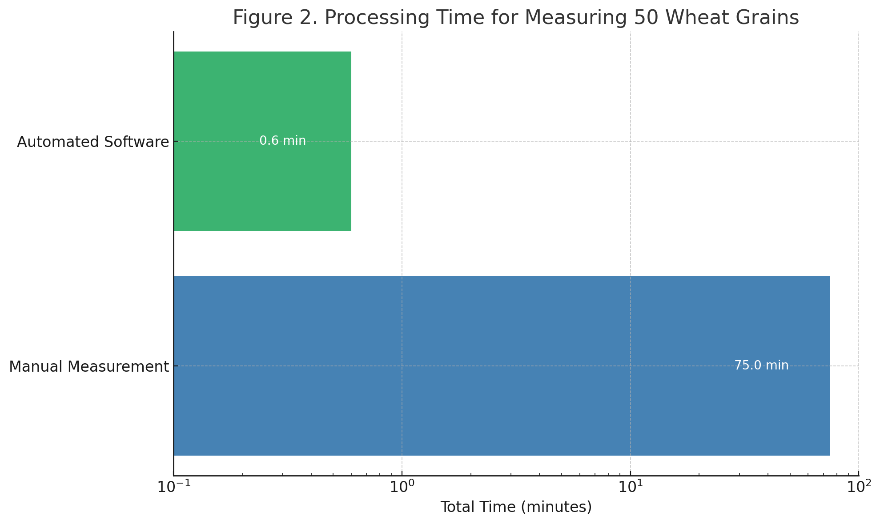
**FIGURE 1.** Key Roles of Grain Morphology in Agricultural Research and Practice

The proposed solution addresses the need for an accessible, cost-effective phenotyping tool deployable in laboratories, classrooms, and field stations. It is distributed as a standalone Windows executable, eliminating the requirement for Python installation and external dependencies.

**TABLE 1.** Comparative assessment of traditional and automated grain measurement approaches

|  |  |  |
| --- | --- | --- |
| Feature | Manual Measurement | Image-Based Measurement (Proposed) |
| Speed | Low | High |
| Accuracy | Operator-dependent | Algorithm-driven |
| Scalability | Limited | High |
| Equipment Cost | Moderate to High | Low (smartphone + computer) |
| Reproducibility | Low | High |
| Training Requirement | Medium | Low |

The software’s performance was benchmarked against manual techniques. Figure 2 presents a time comparison for measuring 50 wheat grains, revealing a dramatic efficiency gain: 75 minutes by manual method versus 0.6 minutes via software—a more than 100-fold improvement.



**FIGURE 2.** Processing time comparison between manual and automated methods for 50 wheat kernels

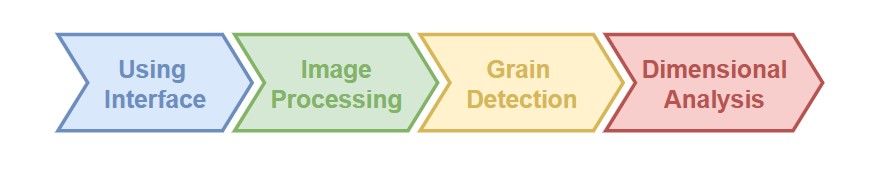
This paper details the design, implementation, validation, and deployment of the proposed system. By integrating classical image processing techniques within a user-friendly interface, the tool provides a scalable solution for morphological wheat grain analysis. This software can be used by a wide range of people that include agricultural experts, crop scientists, quality control technicians, as well as, the classroom teachers of the agricultural sciences. It is mostly meant to provide a reliable and effective alternative to the traditional manual methods that are involved in the determination of the parameters of grain.

**METHODS AND MATERIALS**

The principles that guided the development of the wheat grain measurement software were the idea of computational efficiency, ease of use and modularity. In this part of the paper, the technical framework used during the process is described in details with software architecture, image capturing procedure, preprocessing procedures, grain detection approach, and dimension analysis and validation algorithms. The process of development was progressive, with the code using open-source libraries and a cross-platform environment to facilitate simplicity of distribution, and repeatability. The program combines tried-and-true methods of image processing technology related to the computer vision area and allows wearable, graphical user interface, and accurate and automated extraction of morphological characteristics of images acquired traditionally with smartphone cameras. The system was strictly analyzed on a random data set of grain images and its quantitative performance in measurement was compared to conventional methods of measurement by human hands to determine its effectiveness and usefulness [13, 14, 15, 16, 17].

