

Article

Capacity Constraint Analysis Using Object Detection for Smart Manufacturing

Hafiz Mughees Ahmad ¹, Afshin Rahimi ^{1,*} and Khizer Hayat ²

¹ Department of Mechanical, Automotive and Materials Engineering, University of Windsor, Windsor, ON N9B 3P4, Canada; ahmad54@uwindsor.ca

² IFIVEO CANADA INC., Windsor, ON N9B 3P4, Canada; khizer@ifiveo.com

* Correspondence: arahimi@uwindsor.ca

Abstract: The increasing adoption of Deep Learning (DL)-based Object Detection (OD) models in smart manufacturing has opened up new avenues for optimizing production processes. Traditional industries facing capacity constraints require noninvasive methods for in-depth operations analysis to optimize processes and increase revenue. In this study, we propose a novel framework for capacity constraint analysis that identifies bottlenecks in production facilities and conducts cycle time studies using an end-to-end pipeline. This pipeline employs a Convolutional Neural Network (CNN)-based OD model to accurately identify potential objects on the production floor, followed by a CNN-based tracker to monitor their lifecycle in each workstation. The extracted metadata are further processed through the proposed framework. Our analysis of a real-world manufacturing facility over six months revealed that the bottleneck station operated at only 73.1% productivity, falling to less than 40% on certain days; additionally, the processing time of each item increased by 53% during certain weeks due to critical labor and materials shortages. These findings highlight significant opportunities for process optimization and efficiency improvements. The proposed pipeline can be extended to other production facilities where manual labor is used to assemble parts, and can be used to analyze and manage labor and materials over time as well as to conduct audits and improve overall yields, potentially transforming capacity management in smart manufacturing environments.

Keywords: convolutional neural network; you only look once; YOLO; object detection; deep learning; smart manufacturing



Citation: Ahmad, H.M.; Rahimi, A.; Hayat, K. Capacity Constraint Analysis Using Object Detection for Smart Manufacturing. *Automation* **2024**, *5*, 545–563. <https://doi.org/10.3390/automation5040031>

Academic Editor: Duc Truong Pham

Received: 19 July 2024

Revised: 12 October 2024

Accepted: 24 October 2024

Published: 29 October 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The manufacturing sector has long been a cornerstone of economic development, both driving innovation and providing employment opportunities. According to the World Bank [1], China holds 28.4% of the market share of global manufacturing, with a contribution of nearly USD 4 trillion to the global economy. The United States of America is the second-largest contributor at approximately 16.6%, contributing USD 1.8 trillion. However, the outbreak of the COVID-19 pandemic has exacerbated existing challenges in the manufacturing landscape, notably the critical issue of labor and supply chain shortages, collectively referred to as capacity constraint. Industries have encountered difficulty in finding skilled labor for tasks requiring precise human effort. According to Causa et al. [2], 75% of employers have had a hard time filling open positions, and manufacturing is among the most hard-hit sectors. Additionally, as per the Analysis on labor challenges in Canada, second quarter of 2023 by Statistics Canada in June 2023 [3], 59.3% of manufacturing industries consider rising inflation to be an obstacle over the next three months. Nearly 9 out of 10 organizations surveyed responded that they are having a hard time filling open positions, with most of these comprised of entry-level or mid-level positions. Manufacturing is the most hard-hit, with 93% of organizations struggling to find entry-level employees. The Canadian Federation of Independent Business also reported [4] that as of November 2021,

55% of small businesses in Canada were experiencing labor shortage and difficulty in hiring or retaining staff or in getting staff to work the needed hours. Overall, this shortage is exponentially increasing the existing global supply chain issues.

This confluence of data from various sources underscores a consistent and pressing issue of capacity constraint that requires innovative solutions. Researchers and engineers trying to solve this unique challenge have started to use Artificial Intelligence (AI)- and Computer Vision (CV)-based methods to increase labor productivity and decrease bottlenecks in their production pipelines in order to reduce the impact of supply chain shortages in smart manufacturing [5–8]. Based on an earlier study by Ahmad and Rahimi [9], applications of OD in smart manufacturing play a pivotal role in enhancing quality control, cycle times, safety compliance, and surveillance. Puttemans et al. [10] and Wang et al. [11] employed the DL-based OD model [12] for detecting packages in warehouse environments, highlighting its utility in real-time applications for product packaging. Farahnakian et al. [13] and Li et al. [14] both applied OD models for damage detection and pallet rack identification in industrial warehouse settings.

Despite these advancements in automating manufacturing processes, a significant gap exists in applying OD-based techniques for capacity constraint analysis of individual workstations in production pipelines when production capabilities fall short demand due to shortages in labor, materials, or equipment. The current article addresses this gap by proposing a pipeline for analyzing capacity constraint using a You Only Look Once (YOLO)v8-based model [15] for productivity analysis of each station and identifying bottlenecks in the production pipeline, as even one bottleneck station in a serial pipeline can drastically reduce overall productivity. Our approach begins with the development and training of an OD model meticulously designed to identify objects on the production floor. Subsequently, a CNN-based tracking system is employed to track the lifecycle of these objects. The extracted metadata are then processed in order to provide insights about the productivity of each station in the manufacturing facility. The overall approach provides real-time capacity metrics, and can be further utilized for capacity constraint analysis of both individual workstations and complete production pipelines.

Our key contributions with this study can be summarized as follows:

- A novel non-invasive theoretical framework for analyzing the capacity of manufacturing facilities by using OD methods to categorize workstations into different states.
- Collection and annotation of a real-world dataset from a production floor for use in training an OD model.
- Comprehensive experimentation and evaluation of the proposed framework in a real-world facility over 6 months, demonstrating its practical applicability and effectiveness.

This study involved a collaborative research initiative consisting IFIVEO CANADA INC., a computer vision company, and its client (hereinafter referred to as the client), which specializes in producing assistive medical wheelchairs. The manual assembly of power-assist wheelchairs (henceforth referred to as chairs) is a niche yet crucial manufacturing segment presenting unique challenges.

Our findings revealed significant opportunities for optimizing manual production processes and addressing capacity constraints, with potential implications for improving overall manufacturing efficiency.

In the following sections, we first delve into the state-of-the-art OD methods and their applications in smart manufacturing in Section 2. Next, we propose a theoretical framework in Section 3, while the technical intricacies of our proposed solution are supported by comprehensive analysis and empirical evidence in Section 4. Insights from the manufacturing facility are discussed in Section 5; finally, Section 6 offers concluding remarks.

2. Related Work

CNN-based OD methods and their applications in real-world value-added services are an active area of research, as these models are pivotal in the localization and classification of objects within a given frame.

Historically, traditional approaches relied on carefully engineered hand-crafted features, which was time-consuming and led to less accurate results. However, the advent of CNN-based deep learning models empowered by increased computational capacity through Graphical Processing Unit (GPU) technology has revolutionized computer vision.

Two primary categories of OD methods have emerged, namely, region proposal-based and regression-based methods. A Region Proposal Convolutional Neural Network (R-CNN) [16] starts by proposing regions, which are subsequently classified into predefined categories [17,18]. While these models demonstrate high localization accuracy, they are computationally complex, often falling short of real-time performance due to the need to propose of thousands of regions per image. To address this limitation, one-stage detectors were introduced [19], including the groundbreaking You Only Look Once (YOLO) by Redmon et al. [20], providing real-time performance across various benchmarks. The following subsections discuss the evolution of YOLO models and their applications in smart manufacturing.

2.1. Evolution of YOLO Models

YOLO [20] represents a revolutionary approach to OD and localization by treating the problem as a regression task. In essence, YOLO directly proposes bounding box coordinates and associated class probabilities from the image pixels, presenting a unified and end-to-end trainable model. This monolithic architecture learns directly from the input images during training, eliminating the need for complex multistage pipelines.

