



Bridging the Gaps: A Flexible Automated Object Classification System for PLC, Machine Learning, and Industry 4.0 Education

CHIA-YING KUO¹, YE BAI², CHENG NIU¹, SHENG-JEN HSIEH^{1,3*}

¹Department of Electrical & Computer Engineering, Texas A&M University, College Station, TX 77843, USA

²Department of Mechanical Engineering, Texas A&M University, College Station, TX 77843, USA

³Department of Engineering Technology & Industrial Distribution, Texas A&M University, College Station, TX 77843, USA

*hsieh@tamu.edu

Abstract: Industry 4.0 manufacturing increasingly relies on integrated cyber-physical systems. However, lab experiences in many college classrooms often present vision and controller subsystems as closed appliances. This can limit the ability of students to observe, validate, and troubleshoot behavior from start to finish. This paper presents a laboratory module and instructional station deployed in the classroom that connects computer vision inference in real time to programmable logic controller (PLC) actuation through a deliberately transparent supervisory interface. The platform integrates a standard webcam, a VB.NET inference application using transfer learning, an Excel workbook enabled by macros for explicit mapping of labels to commands, and PLC ladder logic that drives discrete indicator outputs.

The module was delivered over six sessions of two hours each as part of a course on manufacturing automation and robotics. Engineering Technology students worked in small teams, and most achieved the intended closed-loop behavior. Real variations in capture conditions—such as lighting, clutter, pose, and occlusion—provided an immediate basis for debugging across every layer of the system. The results indicate that the module is feasible for delivery within typical laboratory constraints. Furthermore, the module helps learners develop a system-level understanding of how perception is integrated with control, using interface observability as a central design principle.

Keywords: PLC education, Industry 4.0 education, machine vision education, perception to control, systems integration pedagogy

© 2026 under the terms of the J ATE Open Access Publishing Agreement

Introduction

Advanced manufacturing is an important part of the U.S. economy. According to a 2026 U.S. Census Bureau report, total goods exported during 2024 and 2025 were \$174 billion and \$185.7 billion, respectively, representing a 6.3% increase [1]. As the manufacturing sector becomes more automated and continues moving toward Industry 4.0 technologies, this upward trend is expected to continue. A 2024 Deloitte report on the manufacturing industry outlook suggests there will be a shortage of 1.9 million employees in the manufacturing workforce by 2033 [2].

A programmable logic controller (PLC) is a solid-state control system with a user-programmable memory. The PLC reads input conditions and sets output conditions to control a machine or process. PLCs are widely used for applications such as monitoring security, managing energy consumption, and controlling machines and automatic production lines. PLCs are said to be among the most ingenious devices ever invented to advance the field of manufacturing automation [3]. Understanding how PLCs work and how to integrate PLCs with I/O devices is fundamental and essential for engineers and technicians in the industrial



automation and control field. However, the knowledge domain is complex and multifaceted, requiring lab exercises that help learners visualize abstract and complex concepts. Hsieh employed instructional technologies such as an intelligent tutoring system, simulation, and animation tools in Virtual PLC, a web-based system that facilitated learning about PLC instructions and programming concepts [4]. The concept was later expanded to cover interfacing with I/O devices and robot controllers in a system called ASI Tutor [5, 6]. The evaluation of ASI Tutor took place over several semesters and involved faculty at multiple institutions with positive results [6]. To provide hands-on, just-in-time, and experiential learning in the classroom, Hsieh also presented the idea of a portable PLC kit as a companion tool for instructors and students [7]. The kits were built and evaluated in the classroom for several semesters with positive learning outcomes.

One major challenge is that today's manufacturing work requires engineers who can connect Information Technology (IT), Operational Technology (OT), and Artificial Intelligence/Machine Learning (AI/ML). In other words, companies need a "T-shaped" engineer who has a strong technical focus but also understands digital tools and system integration [8, 9]. However, in many undergraduate programs, these skills are not taught together; often, they are separated across different courses and departments. As a result, it has been estimated that around 27% of manufacturers are unable to expand production because they cannot hire enough properly trained workers [10]. This gap is becoming more visible as ML is increasingly embedded in manufacturing workflows.

Machine learning (ML) is central to this shift because modern manufacturing increasingly relies on connected sensors, continuous data collection, and automated decision-making. Across recent Industry 4.0 reviews, ML is consistently framed as the layer that turns high-volume industrial data into actionable information for operational decisions and process improvement [11, 12]. Importantly, ML's impact in manufacturing is not limited to "model accuracy"; it is directly tied to concrete functions such as inspection, predictive maintenance, and optimization, which explains its sustained prominence in both research and practice [11, 13]. In many cases, the main barriers are often adoption and implementation rather than novel algorithms, emphasizing the need for deployable and reliable solutions in real production settings [13]. This aligns with "Industrial AI" perspectives that argue real value depends on integrating analytics with data infrastructure and domain knowledge, not treating ML as a standalone modeling exercise [14].

The economic and labor-market implications of AI/ML adoption further increase the urgency of workforce preparation. Studies suggest AI adoption does not necessarily reduce total jobs, but it shifts work by automating some tasks while augmenting others, creating stronger demand for reskilling and integration-oriented roles [15-18]. Evidence also indicates an initial transition period where firms may experience temporary performance dips before realizing productivity gains as organizational changes stabilize [19]. Industry surveys also highlight difficulty demonstrating return on investment (ROI) and workforce constraints as major barriers to smart manufacturing deployment, reinforcing that the limiting factor is often the availability of workers who can connect data-driven methods to operational control and decision-making [16].

