PROGRESS IN ENGINEERING APPLICATION AND TECHNOLOGY
e-ISSN: 2773-5303

PEAT

Vol. 0 No. 0 (YEAR) 1-6
https://publisher.uthm.edu.my/periodicals/index.php/peat

Manual Process Operational Efficiency Monitoring System Utilizing Camera Vision in The Production Lines Environment

[1] *Department of Electrical Engineering Technology, Faculty of Engineering Technology, Universiti Tun Hussein Onn Malaysia, 84600, Pagoh, Johor, MALAYSIA*

*Corresponding Author: amirul@uthm.edu.my
DOI: https://doi.org/10.30880/peat.2024.00.00.000

Article Info

Received: 27 June 2024
Accepted: Day Month Year
Available online: Day Month Year

Keywords

Camera Vison Technology, Operational Efficiency, Real-time Monitoring, OpenCV, Python Programming, System GUI.

Abstract

The "Manual Process Operational Efficiency Monitoring System Utilizing Camera Vision in Production Lines Environment" project addresses inefficiencies in traditional human inspection by employing advanced camera vision technology. Aimed at enhancing operational efficiency, minimizing human error, and improving production quality, the project uses Python, OpenCV, Tkinter, and MediaPipe to develop a real-time monitoring GUI. Utilizing a Logitech C920 webcam, the system captures and analyzes production activities, providing accurate and comprehensive data. The research objectives include boosting productivity by increasing line speed, reducing costs through lower labor expenses and material waste, and enhancing traceability with detailed data capture. Results show significant improvements in efficiency and accuracy, with a notable reduction in human error and increased consistency in production quality. Detailed data collection supports proactive decision-making, and optimizing production processes. This project demonstrates that integrating camera vision technology can revolutionize monitoring systems, setting new standards for manufacturing efficiency and productivity.

1. Introduction

In today's fast-paced and competitive manufacturing world, excelling in operational efficiency is crucial. Businesses continuously seek innovative methods to enhance production lines, reduce costs, and improve quality. One promising approach is incorporating camera vision technology into production monitoring systems. Traditional methods involving human inspection and measurement face challenges such as time consumption, inconsistent accuracy, and difficulties in handling fast or repetitive tasks.

Camera vision technology addresses these challenges by automating the monitoring process, thus revolutionizing the production environment. The "Manual Process Operational Efficiency Monitoring System Utilizing Camera Vision in Production Lines Environment" project aims to introduce an advanced camera vision system to production lines, thereby enhancing operational efficiency. These systems capture, process, and analyze data from production lines in real-time, providing high accuracy and speed. This technology is adept at identifying defects, precisely measuring products, and monitoring material movement with high speed and accuracy. By automating these tasks, companies can achieve significant improvements in operational efficiency, reduce waste, and ensure consistent product quality. Traditional monitoring methods in production lines are plagued by inefficiencies such as high time consumption, lack of accuracy, and difficulties in handling repetitive tasks, leading to increased costs, higher error rates, and inconsistent production quality. There is a need for a more efficient and reliable monitoring system that can address these challenges. Only 17% of factories have adopted fully automated systems, with 47% using semi-automated systems and 36% using manual systems, leading to human limitations such as fatigue, boredom, distraction, and bias that affect accuracy and reliability. These issues can cause missed defects, incorrect data, and inconsistent performance, leading to costly, time-consuming, and labor-intensive processes, especially for large-scale and complex production lines [1].

The primary objectives of this project are to design a system that tracks the efficiency of employee assembly of the product, implement real-time monitoring and alerts to ensure timely task completion while analyzing employee movements, develop a graphical performance dashboard that visually represents employee movements and task completion data over time to track performance trends and analyze the movement graph to identify bottlenecks or inefficiencies in the workflow, providing real-time monitoring with alerts. This project focuses on the development and implementation of a camera vision-based monitoring system for production lines, leveraging Python programming, OpenCV, Tkinter, and MediaPipe to develop a sophisticated graphical user interface (GUI) for real-time monitoring and data collection, utilizing a Logitech C920 webcam to capture and analyze production line activities, ensuring accurate and comprehensive data collection.

