Journal of Applied Automation Technologies

Research Paper

Constraint-Aware PPO Network for Adaptive Food Surface Defect Detection in Industrial Sorting Scenarios

Radu Păunescu[1,*], Aurel Zăpodeanu[1] and Gheorghe Fieraru[1]

[1] Faculty of Automatic Control and Computers, National University of Science and Technology Politehnica Bucharest, Bucharest, 060042, Romania

*Corresponding author: radu.pa@acs.pub.ro

Abstract. For industrial sorting lines, food-surface fault detection should be highly accurate visually and satisfy production needs. It is challenging to differentiate small bruises, fractures, stains, mold spots, and texture anomalies from the inherent characteristics of food, and real-time sorting must make solid choices in the face of conveyor movement and changing light. A constraint-aware PPO model for adaptive food defect detection is presented in this research. Create visual states using the model's multi-scale surface characteristics, then define inspection as a restricted policy optimization problem. Under latency, missed-defect, and false-rejection limitations, the PPO agent gains knowledge of inspection focus modification, confidence control, and sorting decision refinement. Based on 42,600 food surface photos gathered from the three sorting lines, the suggested method's inference latency is 18.7 ms, its detection accuracy is 96.4%, and its defect recall rate is 94.8%. The strategy improved subtle-defect recall by 7.9% and decreased false sorting by 4.6% compared to YOLO, attention-based, and rule-based baselines.

Keywords: *Food Defect Detection, YOLO, Industrial Sorting Line, Constraint-Aware PPO, Adaptive Inspection*

Received on 24 May 2024, Accepted on 07 November 2024, Published on 14 November 2024

Introduction

Industrial food sorting lines have been progressively included into intelligent food production because they have a direct influence on product division, quality consistency, consumer safety, and production costs. Surface flaws such bruising, fractures, cuts, mold, stains, mechanical pressure marks, peel damage, and unusual texture must be found quickly while sorting at a rapid speed. In a continuous-operation setting, manual inspection is impractical, and conventional machine vision systems are prone to mistakes brought on by variations in light, conveyor vibration, product attitude, etc. By learning texture, color, and shape characteristics from massive picture datasets, deep visual inspection has improved the analytical capabilities of food surfaces [1]. Additionally, studies on fruit and vegetable inspection have discovered that transfer learning and deep feature extraction can enhance fault identification when the target sample is small [2]. However, because the final choice must balance defect recall, false rejection, inspection delay, and actuator reaction time, practical sorting lines still have a gap between the precision of offline detection and deployable sorting stability [3].

Static classification or localization models are currently used in the majority of food and industrial flaw detection techniques. An image is sent into a CNN, YOLO detector, segmentation network, or attention-based model, which immediately generates a defect label, bounding box, or mask. Although this method is practical, it does not clearly depict the sequential decision-making process of industrial sorting. In fact, the system often needs to make decisions on whether to disregard mild textural noise, lower the confidence level for safety-critical faults, re-inspect a suspect area, or activate a sorting actuator in a short amount of time. Although many deep learning techniques for industrial product inspection lack a mechanism to optimize the decision policy under operational restrictions, they do offer strong feature learning capabilities [4]. Inspection may alternatively be framed as an interaction process comprising visual states, actions, rewards, and limitations using reinforcement learning [5]. Constrained reinforcement learning can increase latency, cost, and safety penalties, whereas proximal policy optimization can avoid unstable policy learning by clipping the update [6].

A Constraint-Aware PPO Model is put forth for the detection of food surface defects in industrial sorting lines. It learns an adaptive inspection policy on top of multi-scale visual characteristics rather than taking the role of the visual detector. Under missed-defect, false-rejection, and real-time limitations, the policy establishes the inspection focus, confidence adjustment, defect confirmation, and sorting action. Without increasing the industrial inference delay, stabilize sorting judgments and increase the recall rate of small errors. Separating surface flaws from natural texture changes, coordinating visual confidence with production constraints, and maintaining policy stability under conveyor speed and light variation are the three technical issues with the suggested framework.

Related Work

Industrial Food Defect Detection with Deep Vision

Non-destructive surface-feature acquisition technology is being used for non-destructive testing in several food quality inspection facilities. The initial approach employed thresholding, support vector machines, or other shallow classifiers after manually creating color, shape, and texture descriptors. Although these methods are simple to comprehend, they are sensitive to variations in light and products. To improve food inspection, deep learning can extract multi-level characteristics from the raw image. Deep Transfer-Based Potato Surface Defect Identification In situations when there are few photos of food defects, learning demonstrated that pretrained visual backbones can increase recognition accuracy [7]. The issue of finding flaws in Apple Surfaces for agricultural inspection has also been addressed using deep networks [8]. Deep models are more accurate than handmade pipelines at locating tiny irregular patches for prolonged industrial inspection, according to segmentation-based defect detection and surface inspection networks [9]. Deep visual inspection may be used to production-like data streams when the backbone is small and the post-processing procedure is straightforward, according to research on real-time defect detection of plastic, steel, rail, and gear [10].

In contrast to many other industrial surface duties, food surface defect inspection considers natural roughness as part of the usual look rather than noise. Uneven color, porosity, moisture reflection, edge curvature, and other comparable flaws can be seen in potatoes, apples, citrus fruits, and pastries. In a sorting line, the same product may travel at various angles and speeds. As a result, the defect detection technique should be able to recognize anomalous pixels and assess if the anomaly is serious enough to warrant rejecting the product. The issues of robust representation learning, data augmentation, and transfer learning have recently been the focus of research on visual quality inspection, industrial defect segmentation, and few-shot inspection [11]. After training, the majority of detection models remain fixed. The economic effects of missing faults and incorrect rejections are different, and they do not immediately optimize the downstream sorting cost. This flaw cannot be added to the list as it cannot be found.

PPO-Based Adaptive Decision Models Under Constraints

Reinforcement learning may be used to handle a wide range of issues, including robot motion control, resource allocation, autonomous navigation, inspection path planning, structural health monitoring, and limited control. PPO is popular because it prevents significant policy changes during training by limiting policy updates through a clipped target [12]. Variations of PPO have been used for limited navigation, scanning, and control in several recent robotics and industrial research, despite the fact that the original PPO formulation predates the intended publication window [13]. Constrained Reinforcement Learning improves safety, energy, time, or risk by adding a limitation to the reward function. For network slicing, UAV systems, and limited robot motion planning, reinforcement learning has demonstrated the capacity to make judgments in the face of competing objectives [14]. Visual examination is comparatively possible since the agent must strike a compromise between speed and precision. While a lower threshold will result in more false alarms, a higher threshold improves precision but might not be able to identify minor flaws. PPO can learn a policy that modifies this trade-off according on the operating conditions and visual state.