The history of YOLO models in OD is marked by significant technical advancements [21]. YOLOv1 [20] (2016) introduced a single end-to-end architecture that simultaneously predicted multiple bounding boxes and class probabilities for those boxes, resulting in significantly improved speed compared to previous region proposal-based methods [16]. This was achieved by dividing the input image into a grid, with each grid cell being responsible for detecting objects within it. Each cell can predict multiple bounding boxes and confidence scores for those boxes. The network used a combination of 24 convolutional layers and two fully connected layers, with a final output tensor providing class probabilities and bounding box coordinates. YOLOv2 [12] (2017) and YOLOv3 [22] (2018) brought improvements in both speed and accuracy by introducing anchor boxes to predict offsets rather than using the full bounding box and refining the feature extractor for improved accuracy. The same authors also proposed Darknet-19 and Darknet-53, consisting of 19 and 53 layered networks, respectively, while incorporating multiscale predictions using a Feature Pyramid Network (FPN) [23] for improved detection of small objects.

YOLOv4 [24] (2020) proposed the integration of Cross-Stage Partial Connections (CSPNet) [25], Path Aggregation Network (PANet) [26], and modified Spatial Attention Module (SAM) [27], along with the use of the Mish [28] activation function and Complete Intersection Over Union (CIoU) loss [29] for enhancing feature extraction and bounding box accuracy. YOLOv5 [30] (2020) introduced a more streamlined and simplified architecture along with model scalability enhancements to adjust the model size based on the available computational resources. The authors were also able to improve performance by using novel *Mosaic* augmentation combined with multiple other data augmentation methods for data preprocessing. This results in increased variance in the data, thereby improving detection accuracy. They also used the Sigmoid-weighted Linear Unit (SiLU) activation function [31] instead of Mish [28] employed in the previous versions. YOLOv6 [32] (2021) and YOLOv7 [33] (2022) focused on optimizing the balance between speed and accuracy for edge computing devices. The Efficient Long-Range Attention Network (ELAN) [34] strategy was used for more effective feature fusion, and an auxiliary head was employed for better training stability, increased convergence speed, and reduced training time. The authors

also proposed a Reparameterized Convolutional (RepConvN) block inspired by [35] to improve feature extraction.

YOLOv8 [15] (2023), the latest in the series, represents the culmination of ongoing efforts to optimize OD for both performance and computational efficiency while introducing a more advanced network architecture incorporating recent developments in neural network design. The authors used an anchor-free model inspired by [36] with a decoupled head to independently process objectness, classification, and regression tasks, along with the CIoU [29] and Distribution Focal Loss (DFL) [37] functions for bounding box loss and the binary cross-entropy [38] for classification loss. Figure 1 illustrates the fully visualized YOLOv8 architecture, with different stages of the network shaded in distinct colors for clarity.

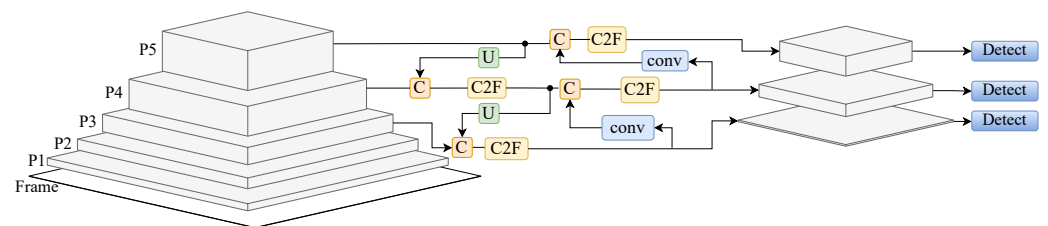


Figure 1. Complete YOLOv8 [15] architecture, consisting of backbone and head; C represents the convolutional block, U is the upsampling block, and C2F is the CSPNet with two convolutional layers. A detailed diagram can be found in [39].

The YOLO series continues to be a prominent example of innovation in CV and DL. Each version has contributed to the rapid progression and adaptability of DL models, helping them to become more efficient and capable and making them suitable for real-world applications. Models can be easily trained to detect required objects in the production facility using annotated image data. Compatibility with portable or standalone computers and mobile hardware devices makes these models ideal for real-time applications in smart manufacturing. The following section demonstrates how these advanced OD models are being leveraged to address challenges in industrial settings.

2.2. Applications of OD in Smart Manufacturing

Researchers have been actively identifying innovative ways to utilize the power of OD methods in smart manufacturing. Recently, Zendejdel et al. [40] used the YOLOv5 [30] model to identify and localize tools on the manufacturing floor as a way to enhance worker safety. Liu et al. [41] proposed a novel Lighter and Faster YOLO (LF-YOLO) model for defect detection based on the X-ray imagery of welds, and also proposed using a Reinforced Multiscale Feature (RMF) module to extract more hierarchical information. Wang et al. [42] proposed a lightweight YOLO-style object detector known as Attention YOLO (ATT-YOLO) for surface defect detection in electronics manufacturing. Zhao et al. [43] modified YOLOv5 [30] to detect steel surface defects and presented a modified architecture that utilizes low-level features for better detection. Puttemans et al. [10] and Vu et al. [44] respectively employed YOLOv2 [12] and YOLOv5 [30] to detect packages in warehouse environments, highlighting its utility in real-time applications for product packaging. Zhao et al. [45] proposed a modified lightweight YOLOv5 that achieved real-time performance in detecting particleboard surface defects. This was achieved by replacing the conventional convolutional layers with depthwise convolution layers and integrating Squeeze and Excitation Network (SENet) [46] layers to optimize the model's parameters. Rahimi et al. [47] proposed modifications to YOLOv3 for detecting large-scale objects in the automobile industry and enhanced the model by altering its architecture and activation function. Ahmad et al. [48] applied YOLOv3 for detecting and tracking cranes in steel manufacturing plants, showcasing the adaptability of these deep learning models for specific industrial surveillance tasks. Liu et al. [49] proposed YOLO for Industrial Manufacturing Field (YOLO-IMF), an improved YOLOv8 algorithm for surface defect detection in the industrial

manufacturing field. Luo et al. [50] modified YOLOv8 for edge computing by reducing the parameters and computational load as well as by modifying the lightweight ShuffleNetV2 network [51] and using it as the feature extractor in YOLO. Ahmad and Rahimi [52] compared the performance of different YOLO models for Personal Protective Equipment (PPE) detection to ensure human safety in manufacturing environments.

Additionally, Krummenacher et al. [53] applied deep learning for wheel defect detection. O'Brien et al. [54] introduced a method for quality inspection during the production of medical devices, addressing the need for high accuracy and low error tolerance in applications involving medical equipment manufacturing. Farahnakian et al. [13] and Li et al. [14] used OD methods for damage detection and pallet rack identification in industrial warehouse settings. Wei et al. [55] and Luo et al. [56] demonstrated the effectiveness of deep learning in detecting humans and industrial tools from a distance, showcasing the versatility of these models in diverse industrial scenarios. Wang et al. [11] advanced product defect detection using a deep learning approach that synergized preprocessing techniques with deep learning, thereby reducing computational load and excluding irrelevant background content.

While these studies demonstrate the effectiveness of OD methods in various aspects of smart manufacturing, there remains a gap in applying these techniques for comprehensive capacity constraint analysis. Our work addresses this gap by proposing a non-invasive framework that leverages state-of-the-art OD models for real-time monitoring and analysis of manufacturing processes.

3. Capacity Constraint Analysis

Capacity constraint in a manufacturing environment is defined as a situation where the production capacity of a business is insufficient to meet demand. This limitation can manifest in various forms, including labor, materials, or equipment constraints. According to the Theory of Constraints [57,58], every system has bottlenecks that dictate the pace of the entire production line, and addressing these bottlenecks can significantly increase overall output. Common bottleneck types include machine bottlenecks due to breakdowns, labor shortages or insufficient staffing, delays in materials supply, complex or inefficient processes, communication gaps, logistical challenges, power supply issues, and capacity mismatches. The initial step in this process involves thoroughly analyzing the constraints in order to pinpoint the actual limiting factors. In this section, we propose a comprehensive framework for analyzing workstations, particularly those operated by manual labor. This framework aims to identify the root causes of capacity limitations so that manufacturers can develop strategic long-term solutions to enhance manufacturing efficiency and output.