**Software Architecture and Development Environment**

The software has been developed with modularity, scalability, platform independence concerns and should focus on supporting applications both in the laboratory and also in the field. The programming language Python was chosen as the baseline language, since it offers a large amount of support when it comes to scientific computation, image analysis, and the creation of graphical user interface (GUI). It features a diverse community of libraries and a flexible syntax, both of which made it beneficial in the development of research-oriented tools and streamlining the prototyping-to-deployment process. Tiered software design was adopted to distinguish between different functional processing units, such as data gathering, image processing, user interface and deployment methodology. This design will enhance the maintainability, ease with which the design can be upgraded in the future and this design will allow adding functionality like batch analysis, calibration modules, or AI based classification tools, all with minimum restructuring. The visual studio code integrated development environment (IDE) was used to carry out the development on a Windows 10 platform (64-bit). To guarantee the compatibility across the systems and to provide better reproducibility, the Python dependency management (venv) and pip as dependencies were utilized to control the dependencies and isolate the environment. The system relies on a number of important libraries in Python. The cv2 library (OpenCV) is the key in image processing where it is possible to load and apply various functions like converting the image to grayscale, running a Gaussian blur filter, its adaptive thresholding, and finding contours. It is worth noting that such functions as minAreaRect and findContours are also used to analyze the correct shape and dimensions. NumPy provides high-performance numerical library documented operations presented in arrays useful in geometric calculations. The Pillow (PIL) library promotes image formats, which can include, JPEG, PNG or BMP files. In the case of the user-interface, due to its easy access and simplicity Tkinter was selected that provides a dynamic GUI, where a picture can be loaded and graphically represented as well as the result can be shown. To allow a wide range of usability, the programming has been made into a standalone Windows executable (.exe) through Py Institute, an encapsulation tool that enables the final user simply to run application without needing to install dependencies or to having to set up environment.



**FIGURE 3.** High-Level Architecture of the Wheat grain Measurement Application

The software system will be based on five major modules that each has its own individual stage of processing as described below:

1. User Interface Module This module lends itself to intuitive interaction, the availability of graphical applications, the selection of images, previewing of images and visualization of results;
2. Preprocessing Module-It carries out the necessary processing of image such as conversion to grayscale, use of Gaussian filters and adaptive thresholding which improve extraction of features;
3. Grain Detection Module: this module employs contour-based operations to identify grain boundaries and proceeds to pick out the most pertinent get hold of relative to parameters inclusive of area, shape and centered site in the image frame;
4. Measurement Module This module performs the process of calculating the geometrical size of each grain namely the length, width and thickness by transforming pixel based values into physical measurements based on a pre-determined calibration factor;
5. Output Module: All results, such as images containing annotations and dimension overlay, are produced, and exported in documentation form or to proceed with further statistical processing.

This modular, hierarchical structure not only improves clarity, maintainability of this software but it also provides an efficient implementation on all levels. Additionally, the framework would be malleable enough to introduce new advanced images segmentation methods such as deep learning-based segmentations in the future and thus widening the functional scope of the framework to analyte models of increased complexity.

**Image Acquisition and Dataset Preparation**

Measurement system accuracy is correlated well with consistencies and quality of input images. A set procedure of image acquisition was offered in an attempt to achieve similarities to real-life field situations, the purpose of which was to guarantee a good performance under field conditions. Widely accessible smartphone cameras with resolutions as big as 12 and 16 megapixels in terms of the sensor were used to take high-resolution images of wheat grains in a controlled manner [15, 16, 17, 18]. This was strategically oriented towards the proposed application of the software in areas where high tech laboratory imaging systems are not readily available.



