



UNIVERSITÀ POLITECNICA DELLE MARCHE
Repository ISTITUZIONALE

A Comparative Analysis on a Limited Image Dataset for Accurately Detecting Improperly Polished Surfaces for Industrial Applications

This is the peer reviewed version of the following article:

Original

A Comparative Analysis on a Limited Image Dataset for Accurately Detecting Improperly Polished Surfaces for Industrial Applications / Bajrami, A.; Palpacelli, M. C.; Lettera, G.; Pantanetti, S.. - ELETTRONICO. - (2024). (20th IEEE/ASME International Conference on Mechatronic, Embedded Systems and Applications, MESA 2024 Genova, Italy 2 -4 September 2024) [10.1109/MESA61532.2024.10704894].

Availability:

This version is available at: 11566/341193 since: 2025-02-25T15:00:09Z

Publisher:

Institute of Electrical and Electronics Engineers Inc.

Published

DOI:10.1109/MESA61532.2024.10704894

Terms of use:

The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. The use of copyrighted works requires the consent of the rights' holder (author or publisher). Works made available under a Creative Commons license or a Publisher's custom-made license can be used according to the terms and conditions contained therein. See editor's website for further information and terms and conditions.

This item was downloaded from IRIS Università Politecnica delle Marche (<https://iris.univpm.it>). When citing, please refer to the published version.

Publisher copyright:

IEEE - Postprint/Author's Accepted Manuscript

©2024 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. To access the final edited and published work see 10.1109/MESA61532.2024.10704894

(Article begins on next page)

A Comparative Analysis on a Limited Image Dataset for Accurately Detecting Improperly Polished Surfaces for Industrial Applications

1st Albin Bajrami

Dept. of Industrial Engineering and Mathematical Sciences
Polytechnic University of Marche
Ancona, Italy
<https://orcid.org/0009-0002-3207-4867>

2nd Matteo Claudio Palpacelli

Dept. of Industrial Engineering and Mathematical Sciences
Polytechnic University of Marche
Ancona, Italy
<https://orcid.org/0000-0001-6931-6618>

3st Gaetano Lettera

Dept. of Industrial Engineering and Mathematical Sciences
Polytechnic University of Marche
Ancona, Italy
<https://orcid.org/0000-0002-6650-1786>

4th Simone Pantanetti

Dept. of Industrial Engineering and Mathematical Sciences
Polytechnic University of Marche
Ancona, Italy
<https://orcid.org/0009-0000-9119-6210>

Abstract—The objective of this study is to identify the optimal object detection architecture for training on a specific type of defect detection, namely incorrectly polished surfaces on aluminium elements. In order to facilitate a meaningful comparison of the various architectures, a maximum training time of approximately one hour was established for each architecture. Using the Darknet framework and a specific dataset, five architectures were compared (for the time being). The parameters of the various architectures, including network size, number of batches, and so forth, were modified according to a well-defined and systematic procedure. The preliminary findings indicate that the YOLOv4-tiny network exhibits superior training performance on this dataset, rendering it an optimal choice for industrial applications. This research provides support to small and medium-sized enterprises (SMEs) by identifying effective object detection architectures for quality control and highlighting avenues for advancing AI-driven defect detection in manufacturing.

Index Terms—AI, Manufacturing, Comparative Analysis, YOLO, Darknet, Object Detection

I. INTRODUCTION

Surface finishing is an important process in industrial manufacturing as it enhances mechanical properties, such as stress and fatigue resistance, and aesthetic properties, such as the polish and feel of products [1], [2]. The emergence of 3D printing technology and the resulting ability to produce products with complex but cheaper geometries have presented new challenges for robotic surface finishing. The advent of new technologies has opened up new opportunities for SMEs (Small Medium Enterprises) to have more flexible and customised manufacturing, in line with user-centric design principles [3]–[6]. However, the limited resources, including personnel, specific skills, lack of managerial leadership, and the small scale of production, represent significant obstacles to the adoption of such technologies [7]–[11].

