



Intelligent Defect Recognition System for Fish Product Packaging Using Machine Vision

*A. N. Sheremetev, E. A. Luttseva**

Kamchatka State Technical University,
683003, Klyuchevskaya st., 35, Petropavlovsk-Kamchatsky, Russia

Abstract. This paper presents a study on the development of an automatic quality control system for fish product packaging based on computer vision and convolutional neural networks. The relevance of the work is driven by the need to reduce human factor influence during visual inspection on conveyor lines, especially under high production rates and given the perishable nature of the product. To address this problem, the YOLO26 model was selected. A two-stage architecture is proposed: at the first stage, the YOLO26n model detects the box region in the frame (the frame is then cropped); at the second stage, a binary classifier built upon YOLO26s determines whether a packaging defect is present. This approach reduces classification errors caused by background differences between the dataset and real-world conditions. A custom labelled image dataset was collected on a prototype conveyor line. An experimental comparison of the YOLO26s, YOLOv8s, and YOLO11s classifiers was performed on the test set. Testing on a set of 108 images demonstrated an accuracy of 96.3% with false positive and false negative rates of 3.7% each. It is concluded that the developed system is applicable to the existing conveyor line prototype and can potentially be scaled to industrial fish processing plants.

Key words: neural networks, computer vision, YOLO26, dataset, fishing industry, packaging defect.

Received: 23.06.2026; Revised: 26.07.2026; Accepted: 24.08.2026; First online: 14.09.2026

For citation. Sheremetev A. N., Luttseva E. A. Intelligent defect recognition system for fish product packaging using machine vision. *Vestnik KRAUNC. Fiz.-mat. nauki.* 2026, 54: 2, 106-118. EDN: ZQKLMN. <https://doi.org/10.26117/2079-6641-2026-54-2-106-118>.

Funding. The work was carried out within the framework of the research project of the GBT of the FSBEI HE "Kamchatka State Technical University" (state registration number 126022517947-0)

Competing interests. There are no conflicts of interest regarding authorship and publication.

Contribution and Responsibility. All authors contributed to this article. Authors are solely responsible for providing the final version of the article in print. The final version of the manuscript was approved by all authors.

*Correspondence:  E-mail: luttsevaea@gmail.com

The content is published under the terms of the Creative Commons Attribution 4.0 International License

© Sheremetev A. N., Luttseva E. A., 2026

© Institute of Cosmophysical Research and Radio Wave Propagation, 2026 (original layout, design, compilation)





Интеллектуальная система распознавания дефектов упаковки рыбной продукции с использованием машинного зрения

*А. Н. Шереметев, Е. А. Лутцева**

Камчатский государственный технический университет, 683003, ул. Ключевская 35, г. Петропавловск-Камчатский, Россия

Аннотация. В статье представлено исследование по разработке системы автоматического контроля качества упаковки рыбной продукции на основе методов компьютерного зрения и сверточных нейронных сетей. Актуальность работы обусловлена необходимостью снижения влияния человеческого фактора при визуальной инспекции на конвейерных линиях, особенно в условиях высокого темпа производства и скоропортящегося характера продукта. Для решения данной задачи была выбрана модель YOLO26. Предложена двухэтапная архитектура: на первом этапе модель YOLO26n детектирует область коробки на кадре (кадр обрезается), на втором этапе бинарный классификатор YOLO26s определяет наличие дефекта упаковки. Данный подход позволяет снизить ошибки модели классификации, связанные с некорректным обучением из-за различий фона между датасетом и реальными условиями. Собран собственный размеченный датасет изображений, полученный на прототипе конвейерной линии. Проведено экспериментальное сравнение классификаторов YOLO26s, YOLOv8s и YOLO11s на тестовой выборке. Тестирование на выборке из 108 изображений продемонстрировало точность 96,3% при доле ложноположительных и ложноотрицательных результатов по 3,7%. Сделан вывод о принципиальной применимости разработанной системы на существующем прототипе конвейерной линии и о возможности её масштабирования на промышленные рыбоперерабатывающие предприятия.

