Advances in Science and Technology Research Journal

Advances in Science and Technology Research Journal, 2026, 20(1), 462–476
https://doi.org/10.12913/22998624/209916
ISSN 2299-8624, License CC-BY 4.0

Received: 2025.07.04
Accepted: 2025.09.26
Published: 2025.11.21

Intelligent tool management with an enhanced six-channel you only look once model and structural similarity index measure analysis

Kacper Marciniak[1*], Paweł Majewski[2], Piotr Lampa[1], Mariusz Mrzygłód[1], Jacek Reiner[1]

[1] Faculty of Mechanical Engineering, Wrocław University of Science and Technology, ul. Ignacego Łukasiewicza 5, 50-371 Wrocław, Poland
[2] Faculty of Information and Communication Technology, Wroclaw University of Science and Technology, ul. Janiszewskiego 11/17, 50-372 Wrocław, Poland
* Corresponding author's e-mail: kacper.marciniak@pwr.edu.pl

ABSTRACT

Tool storage systems are an integral component of the production chain in modern manufacturing facilities. Automated vertical storage systems are commonly employed to store and manage tools and equipment required for rapid replacement or re-tooling during the production process. In such a scenario, any error made by a warehouse operator can disrupt the inventory system, leading to operational issues or even halting the production line. To address the challenges of storage control and operator error identification, this paper proposes a vision-based system capable of detecting changes within the storage space and determining their directionality. The proposed solution leverages a custom synthetic dataset generation process and a hybrid processing method, combining a 6-channel enhanced YOLOv8 (you only look once) model with structural similarity index measure (SSIM) analysis. This approach effectively identifies the location and direction of changes (e.g. object removal or addition) and is characterised by robustness to domain shifts and other disturbances, such as variations in illumination or object relocation, which commonly occur during normal operation. The enhanced model utilises a 6-channel input, integrating 'before' and 'after' images while retaining full colour space information - a capability not achievable with the standard YOLO models. Furthermore, the two-stage processing method that incorporates SSIM analysis significantly improves the recall rate of the developed solution. Comprehensive validation on prepared test datasets demonstrated an F_1-score of 95.1, with Average Precision (AP_{50}) and Average Recall (AR_{50}) of 88.1 and 79.7, respectively.

Keywords: deep learning, object detection, industrial automation, synthetic dataset, tool management.

INTRODUCTION

With current manufacturing standards, the need for rapid access to tools and equipment needed for machine retooling has necessitated the establishment of tool storage facilities in all major manufacturing plants. In order to ensure high-quality tool logistics in companies, automated vertical storage systems are a frequently used solution [1, 2]. In such a storage system, each item is assigned a location within the storage space, which facilitates control over the quantity and state of tools and equipment. Operator errors, such as retrieving wrong items or returning them to the incorrect positions, can disrupt the inventory system and cause discrepancies between the actual stock and the reported virtual stock. Such discrepancies can lead to a number of problems during production and even result in a halt of the production line [3, 4] – this highlights the need to ensure adequate order in storage spaces and to minimise or completely eliminate irregularities in inventory systems. The solution to this problem is a system that monitors inventory changes and detects potential inaccuracies.

A common approach currently employed in storage facilities is the use of RFID (radio-frequency identification) tags [5–7]. By placing a unique tag on each object, it is possible to track its status and accurately supervise the inventory. A significant disadvantage of this approach is the high cost of the required infrastructure as well as the need to place tags on all objects, which may not be possible for systems storing small tools and equipment in manufacturing companies. Another possible solution is the system developed by Aioi Systems (India), in which each storage compartment is covered by a flexible screen on which information is displayed for the warehouse operator (location, type of operation) [8]. Change detection is performed by sensing screen deformation. In the configuration presented, the system cannot detect the direction of the change, only its location.

Vision systems are commonly used to solve the problem of detecting inventory changes. For the detection of changes in images, it is common to use methods such as SSIM (Structural similarity index measure) [9] or absolute difference analysis. However, these classical approaches have a number of shortcomings, such as inability to determine the direction of change, low flexibility (in terms of variability of acquisition conditions and analysed objects), and susceptibility to false-positive errors. The subject of change detection (CD) at the pixel level is often addressed in remote sensing issues. As a solution to this problem, researchers have proposed their own deep neural network architectures based on vision transformers [10–12] or graph networks [13]. These new architectures are highly efficient but require a large amount of training data and a high level of computing power to be developed and are characterised by substantial size and large inference times, making them significantly more difficult to use in an 'on-edge' setting.

In the case of implemented solutions based on machine vision, two solutions could be distinguished. Amazon Technologies (USA) has developed a container content monitoring solution using an overhead camera that takes before and after images and a processing system based on depth feature extractors [14]. Each image is divided into smaller segments (tiles) and encoded in a feature vector. The before and after vectors are compared with each other to detect and localise changes. An example of a machine vision system implemented for commercial use is the system developed by Accel Robotics Corp (USA) to monitor the recovery of goods from a store shelf [15]. It is a solution that uses a two-camera system to first identify the area of change using stereometrics, and then to extract the ROI (region of interest) area and transfer it to a classifier. This solution allows for the determination of the location of the change, its volume, and the class of the collected object. The disadvantages of such approach are that a database of objects has to be created, a training set has to be prepared for the classification task, and there is little flexibility in the event of a change in the appearance of the products, as well as the need for a set-up of two calibrated cameras for each working area.

Considering the problem we have posed, it is worth noting the eagerly addressed issue of semi-supervised anomaly detection by researchers [16]. This approach is based on the development of a training set that contains only sample-free anomalies and the detection of anomalies during production based on the deviation of test samples from the training samples. A distinction can be made between the reconstruction-based [17] and distribution-based [18, 19] approaches depending on the method used to determine the anomaly score. Reconstruction-based approaches use autoencoders to obtain a condensed representation of the image in the bottleneck and determine the anomaly score by calculating the difference between the image at the input and output of the autoencoder. Only anomaly-free images are used to train the autoencoder with the corresponding loss function, e.g. MSE (mean squared error) as a standard approach, or SSIM [17]. For distribution-based approaches, the distribution of deep features is determined, e.g. by estimating a multivariate normal distribution for the anomaly-free samples. Then, the anomaly score is determined based on the distance of the test sample from the distribution. The described semi-supervised anomaly detection methods also have limitations, among which the need for high homogeneity among the anomaly-free samples should be first noted. For the problem posed, it is possible to apply the methods shown, but the model needs to be trained to detect anomalies in the form of changes after each tool pick-up/drop-off, which is not an optimal solution. The proposed approach also does not solve the problem of determining the directionality of changes. With this in mind and considering the solution's versatility, we decided to use supervised methods with a focus on

improving (speeding up) the process of developing training data.

Taking into account existing methods, authors concluded that the problem posed requires a new approach due to: (1) the lack of knowledge regarding all object types that may occur during inference, (2) significant object overlap and similarity (which rules out a counting-based solution), (3) the need for robustness to noise and disturbances (such as object displacement and small changes in acquisition conditions), (4) the requirement for real-time inference in an 'on-edge' setting, (5) the necessity to determine the directionality of changes, and (6) the requirement to pass industrial validation. The authors propose a solution based on a task-specific architectural modification of an existing machine learning (ML) model, enabling an increase in the dimensionality of input data. YOLOv8 models were chosen as they [20, 21] represent state-of-the-art solutions designed for, among other tasks, real-time object detection and instance segmentation. Their high detection quality, short inference times, and ease of development have made models of this architecture frequently used in computer vision tasks.