In this context, the skill set demanded of future engineers is increasingly described as an integration capability across IT/OT boundaries (data interfaces, real-time workflows, system verification, and cybersecurity awareness), rather than ML theory alone [20]. However, guidance on taking ML solutions from ideation to deployment remains limited, which suggests that "how to deploy and integrate" is still a missing piece for both practitioners and learners [21].

Object detection is an important application of machine learning in manufacturing. However, many automated object detection systems are expensive and not sufficiently flexible for use in education [22, 23]. Many are proprietary solutions that require specific cameras and software. Students can learn how to operate such systems, but using these systems to teach core theory—such as how a vision model works, how signals are communicated, and how the logic can be implemented with industrial controllers—can be



challenging. At the same time, industry needs engineers who can integrate high-level software (computer vision and AI) with low-level industrial hardware such as Programmable Logic Controllers (PLCs) [24].

To meet this need, recently, there have been efforts to create learning environments that connect data-driven methods to industrial automation workflows [25, 26]. Low-cost smart manufacturing toolkits and open-source IoT/CPS laboratories aim to integrate sensing, control, and analytics so students can experience end-to-end system integration instead of siloed instruction [25]. Traditional PLC laboratories are also being extended with Internet of Things (IoT) tooling and modern data interfaces to expose students to networking, monitoring, and cross-layer integration skills that mirror Industry 4.0 practice [26, 27]. Learning-factory environments provide realistic cyber-physical contexts and team-based integration experiences, but they can be expensive and operationally complex to scale across institutions [28, 29]. Virtual and remote labs offer a complementary path for access and scalability when physical equipment is limited, but trade off some aspects of hands-on integration with real hardware [30, 31]. Despite these advances, a persistent gap remains between ML instruction and industrial control training: students may learn ML workflows or PLC logic, but they have relatively few opportunities to practice deploying ML outputs into PLC-centered automation and verifying end-to-end behavior under realistic constraints [15, 21].

To address these problems, we believe there is a strong need for a low-cost automated object classification system designed for PLC, Machine Learning, and Industry 4.0 education. With this proposed platform, students can practice the full pipeline—from collecting data and training a basic model to connecting the model output into PLC logic. A well-designed hands-on experience can help students build real integration skills and also improve their confidence and problem-solving ability for working in smart factories in the future [23, 24].

Methods

We developed an instructional platform that connects computer vision to industrial control in a single, continuous workflow. The system links real-time image classification to PLC actuation, giving students hands-on practice with integrating IT and OT within a cyber-physical environment. The setup includes four main parts: (i) a standard webcam, (ii) a VB.NET application that classifies images as they are captured, (iii) an Excel workbook that uses macros to exchange data between the software and hardware, and (iv) an Allen-Bradley ControlLogix PLC that turns those signals into physical light outputs. As shown in Fig. 1, the signal path is designed to be transparent so students can observe and troubleshoot every step of the process.

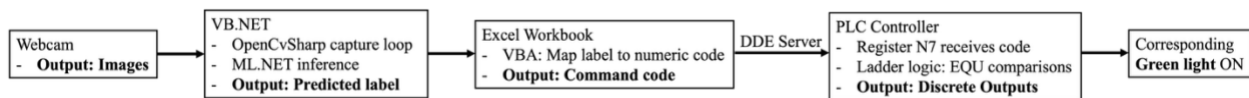


Fig. 1. System architecture.

Machine-learning pipeline

For the classification piece, we used ML.NET to implement transfer learning, leveraging a pre-trained TensorFlow Inception network as our feature extractor (Fig. 2). To keep the training process manageable for a lab setting, we kept the Inception backbone frozen and only trained a lightweight multiclass classifier on top of it.

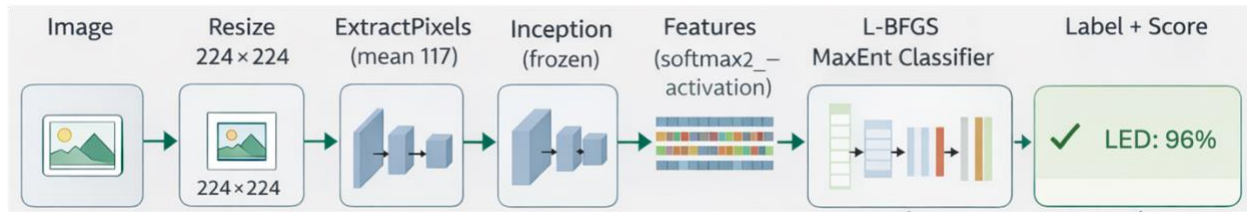


Fig. 2. Inception-based transfer learning pipeline.

We prepared the training data by indexing image paths and labels in TSV files. Before the images enter the model, they are resized to 224×224 and normalized with a mean offset of 117 to ensure they match the backbone's expectations. During training, the frozen model generates feature representations from the "softmax2" layer, which an L-BFGS classifier then uses to learn how to predict different classes. This approach significantly reduces training time, allowing students to focus their energy on building the dataset and understanding how the model makes decisions. Once deployed, the model provides a predicted label and a confidence score, which provide the necessary data for the supervisory system to trigger the PLC (Fig. 1).

Supervisory interface and vision-to-PLC communication

To link the AI's inference to the PLC without adding the complexity of custom industrial protocols, we chose to use Excel as a transparent bridge between the two (Fig. 1). In this setup, the VB.NET application writes the predicted label to a specific cell in a workbook titled PLC_Link.xlsm. From there, students write or edit a simple Visual Basic for Applications (VBA) routine that translates that text label into an integer command code based on the policy in Table 1. We chose this approach specifically for its transparency, which allows students to actually watch the values change in the spreadsheet in real time, making it much easier to verify that the mapping and timing are working correctly before the signal ever reaches the hardware (Fig. 3).

