ACCEPTED MANUSCRIPT

Title: Using deep learning algorithms and a flatbed scanner to assess rice quality

Authors: Zapotoczny Piotr, Graczyk Przemysław Karol

To appear in: Technical Sciences

Received 18 February 2025;

Accepted 3 June 2025;

Available online 10 June 2025.

USING DEEP LEARNING ALGORITHMS AND A FLATBED SCANNER TO ASSESS RICE QUALITY

Zapotoczny Piotr[1], Graczyk Przemysław Karol
University of Warmia and Mazury in Olsztyn,
ul. Heweliusza 14, 10-718 Olsztyn, Poland
*****Visacom Sp. z o.o,***
ul. Wł. Trylińskiego 8/L1, 10-683 Olsztyn

Abstract

This article presents the results of research on the application of deep learning techniques in the automatic assessment of rice grain quality. A measurement methodology and a computer program were developed, which uses a deep learning model to identify individual grains in an image and detect impurities for further qualitative analysis. The program was implemented in Python using the OpenCV 4, Numpy, and Ultralytics YOLO libraries. The study used a flatbed scanner, enabling the identification of approximately 3,000 objects in a single scanner ruler pass.

Keywords: deep learning, rice, segmentation, OpenCV, flatbed scanner

Intraduction

Rice is a staple food for a large part of the world's population, providing key micro- and macronutrients essential for the proper functioning of the body. At the same time, its cultivation and processing play an important role in the economy of many countries (Ziarno and Zaręba, 2008). For this reason, ensuring high quality of rice grains is of particular importance. Rice quality is assessed based on a range of parameters, such as chemical composition, color, shape, and the presence of pathogens. The evaluation of these characteristics is performed using organoleptic or instrumental methods, which, however, require expensive, specialized equipment and qualified personnel. Therefore, the food industry is actively seeking new

[11] Correspondence: Piotr Zapotoczny, Department of Mechatronics, Faculty of Technical Sciences, University of Warmia and Mazury in Olsztyn, ul. Heweliusza 14, 10-718 Olsztyn, e-mail: zap@uwm.edu.pl

analytical methods that are more efficient, cheaper, and easier to apply. In recent years, thanks to the rapid development of machine learning technology, especially neural network models based on deep learning, new possibilities for assessing rice grain quality have emerged. Systems based on optical scanning can be an attractive alternative to traditional analytical methods, such as near and far infrared spectroscopy.

1. Rice

Rice is one of the oldest crops, accompanying humans for over 5,000 years. It is also one of the most important cereals in the world, ranking third in cultivation, behind only corn and wheat. Global annual rice production is approximately 700 million tons, making it a very important nutrient (Ziarno and Zaręba, 2008). The largest rice production takes place in Asia, where 90% of the world's cereal crops are grown. Despite Asia's dominance, Europe also has cultivated fields, mainly located in Southern Europe. This demonstrates the versatility of this plant. One of the basic ways to classify rice grains is to divide them based on differences in shape (Ziarno and Zaręba, 2008; Borowy and Kubiak, 2014):

- long-grain – grain length is 4-5 times greater than its width
- medium-grain – grain length is 2-3 times greater than its width
- short-grain – spherical

However, depending on the degree of purification and processing, rice can take the form of (Ziarno and Zaręba, 2008; Borowy and Kubiak, 2014):

- brown rice – it is not shelled, but devoid of husk and spores
- white rice (polished) – devoid of chaff, germ, which makes it poor in protein, fat and vitamins, but makes it have a longer shelf life
- parboiled – before threshing, it is subjected to steam under pressure, then dried, thanks to which it retains some of the nutrients
- precooked – the grains are completely cooked, and then the water is removed from them
- Instant rice – rice grains are precooked and then dried.

Despite its lower nutritional value (Bienvenido and Juliano, 2019), white rice remains the most popular variety of rice. This popularity can be attributed to its ease of processing and long shelf life. For these reasons, white rice was chosen as the research material for the rest of the work.

1.1 Rice Quality Assessment

The high global consumption of rice makes it extremely important to control its quality and safety as one of the basic components of the global diet. One of the key processes for assessing rice quality is determining the quantity of grains of specific dimensions and shape (Polski Komitet Normalizacyjny, 2004). Traditional measurement methods rely on manual analysis, which is time-consuming and requires qualified specialists with appropriate experience. Moreover, due to the subjective nature of the assessment by the people performing the study, there is a risk of errors that may affect the accuracy of the results. Individual grains are subjected to analysis, based on which the proportion of individual quality fractions in the total mass of the grain batch is calculated. The division is made according to a specific methodology that ensures standardization and comparability of results between studies (Polski Komitet Normalizacyjny, 2004): whole grain – without any broken parts or the part of grain whose length is equal to or greater than $\frac{9}{10}$ the average length of the entire tested sample,

- whole rice – whole grain, the length of which is equal to or greater than $\frac{3}{4}$ the average length of the entire tested sample,
- broken grain – a part of grain whose length is shorter than $\frac{3}{4}$ but greater than $\frac{1}{2}$ the average length of the tested sample,
- broken grain – a part of grain whose length is equal to $\frac{1}{2}$ or less than but greater than $\frac{1}{4}$ the average length of the tested sample,
- grain crumbs – a part of grain whose length is equal to $\frac{1}{4}$ or less than the average length of the tested sample, but is not sifted through a metal sieve with round meshes with a diameter of 1.4 mm,
- fraction below 1.4 mm – the part of the grain that is sifted through a metal sieve with round meshes with a diameter of 1.4 mm.

