NATIONAL ENGITECH DIGEST

VOL. 3 ISSUE 1

e-ISSN: 3030-5969

An OpenCV-Based Rule Validation Framework for Hand-Drawn Data Flow Diagram Symbols in a Web-Based Learning Environment

Ainie Hayati Noruzman[1]

[1] Department of Information and Communication Technology, Politeknik Ungku Omar, Perak, Malaysia

ainiehayati@puo.edu.my

Abstract

Traditional approaches to teaching Data Flow Diagrams (DFDs) rely primarily on static learning materials and lecturer-guided exercises, providing limited opportunities for interactive practice and immediate formative feedback. This paper proposes an OpenCV-based rule-validation framework for recognising and validating hand-drawn DFD symbols in a web-based learning environment developed with Python and Streamlit. The proposed framework applies a sequence of computer vision techniques, including Red, Green, Blue, and Alpha (RGBA) format to Blue, Green, and Red (BGR) conversion, grayscale conversion, Canny edge detection, contour detection, polygon approximation, vertex counting, and rule-based geometric validation, to automatically recognise predefined DFD symbols and provide immediate feedback to learners. A user evaluation involving 31 Diploma in Information Technology students from three Malaysian polytechnics indicated high user acceptance, with the highest ratings given to the immediate feedback mechanism and its perceived contribution to understanding DFD concepts. The findings indicate that the proposed OpenCV-based rule-validation framework is a practical and well-accepted computer vision approach to support interactive DFD learning in Technical and Vocational Education and Training (TVET), although its effect on learning outcomes has not yet been objectively measured.

Keywords: OpenCV; Computer Vision; Rule-Based Validation; Data Flow Diagram; Interactive Learning; TVET.

Introduction

Recent studies have shown that interactive learning environments incorporating immediate feedback can significantly improve student engagement, self-efficacy, and learning outcomes in digital learning environments (Lin et al., 2024). For example, Yoon et al. (2025) demonstrated that automated grading with real-time feedback enhanced student independence and reduced the delay associated with conventional manual assessment. Similarly, Aung et al. (2024) reported that image-based evaluation techniques in web-based learning assistants enabled rapid assessment and improved learning interactions. These findings highlight the potential of integrating lightweight web technologies with automated validation techniques to create more engaging and effective learning environments.

Recent advances in computer vision have further enabled automated recognition and validation of user-generated drawings. OpenCV provides efficient image-processing techniques, including contour detection and polygon approximation, which have been successfully applied to automated educational assessment and engineering diagram recognition (Fontenla-Carrera et al., 2023; Jocovic et al., 2024). However, existing studies have primarily focused on automated grading, engineering sketches, or general image recognition, with limited attention given to the automatic validation of user-drawn Data Flow Diagram symbols within interactive learning environments. Rule-based computer vision was selected over machine learning because it enables transparent, real-time symbol validation without requiring labelled training data or substantial computational resources.

Hence, this paper proposes a design and implementation of an OpenCV-based rule-validation framework for Hand-Drawn Data Flow Diagram Symbols in a Web-Based Learning Environment. The proposed system, developed using Python, Streamlit, and OpenCV, features guided tutorials, interactive drawing exercises, immediate feedback, and real-time validation to facilitate practical education in DFD.

Methodology

Figure 1 illustrates the proposed OpenCV-based DFD symbol validation framework. The framework consists of four sequential stages: exercise definition, student drawing acquisition, OpenCV-based rule validation, and immediate feedback. The workflow enables real-time recognition and validation of hand-drawn Data Flow Diagram (DFD) symbols within a web-based learning environment, allowing learners to receive immediate formative feedback during practical exercises.

Step 1: Exercise Definition

The process begins by defining the exercise, in which learners are tasked with drawing a DFD. The exercises outline the following: the DFD symbol to be used, a sample reference, and predefined validation rules. The rules outline the target symbol's expected geometric attributes and provide a basis for the next step in validation.

Step 2: Student Drawing Acquisition

The learner uses the web-based drawing canvas to draw the requested DFD symbol. The drawing is then stored as an image and sent to the OpenCV processing module for automatic validation upon completion. This captured image serves as input to the computer vision pipeline.