Table 1. Mapping from predicted class to PLC light outputs

Predicted Class (from VB)	Command written to PLC (N7)	Energized Output address	Physical Indicator
DC_motor	0	Local:9:O.Data.0	Green light 0 ON
Fan	1	Local:9:O.Data.1	Green light 1 ON
Motion_sensor	2	Local:9:O.Data.2	Green light 2 ON
LED	3	Local:9:O.Data.3	Green light 3 ON



```
' 1) Normalize B2 text
s = LCase$(Trim$(CStr(Worksheets("Sheet1").Range("B2").Value)))
isDCmotor = (s = "dc_motor")
isFan = (s = "fan")
isMotionSensor = (s = "motion_sensor")
isLED = (s = "led")

' 2) Decide B5 values
' write your logic here!!!
If isDCmotor Then
    b5 = 0
ElseIf isFan Then
    b5 = 1
ElseIf isMotionSensor Then
    b5 = 2
ElseIf isLED Then
    b5 = 3
Else
    b5 = 4
End If

' 3) Write to cells (for visibility/debug)
Worksheets("Sheet1").Range("B5").Value = b5

' 4) DDE poke both BOOL tags (1/0)
On Error GoTo clean
'Change the "MLproject" below to your topic name
ch = Application.DDEInitiate(app:="RSLinx", TOPIC:="MLproject")
Set ValueToPoke = Worksheets("Sheet1").Range("B5")
Application.DDEPoke ch, "Program:MainProgram.N7", ValueToPoke

Application.DDETerminate ch
Exit Sub
```

Fig. 3. Editable VBA code used in PLC_Link.xlsm to map predicted labels to PLC command codes and transmit via DDE.

Once the command code is generated, the workbook sends it to the ControlLogix PLC using Dynamic Data Exchange (DDE). While DDE is an older standard, it is perfect for the classroom because it is easy to set up and keeps the logic process visible and editable for troubleshooting. In practice, Excel "pokes" the data into RSLinx Classic, which then passes it to the PLC. For example, the code initiates a conversation with RSLinx and then sends the integer value to a specific tag, such as Program: MainProgram.N7. The key to replicating this is ensuring the item string in Excel matches the PLC's tag namespace exactly and that the data types are consistent across both platforms.

PLC logic and actuation mapping

On the PLC side, the received command code is stored in an integer register and interpreted using parallel ladder rungs with equality comparisons (Fig. 4). Each rung compares the register against a fixed class code and energizes the corresponding output coil. This implements a deterministic mapping from the predicted class to the behavior of the physical indicators (Table 1). To ground the exercise in a practical context, we frame the mapping as a scenario for smart home automation. In this setup, the vision system identifies common devices, and the PLC indicates the state of the detected device. Specifically, integer values from 0 to 3 select the indicator outputs for a DC motor, fan, motion sensor, and LED, respectively. Because the final output is a physical light, both misclassifications and faults in integration become immediately visible. This allows for systematic debugging across the entire stack, from perception to the final controller output (Fig. 1).

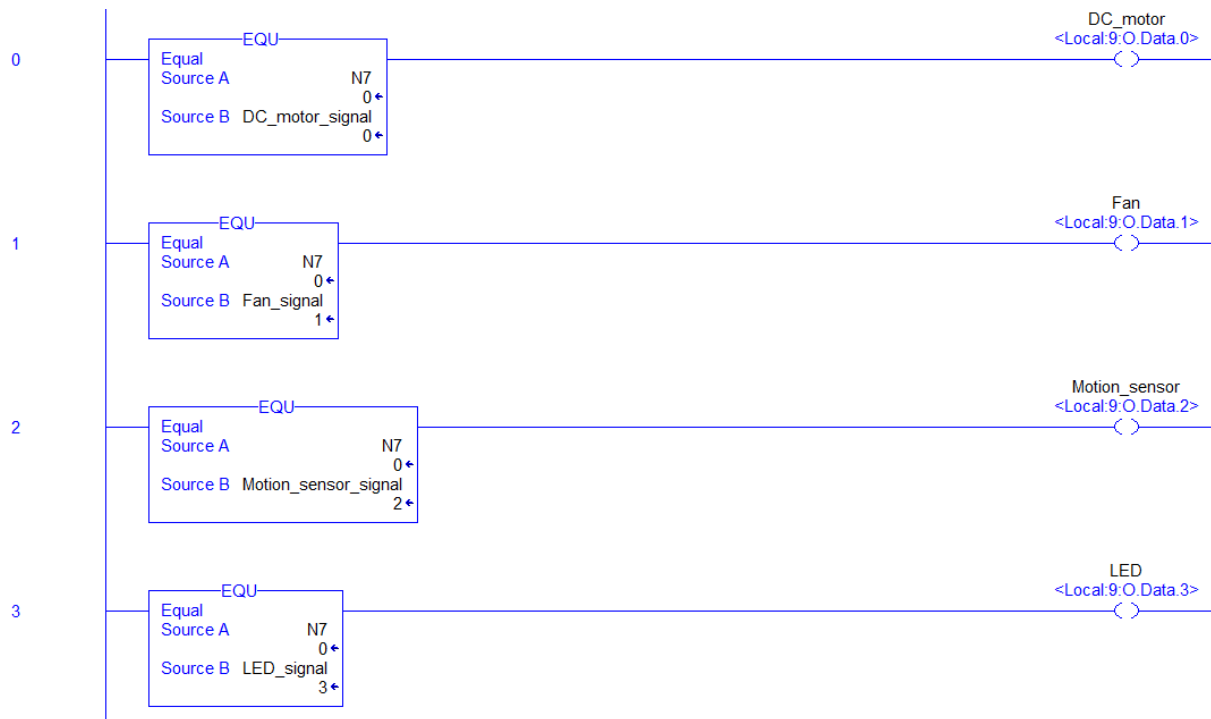


Fig. 4. Ladder-logic structure for mapping vision commands to indicator outputs.