Programming industrial robots can be a time-consuming task and can also raise safety concerns [12]. Collaborative industrial robots, also known as cobots, offer a solution for various industrial applications due to their simpler interface and advanced sensor technology that ensures greater safety [13]–[18]. However, their use in surface finishing, such as polishing, remains limited due to the complexity of the task and the need for human dexterity. In addition, polishing involves risks to operators due to the materials used (generation of dust harmful to humans by inhalation or skin contact) and the musculoskeletal effort required for prolonged and repetitive tasks.

The integration of Artificial Intelligence (AI) with robotics has the potential to significantly enhance efficiency and safety, even in polishing tasks where AI control and decision-making are crucial [19]. Through computer vision, AI can evaluate the quality of polishing in a similar manner to a human operator, optimizing the process until satisfactory results are achieved [20]. However, the adoption of this technology, particularly among SMEs, generally is hindered by the limited availability of useful images and the extensive training time required for AI from zero.

This work is part of the project described in [21], [25], which aims to develop a framework for robotic surface finishing of components produced by 3D printing. To achieve this, it is necessary to integrate AI for vision to recognize surface defects for both quality control and task execution in case the processed piece does not meet the expected standards. To do this in compliance with industrial constraints such as, low computational performance, high accuracy to detection, fast training, and small/micro image dataset, is necessary to compare the effectiveness of different network architectures in training on a restricted dataset for robotic surface finishing.

The primary objective is to find the fastest and most efficient network for a particular dataset, described in section II-A. The decision to use a restricted dataset is driven by the operational constraints typical of SMEs, which often face limited computational resources and small data volumes.

Given these constraints, future research could benefit from exploring Tensor Processing Units (TPUs). TPUs are application-specific integrated circuits developed by Google to accelerate machine learning workloads. Their high performance and energy efficiency make them particularly suitable for industrial applications where computational resources are limited. Evaluating models optimized for TPUs could provide valuable insights into reducing training time and improving detection accuracy, making advanced AI techniques more accessible and cost-effective for SMEs.

II. METHODOLOGY

To disseminate the knowledge acquired during this research, a detailed methodology is presented below. First, is described the dataset used; subsequently is described the framework used to train the deep neural networks (DNNs). This is followed by a description of the systematic approach used in this work and a brief description of the networks used.

A. Dataset

The SPADD Dataset utilized in this work, which is available on GitHub [23], aims to assist in the development of AI models that can identify inadequately polished areas on metal surfaces. Some images contained in this dataset are reported in Fig.1. At an early stage, it provides labelled data to refine the quality of industrial finishing using machine vision algorithms. The dataset is expected to expand in size and diversity.

B. Framework Darknet

Darknet [24] is an open source framework focused on artificial intelligence and visual recognition using neural networks. A recent *fork* of this framework [25], which is easier to configure and suitable for resource-constrained projects, makes it ideal for small and medium-sized enterprises (SMEs). It can be trained on custom datasets and is compatible with CUDA (Compute Unified Device Architecture) for GPU acceleration, making it ideal for real-time applications such as surveillance and industrial applications. The Darknet framework is designed for use with YOLO (You Only Look Once), but it can also be used with other architectures.

C. Network architecture and Procedure

The employed architectures are those already available within the Darknet framework. For this initial phase, only five architectures were utilized and are detailed in Table I.

The main objective is to determine which network achieves the highest performance within a given training time. The time range selected for training the network is between 40 and 70 minutes. This duration is generally considered short, but the purpose of this paper is to evaluate which duration is most effective for training with a limited dataset.

The training procedure follows the schema proposed in the diagram shown in Fig. 2. The procedure starts with selecting a network. To accommodate the two main requirements, i.e., training time between 40/70 minutes and the PC's computing capabilities (12GB of RAM for GPU), it is necessary to modify the maximum batch size, batch size, and subdivision. Default networks have different sizes due to the number of layers and the size of the input images.