Fig. 1 Grain sizes and fractions of grain parts (Polish Committee for Standardization, 2004)

2. Deep machine learning

Deep learning is a computer technique used to analyze and transform data using multi-layered neural networks. Each layer in the network receives input data from the parent layer, processes it, and passes it on, gradually refining the data representation. The process of training neural networks is based on optimization algorithms that minimize prediction errors and increase model accuracy. In this way, the network adapts to perform a given task (Howard and Gugger, 2020; Chollet, 2019; Geron, 2020; Goodfellow et al., 2016). Training neural networks is an iterative process in which network weights are adjusted based on training data to minimize prediction error. Before starting training, it is crucial to properly prepare the model. The process begins with initialization of scales, most often using random values. Then, forward propagation is performed, during which input data passes through successive layers of the network – from the input layer to the output layer. The result of this propagation is the model's predictions, the accuracy of which is evaluated using a loss function that measures the difference between predictions and actual values. The next step is backpropagation, during which the gradient of the loss function is calculated with respect to each scale in the network. Based on these gradients, weights are updated to reduce the error, which is achieved using optimization algorithms such as the simple gradient method. This process is repeated many times over many epochs, each of which involves processing the entire training data set. Thanks to its iterative nature, the network gradually learns to better represent data and solve tasks more effectively.

3. Research Hypothesis and Scope of Work

To achieve the assumed research goal, the following research hypothesis was formulated:
H0: the application of deep learning methods combined with computer image processing enables the identification of touching rice grains with any number of neighbors.

To verify the hypothesis, a program written in Python (Matthes, 2016; Ramalho, 2015) was developed using deep learning and computer image analysis techniques, which perform better than standard image analysis methods (Krizhevsky et al., 2017). The aim of the program was to identify individual rice grains in an image and measure their geometric features, such as length, width, area, and perimeter. To obtain a high-accuracy model, it was necessary to collect appropriate data to train the deep learning algorithm. The algorithm was based on segmenting each kernel as a separate ROI (Region of Interest) area. The data preparation procedure included: (a) scanning rice grain images using a flatbed scanner, (b) labeling images, (c) augmenting input data. Rice grain images were obtained by scanning freely scattered grains on the surface of a flatbed scanner, at a resolution of 600 DPI, to ensure high-quality data for network training (Gonzalez and Woods, 2001). Neural networks based on YOLOv8-seg algorithms best analyze images with dimensions of 640 × 640 pixels (Ultralytics, 2023), which is why after scanning, the images were divided into fragments of the recommended sizes.

Background preparation was crucial to obtain high contrast between rice grains and the substrate. A scanning methodology was developed that eliminated shadows around objects. Finally, the grains were arranged randomly on the scanner plate so that the maximum number of grains touched each other. The images were divided into 5 vertical and 7 horizontal sections, which created a total of 35 fields with dimensions of 640 × 640 pixels. Example scan fragments are shown in Fig. 2.

Fig. 2 Example image after the split operation

4. Image labeling

The first stage of image analysis was the labeling operation, which involved identifying and marking the contours of individual grains. The CVAT program was used for this purpose. The results of the labeling process are shown in Fig. 3.

Fig. 3 Sample image with labeled kernels

The labeling process resulted in the identification of approx. 1826 grains, where each constituted a separate ROI area.

5. Data augmentation

Data augmentation is a technique used to enrich the training set by various transformations and modifications of existing data. This method is particularly useful in image analysis, where obtaining additional data is often costly and time-consuming. Advanced machine learning models, based on complex architectures, require large datasets to achieve high training effectiveness (Elgendy, 2020; Szelski, 2022). A key aspect is to prevent model overfitting, which occurs when data is too homogeneous or insufficient, limiting the model's ability to generalize. Data augmentation, by generating modified samples, supports the learning process, increasing data diversity and promoting the model's ability to generalize, which positively affects its performance (Chollet, 2019). In the analyzed dataset, an augmentation technique was used consisting of rotating the original images by 90° around their axis. This process allowed a fourfold increase in the number of available samples, resulting in 412 training images containing a total of 7304 objects. Thanks to this method, the diversity of the training set was improved, which contributed to increasing the accuracy of the model in the validation phase. An example image subjected to augmentation and the effect of the applied transformations are presented in Fig. 4.