In food sorting, adaptive visual decision-making is still very uncommon. While the detector has been improved by most investigations, the inspection policy has not been optimized. Certain rule-based industrial inspection systems employ manual rejection logic or a confidence threshold. The guidelines are generally consistent, but they break when the conveyor speed, camera location, or product appearance vary. The threshold modification, region re-inspection, and action selection based on input are all learned by deep reinforcement learning. Constraint design is a problem. The learnt policy will examine an excessively vast region and fall short of real-time requirements if the reward just takes detection accuracy into account. Minor flaws won't be disregarded unless the award is predicated on speed. For commercial food fault detection, a constraint-aware reward-and-action model is thus necessary. In the PPO learning process, it must encode visual severity, sorting cost, actuator time, and safety risk [15].

Methodology and Experimental Evaluation Framework

Multi-Scale Food Surface State Representation

The proposed method formulates each product image as a visual decision state rather than a single static detection input. A line-scan or area-scan camera captures food products moving on the conveyor at $0.8-1.6\ \mathrm{m/s}$. Each frame is first corrected by color calibration, illumination normalization, and motion-blur suppression. The visual encoder then extracts multi-scale texture features from the full product region and candidate defect regions. Figure 1 illustrates the multi-scale food surface state representation for sorting-line inspection. The figure should show image acquisition, product localization, candidate defect proposal, texture feature extraction, confidence map construction, constraint descriptor generation, and PPO state output.

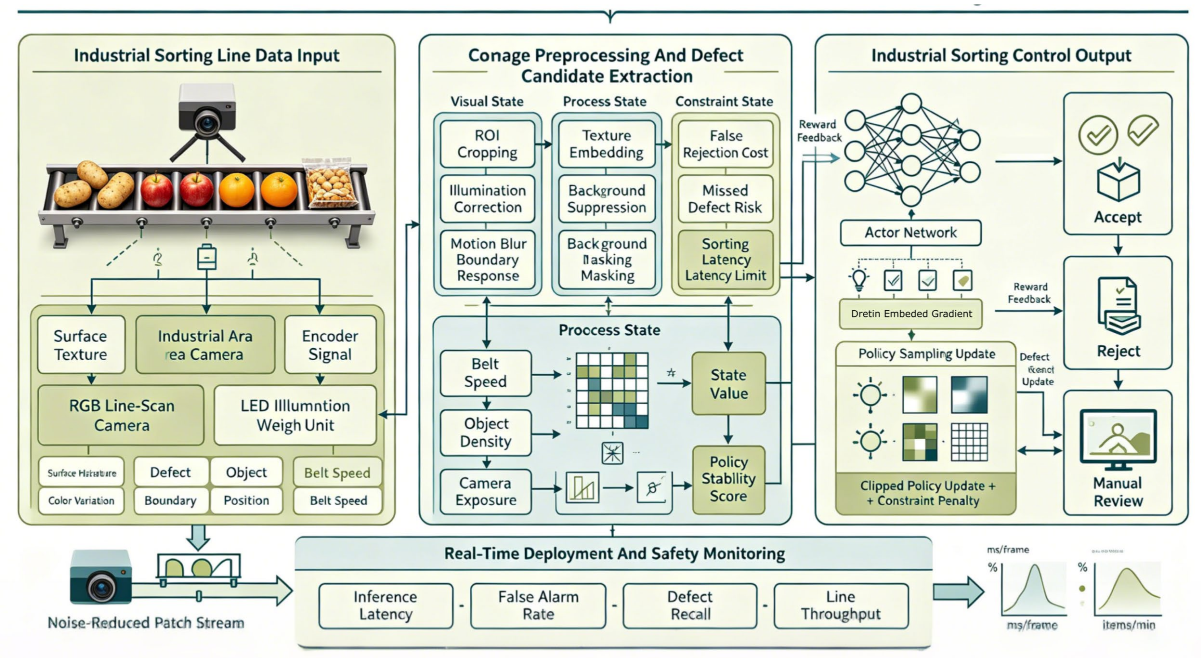

Figure 1. Multi-scale food surface state representation for sorting-line inspection

For image I_t captured at time t, the backbone network produces a pyramid of feature maps at three resolutions. The fused visual descriptor combines local defect texture, global product appearance, and motion-compensated context.

$$f_t = W_f[P_1(I_t) \oplus P_2(I_t) \oplus P_3(I_t)] \quad \text{Eq.(1)}$$

P_1, P_2 and P_3 are feature maps at the shallow, middle and deep levels, respectively. Shallow features keep edges and colour irregularities; deep features learn semantic defect cues. Given the constraints of edge inference in the actual deployment environment, the feature dimension has been reduced to 256.

$$b_t = \phi(B_t, C_t, M_t, V_t) \quad \text{Eq.(2)}$$

https://doi.org/10.64972/jaat.2024v2.343p35e:484-496

The descriptor b_t encodes bounding proposal geometry, confidence distribution, segmentation mask compactness, and conveyor motion velocity. This term allows the policy to distinguish a small high-confidence mold spot from a large uncertain illumination artifact.

$$q_t = \exp[-\alpha_l L_t - \alpha_m M_t] \cdot C_t \qquad \text{Eq.(3)}$$

The visual reliability score q_t penalizes local blur level L_t and mask fragmentation M_t while preserving detector confidence C_t. Reliability is important because sorting cameras often observe partial specular reflection or minor vibration-induced blur.

$$s_t = [f_t \oplus b_t \oplus q_t \oplus v_t \oplus \tau_t] \qquad \text{Eq.(4)}$$

The PPO state s_t concatenates visual feature, defect proposal descriptor, reliability score, conveyor speed v_t, and remaining actuator response time $\tau_{\_1}$ t. This state design lets the policy understand both visual evidence and production constraints.

Constraint-Aware PPO Detection Policy

Industrial inspection behavior is the focus of the Action Space. The detector's output can be sent to the agent, which can then seek local refinement, change the confidence threshold, initiate rejection, or pass the product. The constraint-aware PPO detection policy and sorting decision procedure is shown in Figure 2. Visual state input, the actor-critic network, limited reward computation, clipped PPO update, confidence adjustment, sorting command output, and feedback from missed-defect and false-rejection events should all be included in the image.