Ключевые слова: нейронные сети, компьютерное зрение, YOLO26, датасет, рыбная промышленность, дефект упаковки

Получение: 23.06.2026; Исправление: 26.07.2026; Принятие: 24.08.2026; Публикация онлайн: 14.09.2026

Для цитирования. Sheremetev A. N., Luttseva E. A. Intelligent defect recognition system for fish product packaging using machine vision // Вестник КРАУНЦ. Физ.-мат. науки. 2026. Т. 54. № 2. С. 106-118. EDN: ZQKLMN. <https://doi.org/10.26117/2079-6641-2026-54-2-106-118>.

Финансирование. Работа выполнена в рамках научно-исследовательского проекта кафедры «Промышленное рыболовство» ФГБОУ ВО «Камчатский государственный технический университет» (№ 126022517947-0).

Конкурирующие интересы. Конфликтов интересов в отношении авторства и публикации нет.

Авторский вклад и ответственность. Авторы участвовали в написании статьи и полностью несут ответственность за предоставление окончательной версии статьи в печать.

*Корреспонденция: ✉ E-mail: luttsevaea@gmail.com

Контент публикуется на условиях Creative Commons Attribution 4.0 International License

© Sheremetev A. N., Luttseva E. A., 2026

© ИКИР ДВО РАН, 2026 (оригинал-макет, дизайн, составление)



Introduction

The fishing industry is one of the most dynamically developing sectors of food production, with new technologies for storage, packaging, and processing emerging every year. Quality control under such conditions is critically important, since fish is a perishable product, and any damage to the packaging can compromise its seal, rendering the product unfit for consumption.

Manual quality control has a significant drawback related to the human factor: during prolonged monotonous work, inspectors experience a decline in visual acuity and attention, which can lead to missed small but critical defects for fish products. Moreover, the high speed of modern conveyor lines substantially limits the time available for product inspection. In such circumstances, the use of artificial intelligence technologies becomes appropriate. Neural networks combined with computer vision can eliminate errors and establish a unified standard of inspection that is not subject to the subjective judgment of individual workers.

Review papers, such as [1], show that artificial intelligence is being actively deployed for seafood safety control across all stages of the supply chain. In [2], a comprehensive review of real-time electronic vision technologies for canned fish quality inspection is presented, tracing the evolution from manual methods to automated systems based on hyperspectral imaging and machine learning. With respect to the fishing industry, [3] provides an analytical review of contact-free fish weighing approaches using video cameras and computer vision, demonstrating the broad potential of these technologies in fish processing. In [4], a theoretical development of an automatic fish sorting system based on neural networks is described, while [5] addresses the application of neural networks for the recognition of commercial aquatic organisms under high-fluctuation conditions.

Problem statement

The aim of this study is to develop a system for recognizing packaging defects in fish products based on neural network methods of computer vision.

Classical computer vision methods, which rely on manual feature selection, enable the implementation of simple systems when the defect is clearly expressed, but they perform poorly with deformation-related defects. At the same time, modern computer vision techniques based on convolutional neural networks (e.g., the YOLO family) allow automated visual inspection to be implemented in production environments while accounting for the speed of the conveyor line. However, to achieve high accuracy, they require training on datasets that match specific production conditions.

This work is part of a larger project aimed at creating an automated neural-network-based sorting complex, which involves both software and hardware development. The hardware platform is a prototype conveyor line controlled by a programmable logic controller (PLC) and designed to transport boxed fish products.

The software module analyses incoming images and classifies product quality; defective boxes are then automatically diverted to a reject branch of the conveyor.

Fig. 1 shows the conveyor line prototype developed by the faculty and students of the Department of Control Systems at Kamchatka State Technical University. The equipment for controlling the conveyor line includes a PLC, an HMI panel, a frequency converter, optical sensors, web cameras, and an embedded single-board computer.



