Vision Control for Cable Binding Robot in Offshore and Marine Industry

Jing Zhong Tee
Research and Development
Sembcorp Marine Ltd
Singapore
jingzhong.tee@sembmarine.com

Ye Zhen
Department of Mechanical Engineering
National University of Singapore
Singapore
18019550228@163.com

Chin Boon Chng
Department of Mechanical Engineering
National University of Singapore
Singapore
mpeccbo@nus.edu.sg

Chee Kong Chui
Department of Mechanical Engineering
National University of Singapore
Singapore
mpecck@nus.edu.sg

***Abstract*— The development of a robotic system aimed at enhancing the cable binding process for Floating Production Storage Offloading (FPSO) vessels addresses critical challenges within the labour-intensive shipbuilding industry. The proposed robotic system incorporates computer vision technology which has the ability to detect stainless steel buckles and relay their precise locations to improve the efficiency and effectiveness of the overall process. This study divides the entire process into three steps: object detection, key point prediction, and pose estimation. We have leveraged You Only Look Once (YOLOv8) to detect and predict the key points of the stainless steel buckles. The study has successfully tackled a significant challenge in machine vision, namely detecting the shiny metal parts of the buckle against a similarly shiny background. Furthermore, we validated the effectiveness of our computer vision approach by adopting a training approach that combines real and synthetic data, which mitigates overfitting issues during neural network training.**

Keywords—RGB image, robotics, neural network, deep learning, machine vision.

I. INTRODUCTION

Computer vision technology has advanced significantly in recent years, reaching a level of maturity that enables it to effectively handle various aspects of metal detection across industrial applications. Automated inspection systems leveraging computer vision are now commonplace in the manufacturing industry, offering both high speed and accuracy. Vision-based object detection has particularly matured, with deep learning models like convolutional neural networks (CNNs) demonstrating impressive accuracy in detecting objects, including metallic ones. These models leverage sophisticated algorithms to analyze visual data and identify objects with remarkable precision [1-3]. As a result, computer vision technology has become an invaluable tool in manufacturing, enabling efficient and reliable inspection processes. By automating tasks that were traditionally performed manually, it enhances productivity, reduces errors, and ensures consistent quality control [4, 5].

Today, industrial metal parts are essential components in a wide array of products, inevitably playing a crucial role in various industries. Consequently, our cable binding robotic system project faces the challenge of accurately recognizing, locating, and estimating the pose of stainless steel buckles (buckle), as depicted in Figure 1, during the cable binding process. These challenges stem from the distinctive properties of metal surfaces [6-10]. While some pose estimation methods rely on depth information obtained by actively projecting coded patterns, this approach proves ineffective for shiny metal parts. The reflective nature of metal surfaces can distort or obscure the projected patterns, rendering them misleading or undetectable. Therefore, it's imperative to devise a strategy that doesn't solely rely on RGB images to overcome this hurdle.

Traditional cable binding methods typically involve manual operation using conventional hand-held tools. These methods often require workers to manually tighten and cut metal bands to secure cables. Among these tools, the steel banding tool [18] serves as a popular alternative for tightening and cutting steel bands effectively. In Figure 2, we illustrate some of the existing individual tools utilized in manual cable binding processes, including a wire cutter for band cutting post-binding, a T-slot tool aiding in steel band tightening, and a hammer used to secure the band onto the buckle by stamping.

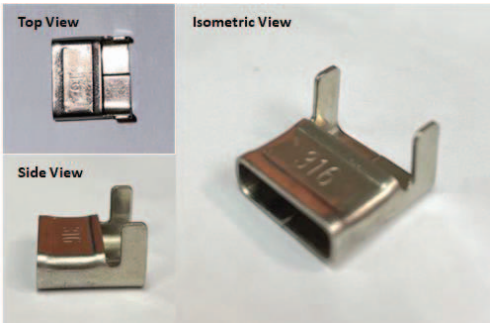

Fig. 1. Overview of the stainless steel buckle (buckle)

