Identification and Classification of Mechanical Damage During Continuous Harvesting of Root Crops Using Computer Vision Methods

ALEKSEY OSIPOV, VYACHESLAV SHUMAEV, ADAM EKIELSKI, TIMUR GATAULLIN, STANISLAV SUVOROV, SERGEY MISHUROV, AND SERGEY GATAULLIN

1Information Security Department, Financial University, Government of the Russian Federation, 125993 Moscow, Russia
2Faculty of Power Engineering, Bauman Moscow Technical University, 105005 Moscow, Russia
3Department of Production Engineering, Warsaw University of Life Sciences, 02-776 Warsaw, Poland
4Department of Mathematical Methods in Economics and Management, State University of Management, 109542 Moscow, Russia
5Department of Applied Informatics, Moscow Polytechnic University, 107023 Moscow, Russia
6Department of Economics, Management and Finance, Ivanovo State Polytechnic University, 153000 Ivanovo, Russia
7Department of Research and Development, Garpix, LLC, 153027 Ivanovo, Russia
8Department of Data Analysis and Machine Learning, Financial University, Government of the Russian Federation, 125993 Moscow, Russia

Corresponding author: Vyacheslav Shumaev (chubchic@gmail.com)

ABSTRACT Detecting sugar beetroot crops with mechanical damage using machine learning methods is necessary for fine-tuning beet harvester units. The Agrifac HEXX TRAXX harvester with an installed computer vision system was investigated. A video camera (24 fps) was installed above the turbine, which receives the dug-out beets after the digger and is connected to a single-board computer. At the preprocessing stage, static and insignificant image details were revealed. Canny edge detector and excess green minus excess red (ExGR) method were used. The identified areas were excluded from the image. The remaining areas were glued with similar areas of another image. As a result, the number of images entering the second stage of preprocessing was reduced by half. Then Otsu’s binarization was used. The main stage of image processing is divided into two sub-stages: detection and classification. The improved YOLOv4-tiny method was chosen for root crop detection using a single-board computer (SBC). This method allows processing up to 14 images of 416 × 416 pixels with 86% precision and 91% recall. To classify root crop damage, we considered two algorithms as candidates: 1. bag of visual words (BoVW) with a support vector machine (SVM) classifier using histogram of oriented gradients (HOG) and scale-invariant feature transform (SIFT) descriptors; 2. convolutional neural networks (CNN). Under normal lighting conditions, CNN showed the best accuracy, which was 99%. The implemented methods were used to detect and classify blurred images of sugar beetroots, which were previously rejected. For improved YOLOv4-tiny precision was 74% and recall was 70%. CNN classification accuracy was 92.6%.

INDEX TERMS Convolutional neural networks, blurred image classification, defect identification, fast detection, machine learning, YOLOv4-tiny.

I. INTRODUCTION

Sugar beet is one of the main crops in the world, which is usually harvested mechanically with multi-row self-propelled harvesters. In this case, the leaves are removed from the beets along with the petioles and the crown, and the root rises from the soil during the harvesting process. The soil from the roots is removed from the clearing zone, and then the roots are transported to the harvester tank and stored. The quality of sugar beet is strongly affected by the conditions of root damage, bruising, and tearing during harvesting [1]. The main reason that can lead to root crops diseases and the loss of their mass is mechanical damage. Root vegetables use energy to heal wounds, which leads to a decrease in the amount of sugar obtained from them [2]. The main factors affecting the magnitude and characteristics of damage to root crops are soil moisture, varietal characteristics, and size of plants, as well as the design and operation of beet harvesters [3]. Sugar beet can get mechanical damage in all processes and components of the harvester [4]. Modern beet harvesters are
high-tech complexes that include a unit for removing and grubbing, a separating complex for cleaning root crops, and a bunker for beets with an unloading conveyor. Each of the mechanisms has many settings depending on the harvester model. Failure in any of them leads to crop loss. In particular, this is due to improper adjustment of the beet head cutter, incorrect adjustment of the depth and distance between the openers of the root crop grubber, incorrect displacement of the harvesting unit relative to the beet rows, the incorrectly selected rotation speed of the separating complex, conveyor, elevator, etc. [5], [6]. A significant part of the losses can be avoided by installing adaptive devices based on computer vision systems.