1.1 Camera Vision Technology

Camera vision, or computer vision, is a field that integrates computer science and engineering to help machines understand and interpret visual information, aiming to mimic and enhance human visual perception through images or videos. This involves developing algorithms and technologies enabling machines, particularly computers, to analyze and derive meaningful insights from visual inputs. In manufacturing, camera vision aids in 3D reconstruction, quality inspection, defect detection, and process control. Image-based 3D reconstruction creates 3D models from 2D images, while quality inspection ensures product conformity by comparing as-built models with specifications. Defect detection identifies flaws on surfaces by analyzing color, texture, shape, or geometry. Defect detection can be performed using camera vision by analyzing the color, texture, shape, or geometry of the images as shown in Fig 1 that detect different colors on the conveyor belt [2, 3].

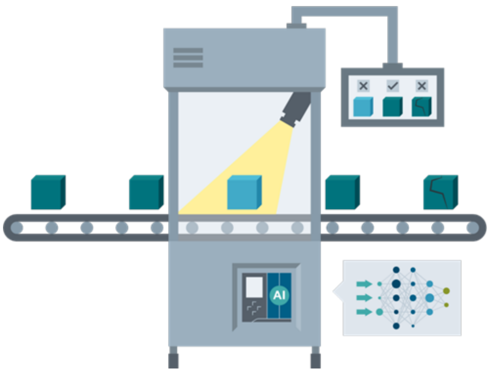

Fig 1 *Color detection using camera vision [4]*

Additionally, camera vision monitors production parameters like temperature and speed, and supervises operators' tasks, providing feedback to guide machines or robots in actions such as positioning and assembly. Motion detection systems track operators' activities, and technologies like face and eye tracking offer tools for vision screening by recording eye movements and gaze points, aiding in diagnosing visual problems. The system processes raw eye-tracking data, performs statistical analysis, and generates graphical interfaces for comprehensive evaluations. Camera vision systems can be marker-based or marker-less, with the latter relying on algorithms to track body parts without physical markers [5]. Various cameras, including depth-sensing and RGB-D cameras like Microsoft Kinect, enhance motion analysis. AI smart cameras, such as the Adlink Neon-2000 series, combine image capture and AI operations, optimizing automation, quality control, and operational efficiency in manufacturing processes [6]. The effectiveness of camera vision depends on factors like camera distance, gait direction, and resolution, which significantly influence accuracy. Integrating these technologies into manufacturing enhances automation, quality control, and overall operational efficiency.

1.2 Monitoring System

Monitoring systems are integral to industrial manufacturing, particularly in production lines, where they enhance processes and streamline operations. These systems conduct quality checks by measuring various parameters, identifying defects, and ensuring products meet high standards. They provide valuable data for resolving issues, accelerating workflows, and optimizing resource utilization. There are two primary approaches to monitoring:

manual and automatic. The manual method involves operators performing visual inspections, manually inputting data, and using tools for meticulous assessments. In contrast, the automatic approach employs advanced technologies such as sensors and cameras to perform real-time quality control, reducing the need for constant manual intervention. This dual methodology combines human expertise with technological innovation for comprehensive monitoring. Automatic monitoring systems collect real-time data without direct operator involvement, including quality control checks, machine health monitoring, and data analytics [7, 8]. These systems offer numerous benefits, such as continuous observation, early fault detection, reduced downtime, and enhanced productivity. They provide real-time insights for process optimization, improve product quality, and enable cost reduction. Predictive maintenance is another advantage, allowing for the strategic scheduling of maintenance activities to prevent unexpected breakdowns [9].