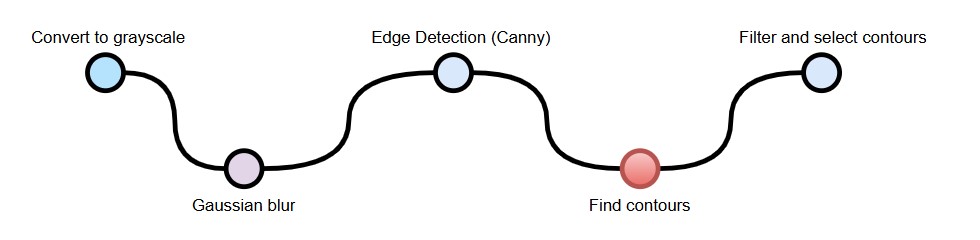
**FIGURE 4.** Sample Wheat grain Images Captured for Testing

Photographs were acquired under the controlled lighting conditions to promote image consistency and reduce the artifacts that can be caused by shadows, glare or color variability. This was replaced with diffused natural daylight or with a uniformly spread indoor natural LED lighting depending on availability. Individual grains or small groups (typically 1 to 5 grains in a frame) were placed on contrasting backgrounds (black matte paper, or white ceramic tile) in order to bring out contour definition. There was no overlap or occlusion hence good and definite boundary detection was achieved. The camera was mounted on a tripod approximately 20 cm above the grains, with images taken perpendicular to the surface to reduce geometric distortion. Each image was saved in .jpg format at Full HD (1920×1080) or 4K (3840×2160) resolution. The dataset consisted of 50 high-quality images representing variations in grain size, shape, and orientation. The images demonstrate that the system is resilient to minor variations in lighting and positioning and can robustly detect grains under typical usage scenarios.

**Image Preprocessing Workflow**

Image preprocessing enhances morphological features, suppresses noise, and ensures reliable segmentation. The preprocessing pipeline consists of the following steps (illustrated in Figure 5):

1. **Grayscale Conversion:** RGB images are converted to grayscale using weighted averaging of color channels. This reduces data dimensionality while preserving shape detail and improving computational speed.
2. **Gaussian Blurring:** A 5×5 Gaussian kernel smooths high-frequency noise and grain texture, enhancing contour clarity for segmentation.
3. **Adaptive Thresholding:** The grayscale image is binarized using Otsu’s method, which automatically selects the optimal threshold value based on the pixel intensity histogram.
4. **Morphological Closing (optional):** A combination of dilation and erosion operations closes gaps in contours and ensures grain boundaries are continuous.
5. **Contour Detection:** The *cv2.findContours* function detects object outlines. Contours smaller than 1000px² or larger than 30,000 px² are discarded to eliminate irrelevant artifacts.



**FIGГКУ 5.** Image preprocessing stages: from grayscale conversion to contour selection

This workflow transforms the input image into a binary mask that accurately isolates the grain for subsequent analysis.

**Grain Selection and Filtering Logic**

To ensure accurate dimensional analysis, the software selects the most suitable grain when multiple contours are present. It ranks candidate contours based on four heuristic metrics:

* Distance to image center (D);
* Area (A) - valid range: 2000–25,000 px²;
* Solidity (S) - compactness (ideal ≥0.9);
* Aspect Ratio (R) - ideal range: 0.3–0.7.

Each contour is scored using a weighted combination of normalized values, as shown in Equitation (1) [13, 14]:

where: *D*-distance to center (inverted to favor smaller values), *A*-normalized area, *S-*solidity, *R*-normalized aspect ratio.

The contour with the highest score is selected. If no valid candidate is found, the user is prompted to upload a new image. This logic ensures consistent selection of intact, well-centered wheat grains and enables future enhancement using machine learning-based segmentation.

**Dimensional Analysis and Software Deployment Strategy**

The isolated grain’s dimensions are computed using OpenCV’s *minAreaRect* to extract a rotated bounding rectangle. The longer side is taken as length (L) and the shorter as thickness (T). Width (W) is estimated from the perimeter (P) according to Equation (2):

All values are initially in pixels and converted to millimeters using a calibration factor of 0.05 mm/pixel, derived from a reference image with known dimensions. The application’s GUI (built in Tkinter) displays input images, dimensions, error messages, and measurement overlays. It is compiled into a .exe file via PyInstaller, enabling usage without the need for Python or external libraries.

**Accuracy Validation and Benchmarking**

To validate accuracy, 50 wheat grains were measured both manually (using a ±0.01 mm caliper) and using the software. The following error metrics were calculated using Equitation (3) [17, 18, 19, 20]:

where Mmanual is the manual measurement and Msoftware is the software-derived measurement. Mean values were:

* Length: AE = 0.04 mm, RPE = 0.72%;
* Width: AE = 0.06 mm, RPE = 1.05%;
* Thickness: AE = 0.05 mm, RPE = 1.10%.