The authors of the article [7] used a digital two-dimensional imaging system coupled with convolutional neural network (CNN) to detect defects in harvested sugar beet. They used high fps video cameras and artificial lighting to produce high quality images. Various detector models based on the CNN, including You Only Look Once (YOLO) v4, region-based fully convolutional network (R-FCN) and faster regions with convolutional neural network features (Faster R-CNN) were developed. The authors obtained good results in detecting damage, however, their system does not allow real-time operation.

The work [8] is devoted to real-time tubers image processing, under normal weather conditions, using a cheap 4 megapixels RGB camera. The computer vision system was installed on a Grimme SE-170/60 harvester. The authors implemented the algorithm based on Mask R-CNN. The method quality is highly dependent on the tuber density, which, in our opinion, does not allow us to recommend it for use without modification. The work [9] is devoted to the problem of development and adjustment of the harvester adaptive systems. The profiling mechanism and hydraulic driving system were developed. The interaction between the soil and the harvester’s profiling mechanism was explored. Profiling control accuracy ranged from 86% to 92%. The harvester’s adaptive system did not use a computer vision system. In our opinion, such adaptive systems have a serious drawback, they cannot be customized to the individual characteristics of each root crop.

In digital agriculture [10], [11], computer vision systems are used to quickly detect and count plants [12]–[15], to determine their ripeness and diseases [16]–[20], as part of systems to protect against weeds and pests [21], [22], to determine the position of cattle [23]. In recent years publications have shown that the problem of identifying diseased or mechanically damaged fetuses on transportation systems such as conveyor belts, drums, turbines and etc. can be successfully solved using algorithms that provide image processing [24]–[31].

When working with such images, it should be borne in mind that not all of their area is suitable for processing. There are static parts that get in the way of tracking root crops. Such parts can be removed using special algorithms. They can use the interframe difference method, the Gaussian background difference method, the ViBe background extraction algorithm [32], and its advanced version ViBe + [33]. These methods work on incomplete images: a convolutional network with a spatial graph (SGCN) [34], a method based on the U-Net architecture [35], and a local descriptor based on several grids and a dimensionality reduction approach [36], [37].

Further, the search for objects in the image can be performed using the following methods: the visual words method, neural network architectures such as R-CNN, Fast R-CNN, Faster R-CNN, and the Viola-Jones method, which can be considered one of the best in terms of the ratio of recognition efficiency/speed of work [38]. This method usually searches for faces and facial features by the general principle of the scanning window, however, there are successful attempts to use it as a detector when detecting potato tubers on a conveyor belt [39].

Most often, convolutional neural networks are used for object classification tasks. However, as the authors of [40] showed, convolutional neural networks are not designed to work with high-resolution images on devices with weak processors. To obtain an acceptable result, it is necessary to use large kernels (for example, 7 × 7 or 9 × 9) or a large number of layers to obtain an acceptable field susceptibility using convolutional layers [41]. Both of these schemes lead to a very significant slowdown in the system. Therefore, most low-end systems are limited to image sizes of less than 41 × 41 pixels to achieve acceptable image processing time. Moreover, the processing of each such frame can reach several seconds. In conditions of a continuously moving harvester, this is unacceptable. Under these conditions, algorithms with a favorable speed/resources ratio and power deserve special attention [42].

The currently popular modifications of the convolutional neural network, such as R-CNN, Fast R-CNN, Faster R-CNN, outperform the visual word method [43] in these characteristics, however, the visual word method should not be underestimated. For example, the histogram of oriented gradients (HOG) descriptor shows good results in low visibility conditions, when neural architectures practically stop working. This is important for the fieldwork with no good lighting.