In assembly processes, monitoring systems ensure every step contributes to high-quality production. They oversee visual inspections, cycle times, and feedback loops, ensuring timely adjustments and maintaining efficiency. Cycle time, or standard time, is crucial for gauging production line efficiency and resource allocation. Manual methods for measuring cycle time involve video recordings and analysis with tools like stopwatches, while automatic methods use camera vision for precise, objective measurements. Calculating standard time involves determining normal time, allowances, and performance ratings. Takt time, another important metric, aligns production rates with customer demand, helping design efficient assembly lines and implement pull systems. By integrating these methodologies, monitoring systems enhance automation, quality control, and overall operational efficiency in manufacturing processes [10, 11].

2. Methodology

Fig 2 illustrates the block diagram for the monitoring system using the camera vision project system, divided into three main parts: sensing, mechanical, and software. The sensing part consists of a Logitech C920 Pro HD Webcam mounted on a tripod, capturing real-time footage of the operator on the assembly line and transmitting the video data via USB to a laptop in the mechanical part. This laptop, running PyCharm with Python 3.12, serves as the interface between the webcam and the software application. The software part involves a Graphical User Interface (GUI) developed in PyCharm, which, when activated, starts the webcam and begins the data collection process. The GUI allows users to monitor data in real time and manage the collected data, facilitating its download into Excel for further analysis. This setup ensures efficient monitoring and analysis of the operator's performance, ultimately aiding in productivity improvements and quality control.

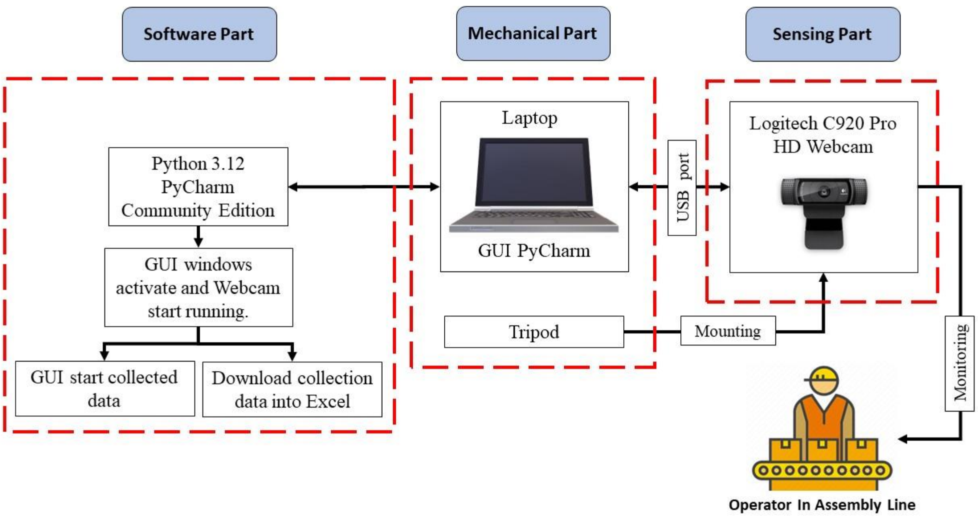

Fig 2 *System Block Diagram*

2.1 System Flowchart

The operational efficiency monitoring system for production lines, as depicted in Fig 3, utilizes camera vision and is initiated by running a pre-configured program in PyCharm. The initial steps involve starting the system, executing the Python code, and activating the GUI windows. The user then sets the standard parameters, such as the expected finish time and standard task time, by pressing the "Set Standard" button. These benchmarks are crucial for measuring efficiency. Once the standards are set, the user presses the "Start" button, activating the system's camera vision to detect hand gestures. The detection of a hand gesture triggers the monitoring of the assembly process. The system assumes the operator is starting to assemble a product, and it begins counting the number of finished products and recording the assembly time for each piece. If no hand gesture is detected, the system records the duration of this out-of-range state, helping to identify idle time or interruptions in the assembly process.