In order to effectively analyze workstations and identify bottlenecks in the manufacturing process, it is essential to consider productivity, which is fundamentally defined as the output per unit of input. Productivity is a critical metric for operational efficiency, especially in contexts with limited labor resources. While different organizations may use various parameters for this analysis, our proposed framework incorporates a holistic set of definitions considering both workforce and materials. This integrated approach ensures a more accurate identification of constraints and facilitates the development of targeted solutions to optimize manufacturing processes.

Definition 1 (Station Productivity). *We define station productivity as the output generated when a worker is actively engaged in working with materials.*

This measure focuses on the station's effectiveness in producing items, rather than solely assessing the worker's efficiency in the quantity of objects produced; the results is a nuanced measure of efficiency that highlights how effectively a station utilizes its resources (both human and material) to generate output.

We can express this as a Boolean function using logical operators by assigning Boolean variables to Material (M) and Worker (W), where True represents presence and False represents absence; then, the overall state can be represented as follows:

$$f_1(M, W) = M \wedge W \quad (1)$$

where \wedge represents the logical and operator.

Definition 2 (Non-Productivity). *Non-productivity at a station occurs when no value is produced due to worker unavailability.*

This situation typically happens when a worker is absent from their station. However, it is important to differentiate between avoidable and unavoidable non-productive time. For example, breaks are a necessary aspect of work that, while non-productive, contribute to overall worker productivity and wellbeing by preventing fatigue and maintaining mental health. Mathematically, non-productivity can be represented as

$$f_2(M, W) = M \wedge \neg W, \quad (2)$$

where \neg , M , and W represent the *logical not* operator, materials, and workers, respectively.

Definition 3 (Downtime). *In this context, downtime refers to periods when a worker is present but lacks the necessary materials to continue production.*

This situation can arise due to supply chain issues, scheduling errors, or unforeseen delays in material delivery. Downtime is a critical aspect of station productivity, as it directly impacts the output despite the availability of workers. The definition can be mathematically expressed as

$$f_3(M, W) = \neg M \wedge W, \quad (3)$$

where M and W respectively represent materials and workers.

Definition 4 (Idle Time). *Idle time is characterized by the absence of both workers and materials at a station.*

This occurs during off-hours or designated break times. Understanding idle time is crucial for workforce planning and ensuring that staffing levels are appropriate to the demands of the production schedule. The definition can be mathematically expressed as

$$f_4(M, W) = \neg M \wedge \neg W, \quad (4)$$

where M and W respectively represent materials and workers.

The above definitions can be mathematically combined by representing productivity, non-productivity, downtime, and idle time as S_1 , S_2 , S_3 , and S_4 , respectively:

$$f(M, W) = S_1 \cdot (M \wedge W) + S_2 \cdot (M \wedge \neg W) + S_3 \cdot (\neg M \wedge W) + S_4 \cdot (\neg M \wedge \neg W) \quad (5)$$

where M represent materials and W represents workers. Visually, Equation (5) can be represented as a simple lookup table, as shown in Table 1, which can be used to find the status of each frame.

Table 1. Different possibilities for station productivity; ✓ represents availability at the station.

State	Symbol	Material	Worker
Productive	S_1	✓	✓
Non-productive	S_2	✓	
Downtime	S_3		✓
Idle time	S_4		

Cycle Time Study

We can find the station status for each frame; however, aggregating that status over time provides the real value for the life of an object in the scene, also known as the cycle time. We want to track when an object appears in and leaves the scene; this can be measured using several methods, as introduced by [59]. Here, we discuss several of these methods along with their respective advantages and disadvantages:

Stop Watches Tasks performed by workers can be manually timed in the manufacturing environment. This traditional method is commonly used for establishing benchmarks in manual operations. However, reliance on human observation limits the scalability and accuracy of this method, and may not accurately represent normal working conditions due to observer bias and the Hawthorne effect. Additionally, manual timing cannot easily be integrated into the broader data ecosystems that smart manufacturing relies on for real-time decision-making.

Video Recording with Offline Analysis This allows for efficient analysis of processes by reviewing recorded footage. Video recording enables detailed post hoc analysis to identify bottlenecks and inefficiencies that may not be visible in real time. However, it suffers from delays in feedback, as the analysis only occurs after the fact. Additionally, storage and management of large video datasets can be challenging.

Breaking Activities into Tasks and Subtasks This can help in understanding task performance and supports line balancing, especially in complex manufacturing processes. By decomposing activities into granular subtasks, manufacturers can identify specific areas for optimization; however, this approach is time-consuming and requires significant initial input to define tasks accurately. Additionally, real-time adaptability to changes in the manufacturing environment might be limited without advanced automation tools.

Working with Predetermined Standard Times This approach offers deep insights into task performance by using historical data and industry standards to establish benchmarks. While predetermined times provide a solid foundation for efficiency analysis, they may not align perfectly with actual timings due to variability in human and machine performance. Moreover, this method may not capture the nuances of novel or highly customized manufacturing processes, requiring continuous updating of standard times.

Sensor-based Tracking This involves using data from Industrial Internet of Things (IIoT) sensors and workflow systems for real-time productivity analysis, the results of which can be fed into predictive analytics models to forecast potential delays and optimize production schedules. However, while this method is efficient, it does not provide insights into the root causes of productivity changes, as sensors typically provide raw data which lack context. Integration with other data sources such as quality control systems is necessary in order to gain a comprehensive understanding.

Visual Tracking This method combines non-intrusive real-time data collection with identification of improvement opportunities by visual tracking of objects and processes. In smart manufacturing scenarios, visual tracking can be implemented through advanced computer vision systems that monitor the production line, providing real-time feedback to operators and management. These systems can detect anomalies, track the flow of goods, and even analyze worker movements for ergonomic improvements. Although the upfront investment in visual tracking technology can be substantial, it is increasingly becoming cost-effective due to advancements in AI and ML. Additionally, the integration

of visual tracking with other smart manufacturing systems can lead to a more holistic view of production efficiency.

While most methods require manual calculation and human input, visual tracking is real-time, does not need human feedback after development is complete, and yields accurate results. It can be implemented by OD-based methods, and can be used to accurately and effectively conduct cycle time studies. Within our proposed methodology, we use the YOLOv8 model [15] as a state-of-the-art OD model, as previously discussed in Section 2.1.

4. Methodology

The proposed methodology consists of a two-step approach: (1) obtaining data for use in training an OD model (in the preceding section, we highlighted the state-of-the-art performance of YOLOv8, which is used as the primary OD model in this study), and (2) testing the model by conducting a capacity constraint analysis in a real manufacturing facility.

4.1. Dataset Description

Manual annotation of the dataset employed in training and evaluation of the OD model was carried out using videos sourced directly from the four production line stations within the client's facility over the course of roughly two months. Each station was responsible for different operations in the facility. The camera viewing angles covering each station were very different due to the nature of the neighboring mounts used to hold the equipment. The variability in the collected data (as seen in Figure 2) provided the necessary variation required to train an effective OD model. As the client's facility runs only for one 8.5 h shift every day, the lighting conditions in the indoor environment remained consistent at each station. A total of 33,956 individual frames were extracted from these recordings at a frequency of 0.3 frames per second (*fps*). As the scene in each station did not change and only the objects moved, we performed stratified data splitting into training and validation sets to avoid data leakage, as data from each day should only be in either split. Thus, we separated each day of data randomly out of five working days each week for use as validation data. The total images used for training/validation ended up as 29,070/4886, consisting of two object classes, i.e., workers and chairs.