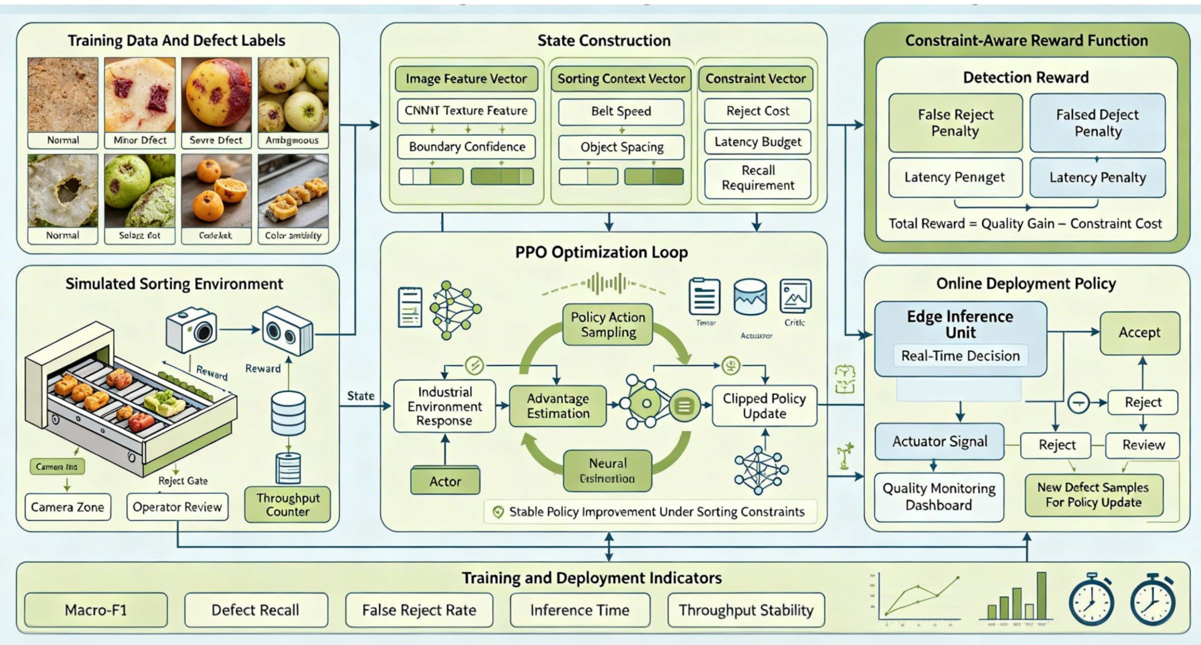

Figure 2. Constraint-aware PPO detection policy and sorting decision workflow

The policy network maps the state to a discrete-continuous hybrid action. The discrete part controls the inspection decision, and the continuous part adjusts the confidence threshold.

$$a_t = \{d_t, \Delta\gamma_t\} \sim \pi_\theta(a_t \mid s_t) \qquad \text{Eq.(5)}$$

The decision variable d_t includes pass, reject, refine, and confirm actions. The threshold adjustment is constrained within $[-0.12, 0.12]$ to avoid unstable policy jumps between adjacent products.

$$c_t = \lambda_1 T_t + \lambda_2 F_t + \lambda_3 R_t \qquad \text{Eq.(6)}$$

The constraint cost c_t combines inference time T_t, false-rejection penalty F_t, and missed-defect risk R_t. The three terms are normalized to $[0,1]$ so that no single industrial constraint dominates the policy early in training.

$$r_t = \eta_1 D_t - \eta_2 F_t - \eta_3 R_t - \eta_4 T_t \quad \text{Eq.(7)}$$

The reward assigns positive value to correct defect decisions D_t and penalizes false rejection, missed detection, and excessive latency. In training, η_3 is set higher than η_2 because missed contaminated products are more critical than rejecting a small number of acceptable products.

$$A_t = \sum_{k=0}^{K} (\gamma\lambda)^k \left[r_{t+k} - V_\psi(s_{t+k})\right] \quad \text{Eq.(8)}$$

The generalized advantage estimate stabilizes policy learning across product sequences. A 32 -step rollout is used because a sorting batch contains visually diverse products, and longer rollouts increase memory cost without improving policy stability.

$$\rho_t(\theta) = \frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta_{old}}(a_t \mid s_t)} \quad \text{Eq.(9)}$$

The probability ratio controls how much the new policy differs from the old policy. This is central to PPO because sorting policies should improve gradually; sudden threshold shifts can cause large false-sorting bursts.

$$L_P = E[\min(\rho_t A_t, \mathrm{clip}(\rho_t, 1-\epsilon, 1+\epsilon) A_t)] \quad \text{Eq.(10)}$$

The clipped surrogate objective prevents aggressive updates. ε is set to 0.18 after validation, because $\varepsilon = 0.30$ produced faster reward growth but unstable false-rejection rates under illumination disturbance.

Industrial Dataset, Evaluation Metrics, and Baseline Setup

The experimental dataset contains surface images from potatoes, apples, citrus fruit, and packaged bakery products. It includes 42,600 images, 9 defect types, and three industrial lines with different lighting conditions. The training, validation, and testing split is 70%, 10%, and 20%, with product-batch separation to avoid leakage. The baselines include rule-based thresholding, ResNet classifier, YOLO detector, attention-enhanced detector, and detector-only confidence tuning.

$$L_V = E\left[\left(V_\psi(s_t) - G_t\right)^2\right] \quad \text{Eq.(11)}$$

The critic loss fits the expected return G_t and supports stable advantage estimation. A shared encoder is used for actor and critic, while the final policy and value heads are separated.

$$L_C = E[\max(0, c_t - c_{lim})] \quad \text{Eq.(12)}$$

The constraint loss penalizes violation of the allowed industrial cost budget. The threshold c_{lim} is set according to the maximum acceptable latency of 25 ms and a false rejection tolerance of 3.5%.

$$L = -L_P + \beta_v L_V + \beta_c L_C - \beta_e H(\pi_\theta) \quad \text{Eq.(13)}$$

The final learning objective combines policy improvement, critic regression, constraint penalty, and entropy regularization. The entropy term encourages exploration during early training but is annealed after 60 epochs.

$$E_s = 1 - \left(w_m M_d + w_f F_r + w_l L_i\right) \quad \text{Eq.(14)}$$

The sorting efficiency score E_s summarizes missed-defect rate, false-rejection rate, and latency index. It is reported together with detection accuracy, precision, recall, F1, constraint violation rate, and throughput.

Results and Discussion

Detection Performance and Category-Level Recognition

Both sorting stability and detection accuracy have been enhanced by the New Model. Rule-based inspection, ResNet, YOLO, attention-enhanced detection, detector-only confidence tuning, and the constraint-aware PPO model are compared in terms of overall performance in Figure 3. The suggested model has a detection accuracy of 96.4%, which is 2.7% better than the best non-RL baseline, as seen in Figure 3(a). The defect recall rate is displayed in Figure 3(b); by lowering the confirmation level for minor high-risk faults, the suggested strategy

https://doi.org/10.64972/jaat.2024v2.343p35e:484-496

obtains 94.8%. The accuracy contrast is seen in Figure 3(c), where constraint learning has decreased false alarms. The false sorting rate is 7.9% for the YOLO baseline and falls to 3.3% with the suggested approach [16]. Figure 3(d) illustrates this.