Currently, the cable binding process on-site involves manual labour, with workers manually securing cables to cable ladders present on every ship vessel. These cable ladders serve as elevated structures along which cables are laid down and fastened using steel bands and buckles. To ensure the secure attachment of cables onto cable ladders, particularly in shipbuilding related to the oil and gas industry, steel bands are utilized for binding purposes. Figure 2 illustrates the cable binding process. The cable binding process is highly labour-intensive, with workers typically able to complete only around a hundred meters of cable binding per day. This process involves securing multiple cable bundles across the width of the cable ladder. Specifically, every alternate rung of the cable ladder is secured using a steel band and a buckle. This labourious

979-8-3503-5409-6/24/$31.00 ©2024 IEEE
DOI 10.1109/CAI59869.2024.00272

process extends over considerable lengths and heights onboard the vessel.

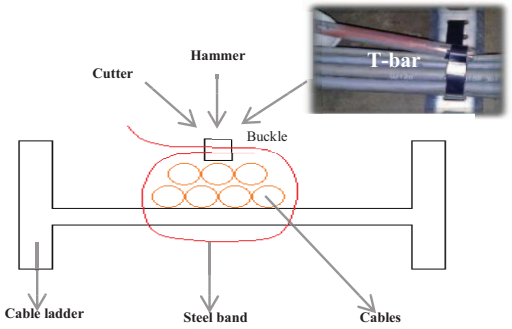

Fig. 2. Overview of cable binding process

Another challenge in the cable binding process arises from its overhead execution, necessitating the erection of scaffolding for safety compliance. This not only adds to the logistical complexity but also entails temporary halting of ground-level operations until the binding is finished. Given the arduous and labour-intensive nature of this conventional approach, Figure 3 illustrates the implementation of a robotic system integrated with computer vision technology. This advancement is anticipated to markedly diminish both the expended man-hours and the overall costs associated with the process.

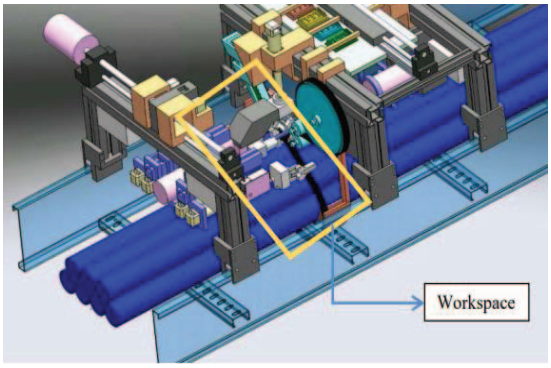

Fig. 3. Workspace area to be viewed by webcam

This robotic system is engineered with five manipulators tailored to operate effectively within constrained spaces. Given that the application entails multiple manipulators equipped with individual end effectors, they operate sequentially to execute the cable binding process step by step. The analytical kinematics of this robotic system integrate counter and digital signals to communicate with the controller and motor drivers, facilitating precise manipulation of the manipulators and end-of-arm tooling to execute specific tasks at predefined positions. Upon receiving signals from the controller, the manipulator's motor drive extends its tooling arm outward to perform designated tasks such as grasping the buckle and steel band, threading, cutting, and tightening the band. To enhance these tasks, a computer vision system conducts a final verification of the buckle position, alerting the operator via the computer interface to proceed to the next cable binding process once confirmed.

National University of Singapore (NUS) and Sembcorp Marine Ltd. (SCM)

In the context of addressing the challenges posed by estimating the pose of shiny metal parts, a similar approach which is the current state of the art is described in [11], This approach involves leveraging advanced computer vision techniques and deep learning methodologies, such as the Mask R-CNN (Region-based Convolutional Neural Network), specifically tailored for object recognition and pose estimation using red, green and blue (RGB) images. Currently, You Only Look Once (YOLOv8) is a popular deep learning model for object detection tasks [12-17]. Although YOLOv8 is renowned for its effectiveness in object detection tasks, it has not been tested to accurately recognize, locate, and estimate the pose of shiny metal parts. Hence its overall performance and reliability in real-world applications is uncertain.