The main highlights of our research include: (1) a synthetic dataset generation system, (2) a custom-modified machine learning solution based on the YOLO architecture allowing processing of six-channel images, (3) hybrid image processing with SSIM analysis providing a high level of recall. The proposed methods allow for rapid solution development and high scalability whilst also ensuring high robustness to domain shifts. The task-oriented modification of already existing and proven ML solutions enables easy deployment in industrial applications and further development. The proposed modification is characterized by significant flexibility and the potential for straightforward adaptation to other applications requiring inference on multi-channel images.

MATERIAL AND METHODS

Problem definition

The described problem concerns a vision system that is a component of a tool storage inventory monitoring system (Figure 1). The inference results of the vision module – the location of changes and their type – are compared with operations recorded in the IT inventory system controlling tool withdrawals and returns. The system associates each tool with a specific segment (box) of the tray where it is located. This allows for the detection of irregularities during warehouse operations performed by the operator and enables appropriate responses, such as displaying information on a screen or indicating the object's location using a multi-coloured light indicator.

The main task of the developed vision system is detecting changes between two images in the RGB (red-green-blue) colour space, depicting states before and after changes to the storage area, along with determining the direction of the change – either the addition or removal of an object (Figure 2).

In developing a solution for the presented research problem, certain aspects of the vertical tool storage system (Figure 3), with which the system is to be integrated, were considered:

- the system stores multiple trays, each containing tools collected in cardboard boxes of fixed sizes;
- the operator can make changes only within a selected tray, and only after it has moved out of the lift;
- once the 'return to magazine' option has been selected, it is no longer possible to make changes within the tray working area.

After analysing the above characteristics of the target system, the following design and operating principles for the vision system were adopted:

- the 'before' image is captured immediately after the selected tray leaves the lift of the storage system, before any operator action is taken;
- the 'after' image is captured after the operator has performed operations inside the tray and selected the 'return to magazine' option;
- the vision system is integrated into the structural design of the storage system, ensuring a constant orientation and location of the trays relative to the camera between shots.

By adopting the described principles, constant acquisition conditions were ensured. Despite this, several problems were identified that could negatively affect the reliability of the developed system, such as: (1) the possibility of significant overlap of tools in individual boxes, (2) the presence of many similar objects in the container, resulting in a low level of variability between images, and (3) changes in the position and orientation of tools inside the tray, caused by vibration or other activities

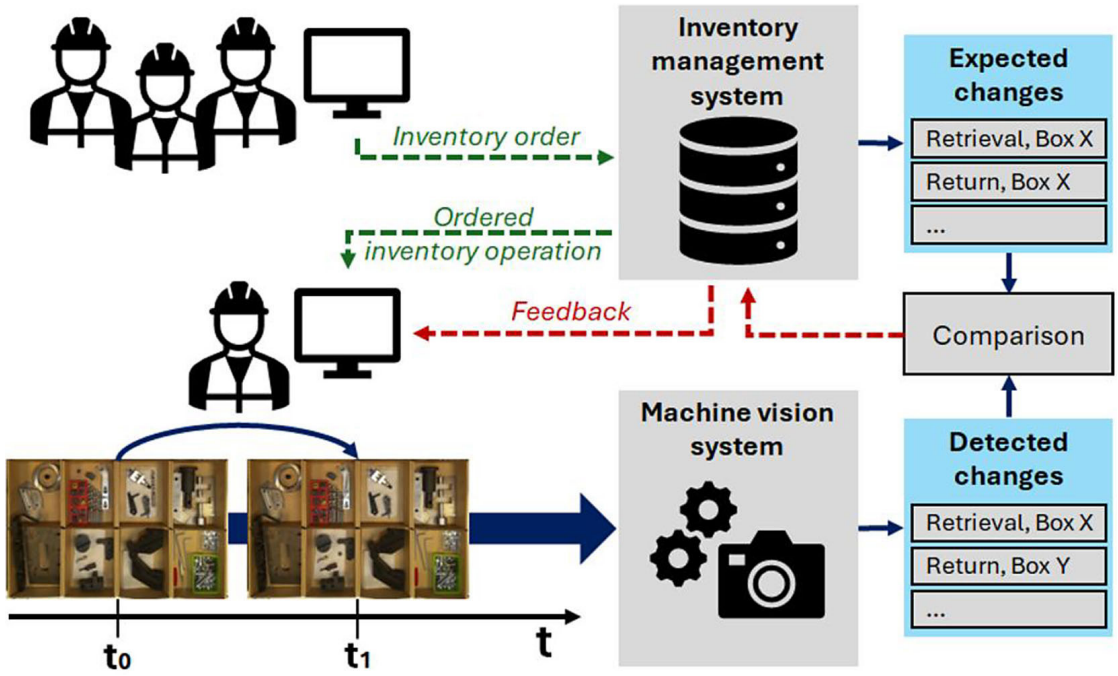

Figure 1. Schematic of the operation of the tool inventory management system with the proposed vision component

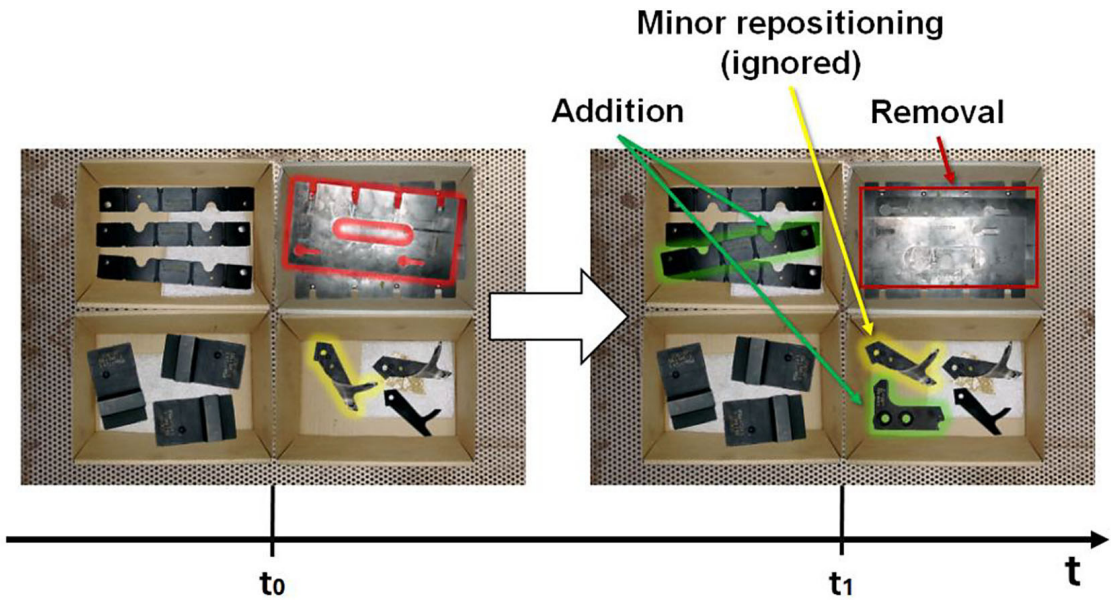

Figure 2. Visual representation of the problem

performed by the operator inside the tray (further referred to as disturbances).