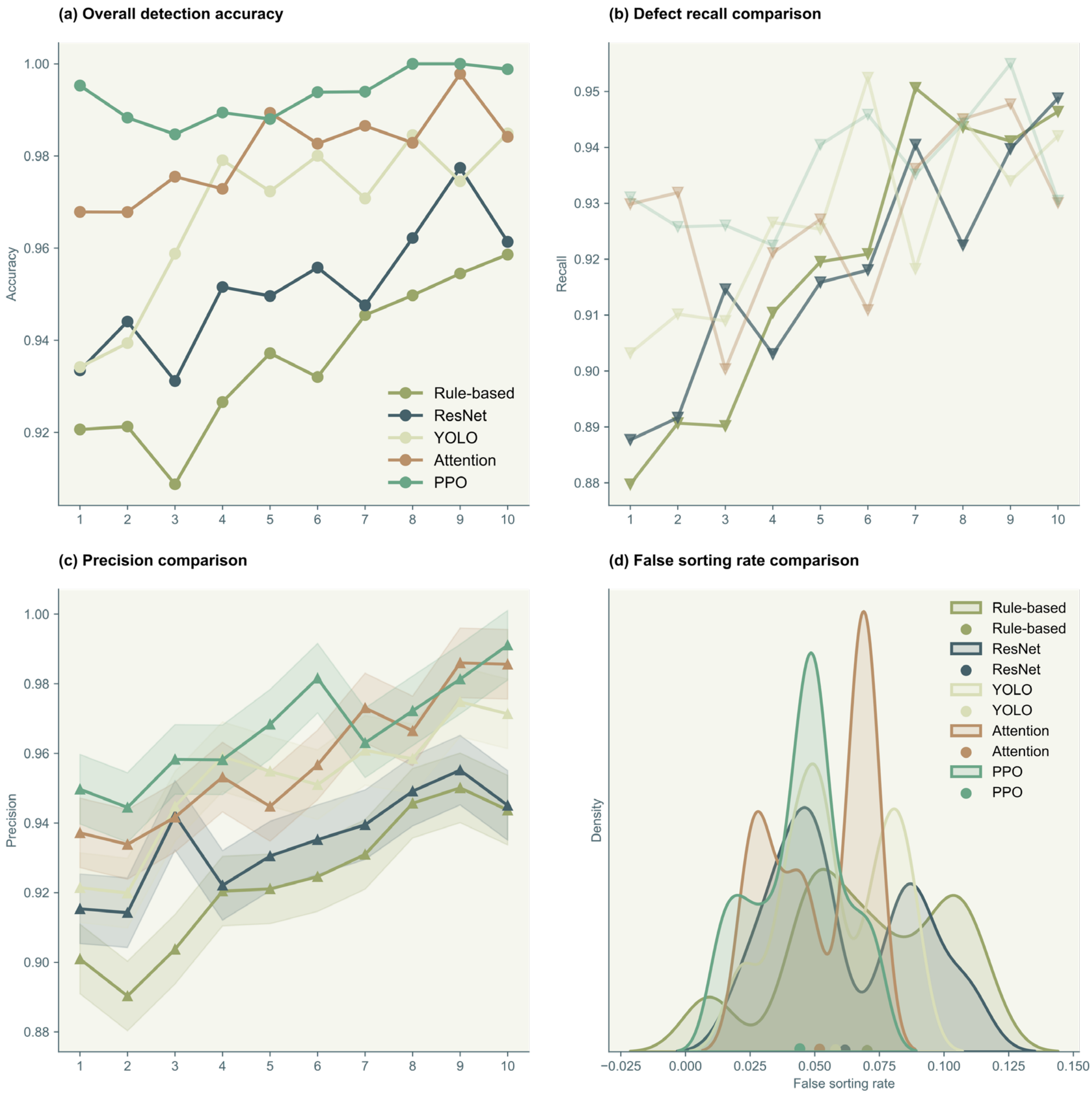

Figure 3. Food surface defect detection performance comparison. (a) Overall detection accuracy. (b) Defect recall comparison. (c) Precision comparison. (d) False sorting rate comparison.

The category-level recognition is displayed in Figure 4. Bruise detection is shown in Figure 4(a), and recall has increased from 88.2% to 95.1% following the PPO policy's request for optimization for low-contrast pressure marks. Due to the decreased misclassification of elongated natural texture as a crack, the precision for crack identification has risen by 4.1 % points, as seen in Figure 4(b). The outcome demonstrates that adaptive policy learning works best when there is ambiguous evidence of defects rather than unambiguous ones [17].

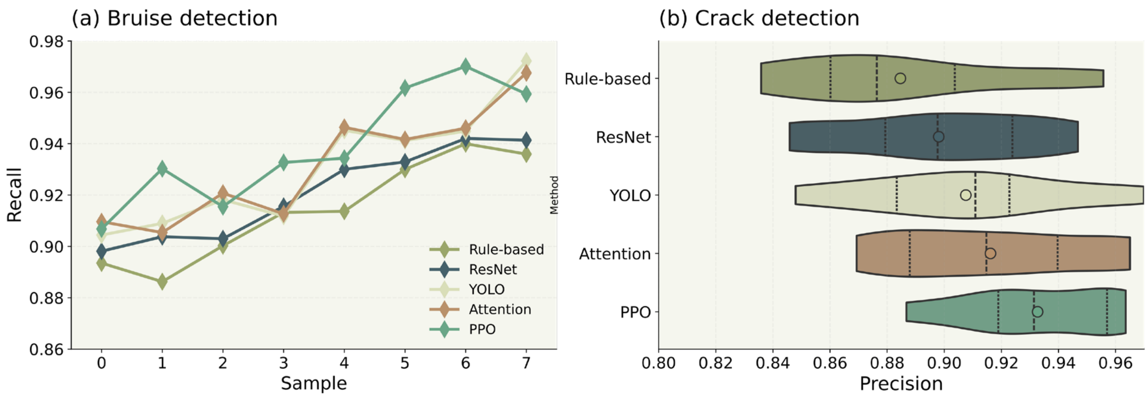

Figure 4. Category-level recognition under diverse defects. (a) Bruise detection. (b) Crack detection.

Constraint Learning and Industrial Robustness

Figure 5 below illustrates the constraint-aware policy's learning behavior. The convergence of rewards is depicted in Figure 5(a), and after around 72 epochs, the PPO policy has stabilized. The constraint violation rate, shown in Figure 5(b), drops from 18.6% during early training to 2.8% following convergence. Confidence threshold adaption is seen in Figure 5(c), where the model lowers the threshold for tiny flaws that are visually dependable and raises it for noisy texturing. The sorting policy stability for five random seeds is shown in Figure 5(d); the suggested model has a standard deviation of 0.38 % points in F1, which is less than the 0.91 % points for unconstrained PPO [18].