One-pass detectors SSD (Single Shot MultiBox Detector), RetinaNet, and YOLO are considered popular high-speed methods of image classification for object recognition [44]–[47]. At the same time, tests show that modern architectures YOLOv3, YOLOv4 surpass their counterparts in performance and accuracy [48]–[52]. Although YOLO’s architecture requires relatively large processing resources of the image analysis system, its high speed results in the possibility of using it in real-time systems [53]. Also, due to the use of the grid approach, these networks can be used to crop the background [48]. The modular structure of these algorithms allows them to be modified for specific tasks, further improving performance and quality metrics. So, for example, the version with densely connected convolutional
networks is intended to improve the reuse of functions during transmission in the model [54], [55], and the version YOLOv3-tiny takes up a small amount of memory and gives a higher FPS [56]. The YOLOv4-LITE version uses MobileNetv2 as the model’s backbone and uses the Do-Conv convolution [57]. The Fusion-YOLO version uses CSPDenseNet and function fusion modules to maximize gradient flow differences, and the CSPResNeXt network to reduce excess gradient flow [58].

Computer vision systems used in agriculture should be as adaptive as possible and easily integrated into production processes [59]. They should not introduce additional restrictions on the processing speed, the location of agricultural equipment units, and be as informative as possible.

The main limitation of using optical methods is blurring of image in working conditions, resulting from vibrations of working elements of agricultural machines, transferred to optical system of camera. The problem of blurry images was solved in [60]. Using coffee beans as an example. They developed an algorithm based on deep neural networks, which showed good results with a slight blurring of grain boundaries. However, with the significant blurring of images, this method does not give satisfactory results.

Other researchers often mention the combination of YOLO and GAN-type architectures. Thus, in [61], the YOLOv4 Model combined Attention Mechanism and DeblurGanv2 is used for real-time gesture recognition. Reference [62] used deblurring methods for remote sensing tasks: Reference [63] introduces a two-phase framework of deblurring and object detection, by adopting a slimmed version of the deblurring generative adversarial network model and a YOLOv2 detector. The work [64] is devoted to Deblur-YOLO: real-time object detector with efficient motion deblurring.

Therefore, according to the authors, methods based on the YOLO architecture are currently best suited for detecting damage to sugar beets using digital cameras installed on the harvester and will be analyzed in this work. There are relatively few publications comparing fast algorithms used in vision systems designed to assess the condition of root crops. The purpose of this study is to find the optimal detector for sugar beetroot and to determine the optimal method for classifying defects in the mechanical processing of the root crop.

II. MATERIALS AND METHODS

A. DATA COLLECTION AND PREPROCESSING

For the experiment, we used a beet harvester model Agrifac HETX TRAXX with a hitch, designed to process 12 rows of beets at the same time. Digging up the beets begins with the foliage removal. This process is carried out in two stages, first, the main part of the tops is removed by the topper, so that it does not interfere with the further root crops processing. The operator from the cabin regulates the scalping system after the topper (see Figure 1). It removes the remaining parts of the leaves as accurately as possible. After the tops are cut, the roots are dug up by a lifting system. The working part of it is the digging shares, which are adjusted using the digger wheels. The dug-out beets are fed to turbines, which provide a continuous flow of root crops for further cleaning and transportation. Damage to beets can be caused by the wrong haulm cut or by a broken tail.

After investigating the possible attachment points for video cameras, the availability for installation and maintenance of computer vision systems, the information content of the resulting image, and the possibility of minimizing the time for reconfiguring the cutting, rotating, and other mechanisms, several promising observations points were selected. One of them is located directly in front of the driver’s cab. A video camera with a frame rate of 24 fps was installed in it. The image from the video camera is shown in Figure 2. This figure shows an image of a rotating turbine, using centrifugal force to separate the roots from the ground.
Figure 4 shows three superimposed consecutive images of this turbine, processed using the Canny edge detector.