Although computer vision for metal detection has made significant strides in various industrial applications, the efficacy of vision-based metal detection remains contingent upon several factors, each requiring validation to enhance the reliability of our proposed algorithms for buckle detection. The reflective and glossy characteristics of metal surfaces present challenges due to specular reflections. Addressing such instances may necessitate the utilization of advanced algorithms, potentially integrating visual data with analytical kinematics. Additionally, the buckle often appears partially obscured or distorted due to obstructed views or shadows on the surface, rendering it invisible in captured images. This underscores the need for further refinement and adaptation of detection methods to accommodate such scenarios. Moreover, examining all the geometric aspects of the buckle proves excessively labourious, requiring adjustments to both light direction and camera angle. To tackle this challenge, we will conduct an investigation aimed at testing the detection process without the need for altering light illumination or camera view.

In computer vision, estimating the pose of shiny metal parts presents challenges due to potential inter-reflection and overlapping. Furthermore, labeling metal parts for object detection is prohibited. Moreover, the similarity in material between the metal parts and the shiny background reduces visibility for detection. Recently, deep learning, particularly Mask R-CNN (MRCNN), has demonstrated effectiveness in addressing these challenges by accurately estimating the pose of texture-less shiny metal parts [11]. While MRCNN stands as the current state-of-the-art solution for this application, YOLO offers a compelling alternative. YOLO is a single-stage object detection system known for its superior speed, ease of implementation, and compatibility with less powerful hardware. Unlike multi-stage approaches like MRCNN, YOLO predicts bounding boxes and class probabilities directly in a single forward pass through the neural network, making it an efficient choice for real-time applications with resource constraints [12-17].

Given that YOLO hasn't been specifically designed to detect objects like metal parts against a shiny metal background, this paper will focus on developing vision control algorithms by implementing YOLO in this specific scenario. The study outlined in the paper will involve iterative training of the YOLO model to enhance its accuracy in detecting shiny stainless steel buckles, aiming to achieve results similar to the current state-of-the-art methods.

The upcoming test will showcase the application of visual control on the cable binding robotic system through

the utilization of generated training datasets and the implementation of the YOLOv8 platform to enhance computer vision capabilities for buckle detection in this context. This will be detailed in the subsequent section. In this research project, the key contributions of this paper are as follows:

1. Development of a custom trained dataset to evaluate the robustness of integrating YOLOv8 for buckle detection across various scenarios outlined earlier.

2. Experimental validation of the trained custom datasets through a comparison between datasets generated from photographs and those derived from 3D computer-aided design (CAD) models.

3. Demonstration of real-time video processing tests, validated through dataset comparisons.

The paper's organization is structured as follows. Section 2 provides an overview of computer vision control, experimental setup, and dataset training methods. Test scenarios are elucidated in Section 3. Section 4 presents simulation results along with discussions on all test scenarios. Finally, Section 5 offers conclusions and outlines potential avenues for future research.

II. Methodology with System Overview

A. Overall System Configuration

In the configuration of this robotic system, the analytical kinematics supports the utilization of a joint space control scheme. This choice simplifies algorithmic complexity within the control architecture and facilitates swift operation for preplanned motion control. Leveraging computer vision enables the robotic arm's joint space to navigate within the constraints of the workspace. Given the critical role of computer vision in detecting the buckle's position within the kinematic model, its implementation will undergo testing across various scenarios anticipated during operation. These tests aim to demonstrate the system's reliability under realistic conditions.

Computer vision serves as the automated feedback control mechanism for the robot, operating based on the comparison between the input image and a reference image. Illustrated in Figure 4, this approach differs from operational space control, where inverse kinematics are embedded within the feedback loop. Here, the process involves solving the kinematics model initially to derive the desired input joint space, qd, which is then controlled based on the actual tracked value of the joint output, q. However, in general manipulator applications, this method may result in inaccurate output in the operational space, xe, due to structural uncertainty, as the feedback remains independent of xe.