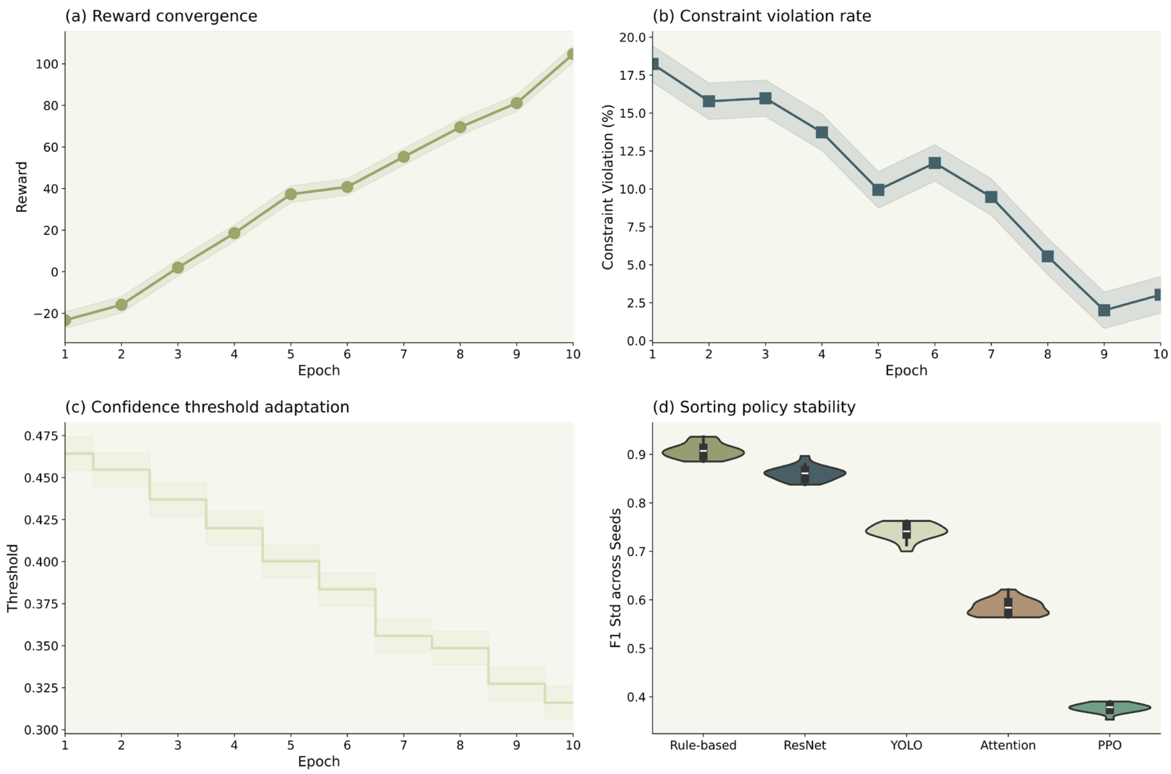

Figure 5. Constraint-aware PPO learning behavior. (a) Reward convergence. (b) Constraint violation rate. (c) Confidence threshold adaptation. (d) Sorting policy stability.

Figure 6 illustrates robustness to sorting-line disruption. Conveyor speed fluctuation between 0.8 and 1.6 m/s is assessed in Figure 6(a). The detector-only baseline is 89.4%, while the suggested model maintains the F1 at 93.0%

https://doi.org/10.64972/jaat.2024v2.343p35e:484-496

at high speeds. The lighting fluctuates between -20% and +20%, as seen in Figure 6(b). Because the reliability descriptor keeps the agent from unduly relying on low-quality confidence maps, PPO policy is stable [19].

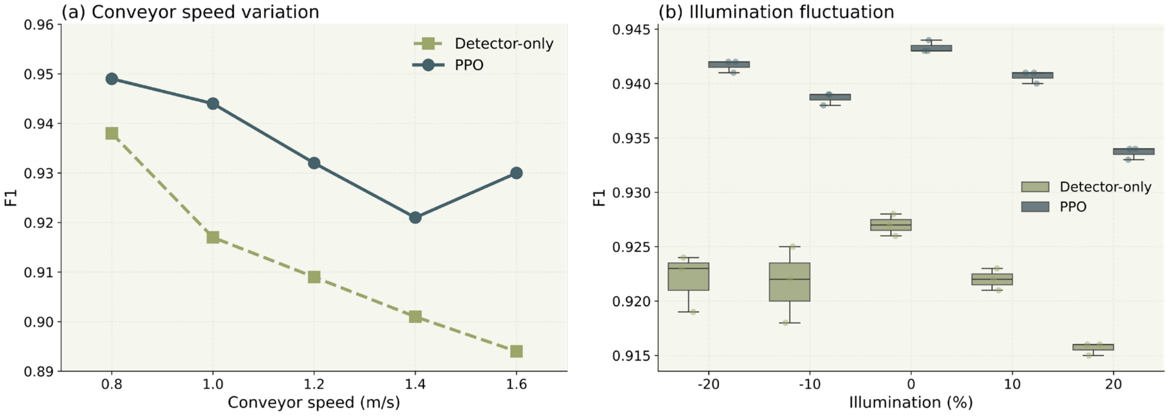

Figure 6. Robustness under sorting-line disturbance. (a) Conveyor speed variation. (b) Illumination fluctuation.

Edge Deployment and Cross-Line Adaptation

The following are the deployment results: Figure 7 Figure 7(a) illustrates inference latency; the suggested model satisfies the 25 ms sorting criterion with a delay of 18.7 ms per product on an edge GPU. The sorting throughput is shown in Figure 7(b), and without slowing down actuator response, the suggested approach may reach a rate of 3.2 products per second per lane. The cross-line transfer accuracy is shown in Figure 7(c). The model achieved 93.6% accuracy after fine-tuning with just 12% annotated pictures from a fresh line, and policy adaptation outperformed fixed confidence criteria [20].