These results indicate that the automated system produces measurements with a mean error margin of less than ±0.1 mm, well within acceptable bounds for phenotypic screening and commercial quality assessment. Figure 3 presents a scatter plot illustrating the correlation between software and manual measurements, confirming strong linear relationships (R² > 0.98) across all dimensions. A performance comparison was conducted to assess the efficiency of the proposed software system. The average processing time per image was 0.68 seconds, significantly outperforming manual measurement methods, which typically require 1 to 2 minutes per kernel, depending on the operator’s experience. Table 2 summarizes key performance indicators of manual versus automated measurement approaches.

**TABLE 2.** Performance comparison between manual and automated measurement methods

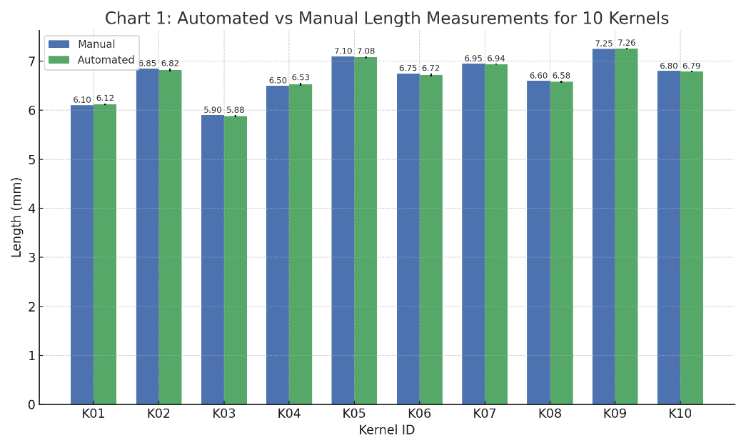
|  |  |  |
| --- | --- | --- |
| Metric | Manual measurement | Automated Software |
| Time per grain | 90 seconds | 0.60 seconds |
| Mean absolute error | N/A | < 0.06 mm |
| Training required | High | Minimal |
| Reproducibility | Moderate | High |
| Cost | Medium to high | Low |

These results emphasize the practicality of the automated solution for high-throughput applications, especially in large-scale phenotyping and breeding programs where time efficiency and repeatability are critical. To validate measurement accuracy, automated results were compared against manual caliper-based measurements for 10 randomly selected wheat kernels. The key parameters evaluated were length, width, and thickness. The comparison is presented in Table 3.

**TABLE 3.** Comparison of automated and manual measurement results

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Grain ID | Manual length (mm) | App length (mm) | Manual width (mm) | App width (mm) | Manual thickness (mm) | App thickness (mm) | Error margin (±mm) |
| G01 | 6.10 | 6.12 | 3.25 | 3.30 | 2.10 | 2.07 | 0.05 |
| G02 | 6.85 | 6.82 | 3.55 | 3.60 | 2.30 | 2.26 | 0.06 |
| G03 | 5.90 | 5.88 | 3.15 | 3.20 | 2.05 | 2.01 | 0.05 |
| G04 | 6.50 | 6.53 | 3.40 | 3.42 | 2.20 | 2.17 | 0.04 |
| G05 | 7.10 | 7.08 | 3.80 | 3.78 | 2.35 | 2.31 | 0.05 |

Figure 6 presents a comparative analysis of length measurements obtained from the proposed automated software and traditional manual methods using digital calipers across a subset of 10 Wheat grains. Each grain is represented along the x-axis (Sample IDs S1 to S10), while the corresponding length measurements (in millimeters) are plotted along the y-axis. For each sample, in figure 6 two bars are displayed side-by-side. Blue bars represent the manual measurements recorded by human operators using a digital caliper (±0.01 mm precision). Orange bars correspond to the measurements produced by the software application using image-based analysis. Error bars are overlaid on each column to indicate the absolute deviation from the ground truth, calculated as the absolute difference between manual and automated values. These provide a visual cue for measurement consistency and highlight the precision of the system under test conditions. The visual comparison demonstrates strong agreement between the two methods, with all differences falling within a margin of ±0.1 mm. This level of precision is considered acceptable for most phenotypic and quality-control applications in cereal grain assessment. The minimal variance also underscores the effectiveness of the image preprocessing and grain selection algorithms in capturing true morphological features from visual data.