D. Workstation Specifications

In this study it was used a workstation that was configured for intensive neural network training tasks. The environment runs on an Ubuntu 22.04.3 LTS operating system and is equipped with the NVIDIA GeForce RTX 3060 GPU, providing high performance that is essential for processing the parallel computational tasks required in network training. The Ampere architecture of the GPU and its support for CUDA version 12.2 significantly accelerate computation. The table in reference II summarises the specifications of this workstation.

E. Data Augmentation

To enhance the robustness and generalization capability of the neural networks, various data augmentation techniques were employed. These techniques help simulate a diverse range of scenarios that the model might encounter in real-world applications. The following augmentations were applied to the training dataset:

- **Color Adjustments:**

- **Saturation:** Variations in color saturation were introduced to ensure color consistency is not a factor in model performance.
- **Exposure:** Exposure levels were randomly altered to simulate different lighting conditions the model might face.
- **Hue:** The hue of images was shifted randomly to enhance the model's tolerance to color variations.

- **Miscellaneous Techniques:**

- **Flip:** Both horizontal and vertical flips were utilized to augment the dataset and prevent directional biases.
- **Mosaic:** Multiple images were combined into a single training example to help the model handle complex scenes.
- **Mixup:** Images and their labels were blended to encourage the model to focus on multiple features simultaneously.
- **Random:** Random transformations, including slight rotations and shifts, were applied to emulate real-world unpredictability.

These augmentations were systematically applied during the training process, ensuring that each epoch received a varied set of inputs. This approach helps in creating a more resilient model capable of accurately detecting defects in a wide array of conditions.

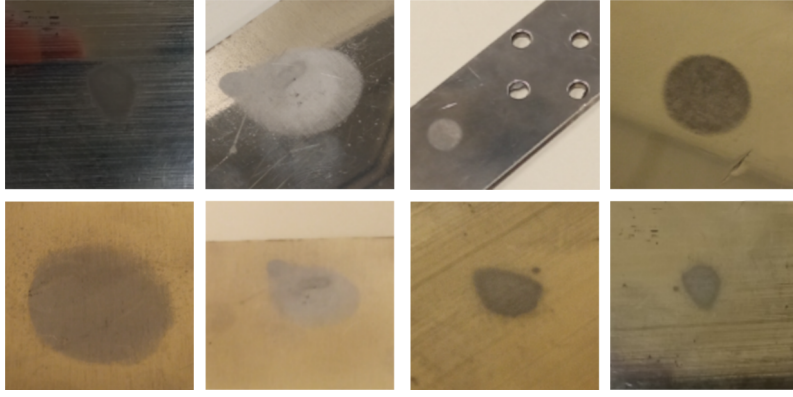


Fig. 1. Examples of images from the SPADD repository showing defects in unpolished aluminum.

TABLE I
SUMMARY DESCRIPTION OF THE NETWORKS USED IN THIS PRELIMINARY WORK

Network Name	Number of Layers	Size	Weight [MB]	Notes
YOLOv4	162	512x512	256	3 YOLO layers
YOLOv4-tiny	38	416x416	23.5	2 YOLO layers
YOLOv7	143	640x640	146.2	3 YOLO layers
YOLOv7-tiny	99	416x416	24.1	3 YOLO layers
cd53paspp-gamma	180	512x512	237	3 YOLO layers. Sperimental

TABLE II
THE TABLE LISTS THE SYSTEM CONFIGURATIONS OF THE WORKSTATION

Operating System	Ubuntu 22.04.3 LTS
Kernel Version	6.2.0-37-generic
GPU Model	NVIDIA GeForce RTX 3060 Lite Hash Rate
GPU Architecture	Ampere
GPU Driver Version	470.223.02
CUDA Version	12.2
CUDA Cores Number	3584

III. PERFORMANCE CRITERIA

The effectiveness of the defect detection models was evaluated using metrics provided by the Darknet framework. These metrics offer a comprehensive view of a model's performance. Before discussing the specific metrics, it is important to define the key terms used in their calculation.

- **TP (True Positives):** The number of positive cases correctly identified by the model as positive.
- **FP (False Positives):** The number of negative cases incorrectly identified by the model as positive.
- **TN (True Negatives):** The number of negative cases correctly identified by the model as negative.
- **FN (False Negatives):** The number of positive cases incorrectly identified by the model as negative.