Experiment

The module served as the final laboratory activity for an undergraduate course focused on PLC programming and control; 65 students participated in this activity. Before starting this project, students had already practiced fundamental ladder logic patterns such as comparators and discrete I/O. They had also been introduced to concepts for supervisory communication, including the use of Excel for DDE linkage.

The purpose of this final module was to merge those operational technology skills with a practical information technology pipeline. By using computer vision for classification, students could see exactly how a software inference result is transformed into deterministic controller behavior through explicit logic and interface constraints. We organized the laboratory sequence into three distinct stages: (1) expanding the dataset and training the model offline, (2) performing inference with a webcam in real time while observing the supervisory layer, and (3) achieving closed-loop integration with PLC actuation. The physical topology for these connections is illustrated in Fig. 5.



Fig. 5. Physical connection topology.

Part 1: Dataset expansion and model training

Students expanded a provided starter dataset by adding a class for an LED and then trained and evaluated an image classifier. Each team collected between 8 and 12 images of the LED, either through capture or download, and placed them into the required directory structure before updating the index files for the training pipeline. The training and test sets were defined using two tab-separated files, and teams chose a split for training and testing to examine how data availability and split decisions influence performance.



Because the starter dataset already included baseline categories for a DC motor, fan, and motion sensor, the newly added LED class typically became the smallest and most sensitive category. Students ran the training pipeline and recorded key outputs such as training status messages, LogLoss, and prediction scores. They then documented any misclassification behavior and its likely causes, such as limited diversity in the images, background clutter, or variation in pose. The interface used for training and testing in this stage is shown in Fig. 6.

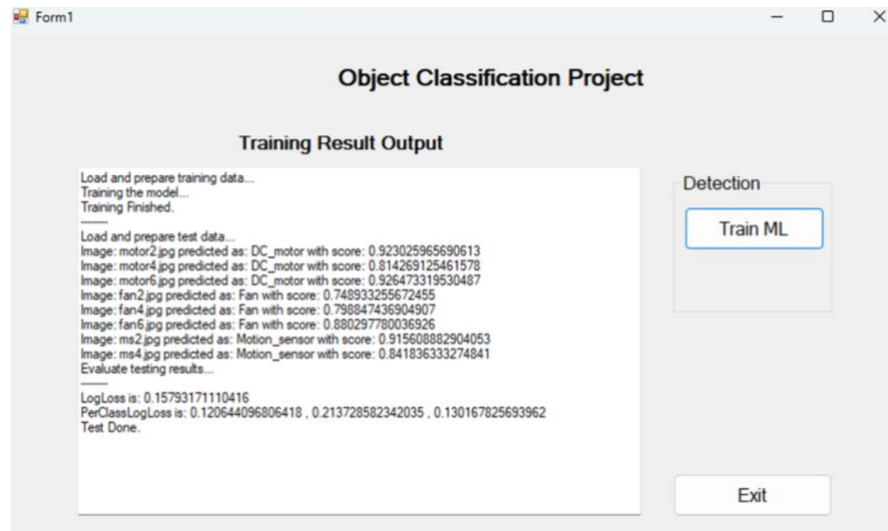


Fig. 6. Interface of the training and testing process.

Part 2: Real-time webcam inference and supervisory observation

Students transitioned from offline evaluation to inference in real time using a webcam feed. They transferred the trained model assets into the project structure for the live application, verified the webcam capture, and initiated the classification process. The application displays the predicted label during capture and writes the result directly to the PLC_Link.xlsxm workbook. Students configured the system by specifying the correct path for the workbook within the VB.NET project. This stage of the lab emphasized sensitivity during runtime; students observed firsthand how variations in lighting, occlusion, and background clutter affect the stability of predictions and the behavior of the scores. The interface for this process is shown in Fig. 7.

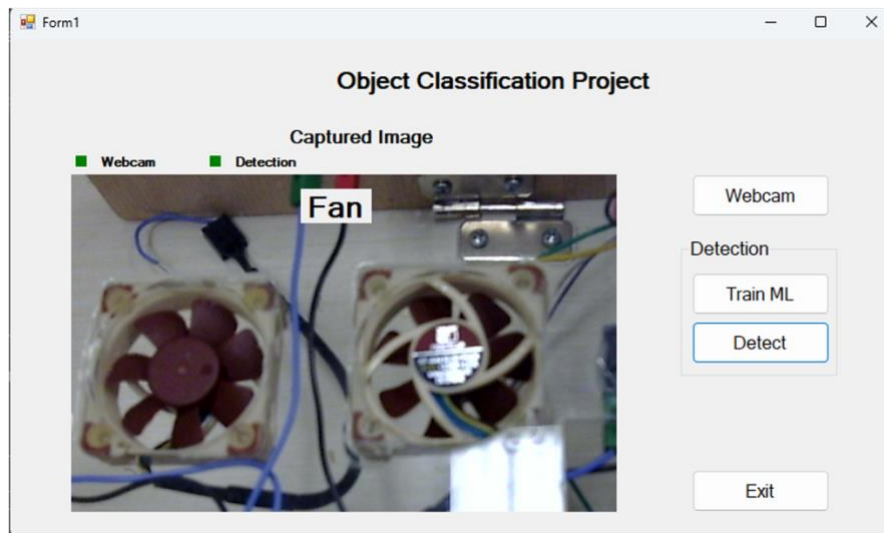


Fig. 7. Interface for webcam capture and object classification process.