The monitoring process continues until the "Stop" button is pressed, at which point the system ceases all data collection activities. The results, including the efficiency percentage, are displayed in the GUI windows for immediate review. For detailed analysis, the user can press the "Show Data" button to view a table of recorded data or the "Show Graph" button to visualize the hand detection data graphically. Additionally, the system allows the user to download out-of-range durations and assembly time data into a Microsoft Excel file for further analysis and record-keeping. Finally, pressing the "Reset" button returns the system to its initial state, ready for a new monitoring session. This comprehensive flow ensures continuous monitoring and recording of operational efficiency, providing actionable insights for enhancing production line performance.

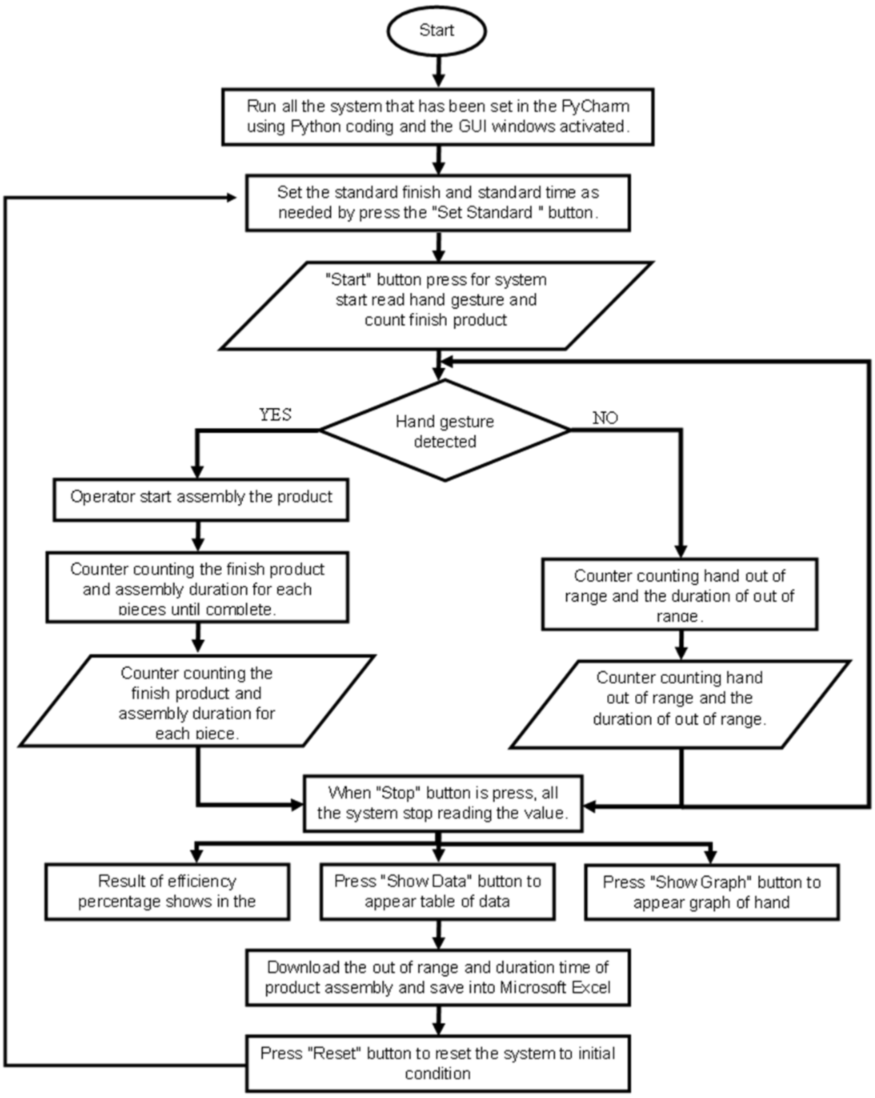

Fig 3 *System Flowchart*

2.2 Experimental Setup

The experimental setup is designed to monitor an assembly operator's workstation effectively, providing a comprehensive view for real-time observation and subsequent analysis. As depicted in Fig 4, a Logitech C920 webcam mounted on a tripod is positioned optimally to fully capture the workspace without any blind spots. The webcam is connected to a laptop featuring a graphical user interface (GUI), enabling both live monitoring and video recording of the operator's activities. This nearby laptop