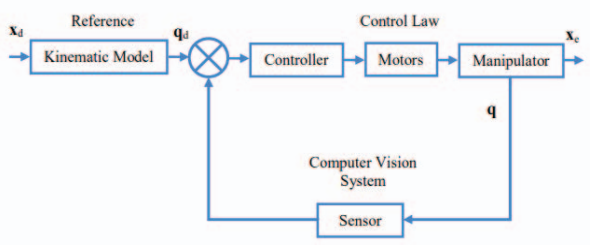

Fig. 4. Control structure of the computer vision control system

B. Configuration of Set up Experiment

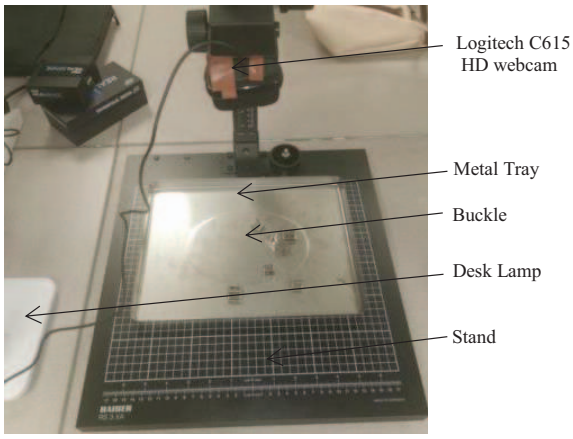

Fig. 5. Overview of the Set-up Experiment in Laboratory

In our laboratory experiment setup, we utilize a stand, a Logitech C615 HD webcam, a metal tray, and a desk lamp with adjustable lighting. Images of buckles are obtained and prepared by capturing them from various perspectives, achieved by adjusting the stand's height. To simulate different outdoor lighting conditions, we vary the lighting intensity from the desk lamp. Additionally, we modify the background for the buckle in the captured images, incorporating both a shiny stainless steel surface and a dark wooden surface. These adjustments aim to provide a comprehensive dataset for testing the reliability of our computer vision system under diverse environmental conditions.

C. Components Design with methods

For YOLOv8 model training, we curated a specialized dataset divided into training and validation sets. This dataset comprises real-world images captured under diverse backgrounds and lighting conditions, along with synthetic images generated using Blender software. Leveraging Blender's 3D rendering engine, we created images of buckles positioned at different angles and under various lighting conditions. Subsequently, we trained the YOLOv8 model, which offers flexibility in parameter sizes, by fine-tuning initial and final learning rates, as well as parameters like intersection over union (IoU). This iterative process culminated in a neural network model and optimized parameters capable of accurately predicting buckle positions and key points.

This strategy aimed to overcome limitations associated with insufficient original data and to counteract potential overfitting issues during image training. For the real data subset, we utilized a total of 180 images, with 145 allocated for the training set and 35 for validation purposes. Conversely, the synthetic data subset consisted of 1000 images, segmented into 790 for training and 210 for validation. This hybrid training approach enabled us to leverage the strengths of both real and synthetic datasets, enhancing the robustness and generalization capabilities of the trained model.

III. Object Detections with Test Scenarios

In our test scenarios, we employ the YOLOv8 network for object detection, recognizing the challenges inherent in traditional metal part segmentation. We utilize CSPDarkNet

as the backbone for feature extraction, enhancing the network's capability to discern intricate details. We utilize the synthetic data to train the metal part detection network. Figure 6 showcases a 3D model of a buckle, representing the object of interest in our detection tasks.

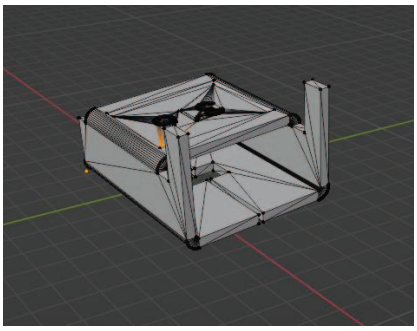

Fig. 6. 3D-Max model in the virtual 3D environment

Synthetic data generation should encompass a range of conditions, including variations in background, lighting, and reflectance effects, as detailed in reference [19]. Blender, a rendering engine, is adept at simulating such conditions. As depicted in Figure 7, this software renders synthetic data by situating the 3D-Max model within a virtual 3D environment governed by physical conditions. Figure 8 illustrates the setup of the image generator. On the left side, the diagram depicts the configured virtual scene environment. In the middle, a Blender script generates synthetic images of buckles with diverse angles, positions, and lighting intensities. On the right side, options for the surface material of the buckles in the rendered images are presented.