Part 3: PLC integration and end-to-end verification

In the final stage, students connected the supervisory layer to the PLC through Excel VBA and DDE. They began by implementing or verifying the ladder logic that reads the incoming integer register and energizes discrete outputs based on equality comparisons for each category (Table 1). Next, students configured the workbook by updating the DDE topic to match their communication setup in RSLinx Classic and modified the VBA mapping section so that label strings were translated into integer commands. Once the configuration was complete, teams validated the closed-loop behavior from start to finish. The predicted label updated in the VB.NET interface, the corresponding integer command appeared in Excel, the PLC register updated online, and the mapped output energized in real time (Fig. 8). In the smart home configuration, each recognized device class drives a distinct green pilot light. This makes the mapping from perception to control directly observable at the operational technology layer and supports systematic debugging if misclassifications or communication issues occur. Fig. 9 illustrates the demonstration setup used for this validation, showing the integrated hardware and the correspondence between the vision output and the PLC states.

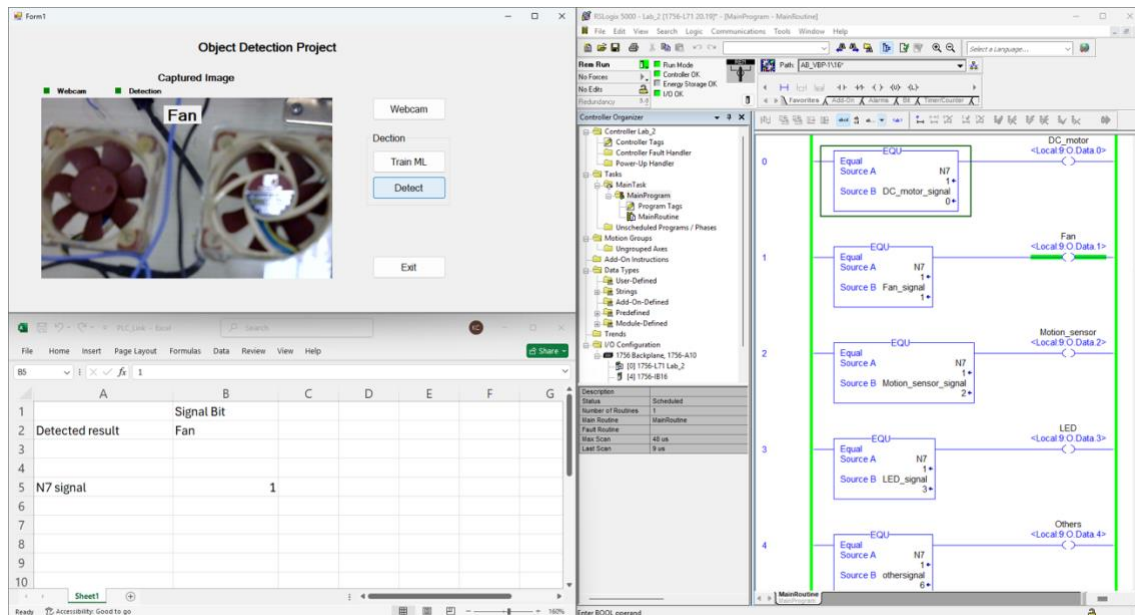


Fig. 8. Real-time inference interface used in the integrated laboratory module. The VB.NET application (upper left) displays the live camera feed and classification results, the Excel supervisory layer (lower left) maps the predicted label to an integer command (N7), and RSLogix 5000 (right) shows the ladder logic executing in real time on the PLC in response to the transmitted command.

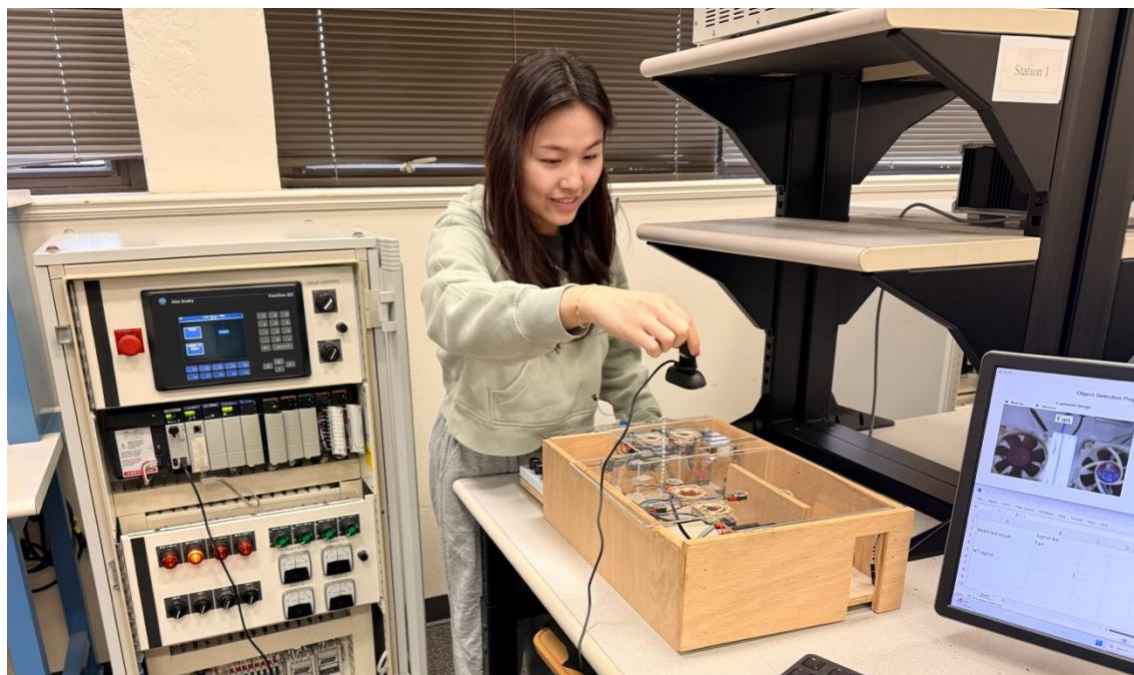


Fig. 9. Classroom demonstration of the end-to-end perception-to-control pipeline. The detected smart-house component is classified in real time and mapped to a PLC-controlled indicator output through the supervisory interface.