Proposed machine learning solution

Two approaches have been proposed as a solution to this problem: (1) a vanilla (unmodified) YOLOv8 with 3-channel input, or (2) an extended YOLOv8 model capable of inference on 6-channel images. The first method involves using existing instance segmentation models operating on specially prepared input images (Table 1). For the 6-channel approach, referred to as YOLO6C, it was proposed to use the YOLOv8 model with architectural and framework modifications to enable inference on 6-channel $R_1G_1B_1R_2G_2B_2$ images. The input image for this model consisted sequentially of the RGB channels of the 'before' image followed by the RGB channels of the 'after' image.

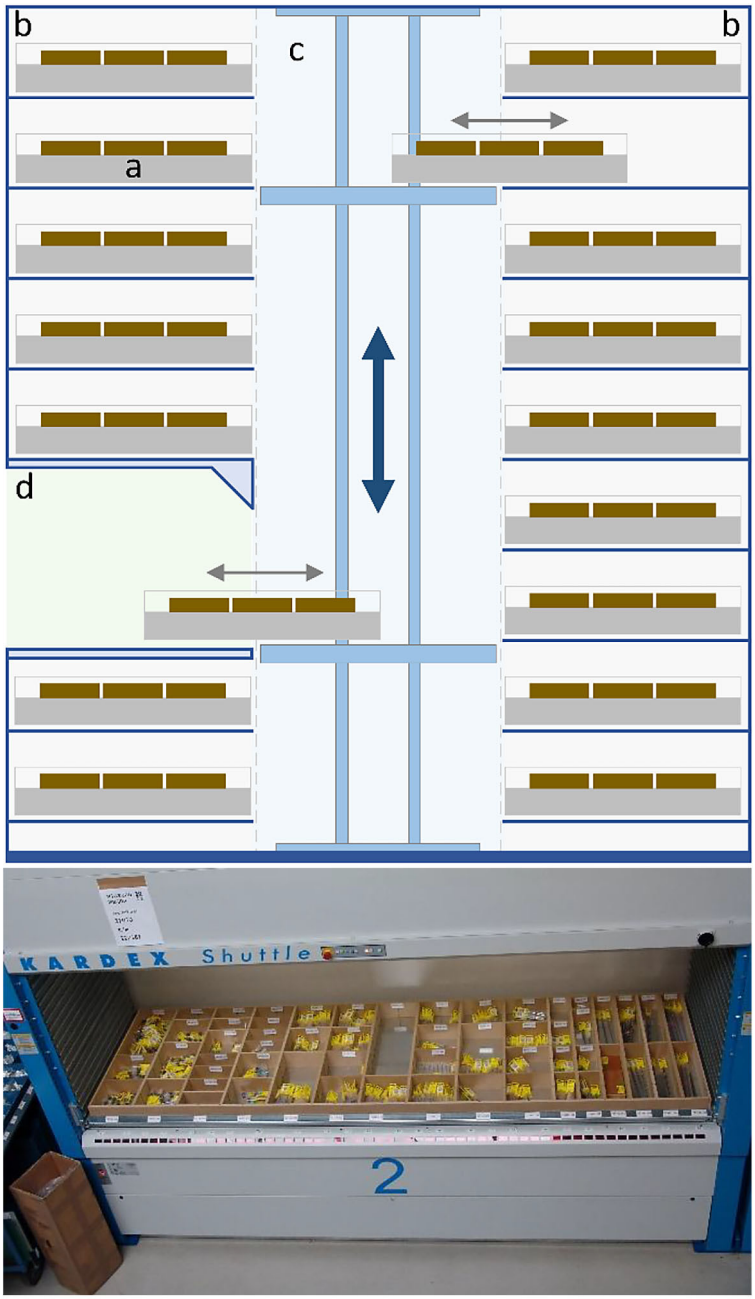

Figure 3. Example of a vertical storage system — vertical lift module (VLM): (a) tray, (b) storage zone with multiple shelves, (c) lift, (d) work area for tool deposition and retrieval [22]

Multistage processing

The application of the developed solution necessitates achieving a high recall, placing great importance on detecting changes occurring in the tool magazine, while allowing for a certain level of false positive errors. Therefore, an additional processing step was proposed to capture changes missed by the ML model. To detect these changes, SSIM values for individual pixels were used, describing differences in both colour space data and structure between the two RGB 'before' and 'after' images. To avoid redundant detection of changes already identified by the first-stage ML model, the input images are masked in the areas covered by the detected segments (Figure 4). The SSIM values are pre-processed to filter out noise and enhance the data, and then thresholded using a predetermined value, manually selected for the target domain.

Data acquisition

The images of objects, containers, and backgrounds used to generate the training dataset

Table 1. Channel diagram of the input image for the YOLO3C approach

Model type	Input - channel R	Input - channel G	Input - channel B
YOLO3C-A	Grayscale(I_{Before})	Zero	Grayscale(I_{After})
YOLO3C-B	Grayscale(I_{Before})	Mean(I_{Before}, I_{After})	Grayscale(I_{After})
YOLO3C-C	Grayscale(I_{Before})	AbsDiff(I_{Before}, I_{After})	Grayscale(I_{After})
YOLO3C-D	Grayscale(I_{Before})	SSIM(I_{Before}, I_{After})	Grayscale(I_{After})