serves as the control hub, allowing the supervisor to monitor the operator's performance in real-time. The laptop's GUI facilitates real-time observation and enables the supervisor to set standard values for product specifications and time requirements for daily tasks, allowing for immediate adjustments and feedback. This ensures the operator's performance aligns with the set benchmarks. Additionally, the recorded footage provides valuable data for in-depth analysis of the operator's efficiency, ergonomics, and potential workflow issues.

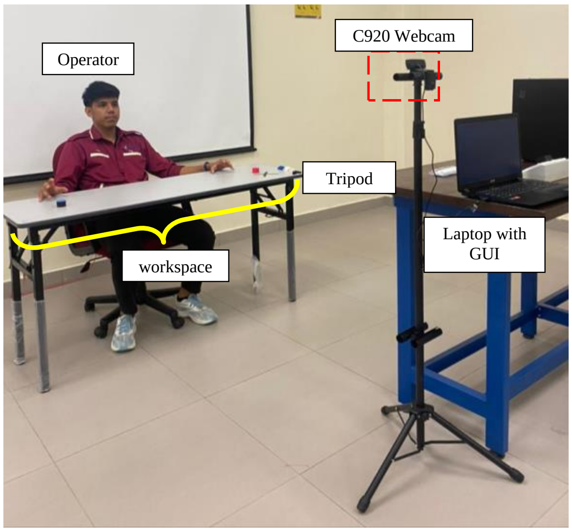

Fig 4 *Experimental Setup*

2.3 GUI Windows Design of Operational Efficiency Monitoring System Using Camera Vision in Production Lines

The GUI of the operational efficiency monitoring system for production lines, implemented in Python using PyCharm 3.12 Community Edition, integrates several essential libraries: OpenCV, MediaPipe, Matplotlib, Tkinter, and Datetime as shown in Fig 5. OpenCV captures and processes real-time video frames for hand gesture detection, while MediaPipe enhances this functionality with advanced hand gesture recognition using machine learning models. Matplotlib generates detection graphs that visually represent hand gesture detection status over time. Tkinter manages the GUI layout, including buttons, labels, tables, and the video display area. DateTime handles time-related data, recording operation times and calculating task durations.