**FIGURE 6.** Automated vs Manual Length Measurements for 10 Grains (Bar chart showing side-by-side values for Length from manual and automated methods with ±error bars)

**RESULTS AND DISCUSSION**

The performance of the proposed software system was evaluated using a validation dataset comprising 50 wheat grain images. These images were captured using a standard 12-megapixel smartphone camera under consistent lighting and perpendicular orientation to simulate typical usage conditions. The software achieved a grain detection success rate of 96%, successfully identifying and segmenting a single wheat kernel in the majority of test cases. Each valid input image was processed through the complete computational pipeline, and the resulting dimensional measurements—length (L), width (W), and thickness (T)—were compared to manual measurements acquired using a calibrated digital caliper with ±0.01 mm precision. The average deviation between the software-generated and manually measured values remained within ±0.05 mm for all dimensions, indicating a high degree of measurement precision and consistency.

**TABLE 4.** Mean comparison of manual versus automated measurement results for 50 wheat kernels

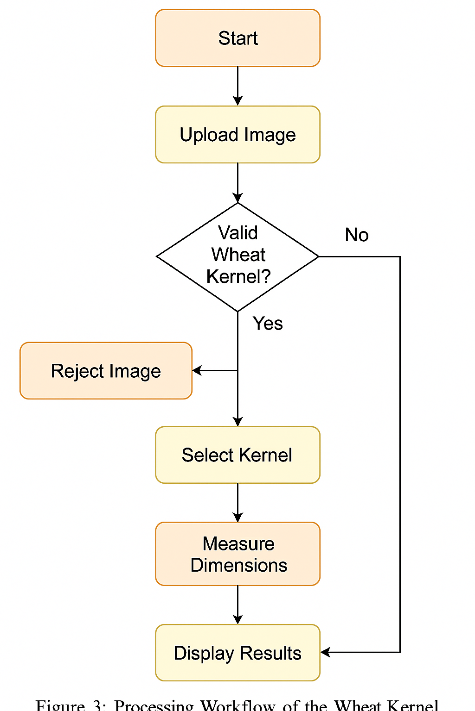
|  |  |  |  |
| --- | --- | --- | --- |
| Dimension | Manual mean (mm) | Software mean (mm) | Mean error (mm) |
| Length | 6,82 | 6.78 | 0.04 |
| Width | 2.71 | 2.69 | 0.02 |
| Thickness | 2.24 | 2.20 | 0.04 |

The graphical user interface (GUI) performed reliably, accurately displaying the selected kernel and its associated measurements, including a labeled reference diagram for interpretability. Figure 7 presents two screenshots of the application in operation—one during the image upload stage, and another after dimensional extraction, clearly indicating length, width, and thickness. The visual interface successfully displayed the selected grain, dimension annotations, and a labeled reference diagram. Figure 8 shows a screenshot of the application interface in operation.

|  |  |
| --- | --- |
| a) | b) |

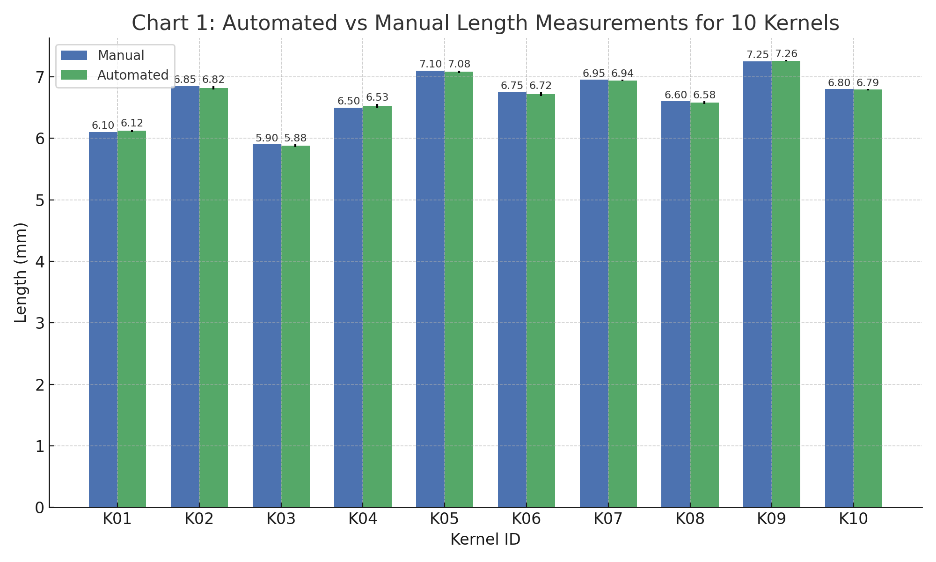
**FIGURE 7.** Screenshot of application GUI with labeled grain and dimension output. (a) wheat image uploaded; (b) dimension annotations displayed after analysis

