Text Recognition using YOLO v3 of Control Logic Diagram

Dongil Lee [a*], Min woo Kang [a], Jung han Lee [b]
[a]*KHNP, Central Research Institute, 70, 1312-gil, Yuseong-daero, Yuseong-gu, Daejeon, 34101, South Korea*
[b]*HUPEC. Co., Ltd., #1810, 10, Dongtan-daero 21-gil, Hwaseong-si, Gyeonggi-do, 18471, South Korea*
[*]*Corresponding author: diturtle@khnp.co.kr*

1. Introduction

Among the control logic drawings used in the Instrumentation & Control (I&C) system of a Nuclear Power Plant (NPP), there are two (2) types of representative drawings: Control Logic Diagram (CLD) and Vendor Logic Diagram (VLD), which is a manufacturer's drawing.

Symbols, lines, and numerous characters exist in the control logic drawing. Characters not only indicate the characteristics of a symbol, but also express a link between drawings and drawings.

Character recognition is essential in order to recognize hardcopy's control logic drawings based on artificial intelligence and reproduce them as vector-type regular drawings.

Representative technologies used for character recognition include OCR and Tesseract [1]. However, although the control logic drawings in the NPP were computerized, many drawings were scanned and reproduced, and the quality ranges from good to bad.

In this paper, we are going to explain the results of applying YOLO v3 to overcome the limitations in recognizing the characters of drawings with Tesseract.

2. Character Recognition using Tesseract

The character recognition function was implemented using Tesseract, which is the first commonly used optical character recognition open-source library.

However, since the size of the characters in the control logic drawing is very small, Tesseract, which extracts the characteristics of the character image and recognizes the character using the characteristics, did not obtain a high discrimination result.

The following table measures the characters recognized using Tesseract.

Table I: Problem Description

Drawing	Input letter	Mis-recog	Line	**Recognize**	**Accuracy**
1	405	36	53	**335**	**82.72%**
2	339	61	68	**200**	**59.00%**
3	452	72	54	**333**	**73.67%**
4	410	22	57	**289**	**70.49%**
5	1,030	70	74	**754**	**73.20%**
6	1,207	95	67	**1017**	**84.26%**
7	395	19	60	**269**	**68.10%**
8	329	46	66	**172**	**52.28%**
9	1,026	140	93	**760**	**74.07%**
10	2,363	215	93	**2105**	**89.08%**

According to the results of Table 1, the accuracy of the recognized characters compared to the input letters was 52.28% to 89.08%. In addition, there are many cases where letters are missed, misrecognized, or lines are recognized as letters.

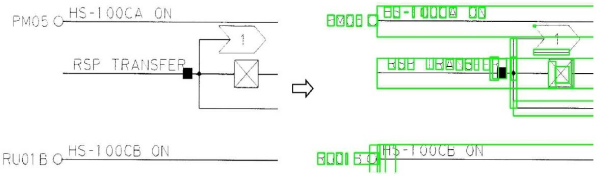

Fig. 1. Tesseract recognition (original (L), recognition (R))

Fig. 1 shows that the control logic drawing was recognized using Tesseract, and the recognized part is marked with a green border. The recognized character extracts the result as shown in Fig. 2. However, it can be seen that "PM05" is displayed as "PMOEO" and "HS-100CB" is not recognized.

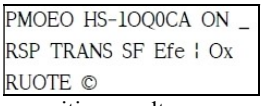

Fig. 2. Tesseract recognition result

3. Application of YOLO v3

Since there are many problems with accuracy to recognize control logic with Tesseract, the following two (2) methods were applied to recognize characters.

a. Apply YOLO v3 by processing characters as one object unit [2]
b. Splitting the drawing for better accuracy