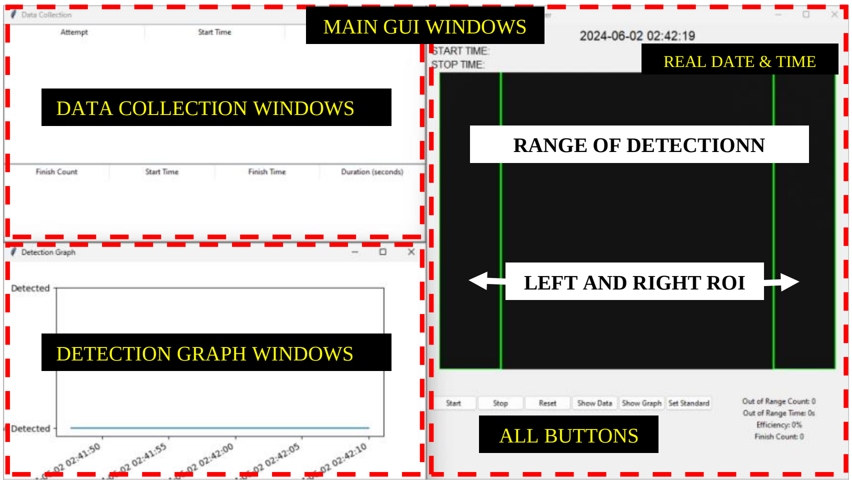

Fig 5 *Main GUI Windows*

The GUI comprises distinct sections: a video display with visual cues for hand placement, a data collection table for logging task details such as attempt number, start time, finish time, and task duration, and a detection graph section that plots detection statuses once the system begins tracking hand gestures. Before the "Start" button is pressed, the system is in standby mode. The GUI also displays key metrics, including Out of Range Count (tracking the number of times the operator's hand is outside the detection area), Out of Range Time (accumulating the total time the hand is out of range), Efficiency percentage (calculated based on detection data), and Finish Count (representing the number of completed tasks and detected hand gestures). This comprehensive interface facilitates real-time monitoring and detailed analysis of operator performance, aiding in productivity improvements and quality control.

The GUI includes six functional buttons as shown in Table 1:

Table 1 *Buttons and Functional*

No.	Types of Buttons	Function
1.	Start	This button initiates the hand gesture detection process, capturing the current time as the start time and beginning to process video frames for gesture recognition.
2.	Stop	Pressing this button stops the detection process and records the current time as the stop time, ending the data collection for that session.
3.	Reset	This button clears all collected data, resets the GUI elements to their initial states, and prepares the system for a new session.
4.	Show Data	When clicked, this button populates the Data Collection table with the recorded data from the current or previous sessions.
5.	Show Graph	This button displays the detection graph, allowing users to visualize the detection status over the operation period.
6.	Set Standard	This button allows the user to define the standards for hand gestures that need to be detected, such as specific gestures that signify the start or end of a task.

3. Results and Discussions

This chapter presents the results of the system functionality tests, hand gesture detection accuracy, system alert features, and operator efficiency analysis. The data collected during the experiments are analyzed to evaluate the performance of the "Manual Process Operational Efficiency Monitoring System Utilizing Camera Vision in Production Lines Environment." The analysis focuses on the finish count of products, average time, efficiency, and instances of hand not detected with time cumulative.

3.1 System Functionality Test

The system functionality test was conducted to assess the overall performance of the camera vision system. The system's ability to accurately monitor and record the production process was evaluated through several test runs. Each test was designed to simulate real-world production conditions, ensuring that the results are representative of hand detection and actual operational efficiency by 4 volunteers. Each volunteer needs to assemble 5 sets of Lego Bricks in 30 seconds. Besides that, hand gesture detection is a critical component of the system, enabling the monitoring of operator actions and ensuring accurate data collection. The detection accuracy was measured by comparing the detected gestures with the actual gestures performed by the operators. Table 2 presents the results of the hand gesture detection tests, the average time taken for each task and the efficiency of volunteers. The operator efficiency analysis was conducted by measuring the average time taken to complete tasks, the efficiency percentage, and the instances where hands were not detected. The following equations and Table 2 summarize the findings.

$$\boldsymbol{Actual\ Ratio = \frac{Actual\ Finish\ Product}{Actual\ Time\ (second)}} \qquad \boldsymbol{Actual\ Ratio = \frac{Actual\ Finish\ Product}{Actual\ Time\ (second)}}$$

Table 2 *Table Of Data Analysis*

Volunteer	Average Time Taken	Out-of-Range Detection Attempt	Cumulative time out of range	Actual Ratio	Efficiency
1.	10.058s	2	0.19s	0.08	96.82%

2.	10.166s	1	0.33s	0.070	84.81%
3.	14.988s	7	1.47s	0.048	57.88%
4.	5.48s	10	6.47s	0.132	159.25%

From Table The evaluation of the hand gesture detection system within the Manual Process Operational Efficiency Monitoring System was conducted using data from four volunteers, each tasked with assembling five sets of Lego Bricks. The system tracked and recorded instances of hand gestures going out of the detectable range, with results summarized in Table 2. Volunteer 1 had 2 out-of-range detections totaling 0.19 seconds, indicating high detection accuracy with minimal time wasted. Volunteer 2 had 1 out-of-range detection totaling 0.33 seconds, demonstrating good hand movement efficiency. Volunteer 3 experienced 7 out-of-range detections amounting to 1.47 seconds, suggesting potential interruptions needing optimization. Volunteer 4 had the highest out-of-range detections with 10 instances totaling 6.47 seconds, indicating a significant need for improvement.