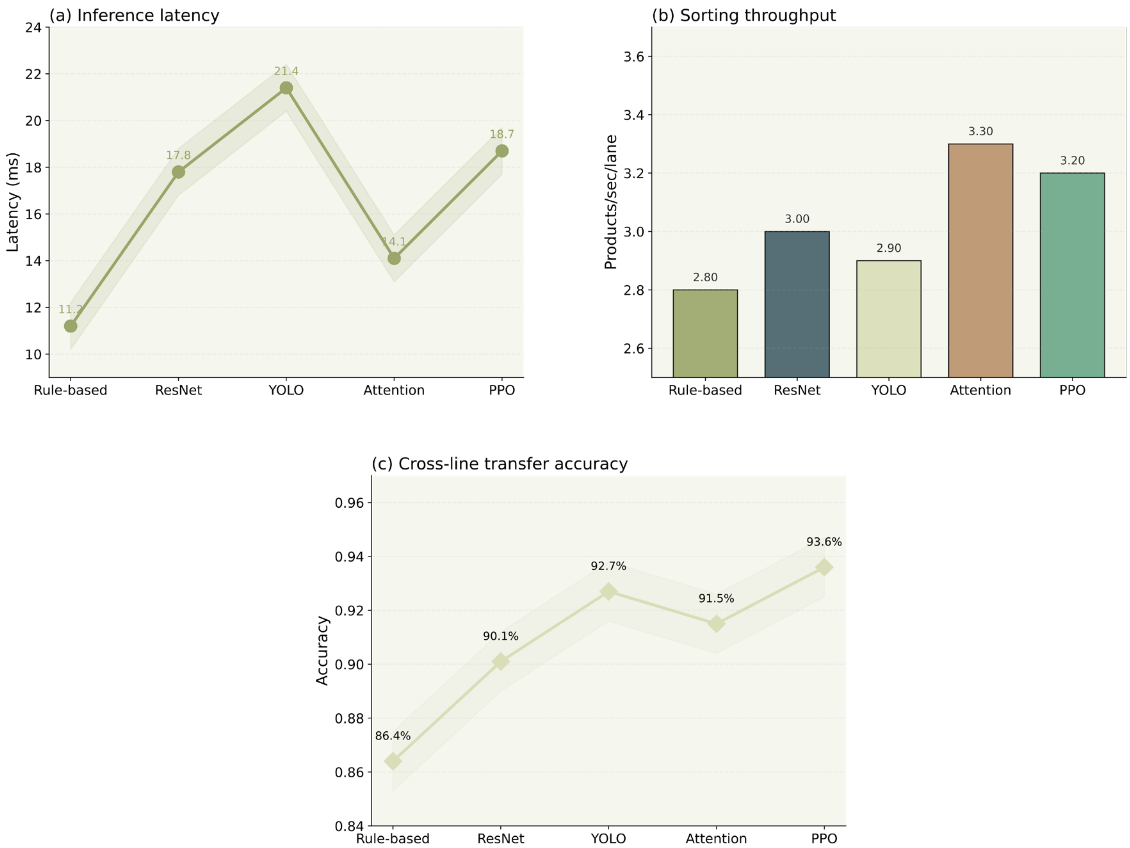

Figure 7. Edge deployment and cross-line generalization. (a) Inference latency. (b) Sorting throughput. (c) Cross-line transfer accuracy.

The behavior of the model is further investigated using ablation studies. The average inference time rises to 31.4 ms when the latency limitation is removed, which makes it inappropriate for the sorting actuator despite a minor gain in recall. The elimination of the false-rejection penalty increased product waste, while the absence of a missed-defect penalty decreased bruise recall by 6.8 % points. In the production setting, the detector-only mode is unable to synchronize threshold and rejection choices, while having high-accuracy local predictions [21]. Given the aforementioned, it has been discovered that favorable working circumstances are not guaranteed by a high visual score [22]. Thus, given explicit production restrictions, the suggested PPO layer acts as a decision regulator to translate visual evidence into sorting actions [23]. Motion blur, uneven lighting, and inherent differences in food texture all exacerbate this issue, and fixed thresholds exhibit instability [24].

The trajectories' characteristics and rules also demonstrate why constraint-aware reinforcement learning works better than straightforward confidence calibration. Because their confidence distributions are similar, low-contrast bruises and shallow fractures cannot be distinguished from innocuous texture alterations using a detector confidence threshold alone. The PPO policy, on the other hand, takes into account conveyor speed, remaining reaction time, proposal compactness, and visual reliability all at once in order to make alternative decisions for situations that are operationally different but visually similar. The agent immediately accepts a high-confidence defect conclusion when the picture is clean. If there is still enough actuator reaction time, blur or local glare initially seek refinement. Only when the missed-defect risk is substantial does it select a cautious sorting action under extreme time constraints. Such behavior, which strikes a balance between feature quality and action cost, might be seen as an adaptation strategy for visual inspection systems on mobile and automated inspection platforms [25]. State-dependent decision strategies are necessary because studies of flaws in fabric and industrial photographs also demonstrate that defect characteristics fluctuate greatly depending on the acquisition conditions [26].

For cross-product testing, the approach offers several practical advantages. The detector-only baseline dropped 8.3 % points in F1, whereas the suggested approach lost 4.9 % points when tested on citrus goods and trained on potatoes and apples. The primary explanation is because, as opposed to visual feature transfer, it involves policy adaptation. The PPO agent has discovered that in various confidence and dependability situations, circular spots, edge fractures, and local discoloration should be handled differently. This is also consistent with research on laser joining and visual quality inspection; in order to prevent brittle acceptance choices, deep models also need additional quality reasoning [27]. The suggested approach improved stain recall in the packaged bakery subgroup from 86.7% to 92.4%, but the precision improvement was minimal since the gradients in natural baking color were still seen as minor flaws. Therefore, while constrained PPO lowers operational errors, it cannot take the place of improved sensing in visually confusing situations. Studies on container inspection and other industrial pipelines have also demonstrated that model architecture and acquisition standardization have an impact on deployment performance [28].

Data-efficient data is another type of data. The detector-only confidence tuning hits 87.6% and the PPO-enhanced model remains at 91.2% F1 with just 30% of the training set. Thus, even with a small number of labeled defect samples, the policy may still take use of the structure in the reward signal. In order to lessen overfitting brought on by a lack of variety and dependability in the training state, research on data augmentation and few-shot learning has been carried out [29]. Studies using limited-sample augmentation and self-attention feature fusion exhibit similar behavior: the model becomes more stable after being subjected to numerous, controlled feature modifications [30]. In addition to picture augmentation, this paper’s-controlled variants incorporate simulated sorting limitations including speed changes, rejection delay, and false-alarm cost. As a result, compared to the static detector, the agent has learnt a greater policy manifold. The notion that features variety should be paired with label or decision consistency to improve generalization has received additional support from spectral-structure and soft augmentation research [31].

The constraint-violation curve is shown in a different figure. In order to improve the visual assurance at first, refining is often employed, but it takes time. The average number of refining operations per product is around 0.72 when the constraint penalty is activated, while recall remains over 94.0%. To put it another way, the learnt policy applies modifications selectively rather than evenly reducing the mistake. Research on few-shot learning using prototypes and graphs has demonstrated that organized feature calibration may preserve class separability with little supervision [32]. The same concept holds true for the new policy: the actor network eventually learns which low-confidence regions are likely to be regular textures and which are worth additional investigation. Although remote sensing and hyperspectral classification investigations are not in the same field,

https://doi.org/10.64972/jaat.2024v2.343p35e:484-496

they share methodological support since they deal with small sample sizes, subtle local changes, and the requirement for high feature discrimination [33].