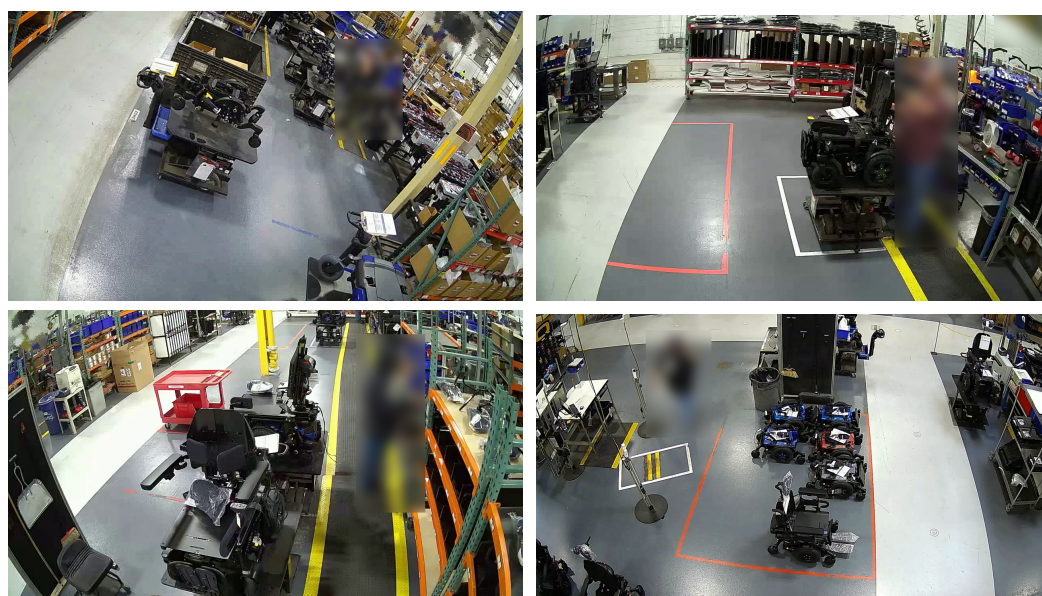


Figure 2. Camera views of the four stations in the production facility. The people in the images are blurred to protect their identity and comply with the request from the industry partner.

4.2. Training

To train the YOLOv8 model, as defined by [15], we employed a transfer learning approach to refine the pretrained model, which was initially trained on the Microsoft Common Objects in Context (MS-COCO) dataset [60]. The open-source model implementation (we used the open-source implementation of the YOLOv8 model available at <https://github.com/ultralytics/ultralytics> (accessed on 10 February 2024)) is available in five different sizes: nano (n), small (s), medium (m), large (l), and extra-large (x), depending on the number of parameters. Table 2 represents the number of parameters in each model. We separately trained nano, medium, and large models in order to compare their accuracy and detection speed. The model output layers was reconfigured to identify two distinct object types. To enhance accuracy, non-maximum suppression [61] was utilized for output refinement. Moreover, the cosine annealing learning rate method [62] was implemented as a scheduler, which we chose due to its demonstrated excellence in various benchmark tests.

To augment our dataset, we incorporated several methods, including random vertical flipping and a mosaic of four frames as per [36]. These techniques expanded the effective dataset size for model training and contributed to a more robust learning process. The training was carried out using two NVIDIA RTX TITAN GPUs (Manufacturer: NVIDIA Corporation, City: Santa Clara, Country of origin: United States) with a batch size of 128 for the nano and medium models and 64 for the large model due to GPU memory limitation.

4.3. Evaluation Metrics

Detection accuracy and inference speed are key metrics for evaluating OD models. Accuracy is often evaluated using Precision (P) and Recall (R), which are derived from the counts of True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN), while speed is measured in *fps*. The formulas for these metrics are as follows:

$$R = \frac{TP}{TP + FN} \quad (6)$$

$$P = \frac{TP}{TP + FP} \quad (7)$$

Furthermore, Mean Average Precision (mAP) assesses the detection accuracy. This measure is calculated for each class based on P and R, then averaged to yield an overall score. To quantify the accuracy of object localization, the Intersection over Union (IoU) metric is used; this metric is calculated between the labeled objects (ground truth) and the model's predictions, as follows:

$$IoU = \frac{Area(b_{pred} \cap b_g)}{Area(b_{pred} \cup b_g)} \quad (8)$$

where b_g represents the ground truth bounding box and b_{pred} denotes the bounding box predicted by the OD model. The IoU threshold functions as a Boolean operator to eliminate FP bounding boxes that score below a certain IoU value. This threshold determines the necessary sensitivity for the localization to be classified as positive or negative (e.g., $IoU \geq \text{threshold}$). Different models may employ varying threshold values in their evaluations, such as 0.25, 0.5, or 0.75. Table 2 lists the evaluation results and inference speed of the different YOLOv8 models, with operations quantified by Floating-Point Operations Per Second (FLOPS).

Table 2. Different model sizes of YOLOv8 Model from [15]; mAP values are calculated for single-scale on the MS-COCO [60] val2017 dataset.

Model	mAP _{0.50–0.95}	GPU (ms)	Parameters (M)
YOLOv8-n	18.4	1.21	3.5
YOLOv8-s	27.7	1.40	11.4
YOLOv8-m	33.6	2.26	26.2
YOLOv8-l	34.9	2.43	44.1
YOLOv8-x	36.3	3.56	68.7

4.4. YOLOv8 Model Performance

We evaluated the performance of three variants (nano, medium, large) of the YOLOv8 model in detecting the *worker* and *chair* classes on a separate test set. The medium model exhibited superior overall performance with 94.4% mAP@50, which was higher than the nano and large models by 1.8% and 0.6%, respectively. Theoretically, the large model should have performed better due to more learnable parameters; however, as only two classes existed within four camera views, more parameters could have caused overfitting on the training set due to more limited variation compared to large-scale general-purpose datasets. Furthermore, a larger batch size can improve learning due to batch normalization, as observed by [63,64].

We present the evaluation results of the different model types trained on our collected dataset in Table 3. In each column, the items with the highest value are denoted in boldface. While the large model has a slightly better P and R on worker detection, it is also compute-heavy, with 40% more trainable parameters. It takes 0.17 milliseconds (ms) more than the medium model on each frame, and this compounds fairly rapidly during real-time inference without adding significant value. Hence, we chose the medium model to conduct our analysis of the client’s manufacturing facility and obtain insights into the productivity of each station.

Table 3. Performances comparison of nano, medium, and large YOLOv8 [15] models, with the highest values in each column highlighted in boldface.

Model Size	P (%)			R (%)		
	All	Worker	Chair	All	Worker	Chair
Nano	89.2	84.4	93.9	87.1	86.0	88.2
Medium	89.9	85.4	94.4	88.8	89.5	88.0
Large	89.8	85.7	93.8	89.0	90.2	87.7
Model Size	mAP50 (%)			mAP _{50–95} (%)		
	All	Worker	Chair	All	Worker	Chair
Nano	92.6	91.7	93.5	64.7	64.8	64.6
Medium	94.4	93.8	95.0	68.8	69.7	68.0
Large	93.8	93.8	93.9	68.9	70.0	67.7

Figure 3 presents the training metrics of the medium model after each training epoch, showing that the mAP@50 reaches a plateau after 65 epochs. P and R are always opposing metrics [65], with increasing P affecting R, as can be seen in Figure 3a,b. Hence, mAP provides a better measure for considering the convergence of the model.

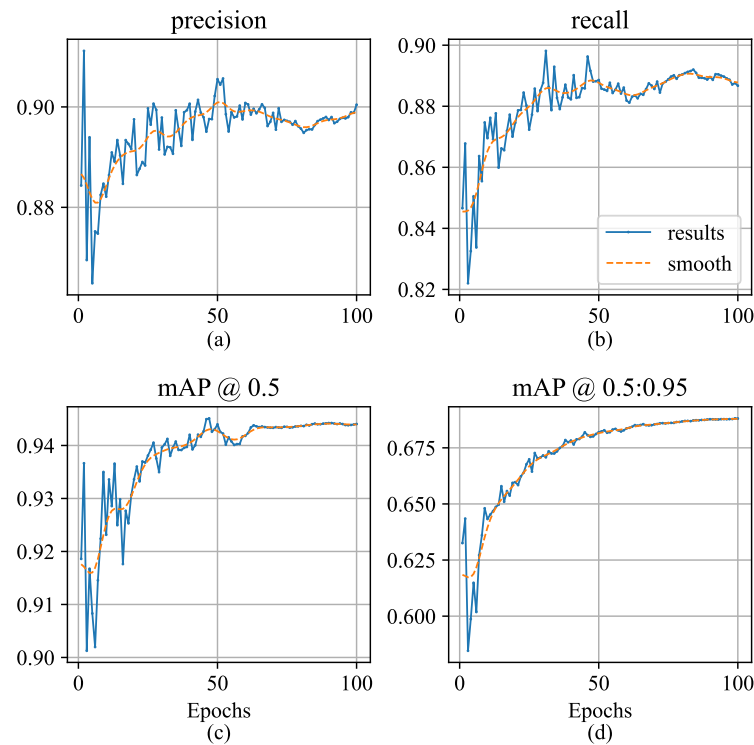


Figure 3. Normalized (a) precision, (b) recall, (c) mAP@50, and (d) mAP50-95 of the YOLOv8 [15] model training.