Each image is represented by a line of its own color. The white line characterizes the static parts of the image. These parts are of no interest to us.

During the initial run, they are fixed, a mask is formed that replaces these parts with black. When this mask is applied, only the dynamic parts remain in the image.

It was found that the optical system used by us with an increase in the speed of movement of root crops associated with the turbine rotation can no longer give clear images of Figure 4.

The contours of root crops in the lower part of the image are not clear-cut and open. However, some areas are quite suitable for processing. A slight displacement of successive images relative to each other is due to the harvester movement. The root crop is still in the ground, its upper part has been cut off and by the appearance of the cut, it is possible to determine how effectively this was done by the haulm removal device. Incorrect setting of this device will result in significant yield loss and must be carefully monitored by our system. When the root crop rotates, the system manages to fix its image from several sides and determine possible damage. This allows you to instantly react to any deviation of the burrowing share from the movement along the beet row.

B. EQUIPMENT SELECTION

When choosing equipment, we settled on the use of single-board computers. This choice greatly simplifies the task of positioning and powering the computer vision system. The equipment of the NVIDIA corporation Jetson Nano was used. Unlike its closest competitor, the Raspberry Pi 4, the NVIDIA Jetson Nano has a 128-core Maxwell GPU clocked at 921 MHz and a large number of interfaces. It supports the NVIDIA CUDA, cuDNN, and TensorRT software libraries and several popular AI frameworks and algorithms such as PyTorch, Caffe, MXNet, etc. It supports CSI and USB cameras.

C. METHOD SELECTION

Object detection algorithms are constantly being improved. If only a few years ago there was no question of reliable operation of algorithms operating in real-time, now such algorithms exist. In this article, we will select the most suitable ones. To detect root crops, we will choose one of the YOLO methods. To classify root crop damage, we considered two algorithms as candidates: 1. bag of visual words (BoVW) with a support vector machine (SVM) classifier and various descriptors; 2. convolutional neural networks.

1) DESCRIPTORS USED

The HOG method assumes that the type of distribution of image intensity gradients makes it possible to accurately determine the presence and shape of objects located on it. The image is divided into cells. In cells, histograms of directional gradients of internal points are calculated. They are combined into one histogram (h = f(h1, . . . , h_k)) after which it is normalized by brightness. We can get the normalization factor in several ways, but they all show approximately the same results, so let’s take one of the options:

$$h_L = h/\left(\|h\|_2^2 + \varepsilon^2\right),$$  

where $\|h\|_2$ is the norm, $\varepsilon$ is some small constant. When calculating gradients, the image is convolved with kernels $[-1, 0, 1]$ and $[-1, 0, 1]^T$, resulting in two matrices $D_x$ and $D_y$ of derivatives along the x and y axes, respectively. These matrices are used to calculate angles and values (modules) of gradients at every image point.

The scale-invariant feature transform (SIFT) descriptor is used to extract feature points from the image, which are later used in classifiers. The key point in finding them is building a pyramid of Gaussians and the difference of Gaussians. Image blurred with a Gaussian filter

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y),$$

where $L(x, y, \sigma)$ is the value of the Gaussian at the point with coordinates $(x, y)$ and blur radius $\sigma$; $G(x, y, \sigma)$ is the Gaussian kernel; $I(x, y)$ is the value of the original image; $*$ is the convolution operation. The difference of Gaussians is an image obtained with pixel-by-pixel subtraction the original image Gaussian from the different blur radius $(k\sigma)$ Gaussian

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma).$$

When moving from one level of the Gaussian and Gaussian difference pyramid to another, the image dimensions are halved. After building the pyramid, key points are determined, which are the Gaussian differences local extremes. False key points are discarded, and for the remaining ones, their orientation is calculated. We determine the gradient value $m$ and direction $\theta$ from the formulae (4) and (5), as shown at the bottom of the next page.

The SIFT method operates the descriptor as a vector. The method takes a $4 \times 4$ square area centered at the particular point and rotates it according to the singular point’s direction. Each area element indicates the gradient value in eight directions.