Step 3: OpenCV-Based Rule Validation

The captured image is processed using OpenCV via a sequence of operations. The captured canvas image was converted from RGBA to BGR and then to grayscale. Canny edge detection was applied using T_{lower} = 50 and T_{upper} = 150, representing the lower and upper edge-detection thresholds, respectively, with an aperture size of 3. External contours were detected and simplified using polygon approximation with $\varepsilon = 0.04 \times P$, where ε is the approximation tolerance and P denotes the contour perimeter. The resulting geometric characteristics were compared with predefined rules to validate the user-drawn DFD symbols. The current framework validates only the geometric contours of DFD symbols; text labels, such as "Customer," are not recognised or validated using Optical Character Recognition (OCR).

Step 4: Immediate Feedback

After the rule-based validation process, the system displays an alert message to the learner indicating whether the submitted DFD symbol is correct or incorrect. If the detected symbol meets the predefined validation rules, the positive feedback message "Congratulations! Your answer is correct." is displayed. If the drawn symbol does not satisfy the rules, the corrective feedback "Not quite right. Try again!" is displayed, and the learner can redraw and resubmit the symbol. This immediate feedback mechanism allows learners to keep learning and correct their mistakes during the drawing activity without waiting for the lecturer's feedback.

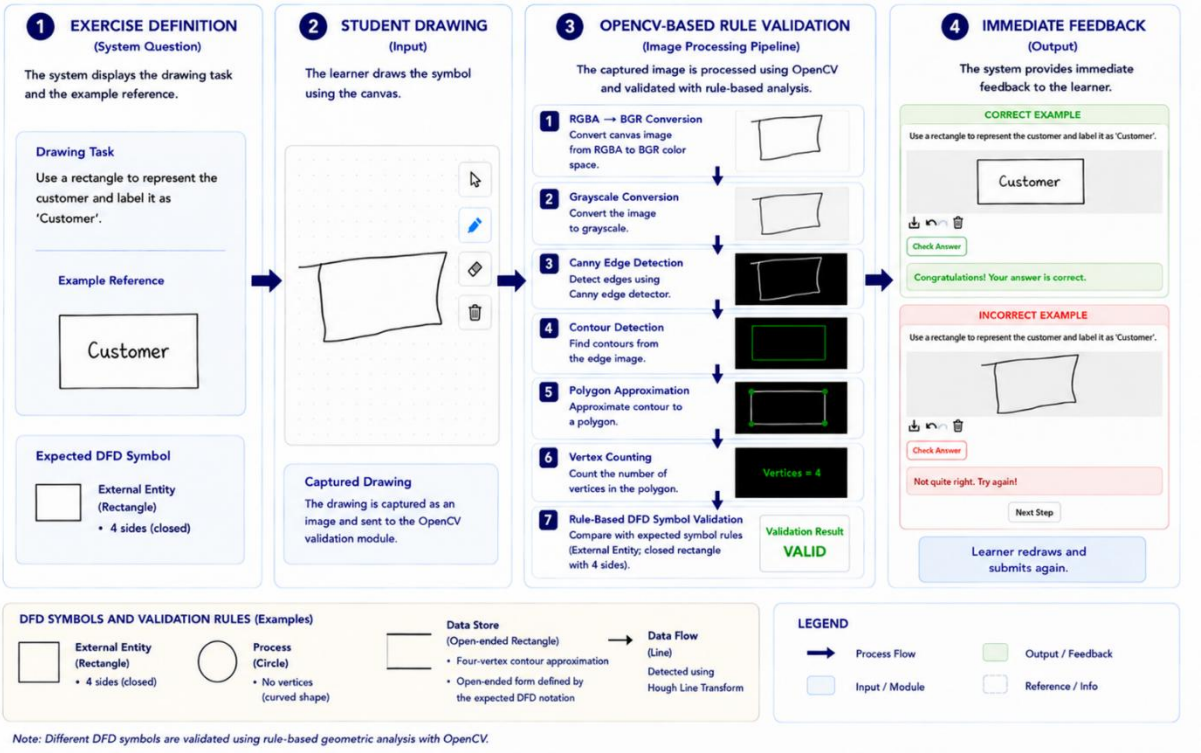


Figure 1. OpenCV-based DFD symbol validation framework.