Assessment approach

Student completion was assessed through demonstration checks and brief written documentation. Teams were required to show successful results from training and testing, demonstrate classification using the live webcam, and show the PLC actuation driven by those predictions. Students also documented observations aligned with the learning objectives, noting whether the predictions were correct, whether Excel updated in real time, and whether the PLC register matched the transmitted command. A short reflection prompt asked students to describe the workflow from inference to PLC actuation to reinforce their conceptual understanding of the integration chain between information technology and operational technology.

Results and Discussion

The laboratory module was delivered in six sessions, each lasting 2 hours, to approximately 10-12 Engineering Technology students per session, as part of the Manufacturing Automation and Robotics course. Students worked in teams of one or two and successfully completed the entire workflow—from expanding the dataset and training the model to real-time inference and PLC actuation—within the scheduled time. As part of the required extension task, teams added a specific class for an LED to the provided image categories, contributing an average of eight images. Because this added class was relatively small, teams frequently observed that the confidence of predictions was sensitive to capture conditions such as lighting variation, background clutter, object pose, and partial occlusion. In practice, this sensitivity occasionally resulted in incorrect indicator outputs. This provided a concrete reason for teams to expand their data coverage and learn how to interpret model outputs within the context of downstream control.

In the integrated demonstration, nearly all teams achieved the intended closed-loop behavior. The predicted device class selected a unique green indicator output, which made the mapping from perception to control deterministic and directly observable through the PLC ladder logic. Across the six sessions, all but one or two teams achieved a complete closed loop during the lab window. The few incomplete demonstrations occurred toward the end of the semester when the activity was an experimental, ungraded module; in those cases, teams primarily ran out of time to finish configuration and verification steps, such as setting the workbook path or mapping labels to codes, rather than encountering flaws in the underlying approach. Overall, the completion rate suggests the module is feasible within standard laboratory constraints while still requiring meaningful work on system integration.

Beyond the functional completion of the tasks, student feedback highlighted two primary instructional outcomes (Table 2). First, many students noted that this was their first experience implementing a workflow enabled by AI and observing how perception-based machine learning can be incorporated into an automation task driven by a PLC. Second, the platform encouraged structured troubleshooting across different layers. Teams iteratively improved their data collection, validated the Excel and VBA mapping logic, and verified PLC addressing and ladder comparisons until the observed behavior matched the project specifications.

Together, these results suggest that the module helps students build a coherent mental model of the path from perception to control. It also allows them to practice the debugging skills that are essential in a cyber-physical system where IT and OT must work together. Future offerings will strengthen the evaluation process by incorporating standardized rubrics and pre- and post-measures to quantify learning gains and proficiency in troubleshooting.

The lab exercise, PLC program, machine learning code, and VBA code are available via GitHub at <https://github.com/TonyHsiehTAMU/MV-PLC-DDE-ML>.

Table 2. Summary of student feedback themes and instructional implications



Feedback theme	Student feedback	Instructional implication
End-to-end “big picture”	Students reported seeing the full workflow from perception to actuation.	Reinforces system integration and reduces “siloed” learning across IT/OT.
Understanding ML and its role in manufacturing	Students reported gaining a conceptual understanding of what machine learning is and how it can be applied in manufacturing/industrial automation.	Addresses limited AI exposure in department curricula and grounds ML in a familiar control context.
Career relevance/ modern technology exposure	The activity felt directly relevant for jobs and broadened their view of modern industrial technologies.	Strengthens motivation and perceived usefulness: key predictors of engagement and retention.
Integrating 4-year coursework into one mental model	Students reported that the module helped them connect prior topics into a coherent system.	Supports capstone-level integration outcomes and “systems thinking” competence.

Conclusion

This study presented a laboratory module that introduces the convergence of IT and OT by linking computer vision inference in real-time to PLC actuation through a transparent supervisory interface. The module positions AI not as a standalone software topic but as one component of a complete workflow from perception to control. Within this framework, students practice expanding datasets, training models using transfer learning, performing real-time inference, and implementing both supervisory mapping and deterministic controller logic. Delivered over six sessions of two hours each with Engineering Technology students working in small teams, the module reliably produced closed-loop behavior within the laboratory window. Furthermore, the class for the LED added by the students was intentionally small and sensitive to capture conditions, providing a concrete basis for troubleshooting across layers under realistic variability such as lighting, clutter, and pose. Overall, the module supports system-level thinking and debugging focused on integration while remaining feasible within a standard PLC laboratory format.

Future offerings will strengthen both the instructional delivery and the collection of evidence. We plan to incorporate the activity as a graded module with a rubric aligned to program outcomes, covering areas such as dataset quality, supervisory mapping accuracy, and PLC ladder logic. This will be supplemented by brief checks of concepts before and after the lab, along with structured reflection. We will also report quantitative indicators appropriate for technical education, such as image counts per class, confusion patterns for the categories added by students, and the update rate and latency of the system under controlled environmental conditions.