Fig. 1. Prototype of the conveyor line

To achieve the stated goal, the following tasks were set:

- to create a labelled dataset of images of fish product packaging with typical defects;
- to train a neural network for real-time defect detection;
- to test the obtained model on an additional data sample under real-world conditions (on the departmental testbed).

Dataset

The input data for the developed system are frames obtained from video streams of two cameras positioned above the conveyor.

The core of the approach is a two-stage image inspection using convolutional neural networks. At the first stage, boxes are detected – they are the critical element ensuring the integrity of the fish product packaging. The detection model extracts (crops) the region containing the box for analysis, after which the cropped region is passed to the second stage, where the classification model is applied.

This approach helps reduce classification errors caused by background differences between the dataset and real-world conditions, since only the box itself is fed into the classifier.

To increase system reliability, two cameras are installed on the laboratory testbed at different angles relative to the conveyor belt. This allows capturing the box from multiple viewpoints and minimising the risk of missing a defect. Images from each camera are processed independently, and the final decision on rejection is made using a logical “OR” rule: if at least one camera classifies the box as defective, it is considered rejected.

The task of detecting boxes on a conveyor line is highly specific and requires a custom-collected dataset, as publicly available datasets for such tasks are either non-existent or do not meet the necessary requirements.

In the authors’ previous work [6], the possibility of using a labelled dataset from the open Roboflow repository was considered. However, the achieved defect detection accuracy was only 59%, which confirmed the inefficiency of off-the-shelf datasets for this task and the need to build a custom one.

For this reason, a proprietary dataset was collected and annotated for the specific task. The dataset preparation was carried out in several stages: data collection, annotation, splitting into training, validation, and test sets, as well as the formation of two separate datasets in different formats for each stage of the analysis system.

Data collection was carried out directly on the conveyor line prototype (Fig. 1) under conditions close to real operation: two web cameras mounted above the belt were used, and recording was performed under both natural and artificial lighting. The prototype design ensures that exactly one box is present in the frame at any time. To create defective samples, the authors artificially produced various types of cardboard packaging damage: tears, punctures, and corner deformations. All these types of damage are grouped into the “defect” class. A total of 432 images were obtained (combined from both cameras), of which 211 contain boxes without visible defects (class “normal”) and 221 contain boxes with damage (class “defect”). The dataset is nearly balanced across the target classes.

Annotation was performed by the authors using the CVAT (Computer Vision Annotation Tool) [7]. Since the defects were intentionally and controllably created at the sample preparation stage, the ground-truth class (normal or defect) was known for each image in advance. The annotation process therefore consisted of formal label assignment and did not require expert judgement.

The annotated data were split into three subsets: training, validation, and test. Stratified splitting was performed to preserve class proportions: the training set comprised 291 images (150 defective, 141 normal), the validation set – 33 images (17

defective, 16 normal), and the test set – 108 images (54 defective, 54 normal). The training set is used directly for model weight optimisation. The validation set is used to monitor model performance across epochs and to trigger early stopping. The test set serves for the final assessment of generalisation ability. Despite the relatively small size of the validation set, it is balanced and sufficient for overfitting control, since the main quality evaluation is performed on the test set. Cross-validation was not applied. All subsets are disjoint: images from the validation and test sets were not used during training.

The limited size of the dataset is due to the restricted number of physical boxes available: several box instances (both defective and non-defective) were used, which were repeatedly captured under varying conditions – different angles and lighting. This volume is sufficient for the initial validation of the key algorithmic solutions and for assessing their potential. The results obtained provide a foundation for the next phase of the study, where the dataset will be expanded with real production footage.

Based on the annotated images, two independent datasets were formed, each corresponding to its own neural network model.

The dataset for the first analysis stage (box detection) contains original frames and text files with the coordinates of the bounding boxes of the "box" class (Fig. 2). Since each frame contains exactly one box, the total number of annotated instances of the 'box' class is also 432.



Fig. 2. Example of the dataset for the box detection stage

The dataset for classification was built from the same images, but they were first passed through the detection model to crop out the extra content and retain only the box region (Fig. 3).