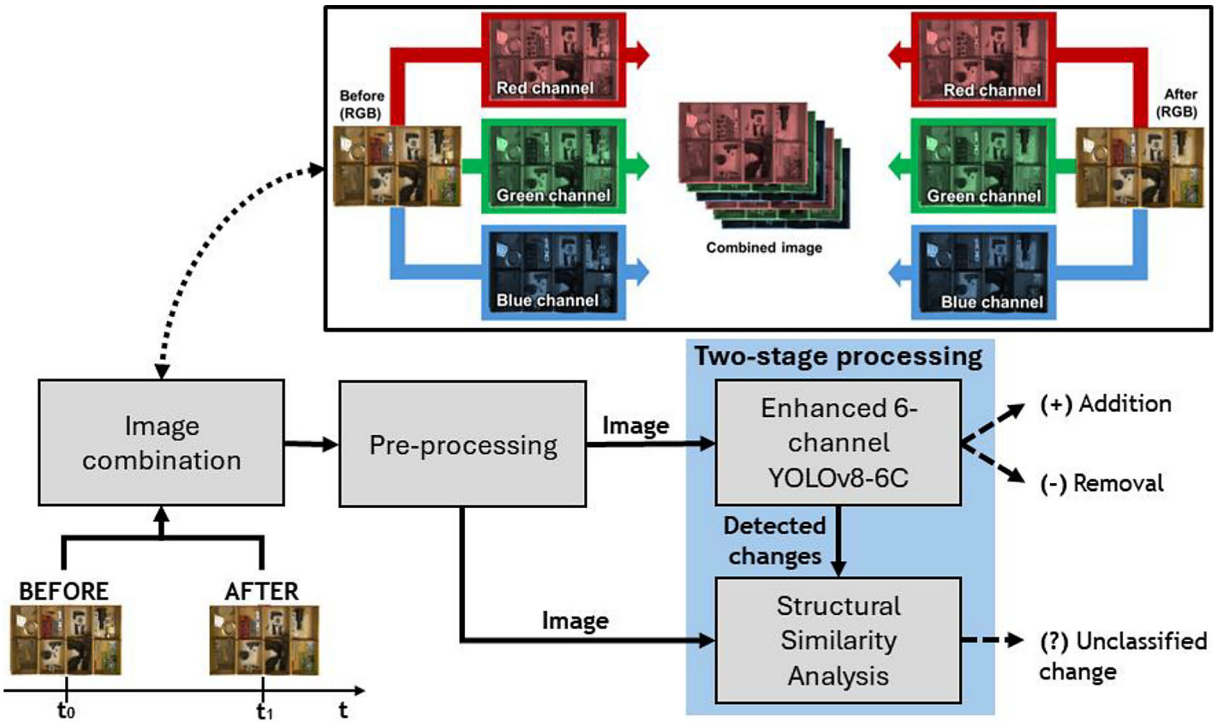

Figure 4. Processing scheme of the developed solution

were captured using a Lori Plus webcam (Natec, Poland) with GC2053 sensor (Galaxycore Microelectronics, China) providing a resolution of 1920 × 1080 pixels. The images of objects and containers were taken with the camera mounted on a stand, utilizing an LFDW201 flat dome illuminator (Wenglor, Germany) that emits diffused light with a colour temperature of 6500 K, matching natural daylight. Background images were captured under the ambient lighting conditions available at the recording location.

Test images were recorded in various configurations and using different equipment. The 'target' conditions for image acquisition on the storage tray were simulated in a specially prepared setup (Figure 5). For this purpose, a Raspberry Pi Camera v2 (Raspberry Pi Foundation, United Kingdom) equipped with a Sony IMX219 sensor (Sony Corporation, Japan) providing a resolution of 3280 × 2464 pixels was used. The setup also included an integrated 3.04 mm focal length lens and an LBDW201 bar illuminator (Wenglor,

Table 2. Comparison of the proposed methods of dataset generation

Parameter	Automatic	Customised
Used objects	COCO dataset objects	Industrial tools and equipment
Generation method	Copy-and-paste	Customised method
No. of objects	1266	146
No. of backgrounds	1500	71
No. of containers	N/A	18

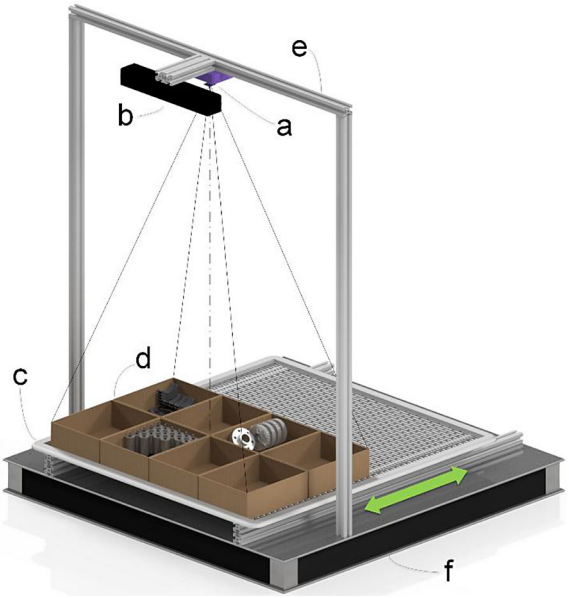

Figure 5. Acquisition setup; (a) camera, (b) illuminator, (c) movable shelf, (d) containers with objects, (e) support frame, (f) laboratory table

Germany). To reduce the impact of noise on subsequent system performance, the test images were downscaled to the target input size of the models using bilinear interpolation.

The setup included a movable shelf on which cardboard boxes with objects were placed, enabling the simulation of the tray sliding in a real storage system and clearly defining the moment of reference image acquisition 'before' any changes were introduced in the storage. Subsequent image acquisitions were manually triggered 'after' introducing changes. However, the developed setup also offers the ability to continuously monitor the storage tray, which, although not utilized in this study, could be employed in future research.

Synthetic image generation

Creating an extensive and diverse training dataset is a significant challenge when developing any system based on machine learning techniques. To address this, the authors proposed an automatic system for generating image pairs, enabling rapid data generation and facilitating the development of the proposed solution. This paper presents and analyses two approaches to automated training data generation: 'Automatic' and 'Customised' (Table 2).

The first method is simple and efficient. Images are created by combining backgrounds and segmented objects from the widely used common objects in context (COCO) dataset [23], thus eliminating the need for additional data acquisition. The 'copy-and-paste' technique [24], commonly used for dataset augmentation, is employed to generate these images.

The 'Customised' approach extends the first method by incorporating domain knowledge about the target station, such as placing tools inside containers and arranging them in a grid (Figure 6). Additionally, this approach uses images of various tools – ranging from industrial warehouse items to laboratory equipment – captured at the image acquisition station. Custom photographs of industrial surfaces, including stainless steel, plastics, and rubber, were used as backgrounds.

Before detailing the dataset generation method, it is necessary to explain the data storage approach

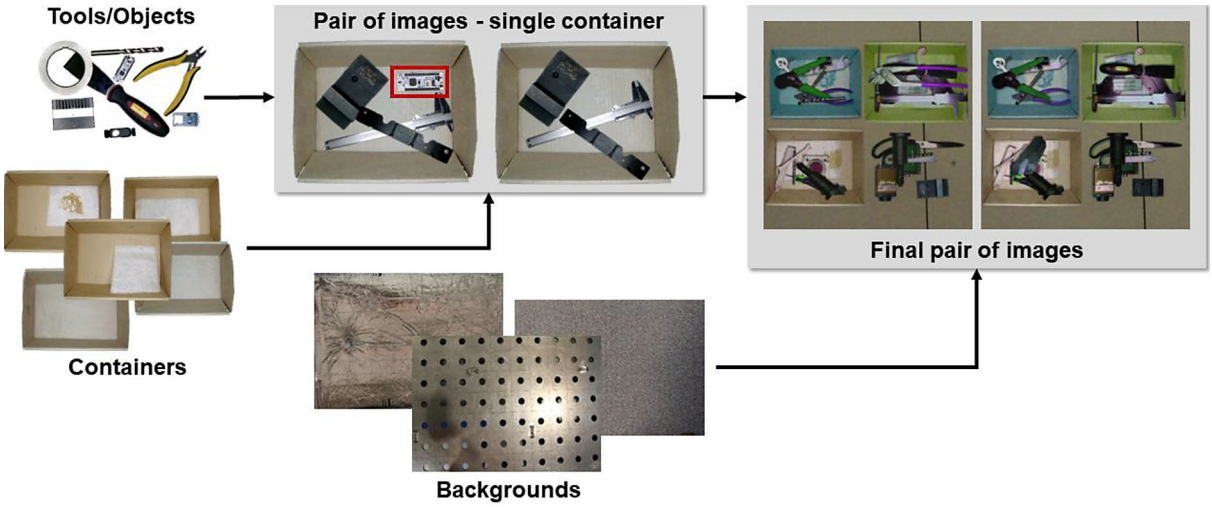

Figure 6. Simplified scheme for the image generation process

for this particular task. As the system operates by analysing two images taken at different time points, the dataset should consist of pairs rather than individual images. Each pair, differing in the presence or absence of an object, can be stored in two configurations: normal [Image 1, Image 2] or inverted [Image 2, Image 1]. This inversion changes the directionality of the change (indicating whether an object is added or removed). Saving each pair in both configurations ensures class balance.