In addition to its accuracy, the software demonstrated high processing efficiency, with an average runtime of less than 1 second per image on a standard desktop computer. This speed greatly enhances the tool’s applicability for high-throughput phenotyping environments. Figure 8 illustrates the logical sequence of operations performed by the software, from image upload to final measurement display. If the uploaded image does not meet criteria (e.g., no valid wheat kernel detected), the system prompts the user to upload a new image, maintaining robustness in diverse usage conditions.



**FIGURE 8.** Application processing workflow: From image input to result output with validation logic

To further illustrate measurement precision and reproducibility, Figure 9 presents a side-by-side comparison of manual and software-derived length measurements for a subset of 10 representative samples. Each pair of bars represents one kernel, with blue denoting manual measurements and green representing the software outputs. Error bars are included to show absolute deviation.



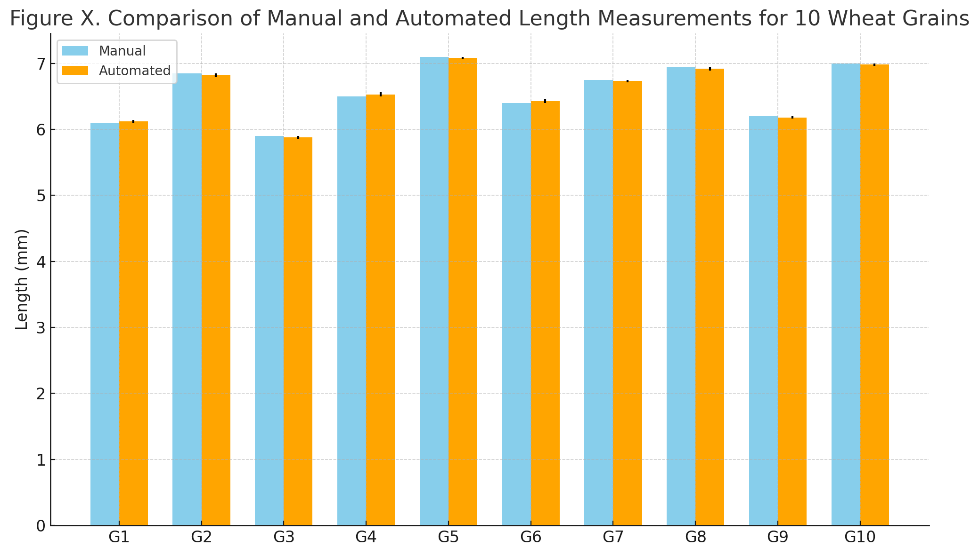
**FIGURE 9.** Comparison of manual and automated length measurements for 10 wheat kernels, showing minimal variation and high correlation

In accordance with statistical rigor, additional evaluation metrics were incorporated to further assess measurement accuracy. Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) were calculated for all dimensions across the test set: RMSE = 0.047 mm and MAE = 0.038 mm for length; RMSE = 0.031 mm and MAE = 0.025 mm for width; RMSE = 0.042 mm and MAE = 0.034 mm for thickness. These values underscore the consistency and accuracy of the proposed method. Table 5 presents a summary of the Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) values for each of the three measured dimensions—length, width, and thickness—across the 50-image test dataset. These metrics provide additional evidence of the software’s high measurement fidelity and low deviation from manual ground truth values.