4.5. Complete Pipeline

The complete pipeline consists of different stages, as shown in Figure 4. A video frame is initially processed through an OD model for detecting objects in the scene. As explained in Section 4.2, we used the trained YOLO model, which processes only one frame at a time and provides no temporal information. To obtain this information and assign a unique Identifier (ID) to each object, we used the state-of-the-art Deep Simple Object Tracking (DeepSort) [66] model, which uses Kalman Filtering (KF) [67] to maintain tracking continuity from the previous state and predict the location of the bounding box in the subsequent frame. This algorithm leverages a deep CNN architecture in the prediction process, combining the strengths of KF with the representational power of CNNs.

This unique ID helps to distinguish between different objects, and can be aggregated to obtain the timestamps for the start and end of the object's life cycle. Each object is assigned a region on the floor to identify the item under process. These metadata are then processed through the proposed framework (as defined in Section 3), where we assign a station status from S_1 , S_2 , S_3 , and S_4 to each frame. These statuses are then aggregated and logged into the database, which is used for bottleneck detection in the entire pipeline by highlighting the reason as either non-productive, downtime, or idle time.

The lifecycle of each object is also logged in the database, and specific alerts are generated if the object's lifecycle increases by a predefined threshold (often calculated as some multiple of the average processing time depending on the use case). These alerts can help line managers and manufacturers to optimize the supply chain by providing the required parts for installation and improving labor productivity.

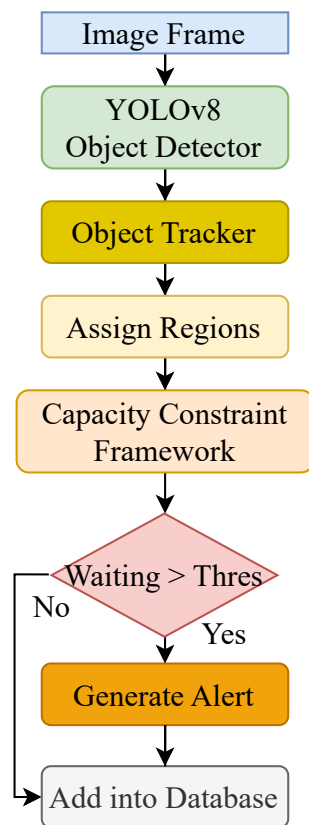


Figure 4. Complete pipeline of the proposed methodology.

5. Insights into Manufacturing Facility

The synergistic combination of YOLO and our proposed framework aims to highlight the specific challenges faced by the client in manufacturing assistive medical wheelchairs based on OD, localization, and tracking throughout the production process. With permission from the existing workers in the clients' facility, we strategically deployed cameras on the manufacturing floor to capture live video feeds for analysis. Figure 5 showcases the frame captured from an installed camera on-site with the view of a station where workers are assembling a chair.

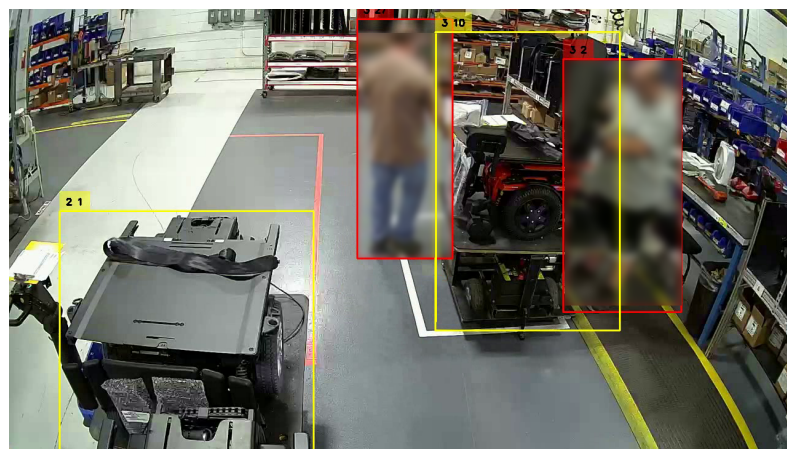


Figure 5. View of floor lines, with power wheelchairs and workers represented by yellow and red boxes, respectively. The people in these images are blurred to protect their identities and comply with the request from the industry partner.

Challenges

The wheelchair manufacturing process commences with specifications from healthcare professionals tailored to individual patient needs. These specifications dictate the assembly at various stations, each manned by a dedicated worker. Despite a standard processing time at each station, delays are common due to labor shortages and workers having to move between stations. Communication gaps exacerbate these delays, especially when the floor managers may be unaware of inventory issues. Transition periods, notably during shift changes, create inefficiencies and bottlenecks, impeding the production timeline. The presence of faulty items in the production line necessitates additional quality control time, disrupting the manufacturing flow. As indicated by data from the Manufacturing Execution System (MES), this inability to promptly identify and address bottlenecks further delays production. The MES data are often inaccurate, as they are primarily reliant on manual worker input. While informative, continuous manual time studies are impractical and may lead to skewed productivity metrics.

Given these challenges, we propose a non-invasive system for capacity constraint analysis. This system tracks the station and cycle time of chairs, allowing for effective labor and inventory planning. Over six months (July to December), we collected videos from four workstations identified by MES data as a primary bottleneck. For simplicity, we can label them as A, B, C, and D, with Station C being the critical one. As the facility works only for the morning shift, data were collected for 8.5 h daily. We excluded three standard 25 min break times and detected workers and chairs using the trained YOLOv8 model as outlined in Section 3. Each frame's status was aggregated to extract various metrics. Figure 6 shows that Station C is 27.9% unproductive, indicating a critical labor shortage with a high percentage for the whole duration. Figure 7 reveals a consistent trend in productivity over the months, contrary to the expected increase towards the year's end due to high demand. This consistency points to a critical labor shortage. Figure 8 depicts normalized hourly productivity data. Notably, productivity is higher in the morning than in the afternoon. Figure 9 represents the daily insights into the client's capacity constraint. They are productive only for 60% to 65% of the time during the 8.5 h shift, and the remaining time is mostly unproductive when a worker is not available at the station to work on the chair.

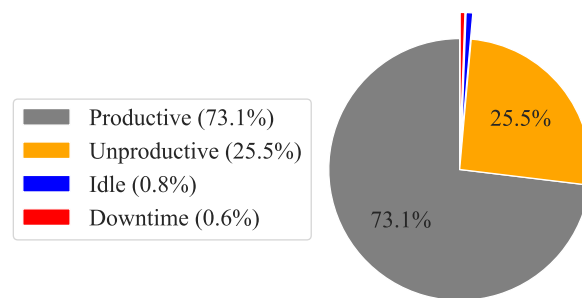


Figure 6. Pie chart representing the status of Station C over 6 months.

Further analysis of the lifecycles of individual chairs is presented in Figure 10, where the box plot represents the five-number summary of the processing time of the chair each week. We have filtered out processing times of less than two minutes due to instances where a person is occluding the chair and the model is not able to detect it for a certain time; the tracker loses detection and completes the life cycle, then assigns a new ID after the chair is visible, resulting in two distinct IDs being assigned for the same chair. It is evident that the median time (represented by the red colored line) is reduced to 15 minutes on some days, while it also increases to 23 minutes (week 42), resulting in a 53% increase in processing time and a decrease in daily output. Upon manual verification of the video data, the worker in charge was on vacation on week 42, and another worker from a different

station was taking care of the station. A similar trend is also visible during days when a worker handles multiple stations or during a severe inventory situation (week 39).