For the 'Automatic' method, image pair generation is significantly simplified. In the first step, a set of objects (segmented from the COCO dataset) is selected and randomly placed against a random background. The last object is placed only in the first image, thereby creating a difference (change) between the generated pair.

The 'Customised' generation process is more complex and consists of multiple stages. Similar to the 'Automatic' approach, a set of segmented tools is randomly selected. However, instead of being placed on a random background, the tools are positioned on a container image (e.g., a cardboard box), ensuring they remain within container boundaries. The process is repeated to create several pairs of container images, which are then assembled into grids ranging from 2 × 2 to 4 × 4 and overlaid onto a random background.

To maintain high diversity in the generated images, various random geometric and colour-space modifications are applied to the objects, such as rotation, translation, vertical or horizontal flipping, resizing, and colour adjustments (brightness, saturation, hue, and gamma). Additional transformations include RGB channel shuffling and artificial light reflection effects. Randomly selected objects were moved and/or rotated between images in a generated pair to simulate minor displacements that may occur during regular warehouse operations. Artificial light reflection was simulated by thresholding the object's 'value' channel and blending it back with a random weight.

In addition to object-level augmentations, several random modifications were applied to the final images, including vertical or horizontal flipping, JPEG compression, noise addition (RGB or salt-and-pepper), colour corrections (brightness, saturation, hue, and gamma), median blur, and perspective warping. Slightly different colour-correction parameters were applied to the 'before' and 'after' images to enhance the model's robustness to changes in lighting conditions. Using each method, 2.500 image pairs were generated, forming the basis for further ML model development.

Experiments and ablation studies

To perform an extensive and comprehensive evaluation of the proposed solutions, a series of experiments was conducted:

1. Evaluation of the individual ML solutions described in subsection Proposed machine learning solution,
2. Comparison of models trained on datasets generated using the fast automatic approach, the customised method, and mixed data (50:50 ratio),
3. Comparison of one-stage and two-stage approaches,
4. Evaluation of the final model.

The evaluation for individual ML solutions, the impact of training data, and the final model assessment was carried out for two classes: 'added' and 'removed'. For the comparison between the one-stage and two-stage approaches, metrics were determined for the combined class labelled as 'change'. For the evaluation of the proposed solutions, five test sets differing in camera type, contained objects, or background were prepared – one set corresponding to the target conditions and four additional domains (Figure 7, Table 3). A total of 189 image pairs from different domains were prepared.

Evaluation of developed solutions

The evaluation was carried out as cross-validation. A total of 500 randomly selected images were divided into 5 independent segments,

Figure 7. Example images from datasets representing various test domains

Table 3. Overview of prepared test datasets

Test dataset	No. of image pairs	Camera	Description
Target	69	PiCamera v2	Corresponding to the target domain with tools arranged inside 8 cardboard boxes
Additional 1	34	Lori Plus	Tools placed directly on a metal surface
Additional 2	23	Lori Plus	Tools arranged inside 6 cardboard boxes
Additional 3	45	Lori Plus	Objects other than tools, arranged on a background of coloured posters
Additional 4	27	Lori Plus	View of a workstation with tools and small electronic equipment

forming 5 unique datasets. To maintain class balance after the data split, each pair of images was saved in both normal and inverted forms. Consequently, each of the five datasets consisted of 400 image pairs for model training and 100 for validation during training.

Pre-trained models were used during training: YOLOv8n-seg for the YOLO3C solutions or a modified version of it for YOLO6C. The training parameters were set as follows: batch size = 10 and epochs = 75.

Each of the 5 models was evaluated on each of the 5 test sets. The mean values and standard deviations of the metrics were determined for each test domain as well as for the entire test set.

The confidence threshold (working point) was set to 0.5. Using the pycocotools library, average precision (AP50) [25] and average recall (AR50) [26] were determined. The F1-score [27] metric was calculated as shown in Equation 1:

$$F_1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (1)$$

where:

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

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

and *TP*, *FP*, and *FN* represent the total number of true positive, false positive, and false negative predictions, respectively.

Evaluation and testing were carried out on a workstation equipped with an Nvidia RTX4080 GPU (Nvidia Corporation, USA) and an Intel Xeon Silver 4110 CPU (Intel Corporation, USA).

Analysis of the effects of disturbances

As mentioned earlier in section, the developed solution must be robust to disturbances such as minor tool displacements, rotations, and illumination changes – it should effectively ignore these variations. To assess this robustness, an experiment was designed to test the model's behaviour under variable translation, rotation, and brightness conditions.

A test dataset was created comprising 25 non-symmetrical objects of size 480 × 480 pixels, positioned at the centre of randomly selected backgrounds of size 1760 × 1760 pixels. The following test cases were performed: rotation – objects were rotated around their centres from −180 to 180 degrees in increments of 5 degrees, translation – objects were shifted along the X and Y axes from −240 to 240 pixels in increments of 4 pixels, and brightness variation – the brightness between image pairs was varied in the range of −50% to +50%.

As the performance quality metric, the ratio of the number of correct negative predictions to the total number of predictions was selected. This metric, known as the true negative rate (TNR), is defined by Equation 2:

$$TNR = \frac{TN}{TN + FP} \times 100\% \quad (2)$$

where: *TN* denotes the number of true negative predictions and *F P* the number of false positive predictions.

Preparation and evaluation of final model

The final version of the model was trained on a dataset consisting of 2000 unique pairs of images. Training was conducted with a batch size of 8 and for 50 epochs, using weights from the pre-trained *YOLOv8n-seg* model. The evaluation was carried out for two tasks: (1) detection and classification of changes according to their direction ('added' / 'removed'), (2) detection of any changes without classification (a combination of the 'added', 'removed', and 'change' classes). The model was evaluated on a dataset comprising images from all test domains, using the methods and metrics described in Table 3.

RESULTS AND DISCUSSION

Evaluation of developed methods

The developed methods were evaluated separately on each test domain as well as on the entire set of test images. The results, presented as mean values and standard deviations, were aggregated in Table 4. The highest metrics were obtained for the YOLO6C approach, with the following results for evaluation on all test images: AP_{50} = 82.55 ± 0.82, AR_{50} = 67.39 ± 0.38, and F_1 = 89.42 ± 0.76. The second highest values were achieved by the YOLO3C-D approach using SSIM masks: AP_{50} = 82.46 ± 0.81, AR_{50} = 66.00 ± 0.67, and F_1 = 87.36 ± 0.74.

Due to the relatively similar results obtained for both approaches on the entire test dataset, the inference quality of the models was also compared across individual test domains. The F_1 metric values for the YOLO6C solution were higher across all tested domains, indicating better inference quality at the adopted working point. In terms of AP_{50} and AR_{50} metrics, neither model consistently outperformed the other. However, YOLO6C achieved superior results in the target domain, with AP_{50} = 88.27 ± 1.20 and AR_{50} = 78.94 ± 1.01, compared to AP_{50} = 87.05 ± 1.01 and AR_{50} = 73.16 ± 1.00 for YOLO3C-D.