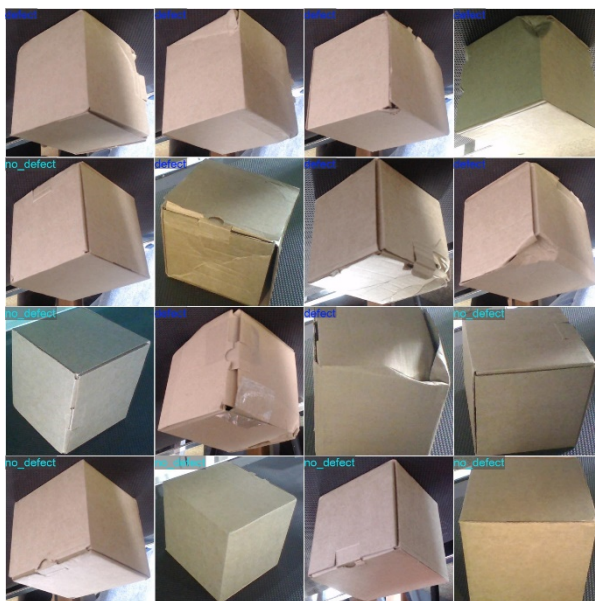


Fig. 3. Example of the dataset for the classification stage

Neural network training

For neural network training, a model from the YOLO family [8] was selected.

YOLO (You Only Look Once) is a family of real-time object detection and image segmentation models. Their key feature is processing a frame in a single pass, ensuring high speed without sacrificing accuracy. Unlike classification-oriented architectures (such as EfficientNet or ResNet), which require additional mechanisms for object localisation, YOLO jointly solves detection and classification tasks, making it an optimal choice for real-time systems. For non-commercial use, the models are distributed under the AGPL-3.0 license, which provides open source and free use for educational and scientific purposes. For commercial use without disclosing one's own source code, a separate enterprise license is available. Thus, the model can be used not only in educational but also in commercial projects.

Comparative studies show that with each new version of YOLO, there is a steady improvement in key metrics: accuracy, speed, and computational efficiency. Review papers such as [9,10] demonstrate a consistent trend of increasing accuracy and efficiency of YOLO family models across versions. In [11], the authors investigated the use of convolutional neural networks of the YOLO family in mobile computer vision systems and showed the high effectiveness of these models for object detection tasks in images.

YOLO26 is the latest model publicly released in January 2026. According to the preprint review [12], YOLO26 retains the versatility of the family, supporting five computer vision tasks simultaneously (object detection, segmentation, classification, pose estimation, oriented object detection), but it is the first model in the line to implement native end-to-end inference for all these tasks without using the NMS algorithm.

Training convolutional neural networks is a resource-intensive task: the main load falls on the graphics processing unit (GPU), since the convolution operation is efficiently

parallelized on CUDA cores. Accordingly, a personal computer with the following specifications was used for training:

- central processing unit: AMD Ryzen 9 7900, 12 cores, 24 threads;
- random-access memory: 32 gigabytes;
- graphics processing unit: Nvidia GeForce RTX 5070 based on the Blackwell architecture, with Tensor and CUDA cores optimized for AI tasks, and 12 gigabytes of video memory.

Before starting the neural network training process, the software environment was fully configured, including the installation and setup of all necessary components for efficient use of GPU CUDA cores. As part of this setup, the NVIDIA CUDA Toolkit version 12.1 was installed, providing full support for computations on the graphics processor.

The models were trained using the Ultralytics library and the Python programming language.

The first-stage model was trained for single-class detection (“box”). The parameter choices were based on dataset size and available computational resources. The following settings were used: model YOLO26n (it has the fewest parameters and high inference speed; chosen because the detector only needs to locate the box, not analyse defects); input resolution 640×640 (standard for the YOLO family); batch size 16 (determined by the GPU memory limit of 12 GB, since a larger value would cause out-of-memory errors); number of epochs 80; training device CUDA; learning rate and augmentation parameters were left at default values. Dropout regularisation was not used, as single-class detection benefits from built-in mechanisms (augmentations, weight decay).

