**Symbol Recognition System**

Orange Team

Tallinn University of Technology (TalTech), Virumaa College

Project Supervisor:

Team members:

Rokas Pranauskas

Jogaila Janušas

Put your name here

Put your name here

Date: 14.10.2024-25.10.2024

**Keywords:**

* Circuit Diagram Recognition
* Symbol Detection
* Elements
* OpenCV
* Deep Learning
* Artificial Inteleigence (AI)
* Automation
* Neural Networks
* Computer Vision
* Machine Learning
* Python

**Abstract**

This project, developed by the Orange Team at TalTech Virumaa College, aims to create a **Symbol Recognition System** capable of detecting and identifying engineering symbols from circuit diagrams or live camera feeds. Utilizing **computer vision** techniques with **OpenCV** and **artificial intelligence** for enhanced recognition, the system interprets schematic elements like control valves, actuators, equipment, storage, motors, heat exchanges, distillation and other technical symbols.

This program offers real-time symbol detection. Its practical integration in the fields of engineering and automation showcases the potential to streamline processes by reducing manual symbol identification.

This document details the system’s development process, architecture, implementation, and evaluation, providing an insight into the technical aspects and challenges faced during the project. The project emphasizes **object detection methodologies**, **template matching**, and the integration of **AI models** for more efficient symbol recognition.

**Introduction**

**Background**

In today's fast-paced engineering environment, the ability to quickly and accurately interpret circuit diagrams is essential. Traditional methods of manual symbol recognition are time-consuming and prone to human error. The advancement of **computer vision** and **artificial intelligence** technologies presents a unique opportunity to automate this process, improving efficiency and accuracy in engineering workflows.

The **Symbol Recognition System** developed by the **Orange Team** at **TalTech Virumaa College** addresses this need by leveraging cutting-edge techniques in image processing. This project is rooted in the growing field of machine learning and computer vision, where algorithms can be trained to recognize and interpret complex visual data.

**Objectives**

The primary objective of this project is to design and implement a robust Symbol Recognition System capable of:

* **Detecting and Identifying Engineering Symbols:** The system aims to recognize various technical symbols such as control valves, actuators, equipment, storage containers, motors, heat exchangers, and distillation units.
* **Real-Time Processing:** By utilizing live camera feeds, the system provides instant feedback, allowing for dynamic interaction with circuit diagrams. This feature is particularly useful in educational settings and industrial applications, where immediate results can enhance learning and operational efficiency.
* **Streamlining Engineering Processes:** By automating symbol detection, the system reduces the burden of manual identification, thus minimizing errors and accelerating project timelines.

**Significance**

The practical implications of this project extend beyond mere symbol recognition. In the fields of **engineering** and **automation**, accurate interpretation of circuit diagrams is critical for safety, compliance, and operational effectiveness. By integrating this technology into everyday engineering practices, organizations can enhance productivity and focus on more complex design tasks.

This document provides a comprehensive overview of the system's development process, architecture, implementation strategies, and evaluation methods. It delves into the technical challenges encountered throughout the project, exploring solutions that leverage **object detection methodologies**, **template matching**, and **AI model integration**. The ultimate goal is to create a reliable and efficient tool that contributes to the advancement of engineering practices.

**Tasks**

1. **Research and Selection of Algorithms**  
   Conducted a thorough literature review to identify suitable computer vision algorithms and techniques, including template matching and AI integration, for symbol recognition.
2. **Template Creation**  
   Collected and created high-quality image templates of various engineering symbols, such as resistors, valves, and actuators, ensuring a diverse and representative dataset for the recognition process.
3. **System Design**  
   Designed the architecture of the symbol recognition system, outlining the interaction between components such as the camera input, image processing, and user interface.
4. **Implementation of Image Processing**  
   Developed the image preprocessing functions, including contour detection and noise reduction, to enhance the accuracy of symbol detection.
5. **Symbol Detection Integration**  
   Integrated the symbol detection algorithm using OpenCV, implementing template matching and AI methods to improve detection accuracy.
6. **Real-time Video Processing**  
   Implemented functionality for real-time video processing, allowing the system to detect symbols from live camera feeds, enhancing the practical application of the tool.
7. **Testing and Evaluation**  
   Conducted extensive testing using various circuit diagrams and live camera scenarios to evaluate the performance of the system, measuring detection accuracy and processing speed.
8. **User Interface Development**  
   Developed a user-friendly graphical user interface (GUI) for easy interaction with the system, enabling users to upload images and initiate live scanning.
9. **Documentation and Reporting**  
   Compiled comprehensive documentation detailing the development process, system architecture, user instructions, and evaluation results for future reference and improvements.

## **Methodology**

The Symbol Recognition System was developed using the YOLOv8 model, a state-of-the-art object detection framework. The model was trained on a dataset comprising over 10,000 images, allowing for improved symbol recognition accuracy. The training process took over a day to complete, ensuring that the model could effectively recognize a wide range of engineering symbols.

Initially, the team experimented with template matching methods by uploading single screenshots of symbols. However, this approach proved challenging, as the program struggled to reliably detect elements in complex schematics. The recognition process was very slow, and the program had significant difficulty identifying symbols accurately. To enhance the system's performance, the focus shifted to the YOLOv8 model, which provided superior detection capabilities and improved recognition speed.