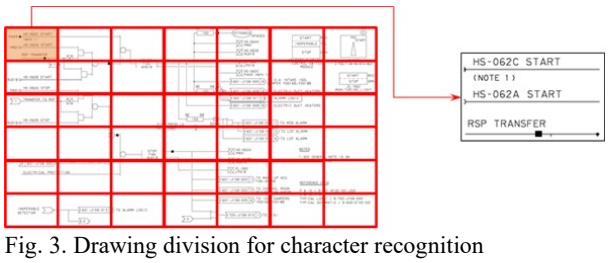

Fig. 3. Drawing division for character recognition

In order to improve the recognition of characters having a relatively small size compared to the size of the drawing, they were divided and processed at a ratio similar to the aspect ratio of the control logic drawing. If the size of the divided image is too small, the processing load is large, so it is set to 500×300. (Fig. 3)

After recognizing characters in the divided image, each coordinate is extracted and the coordinates are calculated from the original drawing.

When dividing a drawing, in order to recognize the characters cut off at the boundary, add 50 each to the x and y coordinate values, then divide and re-recognize. (Fig. 4)

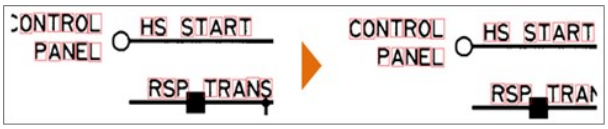

Fig. 4. Recognition of characters cut off at borders

After calculating the coordinates in the original drawing for the recognized characters, the recognized characters overlapped several times are deleted, leaving only one (1) one.

After recognizing characters, the recognition rate is improved through separate learning and setting of different weights for characters that are indistinguishable from straight lines or other symbols.

When YOLO v3 is applied, the results are shown in Table II below. Drawings 7 and 10 have a lot of special symbols, so the accuracy is somewhat low. However, as the learning about special symbols is currently in progress, the recognition of special symbols was not performed. Drawings 7 and 10 showed low results due to the presence of many special symbols.

Table II: Character recognition results of Tesseract and YOLO v3

Drawing	Input	Tesseract	**YOLO v3**	**YOLO acc.**
1	405	335	**403**	**99.51%**
2	339	200	**334**	**98.53%**
3	452	333	**422**	**93.36%**
4	410	289	**380**	**92.68%**
5	1030	754	**981**	**95.24%**
6	1207	1017	**1169**	**96.85%**
7	395	269	**295**	**74.68%**
8	329	172	**321**	**97.57%**
9	1026	760	**989**	**96.39%**
10	2363	2105	**1982**	**83.88%**

The following figure is the result of the text recognized by Tesseract and the text recognized by YOLO v3.

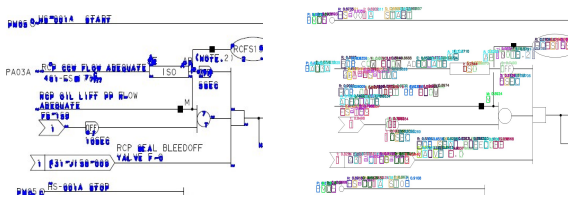

Fig. 5. Tesseract result (L), YOLO v3 result (R)

3. Conclusions

Tesseract, which is widely used for character recognition for control logic drawings (CLD, VLD) used in NPP, was applied, but sufficient results were not obtained to recognize characters in drawings.

However, using YOLO v3, the drawing was cut with an appropriate resolution input, and as a result of recognizing the text as an object, a result almost similar to the actual drawing was obtained.

When the learning of special symbols is completed in the future, recognition technology for all characters in the drawing is completed. When all the characters are recognized, the hardcopy control logic drawing can be converted into a vector-type regular drawing by combining the symbol and the line-recognized drawing information.

REFERENCES

[1] Patel, Chirag and Patel, Atul and Patel, Dharmendra, Optical character recognition by open source OCR tool tesseract: A case study, International Journal of Computer Applications, Vol 55, p50-56, 2012

[2] H. Wang and Z. Zhang, "Text Detection Algorithm based on Improved YOLOv3," 2019 IEEE 9th International Conference on Electronics Information and Emergency Communication (ICEIEC), pp. 147-150, 2019