2) BOVW METHOD

BoVW is used to improve the performance of descriptors. Since the principle of operation of BoVW with all descriptors...
is the same, we will give an example with the HOG (Figure 5). The descriptor itself and examples of its use can be found in [3].

This approach considers the blocks as key parts of the plant, and each block’s HOG represents the local information of the corresponding part. Next, we cluster the HOG of all the blocks in the training set into homogeneous groups using K-means, and the centers will be the mean value of the blocks’ HOG within the cluster. (We then group the HOGs of all blocks in the training set into homogeneous groups using K-means, and the centers will be the average of the HOG blocks in the cluster.) These centers will play the role of visual-words in BoVW.

3) YOLO
At the moment, three YOLO modifications are most actively used: YOLOv3, YOLOv4, and YOLOv5. Compared to most of their competitors, these algorithms show amazing results. However, although they are single-pass, the large number of convolutional layers does not allow them to process a large number of object images at the same time, so we were interested in their lightweight versions YOLOv3-tiny and YOLOv4-tiny. YOLOv3-tiny and YOLOv4-tiny and are simplified versions of YOLOv3 and YOLOv4 respectively. They have a faster detection rate while guaranteeing high accuracy. This allows these networks to be installed on devices with low computing power. In addition, we studied one modification of YOLOv4-tiny (see Figure 6), which seemed to us the most promising improved YOLOv4-tiny [65].

It consists of a backbone, a neck, and a detector head. The backbone is used to extract features from images from video frames, and the improved YOLOv4-tiny uses the CSPdarknet53_tiny network. Compared to CSPdarknet53 [66], CSPdarknet53_tiny uses only three CSP modules and replaces the Mish activation function with the Leaky ReLu activation function, which significantly reduces the complexity and number of parameters of the function extraction network [67]. The neck is used to merge objects into a network of feature pyramids to extract feature maps at various scales [68].

4) CNN IMAGE CLASSIFICATION
The VGGNet network was chosen for this study. VGG-16 consists of five different blocks that are set in series so that the output of each block is defined as the input of the next block (see Figure 7). With this architecture, the network extracts properties such as texture, shape, and color from input images.

VGG-16 contains 13 convolutional layers with $3 \times 3$ kernels and five $2 \times 2$ maximum pooling layers. The activation function for each convolutional layer is the ReLU (Rectified Linear Unit) function. This function performs the following mathematical operation on each input:

$$f(x) = \begin{cases} x, & \text{if } x > 0, \\ 0, & \text{otherwise}. \end{cases}$$  

(6)
To diagnose different types of damaged root crops and separate them from undamaged ones, a modified model was developed to classify four types of conditions, namely under topped, well topped, overtopped and broken tail. The modified model based on the VGG-16 configuration is followed by a block classifier. The classifier block contained two maximum pooling layers with a $2 \times 2$ window. Dropout layers were added after each maximum pooling layer for regularization. A smooth layer was then applied to smooth the output of the second dropout layer. The output of this layer was connected to the batch normalization layer, and the output of the batch normalization was passed to the third elimination layer. Finally, the output of the last dropout layer was traversed through a fully connected layer containing four neurons, each corresponding to a probability of four classes.

**D. PERFORMANCE CRITERIA**

The performance of the model was assessed by comparing the samples of root crops, classified by the algorithm, with their visual classification by experts (the traditional method). The criteria used to evaluate performance on the training and test sets were precision, recall, and $F_1$, which were calculated as described in the equations below:

$$
\text{Precision} = \frac{TP}{TP + FP},
$$

$$
\text{Recall} = \frac{TP}{TP + FN},
$$

$$
F_1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}},
$$

where TP (true positive) indicates the number of correctly detected root crops; FP (False Positive) indicates the number of falsely detected root crops and FN (False Negative) indicates the number of missing roots.