The classifier was trained for binary classification. The specifics of the task, including the limited dataset, required finer tuning. The following parameters were applied: model YOLO26s (accuracy is more important than speed for the classifier, and the “s” model is the optimal choice under given conditions); input resolution and batch size were chosen analogously to the detector (640×640 and 16, respectively); number of epochs 80 with manually specified early stopping; regularisation: dropout=0.3 (to prevent overfitting on the limited dataset); augmentation in the form of random erasing was not used.

For both models, pre-trained weights were used: for the YOLO26n detector – weights pre-trained on the COCO dataset [13]; for the YOLO26s classifier – weights pre-trained on ImageNet [14].

Training results

During training and final model evaluation, a set of metrics for detection and classification tasks is monitored. All metrics are automatically calculated by the Ultralytics library on the validation set at each epoch. For the detection model, the primary focus is on the following metrics [13,15]:

- mAP@0.5 – mean average precision at an Intersection over Union (IoU) threshold of 0.5;
- mAP@0.5-0.95 – the primary metric, averaged over the IoU threshold range from 0.5 to 0.95 with a step of 0.05;
- precision – the proportion of correct detections among all predictions;
- recall – the proportion of correctly detected packages among all annotated ones;
- loss functions (box_loss, cls_loss, dfl_loss).

Fig. 4 shows the graphs of the above metrics and loss functions during training. The X-axis represents epoch numbers, and the Y-axis represents the values of metrics and loss functions.

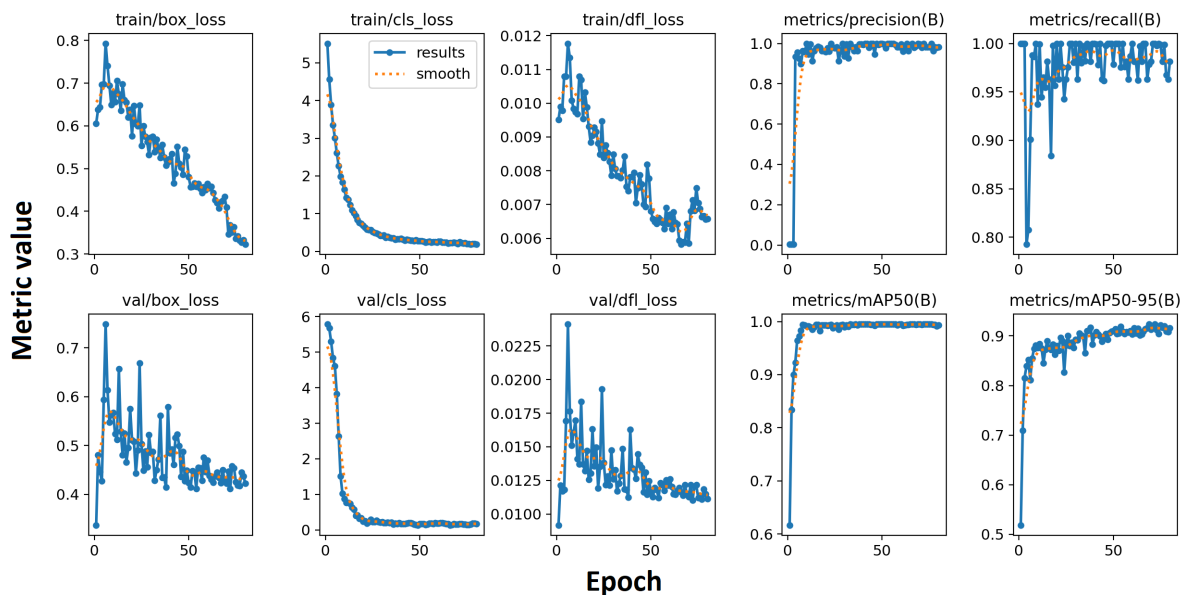


Fig. 4. Training metrics of the detection model

From the graphs, it can be observed that the loss functions gradually decrease, while the mAP metrics increase and tend toward unity. These data indicate successful model training and its readiness for deployment on the prototype conveyor line, as well as the possibility of further fine-tuning if necessary.