Fig. 4 Example data augmentation result

6. Train the model

After collecting the set of training images, the deep learning model training stage began. The better the results obtained at this stage, the greater the model's ability to generalize and adapt to unknown data. The training process included: (a) selecting the appropriate model architecture, (b) selecting hardware adapted to the specifics of data processing and algorithm requirements, (c) implementing the training process.

6.1 Selection of architecture

Choosing the right architecture is a crucial step in creating deep learning applications. In the case of rice grain analysis, the algorithm must be characterized by high precision in detecting individual elements, which requires the use of instance segmentation. An important factor in the context of the application is also the speed of analysis. Instance segmentation, unlike semantic segmentation, not only assigns each pixel of an image to a specific class, but also distinguishes individual instances of the same object in the image (Forsyth, Ponce, 2011). The article analyzed the performance of various models dedicated to instance segmentation, using the COCO database, widely recognized as a standard in benchmarking image processing algorithms. The segmentation quality index (maskmAP) was used as the main measure of effectiveness. The use of deep neural networks, such as those based on convolutional neural networks (CNNs), in object detection and classification significantly improves the effectiveness and accuracy of analysis, which has been confirmed in numerous studies (Aukkapinyo et al., 2020; Dharmik et al., 2022; Gao et al., 2024; Gilanie et al., 2021; On et al., 2023; Li et al., 2022; Ma et al., 2024; Mavaddati and Razavi, 2024; Velesca et al., 2021; Wei et al., 2024; Xu et al., 2023). As a result of the comparative analysis and based on available data on various models, the YOLOv8-seg model was selected for the project. This decision was made based on its high performance and accuracy, which makes it particularly suitable for applications requiring both precision and time efficiency.

6.2 YOLOv8-seg Architecture

YOLO (You Only Look Once) is an object detection algorithm that, in its original version, focused mainly on detection, not including segmentation functionality. Significant extensions have been introduced since the YOLOv5 model, which combine proven YOLO detection methods with modern segmentation techniques (Rath, 2023). The key feature of YOLO remains image processing in a single pass, which enables simultaneous determination of object location and class without the need for multi-stage data processing (Redmon et al., 2016). The YOLOv8-seg model develops these assumptions, using filtering mechanisms present in previous versions, such as thresholding to eliminate low-probability detections and non-maximum suppression technique to reduce multiple detections of the same objects. The innovativeness of YOLOv8-seg lies in abandoning traditional anchors (anchorboxes), which allows for direct prediction of object locations. Additionally, the introduction of special layers enables the generation of segmentation masks for each detected object. Thanks to these solutions, YOLOv8-seg offers precise separation of objects from both the background and each other, which is crucial in many practical applications. The ability to simultaneously detect and segment in real time makes this model particularly attractive in the context of applications requiring high accuracy and computational efficiency.

6.3 YOLOv8 segmentation label format

The previously prepared images with text descriptions were converted to a format supported by the YOLO model. This format assumes that each row in the file describes a single object. The first number in the row indicates the object's class index. In the analyzed case, all objects belong to one class, labeled "Ryż". The next four numbers represent the normalized coordinates of the center in the x and y axes and the width and height of the bounding box. The remaining numbers in the row describe the boundaries of the object being segmented, starting from the sixth number, where each pair of x-y coordinates defines the next boundary point, creating a segmentation mask.

6.4 Data division

The prepared images in YOLO format were divided into three sets: training, validation, and test, according to recommendations (Howard and Gugger, 2020; Chollet, 2019; Geron, 2020; Goodfellow et al., 2016).

- **Training set (train set):** The main part of the data used to train the model. The model adjusts its internal parameters based on this data, striving to fit them as closely as possible.
- **Validation set (validation set):** Used to monitor the model's performance during training, but does not participate in the learning process itself. It allows evaluating the model's ability to generalize and detecting overfitting.
- **Test set (test set):** Used to evaluate the final performance of the model on data not used in any training stage, which allows for an objective assessment of the model's operation.

The dataset, consisting of 412 images, was divided in a proportion of 85% / 10% / 5% into training (350 images), validation (41 images), and test (21 images) sets. Additionally, a configuration file in YAML format was prepared, containing image access paths, the number of classes, and their names. The structure of this file is presented in Figure 5.

Fig. 5 Configuration file structure

6.5 Model and environment selection

Due to the limited amount of data obtained during the acquisition process, in further stages of the work, it was decided to use a pre-trained model based on the YOLOv8-seg architecture. The knowledge acquired by the model during previous trainings allows for better generalization during further learning, which potentially improves the effectiveness of rice grain identification (Howard and Gugger, 2020; Chollet, 2019; Geron, 2020). Convolutional neural networks are characterized by high computational demands and high demand for operating memory. For this reason, the use of graphics processing units (GPUs) is preferred over central processing units (CPUs), due to their ability to parallel process data and optimize complex mathematical operations, which are crucial in deep learning algorithms. Bearing these requirements in mind, the Google Colab platform was used for further experiments, which provides access to advanced graphics cards dedicated to deep learning algorithms. To ensure optimal efficiency while maintaining a favorable training cost ratio, a series of preliminary trainings were conducted on each of the available models. These experiments were performed using standard training parameters, with a duration of 15 epochs. The results of these trainings are summarized in Table 1, which allows for an analysis of the performance of individual configurations.