The GUI development was an iterative process, involving continuous improvements based on feedback and testing. The team engaged in frequent communication to ensure that everyone was aware of progress and could assist where needed. Task delegation was essential to leverage each member’s strengths effectively, fostering a collaborative environment throughout the project.

**Development Process**

The development process was characterized by multiple iterations of design and implementation. Key phases included:

* Initial Planning: Defining the project scope, objectives, and team roles.
* Data Preparation: Collecting and curating the dataset for training the YOLOv8 model.
* Model Training: Training the model and validating its performance on a separate dataset.
* GUI Design: Designing a user-friendly interface that facilitates easy interaction with the system.
* Testing and Evaluation: Conducting thorough testing to ensure the system meets the project objectives and refining the approach based on feedback.

**Challenges and Solutions**

1. **Initial Detection Limitations**:  
   The original approach utilized template matching, which relied on uploading single screenshots of symbols. This method proved to be ineffective for complex schematics, as the program struggled to reliably identify elements. The recognition process was slow, and accuracy was compromised, especially in scenarios where symbols were partially obscured or rotated.

**Solution**:  
To address these limitations, the team transitioned to using the YOLOv8 model, a more advanced object detection framework. YOLOv8 is designed for real-time detection, which significantly improved the speed and reliability of symbol recognition in various scenarios.

1. **Complexity of Schematics**:

Circuit diagrams often contain numerous overlapping symbols, varying sizes, and orientations, making detection difficult. The system initially failed to discern between similar symbols and struggled with background noise, leading to incorrect identifications.

**Solution:**

By training the YOLOv8 model on a diverse dataset of over 10,000 images, the team ensured the system could recognize symbols in different contexts and complexities. This extensive training dataset included various orientations, sizes, and backgrounds, which improved the model's robustness against complex schematics.

1. **Integration of GUI**:  
   The graphical user interface (GUI) was developed to provide an intuitive user experience, but its initial version lacked essential functionalities and aesthetic appeal. Early iterations of the GUI faced usability challenges, making it difficult for users to navigate the system effectively.

**Solution**:  
The team iteratively improved the GUI based on user feedback. Regular testing sessions were conducted to identify usability issues, and enhancements were made accordingly. The final version of the GUI features a clean layout, clear labels, and intuitive buttons, making it user-friendly and effective for symbol recognition tasks.

1. **Collaboration and Task Management**:  
   Coordinating tasks among team members was essential for project success. However, ensuring that everyone was on the same page and effectively utilizing each member's strengths posed a challenge.

**Solution**:  
The team established regular communication through meetings and collaborative tools to discuss progress and address challenges. By identifying each member's strengths, tasks were distributed according to individual expertise, which maximized productivity and fostered a supportive team environment.

1. **Training Time and Computational Resources**:  
   Training the YOLOv8 model required substantial computational resources and time. The initial training sessions were prolonged due to hardware limitations and the complexity of the dataset.

**Solution**:  
To mitigate this issue, the team focused on optimizing the training process by temporarily disabling all non-essential processes on the computer. This allowed the system to concentrate its full computational power on the training task, resulting in faster training cycles and more efficient experimentation with various model parameters.

**Future Work**

Future work may include:

* **Expanding the Dataset:** Increasing the number of training images to enhance model accuracy further.
* **Real-time Performance Enhancements:** Optimizing the detection algorithm to improve speed and efficiency.
* **Additional Symbol Types:** Extending the system to recognize a broader range of engineering symbols beyond the current set.

**Conclusion**

The Symbol Recognition System developed by the Orange Team at TalTech Virumaa College represents a significant advancement in the field of computer vision and engineering automation. This project successfully integrated modern artificial intelligence techniques, specifically leveraging the YOLOv8 model, to enhance the accuracy and efficiency of symbol detection in circuit diagrams.

Throughout the development process, the team faced various challenges, including initial difficulties with template matching methods and the constraints imposed by limited computational resources. However, by shifting to the YOLOv8 model, the project dramatically improved its detection capabilities, enabling it to recognize a broader range of symbols with higher accuracy. This change not only streamlined the detection process but also made it feasible to analyze complex schematics effectively.

Additionally, the design and implementation of the graphical user interface (GUI) underwent iterative improvements, reflecting the team's commitment to functionality. The collaboration among team members was crucial, as it allowed for the distribution of tasks according to individual strengths, promoting a cohesive work environment. Open communication ensured that challenges were addressed promptly and collectively, fostering a sense of teamwork that was essential for the project's success.

The results achieved in this project highlight the potential for automation in engineering workflows, demonstrating how technology can significantly reduce manual identification tasks. This system not only enhances the efficiency of engineers and technicians but also sets the foundation for future advancements in automated symbol recognition and related fields.

**In conclusion**, the Symbol Recognition System developed by the Orange Team not only meets the objectives set forth at the project's inception but also paves the way for further exploration into the integration of AI and computer vision in engineering applications. The insights gained from this project will inform future work, contributing to ongoing research and development in the realm of intelligent automation systems.