For each category in target detection, a PR curve can be plotted based on accuracy and retention. The AP value is the area between the PR curve and the coordinate axis, and the mAP is the average AP value for all categories.

**E. STATISTICAL ANALYSIS**

The tests were repeated 5 times for each pass of the harvester during the experiment. The mean and standard deviation of TP, FN, and FP values were calculated from the measurements. The obtained values were used to calculate Precision and Recall values.

**III. EXPERIMENTS AND RESULTS**

**A. DATA PREPARATION AND EVALUATION METRICS**

To train models for root crops after harvesting, samples of different classes were selected (see Table 1).

A 5-fold cross-validation method was applied to them. The data was divided into two parts. The training sample was related to the testing sample as 80% to 20% of all data, respectively. The distribution structure of training and testing data is shown in Figure 8.

**TABLE 1. Number of samples used for recognition and classification of sugar beetroots.**

| Object               | Beetroot classification | Beetroot detection |
|----------------------|-------------------------|--------------------|
| Sugar beet under     | 2332                    | 2332               |
| topped               | 549                     |                    |
| well topped          | 689                     |                    |
| overtopped           | 569                     |                    |
| broken tail          | 525                     |                    |
| Not sugar beet       | 2103                    | 2103               |

**TABLE 2. Hyperparameters for methods used for classification and detection.**

| Method | Optimizer | Learning Rate | Loss Function | Epoch | Batch |
|--------|-----------|---------------|---------------|-------|-------|
| VGG-16 | Adam      | 1e-4          | cross entropy | 100   | 8     |
| YOLO   | 3e-3      |               |               | 100   | 32    |

For each fold, 80% of the data was used for training, while the rest was evaluated during the testing phase. The data used in the test phase were shifted in each fold. The parameters used in training are shown in Tables 2 and 3.

Image processing was carried out using algorithms developed using Python, Scikit-learn, Tensorflow, and Opencv.

**B. OBJECT DETECTION PERFORMANCE**

The characteristics achieved by the YOLO models with a clear video recording of sugar beet in the images of rotating turbines from the separating complex are shown in Figure 9.

The optimal balance between Precision and Recall parameters was obtained with $\text{IOU} \geq 0.5$. The IOU operation computes a ratio from the area of intersection and area of the union of the predicted bounding box and the ground truth bounding box [69]. Table 4 shows data on methods for $\text{IOU} \geq 0.5$.

The processing time of one 416 × 416-pixel image was analyzed using the four methods presented above. The slowest among them turned out to be YOLOv4. When installed on our SBC, it was able to process no more than...
TABLE 3. Number of samples used for training and testing computer vision methods for recognizing and classifying sugar beet tubers.

| Object               | Train samples | Test samples |
|----------------------|---------------|--------------|
|                      | Plant         | Crop-weed    |
|                      | classification| discrimination| classification| discrimination|
| Sugar beet under     | 1632          | 1632         |
| topped well          | 384           |              |
| topped over          | 482           |              |
| topped broken tail   | 358           |              |
| Not sugar beet       | 1473          | 1473         |

FIGURE 9. The precision-recall curve for clear video recording of sugar beet for the four most advanced YOLO methods.

TABLE 4. Sugar beet detection results when video recorded clearly by four YOLO methods with an IOU value $> 0.5$.

| Models              | Precision | Recall | F1  | mAP  |
|---------------------|-----------|--------|-----|------|
| YOLO-v3             | 92%       | 91%    | 92% | 93%  |
| YOLO-v4             | 92%       | 91%    | 92% | 94%  |
| YOLOv4-tiny         | 83%       | 81%    | 82% | 81%  |
| improved            | 86%       | 91%    | 88% | 86%  |

4 frames per second. The fastest was the YOLOv4-tiny, its result surpassed YOLO-v4 by 2.5 times and amounted to 10 frames per second. However, this method showed the worst result in detecting core infertility, so we chose the improved YOLOv4-tiny, which allows processing about 9 frames per second.