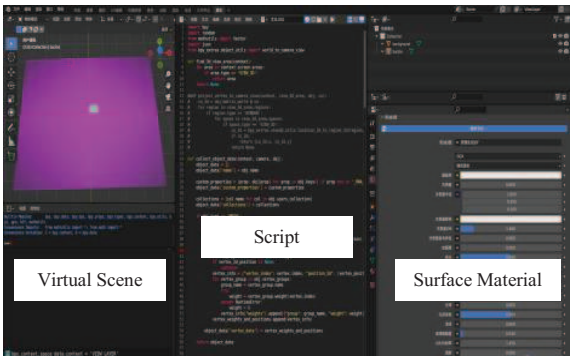

Fig. 7. Blender program interface

In our synthetic data generation process, we incorporate diverse real-world scenes as backgrounds within a virtual environment. By positioning the 3D-max model in various orientations, we generate a series of images depicting metal parts against different backgrounds and poses. Both Figure 8 and Figure 9 showcase rendered synthetic images featuring distinct backgrounds and illumination conditions, representing two scenarios for buckle detection.

Our real-world environment includes metal trays and darkened wooden boards. YOLOv8, classified as a one-stage algorithm, utilizes a single convolutional neural network to directly predict the classes and locations of various targets. Compared to previous iterations, YOLOv8 is faster and more accurate, enhancing the efficiency and precision of our detection tasks.

To generate synthetic images of buckles, we initially created a 3D model of a buckle using 3D-Max software and subsequently imported this model into Blender. Leveraging Blender's Python scripting capabilities, we synthesized scenes featuring multiple instances of the buckle model. Within these scenes, we incorporated multiple light sources and varied camera positions to capture different perspectives. By rendering the buckle surface with metallic properties, it reflected light and cast shadows in the background. In one synthetic image, we set the number of synthesized buckles to 10. To mimic laboratory fluorescent lighting conditions, we configured a panel light source positioned above the scene at a specified height along the Z-axis.

To derive the 2D bounding boxes and keypoint information from the synthetic images, we initially extracted the 3D keypoint data for all buckles within the scene. These 3D keypoints were then projected from the 3D space onto the 2D coordinates of the camera's plane, resulting in corresponding pixel positions along the x and y axes in the synthetic images. we then exported a JSON file from Blender containing the transformed 2D coordinates of all buckle keypoints. There are 12 keypoints previously chosen, annotating them in the synthetic images to prepare the data for future image training. Subsequently, we utilized a program to determine the maximum and minimum values in both the x and y directions, enabling the drawing of bounding boxes around the detected buckles. During the generation of synthetic images, the buckle's surface nodes were configured to use principled (bidirectional scattering distribution function) BSDF, with a specular value of 0.5, roughness of 0.5, and Sheen Tint of 0.5. These settings ensured realistic rendering of the buckles' surface properties.