Table 1.

Results of the initial training of YOLOv8-seg models

After conducting the initial series of trainings, the best results were obtained using the YOLOv8l-seg model, an additional argument for the choice was that this model can be trained on an Nvidia Tesla V100 card.

6.6 Network training

The final stage of object recognition model training involved the final training. The previously prepared images were loaded into the selected model on the Google Colab platform, where the training process was launched with standard parameters and lasted for 200 epochs. The training results are presented in Figures 6 and 7.

Fig. 6 Loss Value Charts

The charts illustrate loss values, which reflect the discrepancy between predictions generated by the model and the actual labels of the training data. Losses serve as a measure of model performance – the smaller the loss values, the better the model's performance. Loss functions evaluate various aspects of the model (Howard and Gugger, 2020; Chollet, 2019; Geron, 2020):

- **train/box_loss** – evaluates the accuracy of predicting object locations
- **train/seg_loss** – measures the effectiveness of assigning pixels to relevant objects
- **train/cls_loss** – determines the quality of object classification.

Analysis of the loss function chart for segmentation indicates that after approximately 120 epochs, the loss values stabilize at a satisfactory level. This trend suggests that further training would not bring significant benefits.

Fig. 7 Average precision (mAP) plots

The charts also present the results of mean average precision (mAP), which evaluates the model's ability to correctly classify and localize objects. An mAP50 value means that predicted bounding boxes (B) or masks (M) are considered correct if their Intersection over Union (IoU) with the actual values is at least 0.5. On the other hand, mAP50-95 is calculated as the average for IoU thresholds in the range from 0.5 to 0.95. The mAP values for masks at IoU 0.5–0.95

stabilize around 180 epochs, which means that further training would not bring significant improvement. Finally, two result files were obtained: best.pt and last.pt. The best.pt file contains the best weights achieved during training, while last.pt saves the weights obtained in the last epoch. For further analysis, the model saved in the best.pt file was used.

Additionally, to fully assess the model's effectiveness, indicators such as: (a) precision, (b) recall, and (c) F1-score were also calculated for both object detection and segmentation. In the final training run, the following values were obtained:

- **Precision:** 0.99 – means that almost all detected objects were correct, and the number of false alarms is negligible.
- **Recall:** 1.00 – the model detected all actual objects in the image.
- **F1-score:** 0.997 – a harmonized average of precision and recall, confirming very high effectiveness in both detection and segmentation.

High and stable values of these indicators confirm that the model is well-suited to the task, characterized by excellent detectability and low susceptibility to misclassifications, which translates into its great potential in industrial quality control applications.

7. Data analysis

After training the model, the algorithm's correctness tests were performed. In this stage of research, two evaluation methods were compared. Measurement of geometry using calipers and a scanner with the developed algorithm.

7.1 Prediction of geometric dimensions

The first stage was to manually measure the length and width of rice grains using calipers (Fig. 8). The measurement was carried out for 50 grains, and the calculated average value was the reference value for determining the prediction error for the model.

Fig. 8 Measurement of geometric features

Then, the same grains were placed on the measurement stage and subjected to analysis using the previously trained model based on the YOLOv8-seg architecture. The prediction result is shown in Figure 9.

Fig. 9 Result of the prediction of the trained model

As a result of image segmentation, correct segmentation into separated ROI areas (grain) and background was obtained. The measurement was performed 4 times, each time obtaining the same dimensions of the grains. The relative error results are presented in Table 2.

Table 2.

Comparison of the measurement of geometric features made with different techniques along with the determination of relative error.

7.2 Fraction division

The next stage of the analysis was to test the functionality of the algorithm designed to classify grains into fractions according to specific standards (Polski Komitet Normalizacyjny, 2004). The analysis was carried out on a more complex image, in which, after prediction and calculation of grain dimensions, they were assigned to appropriate fractions. The algorithm also generates additional data, such as the average length of all grains, and other geometric values of each grain, and a summary of classification results broken down by fractions, which is illustrated in the figure.