To speed up the work of the method, we tested the gluing of two consecutive photographs, with the replacement of blocks that do not include the images of root crops. This allowed us to process 14 frames per second already. Taking into account the fact that each root crop hits the image at least 3 times, this speed of the method is sufficient to ensure the operation of the computer system in real-time.

To classify sugar beet crops according to different types of damage, we used BoVW+SVM with two descriptor options: 1. SIFT and HOG; 2. CNN. The CNN descriptor in the case of normal lighting showed the best results for different classes of root crops damage.

The classification accuracy results of the HOG(SIFT)-BoVW-SVM and CNN models are presented in the normalized confusion matrix, shown in Figure 10.

True positive rates (diagonal values) and false-positive rates (column entries besides diagonals) of each class can be observed in this matrix.

C. IMAGE PROCESSING WHEN VIDEO RECORDING IS BLURRY

At the working speed of the turbine rotation (see Figure 1), part of the image obtained from the 24 fps camera turns out to be blurred. Tracking the same root crop in successive frames from a video camera, you can compare its clear image with a blurry one. A clear image is observed in that part of the rotating turbine, where the root crops only fall on it, and a blurry image is observed when the roots begin to rotate at the speed of the turbine from the separating complex. Taking into account the already trained YOLO network, we have collected a new dataset in which the identified and classified root crop is compared with its blurred image.

Taking into account the uneven distribution of root crops by classes and due to the limited computing resources, from each class, an average of 400 objects were randomly selected for the training sample and 200 for the test sample.

The characteristics achieved by the YOLO models with a blurry video recording of sugar beet on the images of turbines from the separating complex are shown in Figure 11. The results of the four YOLO methods with a balance between the Precision and Recall parameters IOU $> 0.5$ are presented in Table 5.

The classification accuracy results of the HOG (SIFT)-BoVW-SVM and CNN models when the video is recorded...
blurry are presented in the normalized confusion matrix, shown in Figure 12.

IV. DISCUSSION

At this stage in the development of computer vision technologies, object classification is no longer a problem. Convolutional neural networks, decision trees, and other methods began to outperform humans several years ago. However, we drew attention to some limitations of these methods associated with their application in real conditions of harvesting. A beet harvester is a very complex machine, consisting of several units, each of which individually ensures the overall quality of the product. The computer vision system can provide fine-tuning of these units in real-time.

At the preprocessing stage, static and insignificant image details were excluded by the Canny edge detector and excess green minus excess red method. As a result, the number of images before Otsu’s binarization was reduced by half. The main stage of image processing is divided into two sub-stages: detection and classification. The improved YOLOv4-tiny method was chosen for detection. HOG (SIFT)-BoVW-SVM and CNN were used for classification.

YOLOv4-tiny allows processing up to 14 images of 416 × 416 pixels with 86% precision and 91% recall. Under normal lighting conditions, CNN showed the best results, which was 99% depending on the damage class.

For blurred image detection and classification, YOLOv4-tiny precision was 74% and recall was 70%. CNN classification accuracy was 92.6%. These methods allow detecting and classifying blurred images of sugar beetroots, which were previously rejected.

The developed method made it possible to classify images of root crops that were previously rejected. The use of such an algorithm in combination with the harvester onboard computer allows to significantly reduce the amount of crop damage.