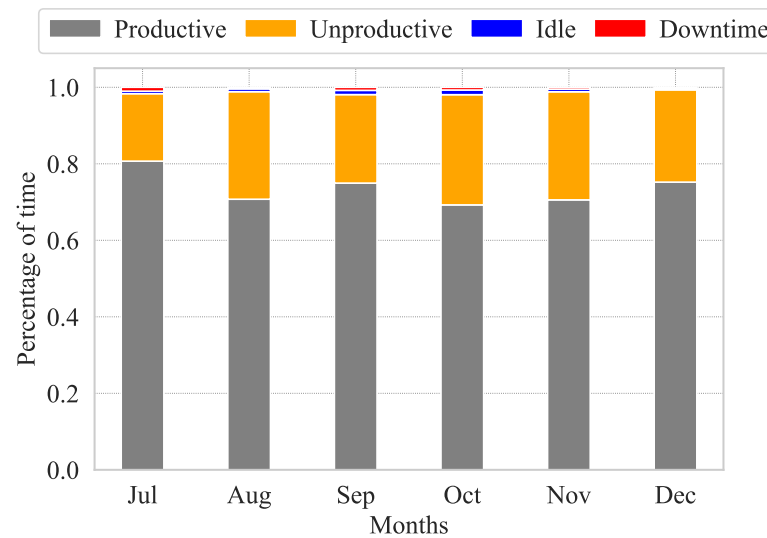


Figure 7. Monthly aggregated status of Station C over 6 months.

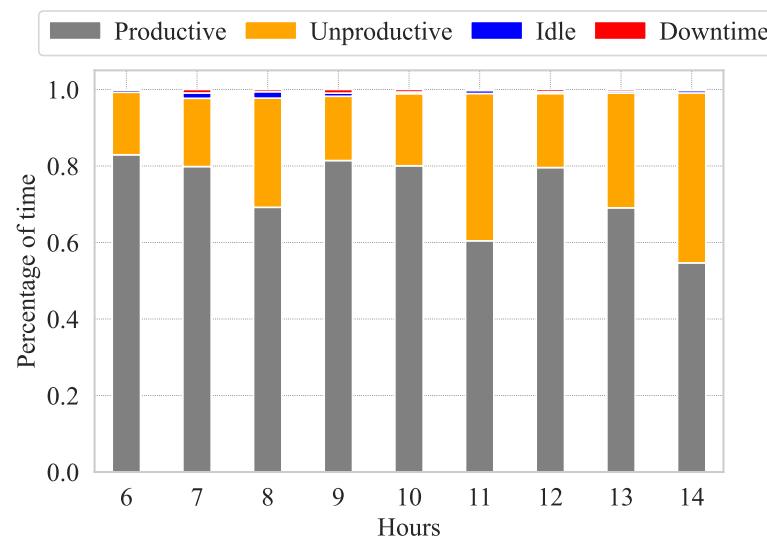


Figure 8. Hourly aggregated data showing the status of Station C over 6 months.

These insights are crucial for workflow optimization. They are most effective when applied in a production environment with real-time metrics and an alert system for increasing processing time, as mentioned in Figure 4. Such a system enables floor managers to promptly address issues, thereby improving efficiency and productivity. For example, if the analysis reveals frequent downtimes due to material shortages, manufacturers can adjust their inventory management strategies; similarly, patterns of non-productivity can inform staffing decisions and training programs. This approach is highly scalable, and can be directly implemented in other manufacturing industries where the production pipeline consists of workstations for individual tasks and human workers are required for the precise assembly of the parts.

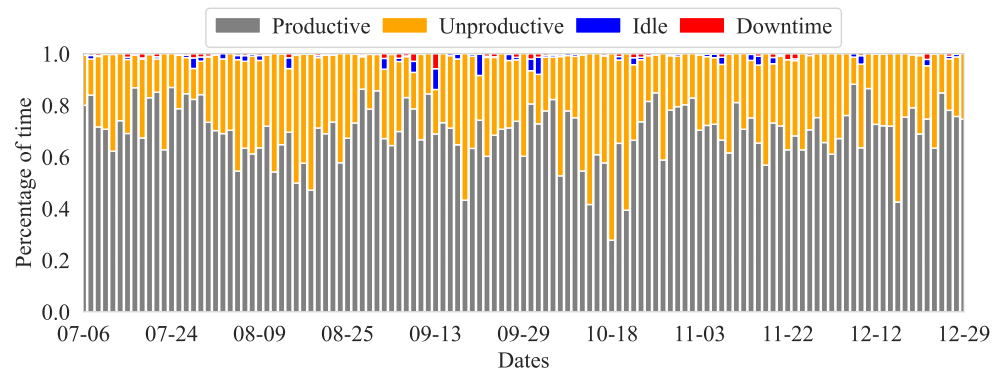


Figure 9. Daily status of Station C in terms of productivity over 6 months.

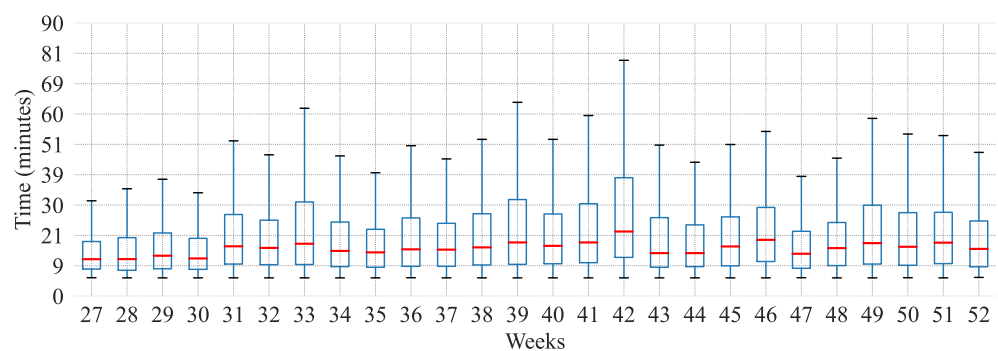


Figure 10. Box plot of the processing time for all the chairs over 6 months, aggregated for each week for Station C. The red line and the lower and upper box boundaries represent the median and the 25th and 75th percentiles, respectively. Please note that these numbers are scaled in order to anonymize them, as requested by the industry partner.

6. Conclusions

In this work, we implemented a non-invasive system for monitoring capacity constraints in manufacturing environments. Data were collected from four manufacturing assembly line stations over six months. Having reviewed the existing literature on OD, we propose a state-of-the-art end-to-end framework using YOLOv8 for object detection, providing insights into labor and inventory management, and revealing notable labor shortages and inefficiencies. Our study underscores the importance of real-time metrics and alert systems for enhancing efficiency and productivity in manufacturing environments. The overall productivity of Station C was calculated to be only 73.1% over 6 months, suggesting significant potential for technological integration in optimizing manufacturing processes.

6.1. Implementation of the Results in Practice

The findings from this study have significant practical implications. By utilizing the proposed framework, manufacturers can gain real-time insights into labor and inventory issues, allowing them to make informed decisions quickly. The proposed framework can be directly integrated into existing assembly line systems, thereby facilitating improvements in labor allocation, reducing inefficiencies, and optimizing overall productivity. In particular, the identified bottlenecks at Station C can serve as a starting point for targeted interventions, ensuring that resources are deployed more effectively and that output matches demand.

6.2. Limitations of the Proposed Framework

While providing promising and groundbreaking results, the application of OD in this context also presents limitations in terms of tracking, especially when occlusion or overlapping objects are present in the frame in the form of a worker occluding the object being worked on or a guest visiting the station to casually chat for an extended period.

Long periods when the object is not visible can inflate the results. Additionally, the accuracy of the analysis depends on the precision of the OD model; thus, training an effective model is critical.

While the proposed framework provides valuable insights, it is crucial to implement it ethically. Worker privacy must be respected, and the collected data should be used solely for productivity analysis, not for individual performance evaluation. Clear communication with workers about the purpose and scope of the monitoring is essential.