Fig. 10 Result of the prediction divided into fractions

7.3 Analysis of a full-size scan

Analyzing high-resolution, large A4 images is a significant challenge because this process requires a substantial amount of RAM on the graphics card. In our case, the memory demand exceeded the capabilities of the available specialized graphics processing units (GPUs) provided by the Google Colab platform. To solve this problem, the original image was divided into smaller fragments of 640 × 640 pixels, which are optimal for the YOLOv8-seg architecture. To prevent problems related to detecting grains at the borders of separated images, which could lead to incorrect recognition of fragments of the same grain as separate objects, an additional strategy of shifting the division points was applied. This allowed eliminating those image fragments that were at the borders during the stitching stage by using them from the next image fragment. The effect of this solution is presented in Figure 11. As a result of the division, a grid consisting of 15 rows and 11 columns was obtained, which gave a total of 165 image fragments intended for analysis using the deep learning model.

Fig. 11 Split the image into 640x640 squares with backshift

After division, all fragments were subjected to prediction, and then the elements were re-merged to reconstruct the entire image. Detected objects located at the division boundaries were removed if they appeared in overlapping areas, which prevented their duplication. Thanks to the use of shifting during the division of squares, the detected objects were correctly assigned to one fragment. The final result of detection and merging is shown in Figure 12. All regions of interest (ROI) were correctly combined, without artifacts resulting from incorrect merging, and the absence of cases of duplicate detections proves the high effectiveness of the applied algorithm and model.

Fig. 12 Prediction result after image stitching

Summary and Conclusion

Based on the obtained results, the assumed research hypothesis can be accepted, indicating that the trained model based on the YOLOv8-seg architecture effectively fulfills its task. It was shown that the application of a deep learning model combined with advanced image processing techniques enables precise geometric analysis of rice grains in images with a complex structure containing many elements. Despite minor discrepancies in prediction results regarding the dimensions of individual grains and the relative error compared to actual dimensions, amounting to less than 2%, the system effectively classifies grains into appropriate fractions. These results confirm that the model is characterized by high accuracy and reliability, which makes it a valuable tool in the analysis of images of complex structures. The confirmed high accuracy and reliability of the model, with a precision of 0.99, recall of 1.00, and F1-score of 0.997 (obtained in the last epoch for the best.pt model), is also reflected in the context of broader research on automated grain analysis. These results are competitive with those presented in the literature, where advanced deep learning methods, such as various variants of the YOLO or Mask R-CNN architecture, form the core of research. For example, in wheat grain detection tasks using modified YOLOv8, mAP levels of 94.7% were achieved (Ma et al., 2024), and in brown rice segmentation, accuracies of around 96% were obtained (Li et al., 2022). Although direct comparison of metrics between different studies is complex due to different datasets and task specificities, the obtained indicators suggest an excellent fit of the developed model to the problem of rice segmentation and geometric analysis. Considering these promising results, further development could focus on exploring even newer neural network architectures, implementing attention mechanisms, or advanced data augmentation to potentially increase robustness and further minimize prediction errors.

Bibliography

Aukkapinyo, K., Sawangwong, S., Pooyoi, P., Kusakunniran, W. 2020. *Localization and Classification of Rice-grain Images Using Region Proposals-based Convolutional Neural Network.* International Journal of Automation and Computing, 17(2), 233–246.

Bienvenido, O., Juliano, A. P. P. 2019. *Rice (Fourth Edition*). Chemistry and Technology.

Borowy, T., Kubiak, M. S. 2014. *Wartość technologiczna i żywieniowa ryżu.* Przegląd Zbożowo-Młynarski, 58(3), 9-11.

Chollet, F., 2019. *Deep Learning: Praca z językiem Python i biblioteką Keras.* Helion.

Dharmik, R. C., Chavhan, S., Gotarkar, S., Pasoriya, A., 2022. *Rice quality analysis using image processing and machine learning*. 3C TIC. Cuadernos de desarrollo aplicados a las TIC**,** 11(2), 158-164.

Elgendy M. 2020. *Deep Learning for Vision Systems.* Manning Publications.

Forsyth, D. A., Ponce, J. 2011. *Computer Vision: A Modern Approach.* Prentice Hall.

Gao, Q., Li, H., Meng, T., Xu, X., Sun, T., Yin, L., Chai X. 2024, *A Rapid Construction Method for High-Throughput Wheat Grain Instance Segmentation Dataset Using High-Resolution Images*. Agronomy, 14(5), 1032.

Geron, A. 2020. *Uczenie maszynowe z użyciem Scikit-Learn i TensorFlow.* Helion.

Gilanie, G., Javed, M., Rauf, B., Cheema, S., Latif, A., Perveen, S., Sajid, M., Saeed, M. 2021. *RiceAgeNet: Age Estimation of Pakistani Grown Rice Seeds using Convolutional Neural Networks*. International Journal of Computational Intelligence in Control**,** 13(2), 831-843

Gonzalez, R. C., Woods, R. E. 2001. *Digital Image Processing (Second Edition).* Prentice Hall.

Goodfellow, I., Bengio, Y., Courville, A. 2016. *Deep Learning.* The MIT Press.