REFERENCES

[1] M. Bentini, C. Caprara, V. Rondelli, and M. Calici, “The use of an electronic beet to evaluate sugar beet damage at various forward speeds of a mechanical harvester,” Trans. ASAE, vol. 45, p. 547, Dec. 2002, doi: 10.13031/2013.8848.
[2] T. Huijbregts, G. Legrand, C. Hoffmann, R. Olsson, and A. Olsson, “Long-term storage of sugar beet in north-west Europe,” Coordination Beet Res. Int., Holeby, Denmark, Tech. Rep. 1–2013, 2013. [Online]. Available: https://www.nordicbeet.nu/wp-content/uploads/2016/04/CORB1-storage-report-2013-final-131004.pdf.
[3] C. M. Hoffmann and H.-J. Koch, “Sugar beet,” in Crop Physiology Case Histories for Major Crops. Amsterdam, The Netherlands: Elsevier, 2021, pp. 634–672, doi: 10.1016/B978-0-12-819194-1.00020-7.
[4] P. Kolodziej, K. Goracki, and M. Boryga. “Impact characteristics of sugar beet root during postharvest storage,” Int. Agrophys., vol. 33, pp. 355–361, Dec. 2019, doi: 10.31545/intag/110810.
[5] S. Xiao, H. Chai, Q. Wang, K. Shao, L. Meng, R. Wang, B. Li, and Y. Ma, “Estimating economic benefit of sugar beet based on three-dimensional computer vision: A case study in Inner Mongolia, China,” Eur. J. Agronomy, vol. 130, Oct. 2021, Art. no. 126378, doi: 10.1016/j.eja.2021.126378.
[6] S. Schwich, J. Schattenberg, and L. Friechs, “Development of a machine learning-based assistance system for computer-aided process optimization within a self-propelled sugar beet harvester,” Amer. Soc. Agricult. Biol. Eng., St. Joseph, MI, USA, Tech. Rep. 2000952, 2020, doi: 10.13031/ijam.202000952.
[7] A. Nasirahmadi, U. Wilczek, and O. Hensel, “Sugar beet damage detection during harvesting using different convolutional neural network models,” Agriculture, vol. 11, no. 11, p. 1111, Nov. 2021, doi: 10.3390/agriculture11111111.
[8] P. Dolata, P. Wróblewski, M. Mrzygód, and J. Reiner, “Instance segmentation of root crops and simulation-based learning to estimate their physical dimensions for on-line machine vision yield monitoring,” Comput. Electron. Agricult., vol. 190, Nov. 2021, Art. no. 106451, doi: 10.1016/j.compag.2021.106451.
[9] Y. Ni, C. Jin, M. Chen, W. Yuan, Z. Qian, T. Yang, and Z. Cai, “Computational model and adjustment system of header height of soybean harvesters based on soil-machine system,” Comput. Electron. Agricult., vol. 183, Apr. 2021, Art. no. 105907, doi: 10.1016/j.compag.2020.105907.
[10] M. Tatarintsov, S. Korchagin, P. Nikitin, R. Gorokhova, I. Bystrenina, and D. Serdechnyy, “Analysis of the forecast price as a factor of sustainable development of agriculture,” Agronomy, vol. 11, no. 6, p. 1235, Jun. 2021, doi: 10.3390/agronomy11061235.
[11] V. Mizonov and E. Shuina, “A simple model for describing the thermal state of a water pipe at low negative temperatures,” Jpn. J. Heat Mass Transf., vol. 23, no. 2, pp. 293–302, Jul. 2021, doi: 10.17654/HM023020293.
[12] F. A. Kateb, M. M. Monowar, A. M. Hamid, A. Q. Ohi, and M. F. Mridha, “FruitDet: Attentive feature aggregation for real-time fruit detection in orchards,” Agronomy, vol. 11, no. 12, p. 2440, Nov. 2021, doi: 10.3390/agronomy11122440.
[13] A. Ammar, A. Koubaa, and B. Benjdira, “Deep-learning-based automated palm tree counting and geolocation in large farms from aerial geotagged images,” Agronomy, vol. 11, no. 8, p. 1458, Jul. 2021, doi: 10.3390/agronomy11081458.
[14] B. Yang, Z. Gao, Y. Gao, and Y. Zhu, “Rapid detection and counting of wheat ears in the field using YOLOv4 with attention module,” Agronomy, vol. 6, no. 6, p. 1202, Jun. 2021, doi: 10.3390/agronomy6061202.
[15] B. Yan, P. Fan, X. Lei, Z. Liu, and F. Yang, “A real-time apple targets detection method for picking robot based on improved YOLOv5,” Remote