For the classifier, the key metric is Top-1 accuracy, which represents the proportion of correctly predicted labels. Fig. 5 shows the training metrics of the classifier. As with the detection model, they demonstrate good learning performance and confirm the model's readiness for use under real-world conditions.

It is important to note that no overfitting was observed for either model: the training and validation loss curves converged synchronously, without divergence, indicating good generalisation capability.

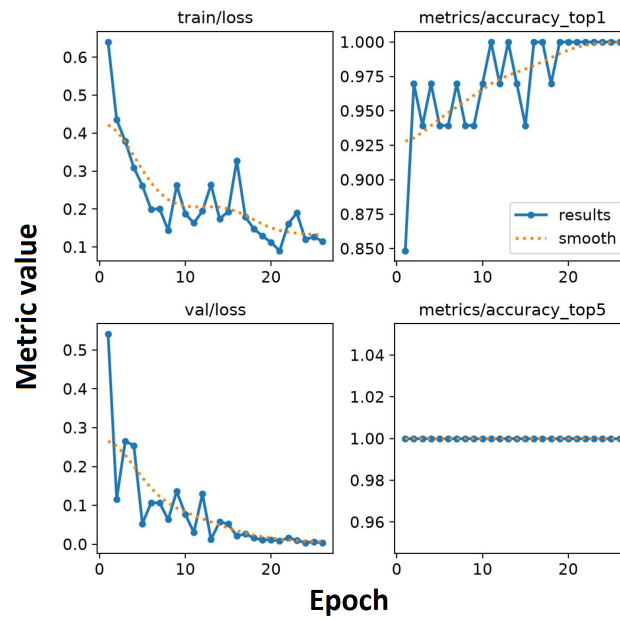


Fig. 5. Training metrics of the classification model

Evaluation on the test set

To evaluate the recognition quality of the trained model, a test set was compiled with examples of both classes (54 normal, 54 defective), not overlapping with the training data, and including both typical and challenging cases (poor lighting, defects at class boundaries).

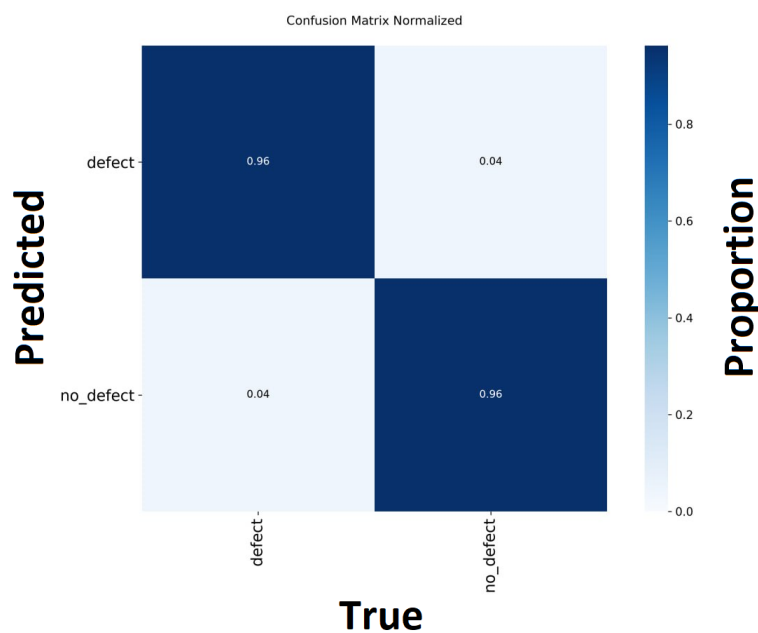


Fig. 6. Confusion matrix for the test set (normalised by true class, values represent proportions)

Fig. 6 shows the confusion matrix for the test set. YOLO26s made 4 errors in total: 2 false negatives (missed defects) and 2 false positives. Thus, the accuracy on the test set was 96.3%, with false negative and false positive rates of 3.7% each.