Below are the definitions of the metrics used:

Precision: The ratio of true positives to the total number of positives identified by the model. It is calculated using the formula:

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

Recall: The ratio of true positives correctly identified to the total number of actual positive cases. It is expressed with:

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

Accuracy: The ratio of correct predictions (both positive and negative) to the total number of observations. It is determined through the formula:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Error Rate: The proportion of all incorrect classifications over the total number of observations. It is calculated as:

$$Error\ Rate = \frac{FP + FN}{TP + TN + FP + FN}$$

Specificity: The proportion of true negatives to the total number of actual negative cases. The formula is:

$$Specificity = \frac{TN}{TN + FP}$$

These metrics are essential for reliably evaluating the effectiveness of models in correctly identifying defect areas in polished components.

mAP (Mean Average Precision): A comprehensive metric for evaluating object detection quality, averaging the precision across different recall levels. The Average Precision (AP) for a single class is calculated by plotting the precision-recall curve and computing the area under this curve. mAP is then the mean of AP values calculated for all classes or over all recall levels. The formula involves integrating the precision with respect to recall:

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i$$

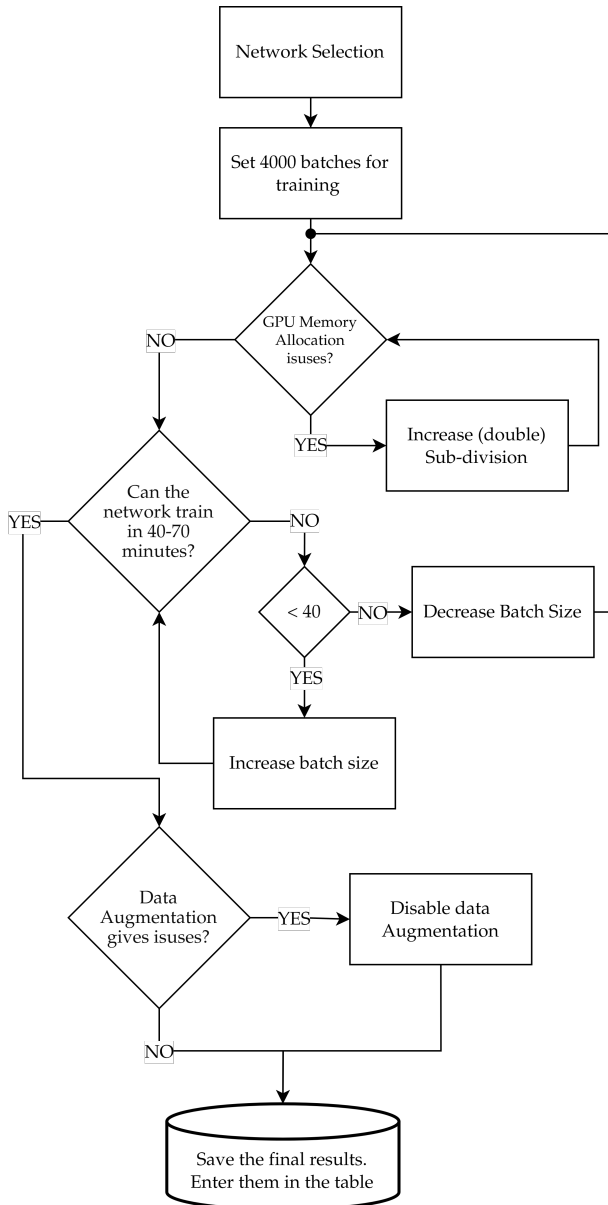


Fig. 2. Flowchart describing the procedure used in this training process.

where AP_i is the Average Precision for class i , and N is the number of classes.

IV. RESULTS

The results of this preliminary study are summarised in table III, which shows the performance of each of the architectures analysed.