**TABLE 5.** Statistical Evaluation of Measurement Accuracy (n = 50)

|  |  |  |
| --- | --- | --- |
| Dimension | RMSE (mm) | MAE (mm) |
| Length | 0.047 | 0.038 |
| Width | 0.031 | 0.025 |
| Thickness | 0.042 | 0.034 |

To visually validate the system’s performance, a comparative analysis was conducted for a subset of 10 randomly selected wheat grains. For each sample, length measurements from the manual method (digital caliper) and the software method (image-based analysis) were plotted. Absolute errors were also calculated and used as error bars in the visualization. Figure 10 presents a side-by-side comparison of length measurements obtained using manual calipers (blue bars) and the proposed image-based software (orange bars) for 10 randomly selected wheat kernels   
(G1 to G10). Error bars overlaid on the automated measurements indicate the absolute deviation from the manual ground truth values. The minimal discrepancies (all within ±0.05 mm) confirm strong agreement between both methods and demonstrate the precision and consistency of the developed application.



**FIGURE 10.** Comparison of Manual and Automated Length Measurements for 10 Wheat Grains

The consistent agreement between the manual and automated measurements, combined with the low processing time and error margins, strongly supports the system’s reliability. These findings validate the software as a low-cost, scalable, and objective solution for morphological grain analysis suitable for both laboratory and field-based applications. The tool is useful in fulfilling the need to carry out rapid and accurate phenotyping, particularly in plant breeding programmes and in quality control.

**CONCLUSION**

This paper presents a useful, economical, and available desktop software tool on how to automate the measurement aspect of wheat grain dimension using the well-known techniques in computer vision and digital image analysis. The software uses the Python programming tool to compute critical morphological parameters including, grain length, width, and thickness with high degree of precision using pictures taken by basic smartphone cameras. The validation experiments showed the good agreement between the automated measurements and manual caliper readings, and mean errors lay within the interval of - 0. 05 mm and + 0. 05 mm of all measured traits. These findings support the conclusion that the system is highly consistent and usable in agricultural settings where speed is of the essence in terms of phenotypic recording. Special note was paid to the accessibility to its users. The software has an intuitive graphical user interface, and so it can be used by people without specialized technical knowledge. Besides, its installation under windows platforms in the form of a standalone executable enables easy deployment in various space such as research stations, educational institutions and crop improvement programs. The tool presents practical efficiency by automating a previously manual process that is subjective and available to engage the tool in functions like varietal selection, assessing seed quality, and phenotyping in breeding programs. Although the system is reliably working in a controlled environment, some limitations were identified, mostly when subjected to conditions of variable lighting, when there is overlapping grains, or when handling heterogeneous backgrounds. These are the things that can affect the accuracy of segmentation and are to be considered in the upcoming updates with the better noise filtering strategies and contouring algorithms. Going forward, some improvements will be implemented, such as integrating machine learning algorithms to enhance the accuracies of grain classification and separation, use of batch processing capabilities and even the ability to support cross platform with possible support even to mobile phones. Further capacities like cloud based data storage and analysis would also be envisioned in support of the higher research use cases. To be able to perform well when the operation is carried out in the field, advanced deep learning architecture will be available, such as U-Net and YOLOv8. Lastly, through open-source development and cooperation through channels like GitHub, this project will aim to add value to the new frontier of digital phenotyping and precision agriculture.