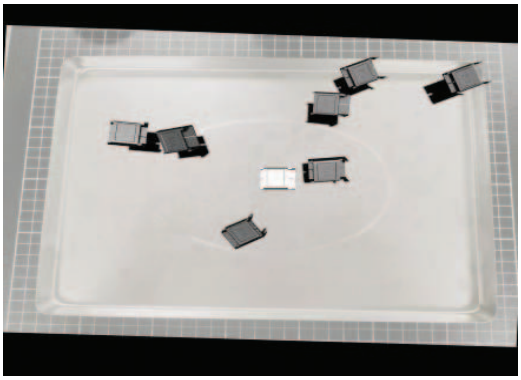

Fig. 8. Synthesis picture with a metal background

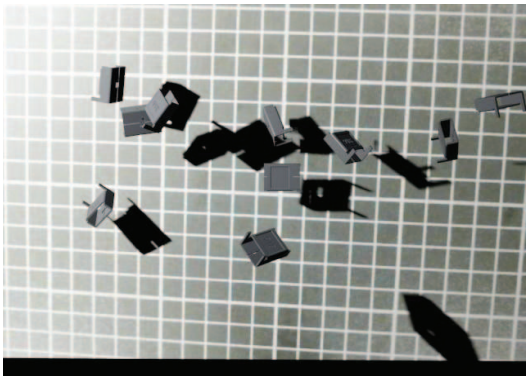

Fig. 9. Synthesis picture with a dark wooden background

In capturing real data, we employed a Logitech C615 Webcam with a picture resolution of 1024×768. The photo shoots were conducted against backgrounds comprising a metal plate and a darkened wooden board. To accommodate the recognition of both large and small buckles, we varied the height of the photo shoots to 10cm and 30cm. The real photos were imported into the Labelme software, where we manually annotated the positions of different buckles within the images. This annotation process facilitated the creation of a labeled dataset, essential for training and validating our computer vision models.

IV. RESULTS

During the experiment, we trained the YOLOv8 model using a hybrid dataset comprising both real and synthetic images. Multiple models were trained concurrently with varying parameter scales. Table 1 shows the model with 5 different parameter configurations, where *Ir0* is the initial learning rate. *Irf* stands for the final learning rate. *Iou* represents the intersection over Union threshold, measuring the overlap between the predicted bounding box and the ground truth. The training employed the Stochastic Gradient Descent (SGD) optimizer. "*Close_mosaic=0*" disable mosaic augmentation for final epochs. "*Dropout=0.0*" means using dropout regularization.

Table I
Model and typical training parameters in experiment

No.	Model Version	Parameters
1	YOLOv8s	lr0=0.01, lrf=0.01, iou=0.7, optimizer=SGD, close_mosaic=0, dropout=0.0, batch_size=4, epoch=30
2	YOLOv8n	
3	YOLOv8m	
4	YOLOv8l	
5	YOLOv8x	

Table 2 presents the experimental results, encompassing Precision, Recall, mAP50, and mAP50-95 metrics for the five model versions evaluated. Analysis of the results reveals that as the number of model parameters and complexity increase, both detection precision and recall exhibit gradual improvements, although the increase in mAP50 is less pronounced. Notably, the Precision metric for YOLOv8l reaches 0.995, meeting the requirements of our work.

Table II
Results in metrics of the experiments

No	Model Version	Precision	Recall	mAP50	mAP50-95
1	YOLOv8s	0.989	0.987	0.993	0.711
2	YOLOv8n	0.980	0.975	0.993	0.690
3	YOLO v8m	0.987	0.988	0.994	0.738
4	YOLOv8l	0.995	0.993	0.994	0.759
5	YOLOv8x	0.992	0.996	0.992	0.752

The YOLOv8 model incorporates several components (box_loss, kobj_loss, cls_loss, dfl_loss, and pose_loss) in its loss function, each serving a specific purpose. Figure 10 illustrates the curves depicting the changes in pose_loss over the course of training steps. Analysis of these curves reveals consistent decreases in box_loss, kobj_loss, cls_loss, and dfl_loss as the number of iterations increases, stabilizing at certain values. However, the pose_loss curve exhibits continual fluctuations, indicating that the prediction of keypoints is not entirely satisfactory. This observation suggests a need for further refinement to enhance model performance.