6.3. Future Direction

Future work will focus on expanding the scope of the current system, particularly by incorporating LMMs to predict future trends in labor and inventory management. These models will allow for prescriptive insights that not only reflect past performance but also suggest actionable steps for future improvements. Additionally, further research will explore the application of the proposed framework to other industries and manufacturing environments along with ways to enhance the system's scalability and robustness for broader industrial use. Integration with emerging technologies such as edge computing and IoT will also be considered to further streamline data collection and analysis.

Author Contributions: Conceptualization, H.M.A.; methodology, H.M.A.; software, H.M.A., A.R., and K.H.; validation, H.M.A.; formal analysis, H.M.A. and A.R.; investigation, H.M.A., A.R. and K.H.; resources, A.R. and K.H.; data curation, H.M.A. and K.H.; writing—original draft preparation, H.M.A.; writing—review and editing, A.R.; visualization, H.M.A. and A.R.; supervision, A.R. and K.H.; project administration, A.R. and K.H.; funding acquisition, A.R. and K.H. All authors have read and agreed to the published version of the manuscript.

Funding: This study was supported by IFIVEO CANADA INC., Mitacs through IT16094, Natural Sciences and Engineering Research Council of Canada (NSERC) through ALLRP 560406-20, Ontario Center of Innovation (OCI) through OCI# 34166, and by the University of Windsor, Canada.

Institutional Review Board Statement: Ethical review and approval were waived for this study because the University of Windsor's Research and Ethics Board (REB) has designated this project exempt from ethical review. This is because the information being collected meets the exemption criteria established in the TCPS2 under Section 2.3. Section 2.3 of the Tri-Council Policy Statement. The TCPS2 (2022) indicates the following: REB review is not required for research involving the observation of people in public places where: (a) it does not involve any intervention staged by the researcher, or direct interaction with the individuals or groups; (b) individuals or groups targeted for observation have no reasonable expectation of privacy; and (c) any dissemination of research results does not allow identification of specific individuals. The exemption is specific to this project and is based on the information provided and the policy in place as of the current date. Similar projects may or may not require ethical oversight in the future.

Informed Consent Statement: Verbal informed consent was obtained from the participants. Verbal consent was obtained rather than written because the consent was given by the employees to their employers for the collection and use of this information, who provided the authors with the data and access to information for this study; additionally, this study is exempt from REB approval requirements, as noted above.

Data Availability Statement: The data used in this work cannot be made publicly available due to the industrial partner's client's protection of proprietary information.

Acknowledgments: A special thanks to Dario Morle and Syeda Sitara Wishal Fatima from IFIVEO CANADA INC. for their helpful insights into the implemented methodology.

Conflicts of Interest: Author Khizer Hayat was employed by the company IFIVEO Canada Inc. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. World Bank. *World Bank Open Data*; Manufacturing: Value Added (% of GDP); World Bank: Washington, DC, USA, 2023.
2. Causa, O.; Abendschein, M.; Luu, N.; Soldani, E.; Soriolo, C. The post-COVID-19 rise in labour shortages. *OECD Econ. Dep. Work. Pap.* **2022**. [CrossRef]
3. Canaj, K.; Sood, S.; Johnston, C. *Analysis on Labour Challenges in Canada, Second Quarter of 2023*; Government of Canada, Statistics Canada: Ottawa, ON, Canada, 2023.
4. Bomal, L.A. Labour Shortage CFIB. 2023. Available online: <https://www.cfib-fcei.ca/en/media/labour-shortages-cost-ontario-small-businesses-over-16b-in-lost-revenue> (accessed on 25 January 2024).
5. Gervasi, R.; Barravecchia, F.; Mastrogiacomo, L.; Franceschini, F. Applications of affective computing in human-robot interaction: State-of-art and challenges for manufacturing. *Proc. Inst. Mech. Eng. Part B J. Eng. Manuf.* **2023**, *237*, 815–832. [CrossRef]
6. Poudel, L.; Elagandula, S.; Zhou, W.; Sha, Z. Decentralized and centralized planning for multi-robot additive manufacturing. *J. Mech. Des.* **2023**, *145*, 012003. [CrossRef]
7. Liu, L.; Zou, Z.; Greene, R.L. The Effects of Type and Form of Collaborative Robots in Manufacturing on Trustworthiness, Risk Perceived, and Acceptance. *Int. J. Hum.-Comput. Interact.* **2023**, *40*, 2697–2710. [CrossRef]
8. Pansara, R. From Fields to Factories A Technological Odyssey in Agtech and Manufacturing. *Int. J. Manag. Educ. Sustain. Dev.* **2023**, *6*, 1–12.
9. Ahmad, H.M.; Rahimi, A. Deep learning methods for object detection in smart manufacturing: A survey. *J. Manuf. Syst.* **2022**, *64*, 181–196. [CrossRef]
10. Puttemans, S.; Callemeyn, T.; Goedemé, T. Building Robust Industrial Applicable Object Detection Models using Transfer Learning and Single Pass Deep Learning Architectures. In Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2018), Funchal, Portugal, 27–29 January 2018; pp. 209–217. [CrossRef]
11. Wang, J.; Fu, P.; Gao, R.X. Machine vision intelligence for product defect inspection based on deep learning and Hough transform. *J. Manuf. Syst.* **2019**, *51*, 52–60. [CrossRef]
12. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6517–6525. [CrossRef]
13. Farahnakian, F.; Koivunen, L.; Mäkilä, T.; Heikkinen, J. Towards Autonomous Industrial Warehouse Inspection. In Proceedings of the 2021 26th International Conference on Automation and Computing (ICAC), Portsmouth, UK, 2–4 September 2021; pp. 1–6. [CrossRef]
14. Li, T.; Huang, B.; Li, C.; Huang, M. Application of convolution neural network object detection algorithm in logistics warehouse. *J. Eng.* **2019**, *2019*, 9053–9058. [CrossRef]
15. Jocher, G.; Chaurasia, A.; Qiu, J. YOLO by Ultralytics. 2023. Available online: <https://github.com/ultralytics/ultralytics> (accessed on 20 January 2024).
16. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 28 June 2014; pp. 580–587. [CrossRef]
17. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
18. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef]
19. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In Proceedings of the Computer Vision—ECCV, Amsterdam, The Netherlands, 11–14 October 2016; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer: Cham, Switzerland, 2016; pp. 21–37.
20. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788. [CrossRef]
21. Terven, J.; Cordova-Esparza, D. A Comprehensive Review of YOLO: From YOLOv1 and Beyond. *arXiv* **2023**, arXiv:2304.00501. [CrossRef]
22. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:804.02767.
23. Lin, T.Y.; Dollar, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 26 July 2017; pp. 936–944. [CrossRef]
24. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:abs/2004.10934.
25. Wang, C.Y.; Mark Liao, H.Y.; Wu, Y.H.; Chen, P.Y.; Hsieh, J.W.; Yeh, I.H. CSPNet: A New Backbone that can Enhance Learning Capability of CNN. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 14–19 June 2020; pp. 1571–1580. [CrossRef]
26. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path Aggregation Network for Instance Segmentation. *arXiv* **2018**, arXiv:1803.01534.

27. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich Germany, 8–14 September 2018; pp. 3–19.
28. Mishra, D. Mish: A Self Regularized Non-Monotonic Activation Function. *arXiv* **2020**, arXiv:1908.08681.
29. Zheng, Z.; Wang, P.; Liu, W.; Li, J.; Ye, R.; Ren, D. Distance-IoU loss: Faster and better learning for bounding box regression. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 12993–13000.
30. Jocher, G.; Stoken, A.; Borovec, J.; NanoCode012; ChristopherSTAN; Liu, C.; Laughing; tkianai; Hogan, A.; lorenzomamma; et al. ultralytics/yolov5: v3.1—Bug Fixes and Performance Improvements. 2020. Available online: <https://zenodo.org/records/4154370> (accessed on 20 January 2024).
31. Elfving, S.; Uchibe, E.; Doya, K. Sigmoid-Weighted Linear Units for Neural Network Function Approximation in Reinforcement Learning. *arXiv* **2017**, arXiv:1702.03118. [[CrossRef](#)]
32. Li, C.; Li, L.; Jiang, H.; Weng, K.; Geng, Y.; Li, L.; Ke, Z.; Li, Q.; Cheng, M.; Nie, W.; et al. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications. *arXiv* **2022**, arXiv:2209.02976.
33. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 7464–7475.
34. Wang, C.Y.; Liao, H.Y.M.; Yeh, I.H. Designing Network Design Strategies Through Gradient Path Analysis. *arXiv* **2022**, arXiv:2211.04800.
35. Ding, X.; Zhang, X.; Ma, N.; Han, J.; Ding, G.; Sun, J. Repvgg: Making vgg-style convnets great again. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 13733–13742.
36. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. YOLOX: Exceeding YOLO Series in 2021. *arXiv* **2021**, arXiv:2107.08430.
37. Li, X.; Wang, W.; Wu, L.; Chen, S.; Hu, X.; Li, J.; Tang, J.; Yang, J. Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 21002–21012.
38. Good, I.J. Rational decisions. *J. R. Stat. Soc. Ser. B (Methodol.)* **1952**, *14*, 107–114. [[CrossRef](#)]
39. King, R. Brief Summary of YOLOv8 Model Structure. 2023. Available online: <https://github.com/ultralytics/ultralytics/issues/189> (accessed on 11 March 2024).
40. Zendejdel, N.; Chen, H.; Leu, M.C. Real-time tool detection in smart manufacturing using You-Only-Look-Once (YOLO)v5. *Manuf. Lett.* **2023**, *35*, 1052–1059. [[CrossRef](#)]
41. Liu, M.; Chen, Y.; Xie, J.; He, L.; Zhang, Y. LF-YOLO: A lighter and faster yolo for weld defect detection of X-ray image. *IEEE Sensors J.* **2023**, *23*, 7430–7439. [[CrossRef](#)]
42. Wang, J.; Dai, H.; Chen, T.; Liu, H.; Zhang, X.; Zhong, Q.; Lu, R. Toward surface defect detection in electronics manufacturing by an accurate and lightweight YOLO-style object detector. *Sci. Rep.* **2023**, *13*, 7062. [[CrossRef](#)] [[PubMed](#)]
43. Zhao, C.; Shu, X.; Yan, X.; Zuo, X.; Zhu, F. RDD-YOLO: A modified YOLO for detection of steel surface defects. *Measurement* **2023**, *214*, 112776. [[CrossRef](#)]
44. Vu, T.T.H.; Pham, D.L.; Chang, T.W. A YOLO-based Real-time Packaging Defect Detection System. *Procedia Comput. Sci.* **2023**, *217*, 886–894. [[CrossRef](#)]
45. Zhao, Z.; Yang, X.; Zhou, Y.; Sun, Q.; Ge, Z.; Liu, D. Real-time detection of particleboard surface defects based on improved YOLOV5 target detection. *Sci. Rep.* **2021**, *11*, 21777. [[CrossRef](#)]
46. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141. [[CrossRef](#)]
47. Rahimi, A.; Anvaripour, M.; Hayat, K. Object Detection using Deep Learning in a Manufacturing Plant to Improve Manual Inspection. In Proceedings of the 2021 IEEE International Conference on Prognostics and Health Management (ICPHM), Detroit, MI, USA, 7–9 June 2021; pp. 1–7. [[CrossRef](#)]
48. Ahmad, H.M.; Rahimi, A.; Hayat, K. Deep Learning Transforming the Manufacturing Industry: A Case Study. In Proceedings of the 2021 IEEE 23rd Int Conf on High Performance Computing and Communications; 7th Int Conf on Data Science and Systems; 19th Int Conf on Smart City; 7th Int Conf on Dependability in Sensor, Cloud and Big Data Systems and Application (HPCC/DSS/SmartCity/DependSys), Haikou, China, 20–22 December 2021; pp. 1286–1291. [[CrossRef](#)]
49. Liu, Z.; Ye, K. YOLO-IMF: An Improved YOLOv8 Algorithm for Surface Defect Detection in Industrial Manufacturing Field. In Proceedings of the Metaverse—METAVERSE, Honolulu, HI, USA, 23 September 2023; He, S., Lai, J., Zhang, L.J., Eds.; Springer: Cham, Switzerland, 2023; Lecture Notes in Computer Science, pp. 15–28. [[CrossRef](#)]
50. Luo, B.; Kou, Z.; Han, C.; Wu, J. A “Hardware-Friendly” Foreign Object Identification Method for Belt Conveyors Based on Improved YOLOv8. *Appl. Sci.* **2023**, *13*, 11464. [[CrossRef](#)]
51. Ma, N.; Zhang, X.; Zheng, H.T.; Sun, J. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In Proceedings of the European Conference on Computer Vision (ECCV), Tel Aviv, Israel, 23–27 October 2018; pp. 116–131.
52. Ahmad, H.M.; Rahimi, A. SH17: A Dataset for Human Safety and Personal Protective Equipment Detection in Manufacturing Industry. *arXiv* **2024**, arXiv:2407.04590.
53. Krummenacher, G.; Ong, C.S.; Koller, S.; Kobayashi, S.; Buhmann, J.M. Wheel Defect Detection with Machine Learning. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 1176–1187. [[CrossRef](#)]

54. O'Brien, K.; Humphries, J. Object Detection using Convolutional Neural Networks for Smart Manufacturing Vision Systems in the Medical Devices Sector. *Procedia Manuf.* **2019**, *38*, 142–147. [[CrossRef](#)]
55. Wei, H.; Laszewski, M.; Kehtarnavaz, N. Deep Learning-Based Person Detection and Classification for Far Field Video Surveillance. In Proceedings of the 2018 IEEE 13th Dallas Circuits and Systems Conference (DCAS), Dallas, TX, USA, 12 November 2018; pp. 1–4. [[CrossRef](#)]
56. Luo, C.; Yu, L.; Yang, E.; Zhou, H.; Ren, P. A benchmark image dataset for industrial tools. *Pattern Recognit. Lett.* **2019**, *125*, 341–348. [[CrossRef](#)]
57. Goldratt, E.M.; Cox, J. *The goal: A Process of Ongoing Improvement*; Routledge: London, UK, 2016.
58. Goldratt, E.M. *Theory of Constraints*; North River Croton-on-Hudson; The North River Press: Mt. Clemens, MI, USA, 1990.
59. Christian Terwiesch. How to Measure and Improve Labor Productivity. Knowledge at Wharton. Available online: <https://knowledge.wharton.upenn.edu/article/how-to-measure-and-improve-labor-productivity/> (accessed on 26 January 2024).
60. Lin, T.Y.; Maire, M.; Belongie, S.; Bourdev, L.; Girshick, R.; Hays, J.; Perona, P.; Ramanan, D.; Zitnick, C.L.; Dollár, P. Microsoft COCO: Common Objects in Context. *arXiv* **2015**, arXiv:1405.0312. [[CrossRef](#)]
61. Neubeck, A.; Van Gool, L. Efficient Non-Maximum Suppression. In Proceedings of the 18th International Conference on Pattern Recognition (ICPR'06), Hong Kong, China, 20–24 August 2006; Volume 3, pp. 850–855. [[CrossRef](#)]
62. Loshchilov, I.; Hutter, F. SGDR: Stochastic Gradient Descent with Warm Restarts. In Proceedings of the 5th International Conference on Learning Representations, ICLR 2017—Conference Track Proceedings, Toulon, France, 24–26 April 2017.
63. You, Y.; Gitman, I.; Ginsburg, B. Large Batch Training of Convolutional Networks. *arXiv* **2017**, arXiv:1708.03888.
64. Bjorck, N.; Gomes, C.P.; Selman, B.; Weinberger, K.Q. Understanding Batch Normalization. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2018; Volume 31.
65. Powers, D. Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness and Correlation. *J. Mach. Learn. Technol.* **2011**, *2*, 37–63.
66. Wojke, N.; Bewley, A.; Paulus, D. Simple Online and Realtime Tracking with a Deep Association Metric. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 3645–3649.
67. Chui, C.K.; Chen, G. Kalman Filtering with Real-Time Applications. In *Springer Series in Information Sciences*; Springer: Berlin/Heidelberg, Germany, 1987. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.