NATIONAL ENGITECH DIGEST

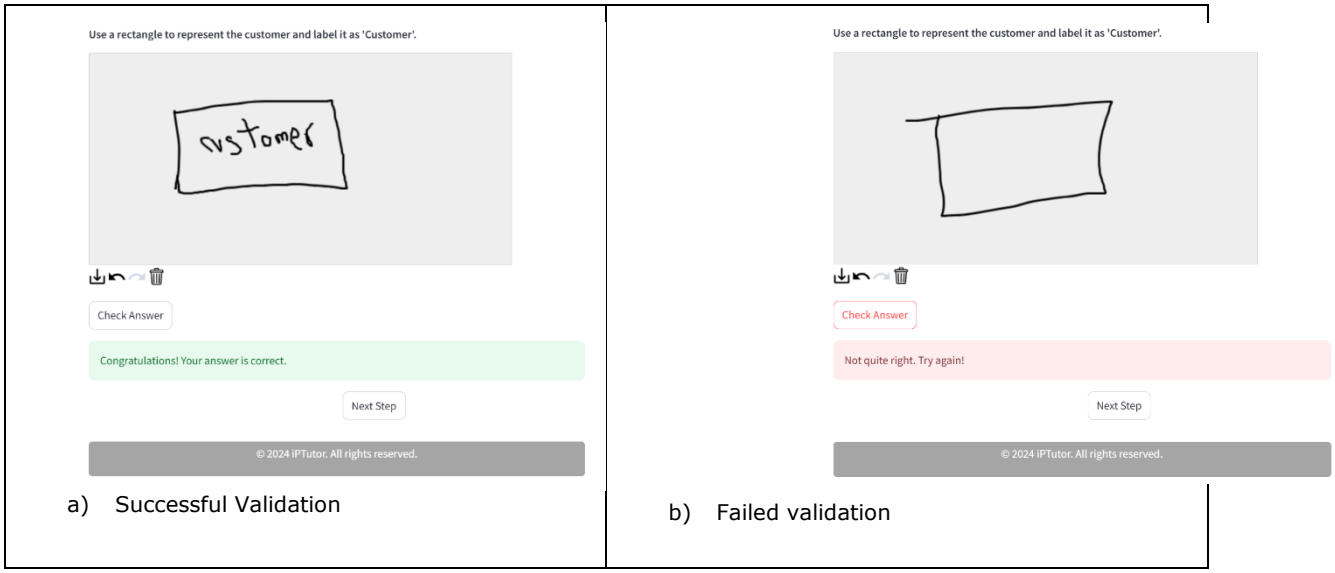
VOL. 3 ISSUE 1

e-ISSN: 3030-5969

Results and Discussion

The proposed OpenCV-based rule-validation framework successfully recognised and validated the user-drawn Data Flow Diagram (DFD) symbols by comparing their geometric properties with the predefined DFD symbol rules. The image processing pipeline (RGBA-to-BGR conversion, grayscale conversion, Canny edge detection, contour detection, polygon approximation, vertex counting, and rule-based geometric validation) enabled real-time identification of hand-drawn DFD symbols in the web-based learning environment. Learners were then given immediate formative feedback to identify and correct drawing errors during practical activities.

Figure 2 illustrates two validation scenarios generated by the proposed framework. Figure 2(a) shows a successful validation example, showing a correctly drawn external entity (a rectangle) labelled 'Customer'. The OpenCV validation process successfully recognised the required geometric characteristics and generated the immediate feedback message, "*Congratulations! Your answer is correct.”* Conversely, Figure 2(b) shows a failed validation example, in which an incorrectly drawn rectangle does not satisfy the expected DFD symbol requirements. In this case, the framework immediately provides corrective feedback, "*Not quite right. Try again!*", allowing the learner to redraw and resubmit the symbol. This immediate feedback mechanism enables learners to identify and correct drawing errors without waiting for a lecturer's assessment.



a) Successful Validation

b) Failed validation

Figure 2. Immediate feedback generated by the proposed OpenCV-based rule validation framework: (a) successful validation and (b) failed validation.

Following functional verification, a user evaluation was conducted involving 31 Diploma in Information Technology students from three Malaysian polytechnics. As summarised in Figure 3, all dimensions recorded mean scores above 4.0 out of 5, indicating generally positive user acceptance. Immediate feedback received the highest rating (M = 4.45, SD = 0.68), followed by its contribution to understanding DFD concepts (M = 4.39, SD = 0.67) and the usefulness of the assistance module (M = 4.35, SD = 0.75). Participants also showed a strong willingness to recommend the platform (M = 8.61, SD = 1.38) on a 10-point scale. These results indicate positive user acceptance of OpenCV-based rule validation as a learning support mechanism that provides immediate formative feedback in interactive DFD learning. However, the results reflect students’ self-reported perceptions; learning outcomes were not objectively measured (e.g., through pre-test and post-test comparison or a control group), so the findings should not be interpreted as evidence of improved learning performance.