Finally, while Excel and DDE offer a fast and observable bridge for integration in the classroom, the same pedagogical sequence can be migrated to modern industrial communication standards like OPC UA and MQTT without changing the core learning objectives. Students would still observe model outputs, implement an explicit policy to turn labels into commands, and verify deterministic behavior in the controller. In this future pathway, the supervisory bridge would be replaced by an OPC UA client or an MQTT publisher and subscriber pattern interfaced to the PLC. This will enable students to compare the configuration burden, observability, and timing behavior across different protocols while preserving the principle of interface transparency as the central design goal.

Acknowledgements. This work was supported by the National Science Foundation's Advanced Technology Education Program (Award no. 2202201). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the National Science Foundation.

Disclosures. The authors declare no conflicts of interest.



References

- [1] U.S. Census Bureau and U.S. Bureau of Economic Analysis, "U.S. international trade in goods and services, October 2025 (FT900)," U.S. Department of Commerce, Jan. 8, 2026. [Online]. Available: [Census.gov](https://www.census.gov)
- [2] Deloitte, "2024 manufacturing industry outlook," *Deloitte Insights*, Oct. 30, 2023. [Online]. Available: [Deloitte.com](https://www.deloitte.com)
- [3] M. P. Groover, *Automation, Production Systems, and Computer-Integrated Manufacturing*, 3rd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2007.
- [4] S. Hsieh and P. Y. Hsieh, "An Integrated Virtual Learning System for Programmable Logic Controller," *J. Eng. Educ.*, vol. 93, no. 2, pp. 169–178, Apr. 2004. doi: [10.1002/j.2168-9830.2004.tb00801.x](https://doi.org/10.1002/j.2168-9830.2004.tb00801.x)
- [5] S. Hsieh, "Interactive and Web-based Animation Modules and Case Studies for Automated System Design," in *Proc. 2024 ASEE Annu. Conf. Expo.*, Portland, OR, USA, Jun. 23–26, 2024. doi: [10.18260/1-2--47671](https://doi.org/10.18260/1-2--47671)
- [6] S. Hsieh and S. Pedersen, "Design and Evaluation of Modules to Teach PLC Interfacing Concepts," in *Proc. 2023 ASEE Annu. Conf. Expo.*, Baltimore, MD, USA, Jun. 25–28, 2023. doi: [10.18260/1-2--42917](https://doi.org/10.18260/1-2--42917)
- [7] S. Hsieh, "Design, Construction, and Evaluation of Portable Programmable Logic Controller (PLC) Kit for Industrial Automation and Control Education," *Int. J. Eng. Educ.*, vol. 39, no. 4, pp. 823–835, 2023. [Online]. Available: https://www.ijee.ie/latestissues/Vol39-4/05_ijee4351.pdf
- [8] R. Langmann, "Training 4.0 in PLC Education," in *Smart Technologies for an All-Electric Society*, vol. 1661, M. E. Auer et al., Eds. Cham, Switzerland: Springer Nature, 2026, pp. 119–130. doi: [10.1007/978-3-032-07316-7_11](https://doi.org/10.1007/978-3-032-07316-7_11)
- [9] M. Elbestawi, D. Centea, I. Singh, and T. Wanyama, "SEPT Learning Factory for Industry 4.0 Education and Applied Research," *Procedia Manuf.*, vol. 23, pp. 249–254, Jan. 2018. doi: [10.1016/j.promfg.2018.04.025](https://doi.org/10.1016/j.promfg.2018.04.025)
- [10] S. Hsieh, M. Barger, S. Marzano, and J. Song, "Preparing the manufacturing workforce for Industry 4.0 technology implementation," in *Proc. 2023 ASEE Annu. Conf. Expo.*, Baltimore, MD, USA, Jun. 25–28, 2023. doi: [10.18260/1-2--43957](https://doi.org/10.18260/1-2--43957)
- [11] H. Castro, E. Câmara, P. Ávila, M. Cruz-Cunha, and L. Ferreira, "Artificial Intelligence Models: A literature review addressing Industry 4.0 approach," *Procedia Comput. Sci.*, vol. 232, pp. 2197–2206, 2024. doi: [10.1016/j.procs.2024.06.430](https://doi.org/10.1016/j.procs.2024.06.430)
- [12] R. Rai, M. K. Tiwari, D. Ivanov, and A. Dolgui, "Machine learning in manufacturing and Industry 4.0 applications," *Int. J. Prod. Res.*, vol. 59, no. 16, pp. 4773–4778, 2021. doi: [10.1080/00207543.2021.1956675](https://doi.org/10.1080/00207543.2021.1956675)