In terms of average assembly times, Volunteer 1 averaged 10.058 seconds per set, while Volunteer 2 averaged 10.166 seconds per set. Volunteer 3 had a significantly longer average time of 14.988 seconds, with a notable delay during the third set. In contrast, Volunteer 4 demonstrated the highest efficiency, averaging 5.48 seconds per set. Efficiency, calculated based on standard and actual ratios, was categorized as poor (<50%), good (<80%), and best (>80%). Volunteer 1 achieved an actual ratio of 0.08 with 96.82% efficiency, Volunteer 2 had an actual ratio of 0.070 with 84.81% efficiency, Volunteer 3 had an actual ratio of 0.048 with 57.88% efficiency, and Volunteer 4 achieved an actual ratio of 0.132 with 159.25% efficiency.

Overall, the first, second, and fourth volunteers demonstrated exceptional performance with best efficiency levels, while the third volunteer showed adequate performance with good efficiency but room for improvement. This analysis helps identify proficiency and areas needing enhancement, guiding supervisors to take appropriate actions to improve overall operational efficiency.

3.2 System Alert Features

The System Alert feature is an essential component of the Manual Process Operational Efficiency Monitoring System, providing real-time alerts to users or supervisors when an operator's hand is not detected within a specified time frame during tasks. As shown in Fig 6, the warning appears prominently in the middle of the main GUI window for easy visibility. When the system fails to detect any hand gestures for a continuous duration of 60 seconds, it triggers a "WARNING: No detection!" alert on the main interface. This notification indicates that the operator may not be within the workspace range or their gestures are not being recognized. The primary objective of this warning is to remind the operator to return to the designated area or adjust their gestures to fall within the detectable range. This feature is crucial for maintaining operational efficiency and ensuring continuous engagement of the operator within the workspace. By minimizing downtime and alerting users to potential gesture detection issues, the system enables timely corrective actions, making it vital in environments requiring consistent monitoring and interaction.

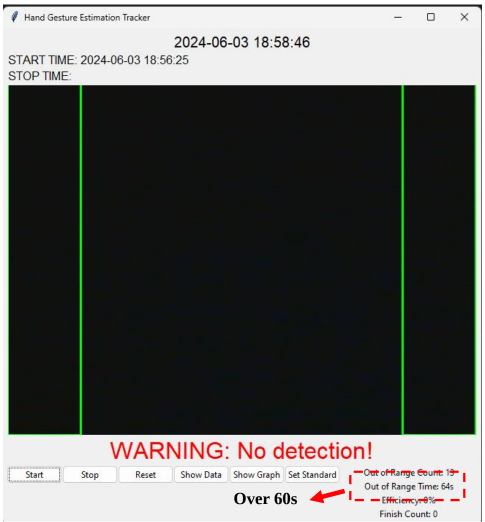

Fig 6 Warning Alerts Popup in The Main GUI Windows

4. Conclusion

In conclusion, the project successfully met all objectives by developing and implementing an effective Manual Process Operational Efficiency Monitoring System utilizing camera vision in a production line environment. The system's real-time monitoring and data analysis capabilities proved valuable in identifying and addressing inefficiencies in the production process, with recommendations provided to further enhance performance and broaden application, ensuring continuous improvement in operational efficiency.