He Y., Fan B., Sun L., Fan X., Zhang J., Li Y., Suo X. 2023. *Rapid appearance quality of rice based on machine vision and convolutional neural network research on automatic detection system*. Frontiers in Plant Science, 14:1190591

Howard, J., Gugger, S. 2020. *Deep Learning for Coders with fastai and PyTorch.* O'Reilly Media.

Krizhevsky, A., Sutskever, I., Hinton, G. E., 2017, *ImageNet classification with deep convolutional neural networks.* Communications of the ACM, Volume 60(6), 84-90

Li, S., Li, B., Li, J., Liu, B. 2022. *Brown Rice Germ Integrity Identification Based on Deep Learning* Network. Journal of Food Quality,

Ma, N., Li, Z., & Kong, Q. 2024. *Wheat grains automatic counting based on lightweight YOLOv8.* INMATEH Agricultural Engineering.

Matthes, E. 2016. *Python: Instrukcje dla programisty.* Helion.

Mavaddati, S., Razavi, M., 2024. *A CNN-LSTM-based Approach for Classification and Quality Detection of Rice Varieties*. Journal of AI and Data Mining, 12(4), 473-485.

Polski Komitet Normalizacyjny. 2004. *Ryż – Wymagania.* PN-ISO 7301:2004. Warszawa: Polski Komitet Normalizacyjny.

Ramalho, L. 2015. *Zaawansowany Python.* APN Promise.

Rath, S. 2023. *Getting Started with YOLOv5 Instance Segmentation. learnopencv.com/yolov5-instance-segmentation* [dostęp: 29.08.2023].

Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016, *You Only Look Once: Unified, Real-Time Object Detection.* Computer Vision and Pattern Recognition 2016, 779-788

Szelski, R. 2022. *Computer Vision Algorithms and Applications (Second Edition).* Springer Nature Switzerland.

Ultralytics YOLOv8. *github.com/ultralytics/ultralytics* [dostęp: 29.08.2023].

Velesaca, H. O., Suárez, P. L., Mira, R., Sappa, A. D. 2021. *Computer vision-based food grain classification: A comprehensive survey*. Computers and Electronics in Agriculture, 187, 106287.

Wei, J., Ni, L., Luo, L., Chen, M., You, M., Sun, Y., Hu, T., 2024, *GFS-YOLO11: A Maturity Detection Model for Multi-Variety Tomato.* Agronomy , 14, 2644.

Xu, X., Geng, Q., Gao, F. 2023. *Segmentation and counting of wheat spike grains based on deep learning and textural feature.* Plant Methods, 19, 77.

Ziarno, M., Zaręba, D. 2008. *Ryż - cenny składnik żywności.* Przemysł Spożywczy, 62(5), 22-26.

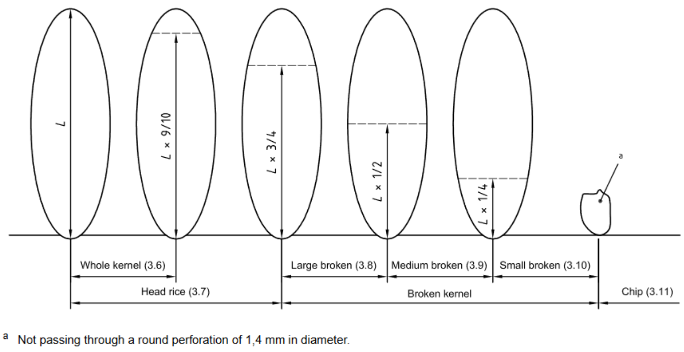

Fig. 1 Grain sizes and fractions of grain parts (Polish Committee for Standardization, 2004)