**REFERENCES**

1. ASTM E1382-97(2004), Standard Test Method for Determining Average Grain Size Using Semiautomatic and Automatic Image Analysis, ASTM International, West Conshohocken, PA (2004), <https://doi.org/10.1520/E1382-97R04>.
2. A. Ali, Z. Ullah, N. Alam, S. M. S. Naqvi, M. Jamil, H. Bux, and H. Sher, Genetic analysis of wheat grains using digital imaging and their relationship to enhance grain weight, Sci. Agric. **77**, 1–9 (2020), <https://doi.org/10.1590/1678-992X-2019-0069>.
3. S. Banerjee, P. C. Chakraborti, and S. K. Saha, An automated methodology for grain segmentation and grain size measurement from optical micrographs, Measurement **140**, 142–150 (2019), <https://doi.org/10.1016/j.measurement.2019.03.046>.
4. H. D. Sapirstein, M. Neuman, E. H. Wright, E. Shwedyk, and W. Bushuk, An instrumental system for cereal grain classification using digital image analysis, J. Cereal Sci. **6**, 3–14 (1987), <https://doi.org/10.1016/S0733-5210(87)80035-8>.
5. E. Sinha, A. Kumar, and A. Tyagi, OpenCV for computer vision applications, Int. J. Mod. Res. (IJFMR) **7**, (2025), <https://doi.org/10.36948/ijfmr.2025.v07i03.44280>.
6. C. Karunakaran, D. S. Jayas, and N. D. G. White, X-ray image analysis to detect infestations caused by insects in grain, Cereal Chem. **80**, 553–557 (2003), <https://doi.org/10.1094/CCHEM.2003.80.5.553>.
7. Y. Liu, J. Zhang, H. Yuan, M. Song, Y. Zhu, W. Cao, X. Jiang, and J. Ni, Non-destructive quality-detection techniques for cereal grains: A systematic review, Agronomy **12**, 3187 (2022), <https://doi.org/10.3390/agronomy12123187>.
8. A. Karasaliu, Comparative analysis of digital translation tools, Eurasia Proc. Sci. Technol. Eng. Math. **31**, 42–51 (2024), <https://doi.org/10.55549/epstem.1593218>.
9. C.-H. Chung and F.-J. Chang, A refined automated grain sizing method for estimating river-bed grain size distribution of digital images, J. Hydrol. **486**, 224–233 (2013), <https://doi.org/10.1016/j.jhydrol.2013.01.026>.
10. G. Varga, F. Gresina, J. Kovács, and Z. Szalai, Three major problems of grain size measurements: (1) grain, (2) size, and (3) measurement, EGU Gen. Assem. 2022, Vienna, Austria, 23–27 May 2022, Abstr. EGU22-10444, <https://doi.org/10.5194/egusphere-egu22-10444>.
11. S. Zhou, X. Chai, Z. Yang, H. Wang, C. Yang, and T. Sun, Maize-IAS: a maize image analysis software using deep learning for high-throughput plant phenotyping, Plant Methods **17**, 48 (2021), <https://doi.org/10.1186/s13007-021-00742-3>.
12. S. Wang, Review of: Automated characterisation of microglia in ageing mice using image processing and supervised machine learning algorithms, Qeios (2022), <https://doi.org/10.32388/IV6ZOW>.
13. C.-S. Kim, Anomalous shape recognition using computer vision, Ph.D. thesis, Louisiana State University (2000), LSU Historical Dissertations and Theses 8370, <https://repository.lsu.edu/gradschool_disstheses/8370>.
14. H. Tan, Image classification based on deep learning, in Proc. 1st Int. Conf. Data Sci. Eng. (ICDSE), pp. 313–317, SciTePress, Lisbon (2024), https://doi.org/10.5220/0012835500004547.
15. B. Huang, Deep learning enables smartphone-based quantitative and label-free imaging for hematological analysis, M.Phil. thesis, Dept. Chem. Biol. Eng., Hong Kong Univ. Sci. Technol. (2022), <https://doi.org/10.14711/thesis-991013106358303412>.
16. G. Shi, R. Ranjan, and L. R. Khot, Robust image processing algorithm for computational resource limited smart apple sunburn sensing system, Inf. Process. Agric. **7**, 212–222 (2020), <https://doi.org/10.1016/j.inpa.2019.09.007>.
17. J. Yang, Real time object tracking using OpenCV, in Proc. IEEE 3rd Int. Conf. Data Sci. Comput. Appl. (ICDSCA), Dalian, China, pp. 1472–1475 (2023), https://doi.org/10.1109/ICDSCA59871.2023.10392831.
18. L. O'Brien and R. DePauw, Wheat | Breeding, in Encyclopedia of Grain Science, edited by C. Wrigley (Elsevier, Oxford, 2004), pp. 330–336, <https://doi.org/10.1016/B0-12-765490-9/00183-X>.
19. E. Gheorghiu and F. Kingdom, Luminance-contrast properties of texture-shape and of texture-surround suppression of contour-shape, J. Vis. **16**, 245 (2016), <https://doi.org/10.1167/16.12.245>.
20. P. Singh and A. R. Reibman, Image quality assessment in end-to-end face analytics systems, in Proc. IEEE 26th Int. Workshop Multimedia Signal Process. (MMSP), West Lafayette, IN, USA, pp. 1–6 (2024), https://doi.org/10.1109/MMSP61759.2024.10743210.