Fig. 10. Pose_loss figure in training set

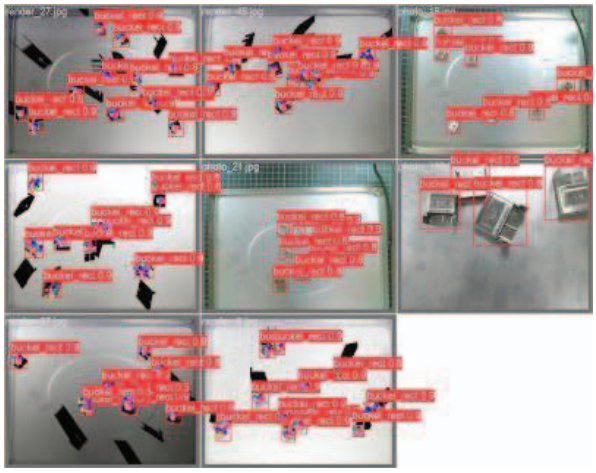

Fig. 11. Predict picture in validation set

Figure 11 displays predicted images from the validation set, while Figure 12 showcases real-time buckle detection in a video. The images from the validation set indicate that the trained model accurately predicts the positions of both large and small buckles, even when they exhibit strong reflections or partial overlaps. However, in synthetic images, although the bounding box positions are highly accurate, the distribution of predicted keypoints around the buckles is not satisfactory, indicating a lack of accuracy in keypoint prediction. Upon analyzing real-time videos using the final trained model, it was observed that the model can predict buckle positions in real-time. However, the prediction of keypoints on the buckles is inaccurate and has not converged well.

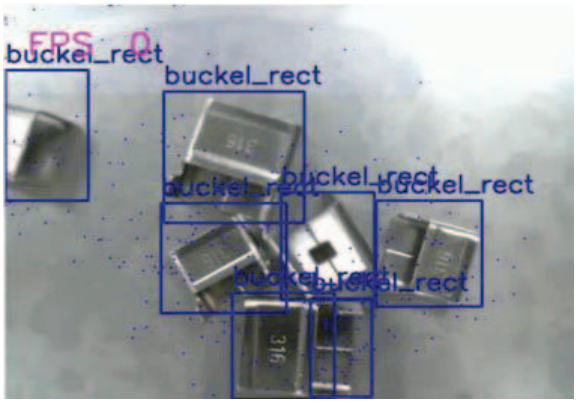

Fig. 12. Real-time buckle video detection

V. Discussion and Conclusion

This robotic system, coupled with computer vision design, offers significant benefits to on-site users by enhancing work quality and crucially addressing the safety risk of workers falling from heights. The shiny color of the buckle necessitates thorough testing for detection, posing a challenge to the capabilities of computer vision. However, our utilization of the latest YOLOv8 architecture, boasting a precision of 0.995, ensures effective buckle detection.

Our model demonstrates proficiency in identifying buckles with specular highlights, overlapping configurations, and low light conditions, thereby proving valuable for the recognition and positioning tasks of metallic industrial parts. Nonetheless, inaccuracies in keypoint prediction present challenges in accurately estimating the pose of metal parts, necessitating further research.

By adopting a training approach that combines real and synthetic data, we not only mitigate overfitting issues during neural network training but also augment the dataset, effectively easing the burden of data labeling. This approach represents a promising direction for the development of future intelligent industries.

Ongoing research and advancements in computer vision, sensor technologies, and machine learning algorithms continuously improve vision-based metal detection systems. The dynamic nature of the field ensures that new techniques are regularly introduced to address emerging challenges, driving innovation and enhancing system capabilities.

Acknowledgment

This research represents a collaborative effort between the National University of Singapore (NUS) and Sembcorp Marine Ltd. (SCM). The second author was a visiting researcher at NUS, sponsored by the China Scholarship Council (CSC) when this work was conducted. He is affiliated with the East China Electronic Technology Research Institute.

References

[1] Z. Huang, H. Xiao, R. Zhang, H. Wang, C. Zhang, and X. Shi, "Multi-scale feature pair based R-CNN method for defect detection," in Proc. 2019 International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Jul. 2019, July, pp. 46-51.

[2] S. Ragi, M.H. Rahman, J. Duckworth, J. Kalimuthu, P. Chundi, and V. Gadhamshetty, "Artificial intelligence-driven image analysis of bacterial cells and biofilms.," IEEE/ACM Transactions on Computational Biology and Bioinformatics, Dec. 2021.

[3] C. Sun, C. Jiang, Y. Che, W.Y. Chen, Y. Zhang, A.M. Jokisaari, L.K. Aagesen, L. Shao, and J. Gan, "Unveiling the interaction of nanopatterned void superlattices with irradiation cascades," Acta Materialia, vol. 239, p.118282, Oct. 2022.