The obtained results confirm high recognition quality on the static set. However, a final conclusion on the system's operation in real time on a moving conveyor requires additional testing with FPS and per-frame processing time measurements on the target hardware (NVIDIA Jetson Orin Nano). At the time of the work, this equipment was not available, so quantitative performance evaluation was not performed and is planned as the next stage of the research.

Comparison with other YOLO versions

To justify the choice of YOLO26, we performed an experimental comparison of the classifiers YOLO26s, YOLOv8s, and YOLO11s. All models were trained on identical splits and with the same hyperparameters described above. Evaluation was performed on the test set of 108 images. The detector (YOLO26n) was not compared with other versions, since the task of detecting a single class ("box") is relatively simple.

Table 1 presents the comparison results: YOLO26s shows the best accuracy (96.3%) and the smallest total number of errors (4) compared to YOLOv8s and YOLO11s (5 errors each), confirming the soundness of its selection.

Table 1

Comparison of classifier metrics on the test set			
Model	Accuracy, %	False Negatives (FN)	False Positives (FP)
YOLOv8s	95.4	3	2
YOLO11s	95.4	4	1
YOLO26s	96.3	2	2

It is also noteworthy that YOLO26 is designed for deployment on edge devices. According to the official documentation [8], the model exhibits lower latency and stable performance on platforms such as NVIDIA Jetson Orin Nano, making it a more preferable choice for the considered task.

Conclusion

In this work, we developed and experimentally validated a two-stage system for detecting packaging defects in fish products using convolutional neural networks of the YOLO family. A custom annotated dataset was collected, reflecting the specifics of the conveyor line prototype. Two models were trained: a detector (box localisation) and a binary classifier (defect detection).

The training results confirm the high quality of the system. The detection model achieved stable mAP@0.5 and mAP@0.5–0.95 values approaching 1.0. On the test set of 108 images, the classification accuracy reached 96.3%, with false negative and false positive rates of 3.7% each.

It should be noted that real-time performance evaluation on the target hardware platform was not performed due to its unavailability at the time of the work. Furthermore, a systematic quantitative analysis of the system's robustness to variations in illumination, viewpoint, and conveyor speed was not carried out. Although the dataset included images acquired under various conditions and testing covered challenging cases (including poor lighting), separate assessment of each factor's impact on accuracy was not performed and is planned for future work.

Thus, the proposed system can be used within the existing conveyor line prototype and, in the future, scaled to industrial fish processing plants. The obtained results are consistent with the findings of [16], which demonstrated the effectiveness of vision systems for quality control automation in food production. The developed system provides a foundation for automating quality inspection: it reduces human involvement and ensures a unified quality standard. Its adaptation to operate at real-time production speed remains the nearest direction for further research.