In an evaluation of five object recognition models, two models, *cd53paspp-gamma* and *yolov7-tiny*, showed remarkably poor performance. Neither model was able to correctly identify any target objects, as evidenced by zero true positives (TP) and mean average precision (mAP@0.50) values close to 0%. This indicates a complete failure to identify the intended targets, despite achieving perfect specificity scores, which

merely reflect the absence of false positives rather than actual identification success.

On the other hand, *yolov4-tiny* excelled, achieving the highest mAP of 99.23%. This model correctly identified almost all target objects, with 138 true positives and only minor inaccuracies, including 2 false positives and 3 misses. Its high precision (0.99) and recall (0.98) underline its reliability and accuracy, making it particularly suitable for tasks requiring precise object recognition.

The *yolov4* model showed moderate effectiveness with a mAP of 43.49%, positioning it well above *cd53paspp-gamma* and *yolov7-tiny* but below *yolov4-tiny*. It showed a reasonable balance with a precision of 0.63 and a recall of 0.43, suggesting a moderate ability to identify true positives, but with some errors and misses.

The *yolo7* had a significantly low mAP of 18.29%, indicating limited effectiveness. Despite its high specificity (1.00) and precision (0.84), the model's extremely low recall (0.10) highlights a significant shortfall in the detection of true positives, which severely limits its practical utility in real-world applications.

yolov4-p6 and *yolov4-p5* models were similarly ineffective, each failing to record any true positives and demonstrating extremely low mAP values of 0.01% and 0.00% respectively. Their inability to identify target objects renders them unsuitable for practical deployment without substantial improvements.

The *yolov3-VOC* and *yolov4-New* models also performed poorly, with neither model achieving any true positives and both displaying a mAP of 0.00%. The negative values in the 'TN' column for *yolov3-VOC* and *yolov4-New* suggest an error in data collection or processing, further complicating any assessment of their utility based on this study.

Overall, *yolov4-tiny* demonstrated the best performance of the models tested, establishing itself as the most reliable option for high accuracy requirements. Conversely, models such as *cd53paspp-gamma* and *yolov7-tiny* may require significant modification or retraining to meet the demands of operational environments.

Fig. 3 shows some examples of defect detection. The detections are based on *yolov4-tiny*, which showed the best performance over the other networks.

V. DISCUSSION AND CONCLUSION

This study compares five YOLO architectures for object detection of inadequately polished areas on different aluminium samples, using images from the SPADD dataset available online refa21. The results show that the YOLOv4-Tiny network performs the best given training constraints between 40 and 70 minutes with the hardware described in table II. This suggests that, in business contexts, the use of this network may be the most appropriate. However, as this study is preliminary in nature, we plan to extend the analysis to a larger number of CNN networks for the full version of this article. This will provide a broader and more detailed comparison.

TABLE III
TRAINING RESULTS OF DIFFERENT RECOGNITION MODELS.

Models	Duration	mAP@0.50 (%)	TP	FN	FP	TN	Precision	Recall	Specificity	F1-score
cd53paspp-gamma	49 min	0.00	0	149	0	N/A	0.00	0.00	1.00	N/A
yolov7-tiny	58 min	0.03	0	166	0	197	0.00	0.00	1.00	N/A
yolov4-tiny	45 min	99.23	138	3	2	289	0.99	0.98	0.99	0.98
yolov4	66 min	43.49	64	85	37	4134	0.63	0.43	0.99	0.51
yolo7	56 min	18.29	16	141	3	3936	0.84	0.10	1.00	0.18
yolov7x	58 min	4.73	0	152	0	701	0.00	0.00	1.00	N/A
yolov4-p6	52 min	0.01	0	55	0	981	0.00	0.00	1.00	N/A
yolov4-p5	60 min	0.00	0	167	0	139	0.00	0.00	1.00	N/A
yolov3-VOC	42 min	0.00	0	55	0	-55	0.00	0.00	1.00	N/A
yolov4-New	66 min	1.82	0	55	0	-11	0.00	0.00	1.00	N/A