Fig. 2 Example image after the split operation

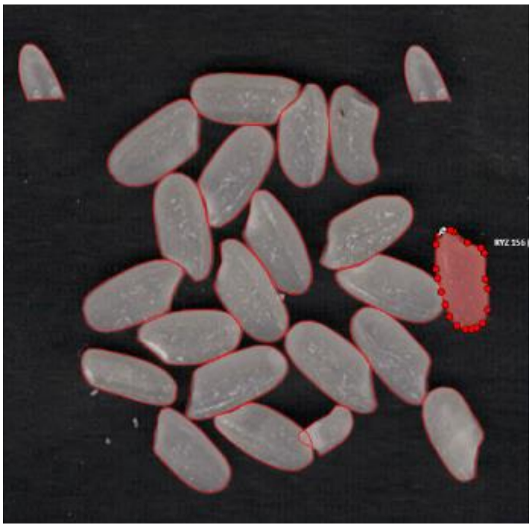

Fig. 3 Sample image with labeled kernels

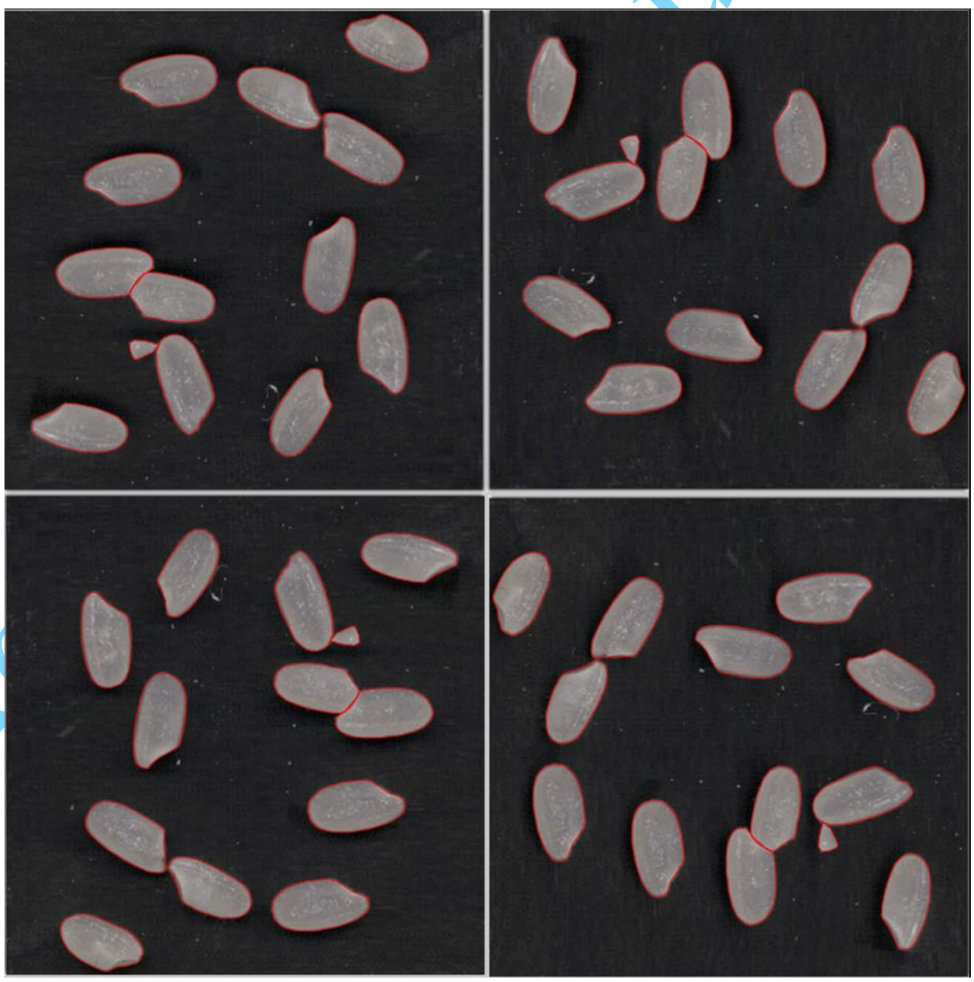

Fig. 4 Example data augmentation result

```
train: ../train/images
val: ../val/images
test: ../test/images

nc: 1
names: ['Ryz']
```

Fig. 5 Configuration file structure

Table 1.

Results of the initial training of YOLOv8-seg models

Model	seg_loss	RAM [GB]
YOLOv8n-seg	0,3846	4,2
YOLOv8s-seg	0,3503	7,5
YOLOv8m-seg	0,3485	12,5
YOLOv8l-seg	0,3282	15,1
YOLOv8x-seg	0,4109	17,5