Asymmetric Cost Design is also necessary, according to False Rejection Analysis. The overall error will be reduced, but more contaminated items will still pass if erroneous rejection and overlooked flaws are penalized equally. Improve mold and crack recall by 5.2% and 4.7%, respectively, and raise the weight of missed-defect risk by 1.6 times that of false rejection. Products with natural spots are over rejected when the missed-defect weight is raised over 2.0. This trade-off is similar to deep visual classification problems: because feature boundaries are not limited, increased sensitivity results in reduced accuracy [34]. High-dimensional models require explicit regularization when classes are visually near, according to review research on deep learning classifiers and remote visual analysis [35]. In the sorting-line model, a production restriction that corresponds to real inspection costs is represented by a mathematical term called the regularizer.

The PPO policy head is appropriate for deployment since it is comparatively modest from an implementation perspective. About 82.6% of the work is done by the Visual Encoder, which only adds 2.9 ms to the Detector pipeline through the actor-critic head. Research on distributed deep learning and high-performance inspection shows that splitting out heavy feature extraction from compact decision modules is a common way to achieve deployment efficiency [36]. Here, the same design is applied: the policy module chooses a sorting action depending on the feature state once feature extraction is completed once. Furthermore, throughout training, the policy entropy steadily drops; otherwise, it would have arrived at a deterministic reject-or-pass policy prematurely. New items have been introduced, and these models must have the flexibility to recognize new surface patterns.

Thus, it is also possible to add more sensors, based on the study above. 3D surface profiles can differentiate between fractures and shadows, while near-infrared or hyperspectral imaging can differentiate between interior bruises and outward color changes. The survey on deep learning for remote sensing image processing states that challenging visual discrimination can be resolved by using multi-scale or multi-spectral data. Increase expenses and synchronization issues by adding sensors. Therefore, for lines where hardware replacement is not immediately achievable, the current RGB-based PPO architecture provides a workable temporary alternative. By employing generative models to increase the distribution of defect states for the policy learner, it may also be used in conjunction with generative augmentation to address uncommon defects like early mold or tiny cut marks. The produced samples should be limited by task-level utility rather than visual realism, according to both classical generative adversarial network research and later augmentation studies. This utility may be directly acquired as recall, false rejection, latency, and sorting efficiency in the suggested setup. As a result, the final application will be a closed-loop inspection control system for defect detectors, and their output will be subjected to action based on visual inference rather than operating as independent image classifiers.

Conclusion

A Constraint-Aware PPO Model is put forth for the detection of food surface defects in industrial sorting lines. Adaptive policy learning and multi-scale visual feature extraction are used in tandem. The inspection focus, confidence threshold, and sorting decision are then modified in accordance with defect evidence and production restrictions. The experiment demonstrates how reinforcement learning may improve the memory rate of small faults by including delay, false sorting, and missed-defect risk into the reward function.

The first technological accomplishment is the transformation of the food defect detection issue into a limited sequential decision-making problem. The PPO policy chooses an action based on conveyor speed, actuator timing, and inspection-cost limitations. The visual encoder produces defect texture, confidence, and reliability information. This design makes more stable choices under changes in illumination, motion blur, etc. than the detector-only baseline. Consequently, compared to a set confidence threshold or a solely image-level classifier, it is better suited for industrial food sorting.

Future research might expand the system by including multi-modal sensing, including 3D surface profiles, near-infrared signals, and hyperspectral imaging. Connect the incentive scheme to the real manufacturing economy, including food safety hazards, downstream reinspection expenses, and product prices. The low-power sorting-line controller's real-time performance may be further enhanced by lightweight policy distillation and embedded deployment.

Author Contributions

Radu Păunescu contributes to conceptualization, methodology, software, validation, analysis, investigation, data collection, draft preparation, manuscript editing, visualization, supervision. Aurel Zăpodeanu contributes to data collection, draft preparation. Gheorghe Fieraru contributes to methodology, software, validation. All authors have read and agreed with the manuscript before its submission and publication.

Funding

This research received no specific financial support from any funding agency.

Institutional Review Board Statement

Not applicable.

References

[1] Wang, H., & Xiao, C. (2021). Potato surface defect detection based on deep transfer learning. Agriculture, 11(9), 863. https://doi.org/10.3390/agriculture11090863

[2] Bergmann, P., Fauser, M., Sattlegger, D., & Steger, C. (2019). MVTec AD: A comprehensive real-world dataset for unsupervised anomaly detection. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 9584-9592. https://doi.org/10.1109/CVPR.2019.00982

[3] Qi, L., Yang, J., & Zhong, Y. (2020). A review on industrial surface defect detection based on deep learning technology. Proceedings of the 3rd International Conference on Machine Learning and Machine Intelligence, 24-30. https://doi.org/10.1145/3426826.3426832

[4] Li, B., Zhang, X., Liang, C., & Wei, T. (2019). Deep learning: Excellent method at surface defect detection of industrial products. 2019 IEEE 3rd Advanced Information Management, Communicates, Electronic and Automation Control Conference, 712-716. https://doi.org/10.1109/IMCEC46724.2019.8984019

[5] Jian, B. L., Hung, J. P., Wang, C. C., & Liu, C. C. (2020). Deep learning model for determining defects of vision inspection machine using only a few samples. Sensors and Materials, 32(12), 4217. https://doi.org/10.18494/SAM.2020.3101

[6] Rathod, V., & Salehi, M. (2020). Vision system with deep learning classifiers for automatic quality inspection. Pattern Recognition and Tracking XXXI, 11400, 114000D. https://doi.org/10.1117/12.2555032

[7] Cai, J., & Zhang, L. B. (2021). Apple surface defect detection based on deep learning. 2021 International Conference on Electronic Information Technology and Smart Agriculture, 491-495. https://doi.org/10.1109/ICEITSA54226.2021.00099

[8] Deng, L., Cheng, W., Feng, Z., & Xiang, J. (2021). Industrial laser welding defect detection and image defect recognition based on deep learning model developed. Symmetry, 13(9), 1731. https://doi.org/10.3390/sym13091731

[9] Bin Roslan, M. F., Ibrahim, Z., & Aziz, A. (2022). Real-time plastic surface defect detection using deep learning. 2022 IEEE 12th Symposium on Computer Applications & Industrial Electronics, 111-116. https://doi.org/10.1109/ISCAIE54458.2022.9794475

[10] Balcioglu, H. E., Sezen, B., Gok, M. S., & Tunca, B. (2022). Image processing with deep learning: Surface defect detection of metal gears through deep learning. Materials Evaluation, 80(2), 44-53. https://doi.org/10.32548/2022.me-04230