Both approaches demonstrated high quality metrics across the target and other test domains, indicating a degree of robustness to domain shift. After analysing the data, the YOLO6C approach was selected for further work due to its superior performance in the target domain. Additionally, approaches based on 3-channel processing of input images to grayscale are more prone to performance degradation when regions of pixels have different hues but similar intensities. This issue does not affect the YOLO6C solution, as no colour space information is lost during preprocessing. While this problem was not observed during testing, it could potentially arise during the system's deployment.

Table 4. Evaluation results for the 5 proposed approaches across all test domains

Model type	Test dataset	F1	AP50	AR50
YOLO3C-A	All images	77.24 ± 2.46	59.85 ± 3.38	43.82 ± 2.31
	Target	79.22 ± 4.64	52.24 ± 5.46	38.54 ± 3.21
	Additional 1	70.13 ± 3.63	65.10 ± 3.88	47.32 ± 2.45
	Additional 2	77.99 ± 4.82	59.31 ± 4.99	39.56 ± 3.70
	Additional 3	77.52 ± 0.51	66.88 ± 1.23	51.32 ± 0.88
	Additional 4	74.77 ± 3.01	55.33 ± 1.80	40.90 ± 1.58
YOLO3C-B	All images	71.44 ± 2.59	51.11 ± 7.28	38.99 ± 5.85
	Target	76.24 ± 3.16	49.50 ± 8.75	37.57 ± 6.61
	Additional 1	62.06 ± 6.18	53.08 ± 9.65	40.34 ± 6.93
	Additional 2	69.33 ± 5.35	50.33 ± 7.77	36.26 ± 6.05
	Additional 3	71.48 ± 1.68	55.19 ± 8.00	43.70 ± 7.08
	Additional 4	70.70 ± 2.73	43.47 ± 6.39	32.25 ± 4.82
YOLO3C-C	All images	77.97 ± 1.53	62.01 ± 2.47	44.04 ± 1.55
	Target	85.66 ± 1.00	59.16 ± 2.38	39.17 ± 1.39
	Additional 1	71.61 ± 3.09	67.95 ± 2.94	51.30 ± 2.00
	Additional 2	78.67 ± 3.50	57.34 ± 1.87	38.04 ± 0.93
	Additional 3	76.07 ± 1.17	67.53 ± 2.72	49.67 ± 1.64
	Additional 4	66.24 ± 4.64	49.69 ± 5.24	38.40 ± 3.47
YOLO3C-D	All images	87.36 ± 0.74	82.46 ± 0.81	66.00 ± 0.67
	Target	91.47 ± 1.24	87.05 ± 1.01	73.16 ± 1.00
	Additional 1	82.51 ± 2.04	79.61 ± 1.20	61.88 ± 0.75
	Additional 2	84.07 ± 2.62	80.25 ± 2.19	61.23 ± 1.79
	Additional 3	89.67 ± 1.43	85.60 ± 0.92	68.05 ± 0.84
	Additional 4	76.35 ± 1.74	65.66 ± 1.93	51.18 ± 0.85
YOLO6C	All images	89.42 ± 0.76	82.55 ± 0.82	67.39 ± 0.38
	Target	93.00 ± 0.88	88.27 ± 1.20	78.94 ± 1.01
	Additional 1	85.87 ± 1.89	78.32 ± 1.47	59.83 ± 0.83
	Additional 2	88.26 ± 2.71	82.57 ± 2.32	63.46 ± 1.00
	Additional 3	89.69 ± 1.33	84.11 ± 1.08	67.56 ± 0.76
	Additional 4	79.02 ± 1.54	65.55 ± 2.37	49.32 ± 0.76

Analysis of the effects of disturbances

Through the conducted tests, the applicability of the model was evaluated under object rotation, translation in the working area, and changes in illumination conditions. Quality metric values were plotted as functions of rotation, translation, and relative brightness changes (Figure 8). The minimum acceptable quality metric threshold was set at 80%. Based on this criterion, the following constraints were established:

- maximum acceptable rotation: 25 degrees;
- maximum acceptable translation in the X or Y axes: 19% of the object size, which in typical cases corresponded to approximately 91 pixels;
- maximum acceptable brightness change: -16% and +26%.

The results presented demonstrate the significant robustness of the model to small changes in the position of objects in the working area that may occur during normal system operation.

Comparison of image generation methods

Evaluations were carried out for two training sets generated using the Customised method and the simpler copy-and-paste automatic approach. The values of the determined metrics are summarised in Figure 9. Models trained exclusively on the set created using the simplified approach with COCO dataset images exhibited significantly lower values across all inference quality metrics in every test domain compared to the customised generation method. The Customised method, as well as the mixed dataset (with a 50:50 ratio), yielded much better inference quality in all test domains, including domains containing objects not present in the image pool used by the data generation system (domains 'additional 3' and 'additional 4'). The study demonstrates the need for data generation methods that take into account the target appearance of images, clearly demonstrating the disadvantages of a naive, fully automated approach. Such a significant drop in the quality of the model for that generation method could be caused not only by the significant disparity with the target test domain but also by high diversity and abstractness of input images, which prevented the architecture used from generalising correctly.

Comparison of the one-stage and two-stage approaches

The evaluation results for change detection for the one- and two-stage approaches are summarised in Figure 10. The approach with additional SSIM processing resulted in a reduction of the F_1 metric in all test domains except the target domain. For the AP50 and AR50 metrics, significant improvements were observed for all test domains. The proposed method of additional post-processing based on analysis of pixel-wise SSIM values has reduced flexibility and, as such, requires fine-tuning through the selection of appropriate parameters. Those values were chosen for the target domain; hence it was decided to restrict further analysis to this test domain. For two-stage processing, values of F1 = 94.62 ± 0.29, AP_{50} = 91.56 ± 0.75 and AR_{50} = 80.78 ± 0.60 were achieved – higher than the values for single-stage processing where the values amounted to F_1 = 92.80 ± 0.19, AP_{50} = 83.47 ± 1.07 and AR_{50} = 74.12 ± 0.68.

The application of the two-stage approach improved the AP_{50} and AR_{50} metrics for all test domains by increasing the sensitivity of the system.

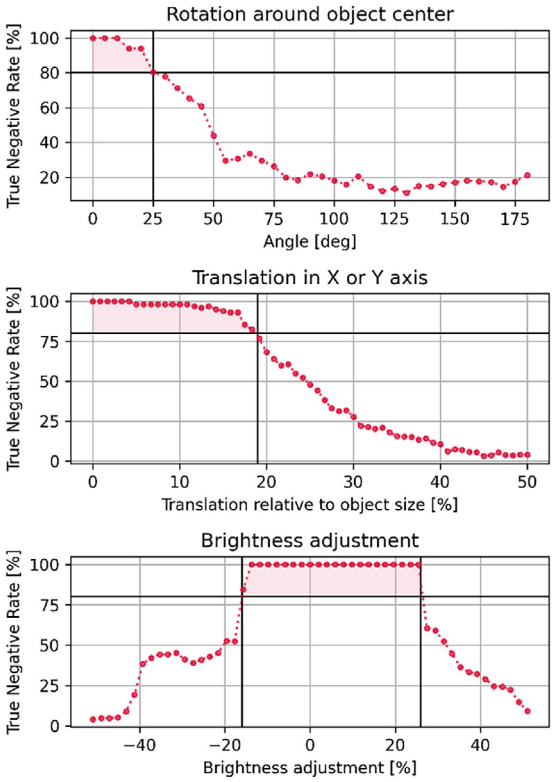

Figure 8. The effect of object's rotation around its own axis translation in the X or Y axes and image brightness change on the true negative rate