References

1. Jiao X., Zhu J., Ye W., Zou H., Yan B., Zhang N., Qiang J., Tao Y., Zhang H., Zhang D., Fan D. Artificial intelligence in smart seafood safety across the supply chains: Recent advances and future prospects, *Trends in Food Science & Technology*, 2025. vol.163, pp. 105161 DOI: 10.1016/j.tifs.2025.105161.
2. Mahatheesan A. J., Sharmilan T. Advancements and Challenges in Real-Time Electronic Vision Technologies for Canned Fish Quality Inspection: A Comprehensive Review, *European Modern Studies Journal*, 2024. vol. 8, no. 4, pp. 414–421 DOI: 10.59573/emsj.8(4).2024.19.
3. Savinov E. D., Ivashko E. E. Analytical review of technologies for contactless "weighing" fish, *Vestnik Astrakhanskogo gosudarstvennogo tekhnicheskogo universiteta. Seriya: Rybnoe khozyaistvo*, 2024. vol. 1, no. 1, pp. 84–94 DOI: 10.24143/2073-5529-2024-1-84-94 (In Russian).
4. Ivanovsky A. N. Application of neural networks in fish sorting systems on fishing fleet vessels, *Vestnik Kerchenskogo gosudarstvennogo morskogo tekhnologicheskogo universiteta. Seriya: Morskie tekhnologii*, 2023. vol. 1, no. 2, pp. 112–118 (In Russian).
5. Alekseev P. P., Kvyatkovskaya I. Yu. Application of neural networks in system of recognition of commercial aquatic organisms during increased fluctuation, *Vestnik Astrakhanskogo gosudarstvennogo tekhnicheskogo universiteta. Seriya: Upravlenie, vychislitel'naya tekhnika i informatika*, 2022. vol. 1, no. 2, pp. 76–86 DOI: 10.24143/2072-9502-2022-2-76-86 (In Russian).
6. Sheremetev A. N., Luttseva E. A. Development of a system for recognizing defects in fish product packaging / *Tekhnicheskaya eksploatatsiya vodnogo transporta: problemy i puti razvitiya*, *Materialy Vos'moi natsional'noi (vserossiiskoi) nauchno-tekhnicheskoi konferentsii* (Petrovsk-Kamchatsky, November 27–28, 2025). Petrovsk-Kamchatsky, Kamchatka State Technical University, 2026, pp. 273–277 (In Russian).
7. CVAT.ai *Data Annotation Platform for Vision AI*, Website <https://www.cvat.ai>.
8. Ultralytics *The ultimate end-to-end computer vision platform*, Website, <https://www.ultralytics.com>.
9. Vychegzhanin S. V., Tatarinova A. G., Myshkin R. E. Comparative analysis of neural network models for detecting unmanned aerial vehicles, *Komp'yuternaya optika*, 2026. vol. 50, no. 2, pp. 17–28 DOI: 10.18287/COJ1728 (In Russian).
10. Beqqali Hassani I., Benhida S., Lamii N., et al. From YOLO V1 to YOLO V11: comparative analysis of YOLO algorithm (review), *International Journal of Electrical and Computer Engineering*, 2026. vol. 16, no. 1, pp. 450–462 DOI: 10.11591/ijece.v16i1.pp450-462.
11. Nebaba S. G., Markov N. G. Convolutional neural networks of YOLO family for mobile computer vision systems, *Komp'yuternye issledovaniya i modelirovanie*, 2024. vol. 16, no. 3, pp. 615–631 DOI: 10.20537/2076-7633-2024-16-3-615-631 (In Russian).
12. Sapkota R., Karkee M. *Ultralytics YOLO Evolution: An Overview of YOLO26, YOLO11, YOLOv8 and YOLOv5 Object Detectors for Computer Vision and Pattern Recognition*, 2026, arXiv: arXiv:2510.09653.

13. COCO Consortium *COCO Dataset*, Website, <https://cocodataset.org>.
14. ImageNet *ImageNet Dataset*, Website, <https://www.image-net.org>.
15. Flach P. *Mashinnoe obuchenie. Nauka i iskusstvo postroeniya algoritmov, kotorye izvlekayut znaniya iz dannykh* [Machine Learning: The Science and Art of Building Algorithms that Extract Knowledge from Data]. Moscow: DMK Press, 2015. 400 pp. (In Russian)
16. Kalabin A. L., Tolstel O. V., Shirkin A. E. Construction of a computer vision system for aligning the contents of packages by a delta manipulator in food production, *Programmnye produkty i sistemy*, 2023. vol. 1, no. 2, pp. 334–341 DOI: 10.15827/0236-235X.142.334-341 (In Russian).

Information about the authors



Sheremetev Aleksey Nikolaevich ✉ – Student, Control Systems Department, Kamchatka State Technical University, Petropavlovsk-Kamchatsky, Russia, ORCID 0009-0002-0200-9159.



Luttseva Ekaterina Aleksandrovna ✉ – Senior Lecturer, Control Systems Department, Kamchatka State Technical University, Petropavlovsk-Kamchatsky, Russia, ORCID 0009-0003-6552-805X.