[11] Chen, Y., Tu, X., Li, Y., & Chen, B. (2021). Defect detection using deep lifelong learning. 2021 IEEE 19th International Conference on Industrial Informatics, 1-6. https://doi.org/10.1109/INDIN45523.2021.9557417

[12] Liu, Y., Wang, J., Yu, H., & Li, X. (2022). A non-invasive system for on-line surface defect detection on special-shaped steel towards real production lines. 2022 IEEE International Instrumentation and Measurement Technology Conference, 1-6. https://doi.org/10.1109/I2MTC48687.2022.9806583

[13] He, Z., & Liu, Q. (2020). Deep regression neural network for industrial surface defect detection. IEEE Access, 8, 35583-35591. https://doi.org/10.1109/ACCESS.2020.2975030

[14] Jia, W., & Wang, H. (2022). Research on industrial production defect detection method based on machine vision technology in industrial internet of things. Traitement du Signal, 39(6), 2061-2068. https://doi.org/10.18280/ts.390618

[15] Zavrtanik, V., Kristan, M., & Skocaj, D. (2021). Reconstruction by inpainting for visual anomaly detection. Pattern Recognition, 112, 107706. https://doi.org/10.1016/j.patcog.2020.107706

[16] Zakharenkov, A., & Makarov, I. (2021). Deep reinforcement learning with DQN vs. PPO in VizDoom. 2021 IEEE 21st International Symposium on Computational Intelligence and Informatics, 000131-000136. https://doi.org/10.1109/CINTI53070.2021.9668479

[17] Liu, Y., Ding, Z., & Liu, X. (2020). A constrained reinforcement learning based approach for network slicing. 2020 IEEE 28th International Conference on Network Protocols, 1-6. https://doi.org/10.1109/ICNP49622.2020.9259378

[18] Rousseas, N., Bechlioulis, C. P., & Kyriakopoulos, K. J. (2020). Optimal robot motion planning in constrained workspaces using reinforcement learning. 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems, 8034-8040. https://doi.org/10.1109/IROS45743.2020.9341148

[19] Wen, L., Duan, J., Li, S. E., & Xu, S. (2020). Safe reinforcement learning for autonomous vehicles through parallel constrained policy optimization. 2020 IEEE 23rd International Conference on Intelligent Transportation Systems, 1-7. https://doi.org/10.1109/ITSC45102.2020.9294262

[20] Kabas, B. (2022). Autonomous UAV navigation via deep reinforcement learning using PPO. 2022 30th Signal Processing and Communications Applications Conference, 1-4. https://doi.org/10.1109/SIU55565.2022.9864769

[21] Park, J., & Kim, J. (2020). Autonomous flying of drone based on PPO reinforcement learning algorithm. Journal of Institute of Control, Robotics and Systems, 26(11), 955-963. https://doi.org/10.5302/j.icros.2020.20.0125

[22] Shen, J., Yu, W., & Chen, X. (2021). Rules-PPO-QMIX: Multi-agent reinforcement learning with mixed rules for large scene tasks. 2021 China Automation Congress, 6731-6736. https://doi.org/10.1109/CAC53003.2021.9728241

[23] Zhang, Y., Wang, C., & Cai, L. X. (2020). SREC: Proactive self-remedy of energy-constrained UAV-based networks via deep reinforcement learning. GLOBECOM 2020 IEEE Global Communications Conference, 1-6. https://doi.org/10.1109/GLOBECOM42002.2020.9348219

[24] Wu, Y., Chen, S., Ji, X., & Chen, C. (2022). Mapless navigation based on VDAS-PPO deep reinforcement learning. 2022 China Automation Congress, 1945-1950. https://doi.org/10.1109/CAC57257.2022.10055939

[25] Han, R., Chen, S., & Hao, Q. (2020). Cooperative multi-robot navigation in dynamic environment with deep reinforcement learning. 2020 IEEE International Conference on Robotics and Automation, 448-454. https://doi.org/10.1109/ICRA40945.2020.9197209

[26] Voronin, V., Sizyakin, R., Zhdanova, M., & Semenishchev, E. (2021). Automated visual inspection of fabric image using deep learning approach for defect detection. Automated Visual Inspection and Machine Vision IV, 11787, 117870N. https://doi.org/10.1117/12.2592872

[27] Han, S., Kim, J., & Choi, H. (2021). Assessment of laser joining quality by visual inspection, computer simulation, and deep learning. Applied Sciences, 11(2), 642. https://doi.org/10.3390/app11020642

[28] Li, D., Chen, D., Jin, B., Shi, L., Goh, J., & Ng, S. K. (2019). MAD-GAN: Multivariate anomaly detection for time series data with generative adversarial networks. Artificial Neural Networks and Machine Learning, 703-716. https://doi.org/10.1007/978-3-030-30490-4_56

[29] Tian, Y., Yang, G., Wang, Z., Li, E., & Liang, Z. (2019). Detection of apple lesions in orchards based on deep learning methods of CycleGAN and YOLOV3-Dense. Journal of Sensors, 2019, 7630926. https://doi.org/10.1155/2019/7630926

[30] Niedbała, G., Kurek, J., Świderski, B., Wojciechowski, T., Antoniuk, I., & Bobran, K. (2022). Prediction of blueberry (Vaccinium corymbosum L.) yield based on artificial intelligence methods. Agriculture, 12(12), 2089. https://doi.org/10.3390/agriculture12122089

[31] Bhargava, A., & Bansal, A. (2021). Fruits and vegetables quality evaluation using computer vision: A review. Journal of King Saud University - Computer and Information Sciences, 33(3), 243-257. https://doi.org/10.1016/j.jksuci.2018.06.002

[32] Mahanti, N. K., Pandiselvam, R., Kothakota, A., Chakraborty, S. K., Kumar, M., & Cozzolino, D. (2022). Emerging non-destructive imaging techniques for fruit damage detection: Image processing and analysis. Trends in Food Science & Technology, 120, 418-438. https://doi.org/10.1016/j.tifs.2021.12.021

[33] Koirala, A., Walsh, K. B., Wang, Z., & McCarthy, C. (2019). Deep learning for real-time fruit detection and orchard fruit load estimation: Benchmarking of MangoYOLO. Precision Agriculture, 20(6), 1107-1135. https://doi.org/10.1007/s11119-019-09642-0

[34] Koirala, A., Walsh, K. B., Wang, Z., & McCarthy, C. (2019). Deep learning-method overview and review of use for fruit detection and yield estimation. Computers and Electronics in Agriculture, 162, 219-234. https://doi.org/10.1016/j.compag.2019.04.017
[35] Lin, E., Chen, Q., & Qi, X. (2020). Deep reinforcement learning for imbalanced classification. Applied Intelligence, 50(8), 2488-2502. https://doi.org/10.1007/s10489-020-01637-z
[36] Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. Journal of Big Data, 6, 60. https://doi.org/10.1186/s40537-019-0197-0