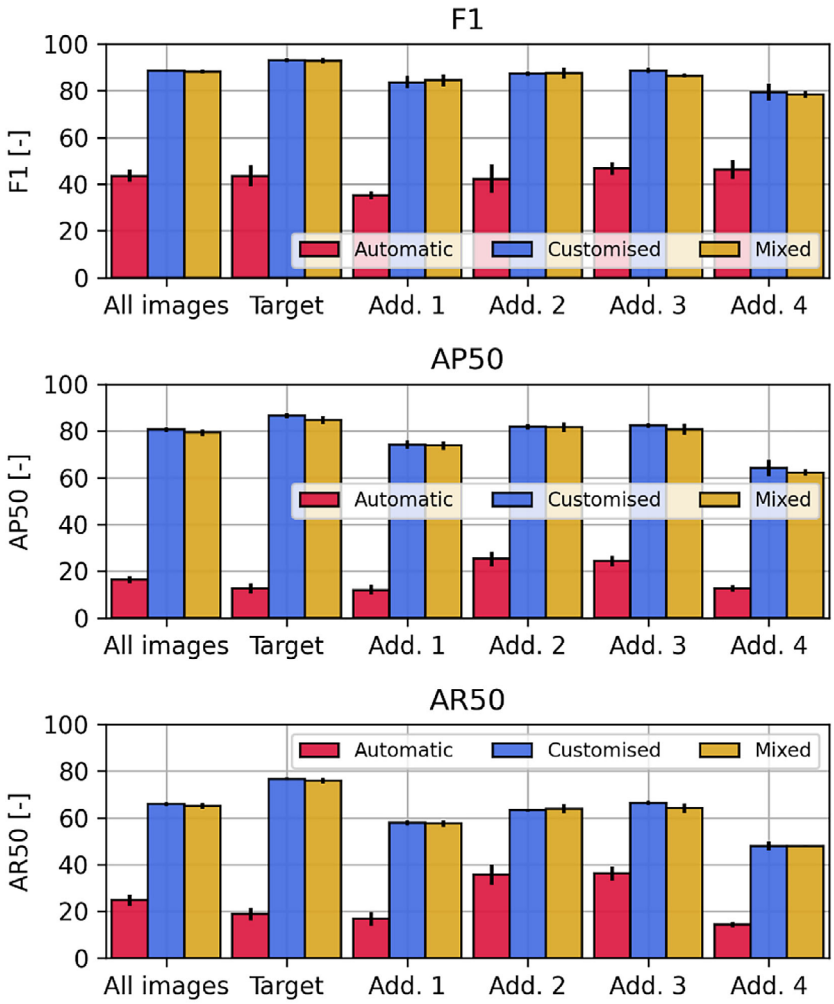

Figure 9. Comparison of evaluation results for YOLO6C models trained using two synthetic datasets

As a result, the number of false negatives was reduced. On the other hand, the introduction of additional processing increased the total inference time as well as decreased the values of the inference quality metrics for the adopted working point (F_1 at thresh conf = 0.5) in non-target domains. However, this is an acceptable drawback given the observed improvement in comprehensive metrics such as AP50 and AR50 and can also be circumvented by further fine-tuning of the solution.

Final system evaluation

The validation of the final version of the developed system was carried out on the 'target' test set. Values of F_1 = 95.09, AP_{50} = 88.06 and AR_{50} = 79.72 were achieved for detection of 'added' and 'removed' classes (first stage only) while values of F_1 = 94.95, AP_{50} = 92.72 and AR_{50} = 83.63 were obtained for the task of detecting any changes without classification ('added', 'removed' and 'change' combined). The average total pre-processing and dual-stage inference time was 0.676 ± 0.039 s. Example inference results are shown in Figure 11.

CONCLUSIONS

The results presented in this paper clearly demonstrate the effectiveness of applying task-specific modifications to state-of-the-art deep learning models for tasks involving change detection and direction classification. The proposed solution retains all the advantages of the YOLOv8 architecture and the Ultralytics framework, such as ease of model preparation and training, along with short inference times, while simultaneously expanding the range of potential applications. By using a synthetically generated dataset instead of a traditional labelled image approach, the time required for preparing training data – and consequently the entire model

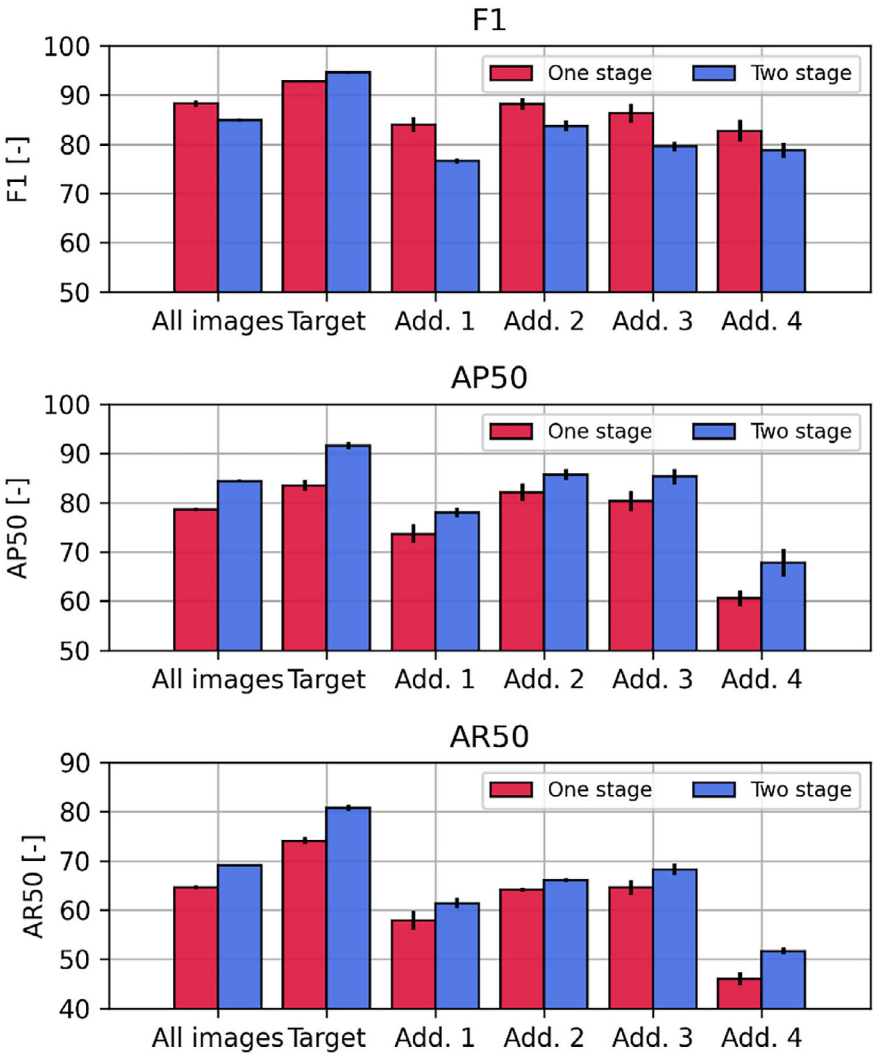

Figure 10. Comparison of evaluation results for one-stage and two-stage approach using YOLO6C models

Figure 11. Visualisation of the inference results of the final system

development process – was significantly reduced. This streamlined approach facilitated the formulation and extensive evaluation of multiple solutions. Studies have confirmed the practical applicability of the developed solution, including its potential integration into a vertical tool storage system. The augmentation methods employed during the generation of learning image pairs significantly improved robustness against negative phenomena encountered under real operating conditions. These challenges include domain shifts, small object displacements or rotations between shots, and slight variations in illumination. A notable advantage of the developed system is its ability to detect changes in stock without prior knowledge of the type or quantity of objects in the workspace, unlike solutions based on object detection and counting.