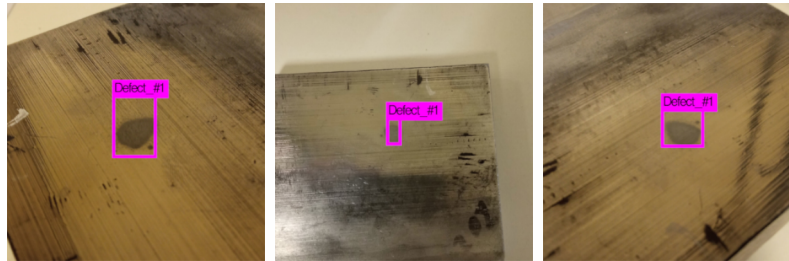


Fig. 3. Detected defects in unpolished aluminum samples from the SPADD dataset.

The positive outcomes of YOLOv4-Tiny offer a strong foundation for additional research. It is suggested that forthcoming studies explore the adjustment and customization of neural networks for particular industrial applications and broaden the scope and variety of the dataset to enhance dependability and efficiency in defect detection.

This work establishes the basis for technological advancement in the area of industrial surface finishing, with direct implications for enhancing quality and production efficiency. Future prospects involve assessing additional neural networks and creating a specialized surface finishing framework to tackle and overcome current industry obstacles.

Future research will explore the integration of Tensor Processing Units (TPUs) into the defect detection framework to potentially speed up the training processes and execute models with high efficiency on edge devices. This offers a practical solution for SMEs facing computational and cost constraints. Specifically, the Coral Dev Board will be utilized in these studies. Comparative studies will be conducted to evaluate the performance of TPU-optimized models against current architectures, with the aim of enhancing the feasibility and scalability of AI-driven defect detection in industrial applications.

REFERENCES

- [1] Bagehorn, S., J. Wehr, and H. J. Maier. "Application of Mechanical Surface Finishing Processes for Roughness Reduction and Fatigue Improvement of Additively Manufactured Ti-6Al-4V Parts." *International Journal of Fatigue*, vol. 102, 2017, pp. 135-142, doi: 10.1016/j.ijfatigue.2017.05.008.
- [2] Lai, M. O., and Sui Fen Koh. "Fatigue Performance of Holes with Different Finishing Processes." *Journal of Materials Processing Technology*, vol. 44, no. 1-2, July 1994, pp. 118-28, doi: 10.1016/0924-0136(94)90043-4.
- [3] Ivanov, D., A. Das, and T. Choi. "New Flexibility Drivers for Manufacturing, Supply Chain and Service Operations." *International Journal of Production Research*, vol. 56, 2018, pp. 3359-3368, doi: 10.1080/00207543.2018.1457813.
- [4] Prabhaker, Paul R., Joel D. Goldhar, and David Lei. "Marketing Implications of Newer Manufacturing Technologies." *Journal of Business & Industrial Marketing*, vol. 10, no. 2, 1995, pp. 48-58, doi: 10.1108/08858629510087373.
- [5] Deniša, M., et al. "Technology Modules Providing Solutions for Agile Manufacturing." *Machines*, vol. 11, 2023, article 877, doi: 10.3390/machines11090877.
- [6] Zhao, Huijiang, et al. "Customization-Oriented Product Flexible Manufacturing Experience System Design Based on VR." *IOP Conference Series: Materials Science and Engineering*, vol. 561, no. 1, 2019, doi: 10.1088/1757-899X/561/1/012098.
- [7] Raj, Alok, et al. "Barriers to the Adoption of Industry 4.0 Technologies in the Manufacturing Sector: An Inter-Country Comparative Perspective." *International Journal of Production Economics*, vol. 224, 2020, article 107546, doi: 10.1016/j.ijpe.2019.107546.
- [8] Baldwin, John, and Zhengxi Lin. "Impediments to Advanced Technology Adoption for Canadian Manufacturers." *Research Policy*, vol. 31, no. 1, 2002, pp. 1-18, doi: 10.1016/S0048-7333(01)00110-X.
- [9] Jun, Minjoon, et al. "Key Obstacles to EDI Success: From the US Small Manufacturing Companies' Perspective." *Industrial Management & Data Systems*, vol. 103, 2003, pp. 192-203, doi: 10.1108/02635570310465670.
- [10] Julien, P., et al. "New Technologies and Technological Information in Small Businesses." *Journal of Business Venturing*, vol. 10, 1995, pp. 459-475, doi: 10.1016/0883-9026(95)00084-L.
- [11] A. Ghobadian et al., "New process technologies: barriers to adoption in small firms," *International Journal of Manufacturing Technology and Management*, vol. 5, 2003, pp. 549-562, doi: 10.1504/IJMTM.2003.003709.
- [12] Gregory F. Rossano et al., "Easy robot programming concepts: An industrial perspective," 2013 IEEE International Conference on Automation Science and Engineering (CASE), 2013, pp. 1119-1126, doi: 10.1109/CoASE.2013.6654035.
- [13] Carole S Franklin et al., "Collaborative robotics: New era of human-robot cooperation in the workplace," *Journal of Safety Research*, vol. 74, 2020, pp. 153-160, doi: 10.1016/j.jsr.2020.06.013.
- [14] Luca Gualtieri et al., "Design of Human-Centered Collaborative Assembly Workstations for the Improvement of Operators' Physical Er-