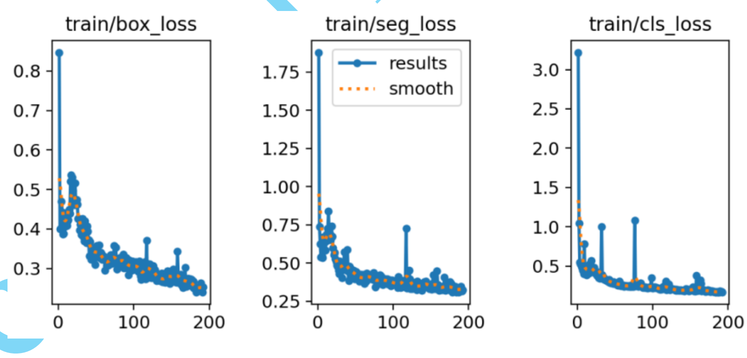

Fig. 6 Loss Value Charts

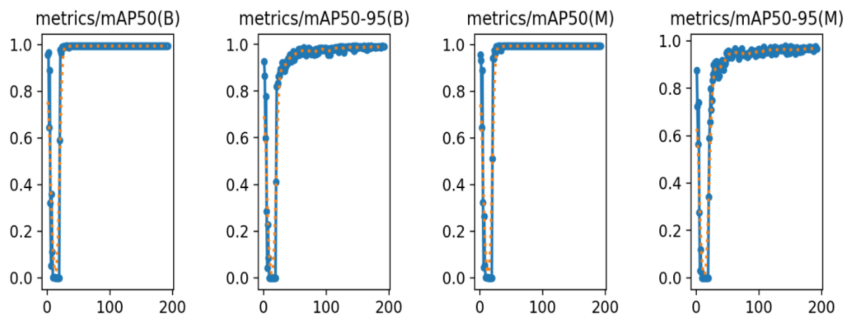

Fig. 7 Average precision (mAP) plots

Fig. 8 Measurement of geometric features

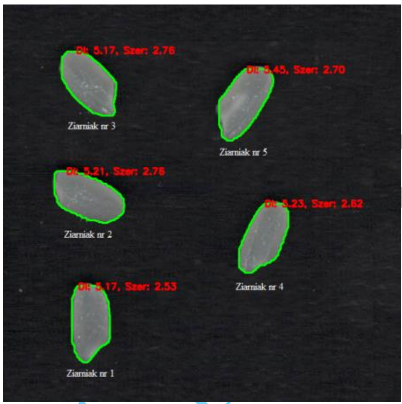

Fig. 9 Result of the prediction of the trained model

Table 2.

Comparison of the measurement of geometric features made with different techniques along with the determination of relative error.

	Caliper	model	Relative error [%]
Length [mm]	5.27	5.25	0.417
Width[mm]	2.69	2.67	0.595

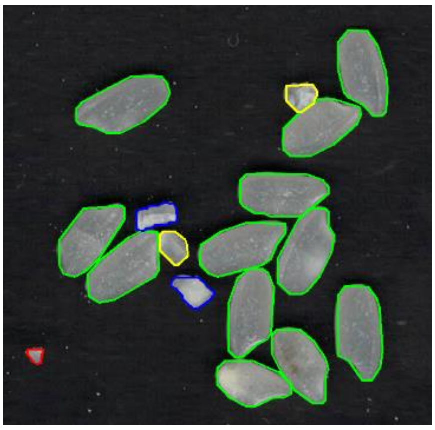

Fig. 10 Result of the prediction divided into fractions

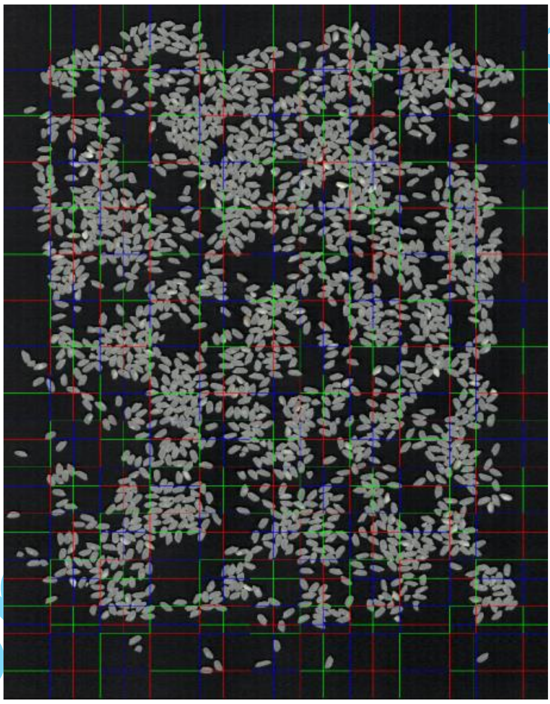

Fig. 11 Split the image into 640x640 squares with backshift

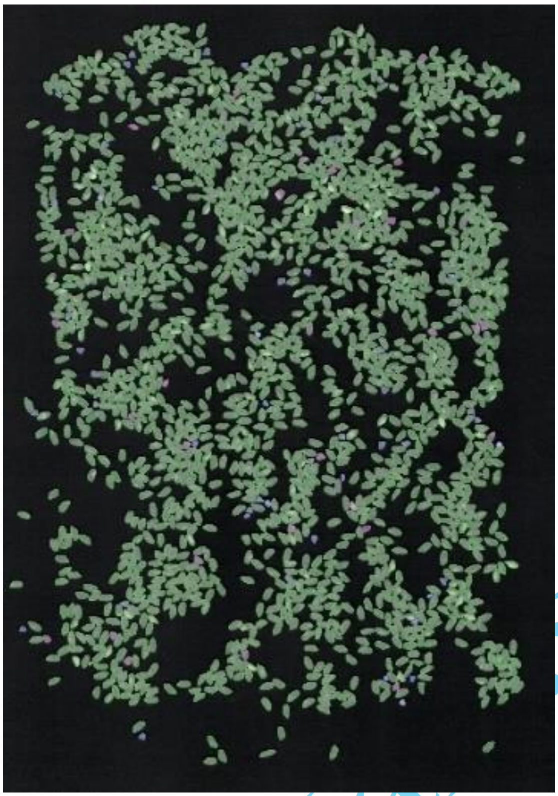

Fig. 12 Prediction result after image stitching