The adoption of a two-stage approach incorporating SSIM post-processing effectively reduced the risk of false-negative errors, substantially improving the AP_{50} and AR_{50} metrics across all test domains. This improvement was achieved with only a slight decrease in the F1 score for non-target domains at the selected operating point.

The proposed solution could be practically deployed within vertical tool storage systems or similar industrial environments. However, several additional factors must be considered to ensure reliable operation. These include compliance with interface standards, compatibility with industrial cameras, and robustness to environmental conditions such as lighting stability and vibrations. While the presented experiments focused on a single setup, large-scale deployment would require handling substantially larger datasets, potentially across multi-camera installations, which increases both computational and data-management demands. Another important challenge is domain drift, arising from the introduction of new object types, or unexpected backgrounds. To address this, lightweight retraining pipelines or active learning strategies could be adopted, enabling the system to adapt with minimal human supervision.

The solution presented in this publication, which extends an existing architecture to handle image data with higher dimensionality than standard RGB, demonstrates considerable flexibility and broad application potential. Such modifications make it possible to adapt widely used architectures like YOLOv8 to process multi- or hyperspectral imagery, as well as composite data types such as RGB combined with polarization information. This capability is particularly relevant for domains where subtle spectral or structural variations carry critical information, including remote sensing, medical diagnostics, and industrial inspection. At the same time, the use of higher-dimensional input introduces challenges related to increased data volume, specialized sensor requirements, and greater computational complexity. Addressing these factors will be essential for practical large-scale deployment, and future work could explore optimization strategies or tailored preprocessing pipelines to ensure efficient real-world integration.

Acknowledgements

We would like to thank Lukasz Wiśniewski from TCM Polska sp. z o.o. for providing the tool samples used in this study.

REFERENCES

1. Kitisomprayoonkul C. Increasing effectiveness of a warehouse by Kardex storage machine: a case study. Tech. rep. Assumption University, 2001.
2. Azadeh K, De Koster R, and Roy D. Robotized and automated warehouse systems: Review and recent developments. Transportation Science 2019 Jun; 53. https://doi.org/10.1287/trsc.2018.0873
3. Pujawan N, Vanany I, and Dewa P. Human errors in warehouse operations: An improvement model. International Journal of Logistics Systems and Management 2017 Jan; 27: 298. https://doi.org/10.1504/IJLSM.2017.10005117
4. Liu H, Hwang SL, and Liu TH. Economic assessment of human errors in manufacturing environment. Safety Science - SAF SCI 2009 Feb; 47:170–82. https://doi.org/0.1016/j.ssci.2008.04.006
5. Li C and He Q. Design for the logistics storage management system based on RFID. 2009 3rd International Conference on Anti-counterfeiting, Security, and Identification in Communication. 2009: 215–8. https://doi.org/10.1109/ICASID.2009.5276932
6. Lee CH and Chung CW. Efficient storage scheme and query processing for supply chain management using RFID. Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data. SIGMOD '08. Vancouver, Canada: Association for Computing Machinery, 2008: 291–302. https://doi.org/10.1145/1376616.1376648
7. Boese F, Piotrowski J, and Scholz-Reiter B. Autonomously controlled storage management in vehicle logistics–applications of RFID and mobile computing systems. International Journal of RF

Technologies 2009; 1: 57–76

8. Tada K et al. Picking assistant system. 2019 Dec
9. Wang Z et al. Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing 2004; 13: 600–12
10. Chen H, Qi Z, and Shi Z. Remote sensing image change detection with transformers. IEEE Transactions on Geoscience and Remote Sensing 2022; 60: 1–14. https://doi.org/10.1109/TGRS.2021.3095166
11. Lv P., Li M., and Zhong Y. A semi-supervised pyramid cross-temporal attention transformer for change detection in high-resolution remote sensing images. IEEE Geoscience and Remote Sensing Letters 2024; 21:1–5. https://doi.org/10.1109/LGRS.2024.3404645
12. Li Z. et al. MS-Former: Memory-supported transformer for weakly supervised change detection with patch-level annotations. IEEE Transactions on Geoscience and Remote Sensing 2024; 62: 1–13. https://doi.org/10.1109/TGRS.2024.3399215
13. Yang B. et al. A graph-based hyperspectral change detection framework using difference augmentation and progressive reconstruction with limited labels. IEEE Transactions on Geoscience and Remote Sensing 2024; 62: 1–14. https://doi.org/10.1109/TGRS.2024.3403237
14. Curlander J.C. et al. Bin content verification. 2023 Oct.
15. Buibas M. et al. Person and projected image item tracking system. 2021 Jun.
16. Bergmann P. et al. MVTec AD–A comprehensive real-world dataset for unsupervised anomaly detection. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 9592–600.
17. Bergmann P. et al. Improving unsupervised defect segmentation by applying structural similarity to autoencoders. arXiv preprint arXiv:1807.02011 2018
18. Rippel O. et al. Gaussian anomaly detection by modelling the distribution of normal data in pretrained deep features. IEEE Transactions on Instrumentation and Measurement 2021; 70: 1–13
19. Defard T. et al. Padim: a patch distribution modelling framework for anomaly detection and localization. International Conference on Pattern Recognition. Springer. 2021: 475–89
20. Jocher G., Chaurasia A., and Qiu J. Ultralytics YOLO. Version 8.0.0. 2023 Jan. Available from: https://github.com/ultralytics/ultralytics
21. Redmon J. et al. You Only Look Once: Unified, Real-Time Object Detection. 2016. arXiv: 1506.02640 [cs.CV]. Available from: https://arxiv.org/abs/1506.02640
22. Kardex. Vertical Lift Module (VLM). Available from: https://www.kardex.com/en/products/shuttle-vertical-lift-modules
23. Lin T. et al. Microsoft COCO: Common Objects in Context. CoRR 2014; abs/1405.0312. arXiv: 1405.0312. Available from: http://arxiv.org/abs/1405.0312
24. Ghiasi G. et al. Simple copy-paste is a strong data augmentation method for instance segmentation. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2021 Jun: 2918–28.
25. Jarvelin K. and Kekalainen J. Cumulated gain-based evaluation of IR techniques. ACM Trans. Inf. Syst. 2002 Oct; 20: 422–46. https://doi.org/10.1145/582415.582418. Available from: https://doi.org/10.1145/582415.582418
26. Hosang J. et al. What makes for effective detection proposals? IEEE Transactions on Pattern Analysis and Machine Intelligence 2016 Apr; 38: 814–30. https://doi.org/10.1109/tpami.2015.2465908 Available from: http://dx.doi.org/10.1109/TPAMI.2015.2465908
27. Van Rijsbergen C. Information Retrieval. London: Butterworths, 1979.