- gonomics and Production Efficiency: A Case Study,” *Sustainability*, vol. 12, 2020, 3606, doi: 10.3390/su12093606.
- [15] Bajrami, Albin, Daniele Costa, Matteo Claudio Palpacelli, and Federico Emiliani. 2024. ”Investigating Collaborative Robotic Assembly: A Case Study of the FANUC CRX-10 iA/L in Industrial Automation at i-Labs” *Eng* 5, no. 2: 532-543. <https://doi.org/10.3390/eng5020029>
- [16] I. Maurtua et al., “Human–robot collaboration in industrial applications,” *International Journal of Advanced Robotic Systems*, vol. 14, 2017, doi: 10.1177/1729881417716010.
- [17] R. Galin et al., “Automation and robotics in the context of Industry 4.0: the shift to collaborative robots,” *IOP Conference Series: Materials Science and Engineering*, vol. 537, 2019, doi: 10.1088/1757-899X/537/3/032073.
- [18] M. Matúšová et al., “The future of industry with collaborative robots,” *MATEC Web of Conferences*, 2019, doi: 10.1051/matec-conf/201929902008.
- [19] T. Segreto et al., “Machine learning for in-process end-point detection in robot-assisted polishing using multiple sensor monitoring,” *The International Journal of Advanced Manufacturing Technology*, 2019, pp. 1-15, doi: 10.1007/S00170-019-03851-7.
- [20] A. R. A. Besari, A. S. Prabuwno, R. Zamri, and M. D. M. Palil, “Computer vision approach for robotic polishing application using artificial neural networks,” *2010 IEEE Student Conference on Research and Development (SCORED)*, Kuala Lumpur, Malaysia, 2010, pp. 281-286, doi: 10.1109/SCORED.2010.5704017.
- [21] A. Bajrami, and M. C. Palpacelli, “A Flexible Framework for Robotic Post-Processing of 3D Printed Components,” *Volume 7: 19th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA)*, 20 Aug. 2023, doi: 10.1115/detc2023-109746.
- [22] Bajrami A, Palpacelli MC. From Dataset Creation to Defect Detection: A Proposed Procedure for a Custom CNN Approach for Polishing Applications on Low-Performance PCs. *Machines*. 2024; 12(7):453. <https://doi.org/10.3390/machines12070453>
- [23] SPADD-Dataset/Yolov3_custom.gif at main · AlbinEV/SPADD-Dataset, GitHub, https://github.com/AlbinEV/SPADD-Dataset/blob/main/Yolov3_custom.gif.
- [24] J. Redmon, “Darknet: Open source neural networks in C,” <https://pjreddie.com/darknet/>.
- [25] Hank-Ai, GitHub - Hank-ai/Darknet: Darknet/YOLO Object Detection Framework, github.com/hank-ai/darknet.