[4] A. Singh, V. Kalaichelvi, and R. Karthikeyan, "Performance analysis of object detection algorithms for robotic welding applications in planar environment," International Journal of Computer Integrated Manufacturing, pp.1-26, Jan. 2023.

[5] Y. Xu, K. Zhang, and L. Wang, "Metal surface defect detection using modified YOLO," Algorithms, vol. 14, no. 9, p. 257, Aug. 2021.

[6] Y. Cao, B. Ding, J. Chen, W. Liu, P. Guo, L. Huang, and J. Yang, "Photometric-stereo-based Defect Detection System for Metal Parts," Sensors, vol. 22, no. 21, p. 8374, Nov. 2022.

[7] L. Yang, Y. Liu, and J. Peng, " An Automatic Detection and Identification Method of Welded Joints based on Deep Neural Network ," IEEE Access, vol. 7, pp. 164952-164961, Nov. 2019.

[8] P. Zhou, G. Zhou, S. Wang, H. Wang, Z. He, and X. Yan, "Visual Sensing Inspection for the Surface Damage of Steel Wire Ropes with Object Detection Method," *IEEE Sensors Journal*, vol. 22, no. 23, pp.22985-22993, Oct.2022.

[9] Y. He, B. Wu, J. Mao, W. Jiang, J. Fu, and S. Hu, "An effective MID-based visual defect detection method for specular car body surface," *Journal of Manufacturing Systems*, vol. 72, pp. 154-162, Feb. 2024.

[10] A.R. Singh, T. Bashford-Rogers, D. Marnerides, K. Debattista, and S. Hazra, "HDR Image-based Deep Learning Approach for Automatic Detection of Split Defects on Sheet Metal Stamping Parts," *The International Journal of Advanced Manufacturing Technology*, vol. 125, no. 5-6, pp.2393-2408 Mar. 2023.

[11] C. Chen, X. Jiang, S. Miao, W. Zhou, and Y. Liu, "Texture-Less Shiny Objects Grasping in a Single RGB Image Using Synthetic Training Data," Applied Sciences, vol. 12, no. 12, p.6188, Jun. 2022.

[12] X. Kou, S. Liu, K. Cheng, and Y. Qian, "Development of a YOLO-V3-based model for detecting defects on steel strip surface," Measurement, vol. 182, p. 109454. Sep. 2021.

[13] J. Li, Z. Su, J. Geng, and Y. Yin, "Real-time detection of steel strip surface defects based on improved yolo detection network," IFAC-PapersOnLine, vol. 51, no. 21, pp. 76-81, Jan. 2018.

[14] X. Yue, J. Chen, and G. Zhong, "Metal Surface Defect Detection Based on Metal-YOLOX," International Journal of Network Dynamics and Intelligence, pp. 100020-100020, Dec. 2023.

[15] X. Zhu, J. Liu, X. Zhou, S. Qian, and J. Yu, "Enhanced feature Fusion structure of YOLO v5 for detecting small defects on metal surfaces," International Journal of Machine Learning and Cybernetics, vol. 14, no. 6, pp. 2041-2051, Jun. 2023.

[16] H. Zhao, Z. Yang, and J. Li, "Detection of metal surface defects based on YOLOv4 algorithm," In Journal of Physics: Conference Series, vol. 1907, no. 1, p. 012043.

[17] L. Posilović, D. Medak, M. Subašić, T. Petković, M. Budimir, and S. Lončarić, "Flaw detection from ultrasonic images using YOLO and SSD," In Proc 2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA), Sep. 2019, pp. 163-168.

[18] Steel banding tool [Online]. Available: https://traditionaltool.com/blog/top-5-steel-strapping-tools/

[19] J. Lecerof, and T. Opheim, "Instance segmentation and pose estimation of assembly parts using deep learning and synthetic data generation," M.S. Thesis, Department of Electrical Engineering, Chalmers Univ. of Technology, Gothenburg, Sweden, 2019. [Online]. Available: https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/2627024