- [13] D. Mazzei and R. Ramjattan, "Machine Learning for Industry 4.0: A Systematic Review Using Deep Learning-Based Topic Modelling," *Sensors*, vol. 22, no. 14, p. 5240, 2022. doi: [10.3390/s22228641](https://doi.org/10.3390/s22228641)
- [14] J. Lee, H. Davari, J. Singh, and V. Pandhare, "Industrial Artificial Intelligence for Industry 4.0-based manufacturing systems," *Manuf. Lett.*, vol. 18, pp. 20–23, 2018. doi: [10.1016/j.mfglet.2018.09.002](https://doi.org/10.1016/j.mfglet.2018.09.002)
- [15] S. H. Khan and A. H. I. Mourad, "Integrating Industry 4.0 in engineering education: Implementation patterns, pedagogical strategies, and industry alignment," *Cogent Educ.*, vol. 12, no. 1, p. 2588010, 2025. doi: [10.1080/2331186X.2025.2588010](https://doi.org/10.1080/2331186X.2025.2588010)
- [16] Deloitte and The Manufacturing Institute, "Creating pathways for tomorrow's workforce today: Beyond reskilling in manufacturing," Deloitte Development LLC, 2021. [Online]. Available: [Deloitte.com](https://www.deloitte.com)
- [17] L. Li, "Reskilling and upskilling the future-ready workforce for Industry 4.0 and beyond," *Inf. Syst. Front.*, vol. 26, no. 5, pp. 1697–1712, 2024. doi: [10.1007/s10796-022-10308-y](https://doi.org/10.1007/s10796-022-10308-y)
- [18] M. Hampole, D. Papanikolaou, L. D. W. Schmidt, and B. Seegmiller, "Artificial Intelligence and the Labor Market," Nat. Bureau Econ. Res., Cambridge, MA, USA, Working Paper 33509, 2025. doi: [10.3386/w33509](https://doi.org/10.3386/w33509)
- [19] K. McElheran, M. J. Yang, Z. Kroff, and E. Brynjolfsson, "The Rise of Industrial AI in America: Microfoundations of the Productivity J-Curve(s)," Center for Economic Studies, U.S. Census Bureau, Washington, DC, USA, Working Paper CES-25-27, Apr. 2025. [Online]. Available: <https://www.census.gov/library/working-papers/2025/adrm/CES-WP-25-27.html>
- [20] L. I. González-Pérez and M. S. Ramírez-Montoya, "Competencies of the engineer in Industry 4.0 context: A systematic literature review," *Production*, vol. 34, p. e20230051, 2024. doi: [10.1590/0103-6513.20230051](https://doi.org/10.1590/0103-6513.20230051)
- [21] T. Chen, V. Sampath, M. C. May, R. Schidelko, and G. Lanza, "Machine Learning in Manufacturing towards Industry 4.0: From 'For Now' to 'Four-Know'," *Appl. Sci.*, vol. 13, no. 10, p. 6140, 2023. doi: [10.3390/app13031903](https://doi.org/10.3390/app13031903)
- [22] C. Chau, V. Su, K. Tran, and C. Nguyen, "Digital Engineering Approach for Automation Education: A Case Study in Using Digital Learning Factory," in *Decarbonizing Value Chains*, H. Kohl et al., Eds. Cham, Switzerland: Springer Nature, 2025, pp. 772–781. doi: [10.1007/978-3-031-93891-7_85](https://doi.org/10.1007/978-3-031-93891-7_85)
- [23] A. M. Rojas and G. Barbieri, "A Low-Cost and Scaled Automation System for Education in Industrial Automation," in *Proc. 24th IEEE Int. Conf. Emerg. Technol. Factory Automat. (ETFA)*, Sep. 2019, pp. 439–444. doi: [10.1109/ETFA.2019.8869535](https://doi.org/10.1109/ETFA.2019.8869535)
- [24] J. S. Siddiqi, A. S. Gandy, and S.-J. Hsieh, "Technical Training for Industry 4.0 Technologies: Low-Cost Gantry Candy Sorting System for Education and Outreach," in *Proc. 2024 ASEE Annu. Conf. Expo.*, Portland, OR, USA, Jun. 23–26, 2024. doi: [10.18260/1-2--48079](https://doi.org/10.18260/1-2--48079)
- [25] J. D. Cuiffi et al., "Factory 4.0 Toolkit for Smart Manufacturing Training," in *Proc. 2021 ASEE Virtual Annu. Conf. Content Access*, Virtual, Jul. 26, 2021. doi: [10.18260/1-2--37176](https://doi.org/10.18260/1-2--37176)



- [26] M. R. Embong, A. Asbollah, and M. Z. Abdul Hamid, "Empowering industrial automation labs with IoT: A case study on real-time monitoring and control of induction motors using Siemens PLC and Node-RED," *J. Mech. Eng. Sci.*, vol. 18, no. 2, pp. 10004–10016, 2024. doi: [10.15282/jmes.18.2.2024.3.0790](https://doi.org/10.15282/jmes.18.2.2024.3.0790)
- [27] A. Awouda, E. Traini, M. Asranov, et al., "Bloom's IoT Taxonomy towards an effective Industry 4.0 education: Case study on Open-source IoT laboratory," *Educ. Inf. Technol.*, vol. 29, pp. 15043–15065, 2024, doi: [10.1007/s10639-024-12468-7](https://doi.org/10.1007/s10639-024-12468-7).
- [28] S. Doboviček, E. Krulčić, D. Pavletić, and R. Godina, "A Scalable Learning Factory Concept for Interdisciplinary Engineering Education: Insights from a Case Implementation," *Educ. Sci.*, vol. 15, no. 12, p. 1574, 2025. doi: [10.3390/educsci15121574](https://doi.org/10.3390/educsci15121574)
- [29] Virginia Tech News, "Learning Factory prepares students for Industry 4.0," Jan. 24, 2022. [Online]. Available: VT.edu
- [30] C. Rejón, S. Martín, and A. Robles-Gómez, "Easy development of Industry 4.0 remote labs," *Electronics*, vol. 13, no. 8, Art. no. 1508, 2024. doi: [10.3390/electronics13081508](https://doi.org/10.3390/electronics13081508)
- [31] E. Gil Herrando, D. Delgado, and R. Aragües, "Virtual lab for online learning in industrial automation: A comparison study," in *Proc. 10th Int. Conf. Educ. New Learn. Technol. (EDULEARN18)*, Palma, Spain, Jul. 2–4, 2018, pp. 6042–6050. doi: [10.21125/edulearn.2018.1442](https://doi.org/10.21125/edulearn.2018.1442)