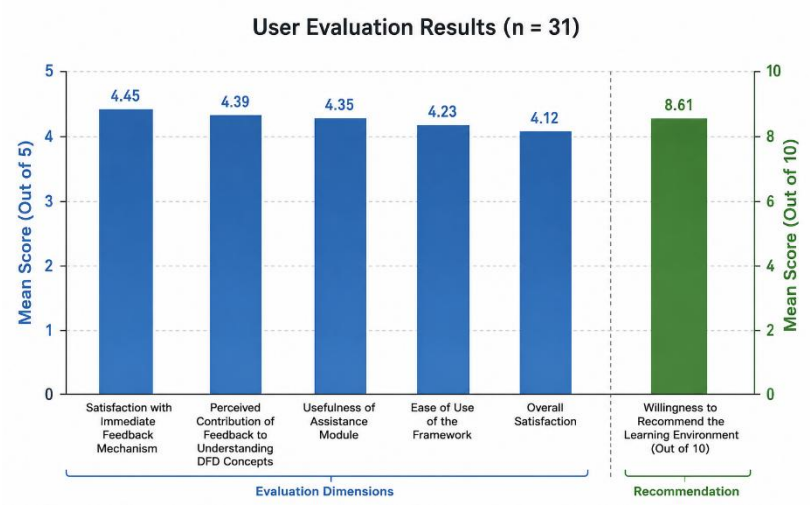


Figure 3. User evaluation of the proposed OpenCV-based rule validation framework (N = 31).

Conclusion

This paper introduces an OpenCV-based rule-validation framework for recognising and validating hand-drawn Data Flow Diagram symbols in a web-based learning environment. The proposed framework successfully employed image processing techniques (including RGBA-to-BGR conversion, grayscale conversion, Canny edge detection, contour detection, polygon approximation, vertex counting, and rule-based geometric validation) for real-time validation of predefined DFD symbols and for formative feedback to learners. The functional verification showed that the proposed framework worked as expected, and the user evaluation showed that the immediate feedback mechanism was well accepted by students, who perceived it as supporting their understanding of DFD concepts. These results suggest that the proposed framework is a practical and well-accepted computer vision solution for supporting interactive DFD learning in Technical and Vocational Education and Training (TVET). This implementation is limited to validating predefined DFD symbols in individual learning exercises, and the evaluation measured user acceptance rather than learning gains. Future work will incorporate OCR-based text-label recognition and syntactic validation of complete DFDs containing multiple interconnected symbols, adaptive feedback during diagram construction, and a controlled pre-test and post-test study to measure learning outcomes objectively.

References

Aung, S. T., Funabiki, N., Aung, L. H., Kinari, S. A., Wai, K. H., & Mentari, M. (2024). An image-based user interface testing method for Flutter programming learning assistant system. *Information*, *15*(8), 464. https://doi.org/10.3390/info15080464

Fontenla-Carrera, G., Fernández Vilán, Á. M., & Izquierdo Belmonte, P. (2023). Automatic identification of kinematic diagrams with computer vision. In A. Vizán Idoipe & J. C. García Prada (Eds.), *Proceedings of the XV Ibero-American Congress of Mechanical Engineering (IACME 2022)* (pp. 425–431). Springer. https://doi.org/10.1007/978-3-031-38563-6_62

Jocovic, V., Marinkovic, M., Stojanovic, S., & Nikolic, B. (2024). Automated assessment of pen and paper tests using computer vision. *Multimedia Tools and Applications, 83*(1), 2031–2052. https://doi.org/10.1007/s11042-023-15767-2

Lin, Y. L., Wang, W. T., & Hsieh, M. J. (2024). The effects of students' self-efficacy, self-regulated learning strategy, perceived and actual learning effectiveness: A digital game-based learning system. *Education and Information Technologies, 29*(16), 22213–22245. https://doi.org/10.1007/s10639-024-12700-4

Yoon, Y., Jeon, Y., Kim, J., Han, S., Kim, H., & Kwon, S. (2025). CADuBoost: Enhancing education in mechanical 3D CAD modeling through automated grading and feedback system. *Computer Applications in Engineering Education, 33*(6), e70096. https://doi.org/10.1002/cae.70096