The first objective of tracking employee assembly efficiency and implementing real-time monitoring and alerts was achieved using camera vision technology to monitor task completion times and provide real-time alerts, enhancing operational efficiency. The system's capability to track hand gestures allowed for precise monitoring and accurate recording of product completion times, overcoming initial challenges by expanding the region of interest. The second objective of developing a graphical performance dashboard was met by processing captured data using Python programming to generate graphical representations of employee movements and task completion data, with a user-friendly GUI window enabling easy visualization and analysis. This feature pinpointed areas for operator improvement, enhancing overall productivity. The third objective of analyzing movement graphs to identify bottlenecks and inefficiencies was accomplished through data analysis capabilities, providing actionable insights and enabling immediate corrective actions. The system's real-time feedback on efficiency percentages and the ability to download and save data into Microsoft Excel facilitated further analysis and reporting, ensuring accuracy and relevance for performance reviews.

Recommendations for future improvements include enhancing hardware capabilities by upgrading RAM and camera frame rates to reduce delays and improve gesture recognition accuracy, and improving lighting conditions to ensure reliable performance of the monitoring system. The project successfully developed a camera vision-based system for monitoring employee efficiency in production lines, achieving real-time tracking, data analysis, performance visualization, and facilitating ease of data export and review, thereby identifying inefficiencies and providing actionable insights to evaluate operator performance.

Acknowledgement

Communication of this research is made possible through monetary assistance from Universiti Tun Hussein Malaysia and the UTHM Publisher's Office via Publication Fund E15216.

References

[1] Jacqueline, "Semi-Automated Material Handling Systems 101 | REB Storage," REB Storage Systems International, 25 Jan 2022. [Online]. Available: https://rebstorage.com/articles-white-papers/semi-automated-material- handling-systems/.

[2] D. Lee, G.-Y. Nie and K. Han, "Vision-based inspection of prefabricated components using camera poses: Addressing inherent limitations of image- based 3D reconstruction," *J. Build Eng.,* vol. 64, p. 105710, 2023

[3] A. Sioma, "Vision System in Product Quality Control Systems," Applied Sciences, vol. 13, p. 751, Jan. 2023.

[4] U. Dag, "Vision Defect Detection Systems," Plugger.ai, 2023. [Online]. Available: https://www.plugger.ai/blog/vision-defect-detection-system.

[5] T. Hellsten, J. Karlsson, M. Shamsuzzaman and G. Pulkkis, "The potential of computer vision-based marker-less human motion analysis for rehabilitation," *Rehabil. Proc.outcome,* vol. 10, 2021.

[6] C. Vavra, "AI vision for monitoring manufacturing and industrial environments," Control Engineering, 06 October 2021. [Online]. Available: https://www.controleng.com/articles/ai-vision-for-monitoring-manufacturing- and-industrial-environments/.

[7] T. Fuerties, C. Mozzati, F. Peysson, A. Alzaga and M. Armendia, "Data Monitoring and Management for Machine Tools," *Twin-Control, Cham: Springer International Publishing,* pp. 123-136, 2019

[8] J.-W. Kim and S. Chi, "Multi-Camera vision-based productivity monitoring of earthmoving operations," *Automation in Construction,* vol. 112, Apr. 2020.

[9] Y. Gua, Y. Sun and K. Wu, "Research and Development of Monitoring System and Data Monitoring System and Data Acquisition of CNC machine tool in intelligent manufacturing," Int. J.Adv.Robot. Syst, vol. 17, 2020.

[10] A. R. Walker, D. Maeda and J. Acharya, "Lightweight video analytics for cycle time detection in manufacturing," *2021 IEEE i=International Conference on big Data (Big Data),* Dec. 2021.

[11] M. Faccio, E. Ferrari, Francesco Gabriele Galizia, M. Gamberi and F. Pilati, "Real-time assistance to manual assembly through depth camera and visual feedback," Procedia CIRP, vol. 81, pp. 1254-1259, Jan. 2019.