

딥러닝 기반 도면 인식

조건웅, 이해연, 윤지혜, 한혜인, 이종석, 심동규
광운대학교

fakem1333@gmail.com, twtewtea@naver.com, jihye9632@naver.com,
hyeneel104@gmail.com, suk2080@kw.ac.kr dgsim@kw.ac.kr

Deep Learning Based P&ID Recognition

Geonung Cho, Hyeeyeon Lee, Jihye Yoon, Hyein Han, Jongseok Lee, Donggyu Sim
Kwangwoon University

요 약

본 논문에서는 P&ID 도면에서 딥러닝을 이용해 기호로 표현된 설비를 인식 및 분류하는 방법을 제안한다. P&ID 도면은 이미지 형식의 파일로 전달 보관되는 경우가 많아 이를 실제 활용하기 위해서는 사용자가 직접 확인해야하는 문제가 있다. 이러한 문제를 해결하기 위해 딥러닝 네트워크 모델인 YOLO_v3 를 이용하여 도면 상의 기호로 표현된 설비를 인식하고, 이를 기반으로 설비에 대한 물량 정보 등을 사용자에게 제공하여 도면 활용의 효율성을 높이는 방법을 제시한다.

1. 서론

P&ID(Piping & Instrumentation Diagram)는 배관 및 공정 장비를 계측 및 제어 장치와 함께 보여주는 도면으로 설계, 건설, 유지 보수 및 운전에 활용되는 핵심 도면이다. 공장 등의 건설을 위한 발주 시에 발주자는 AutoCAD[1]등의 CAD 툴을 이용해 작성한 P&ID 도면을 입찰자에게 이미지 파일 형식 또는 전자 문서 형식(PDF)으로 변환하여 제공한다. 도면을 제공받는 입찰자는 입찰 금액 결정 시에 도면 이미지 파일 상에 존재하는 모든 기호로 표현된 설비, 배관에 대해 수기로 물량 산출을 진행해야 한다. 그러나, 이처럼 수기로 진행하는 물량 산출 과정은 실수로 인하여 입찰 금액과 실제 공사비의 차이가 발생할 수 있고, 이는 회사에 손실을 발생시키기 때문에 많은 재확인을 필요로 한다. 이 과정에서 한정된 시간 내 수작업의 오류를 줄이고자 많은 인력이 투입되어 인력의 낭비가 발생한다.

이러한 문제를 해결하기 위하여 본 논문에서는 딥러닝 네트워크 모델인 YOLO[2] 및 Tesseract[3] OCR 를 이용하여 P&ID 도면 이미지에서 도면 상에 기호로 표현된 설비, 배관들을 자동으로 인식하고, 인식된 설비에 대한 정보를 사용자에게 제공하는 방법을 제안한다.

본 논문의 구성은 다음과 같다. 2 장에서 제안하는 도면 인식 방법에 대해 기술하고, 3 장에서는 실험 결과와 함께 제안하는 방법의 성능을 보인다. 최종적으로 4 장에서 본 논문에 대한 결론을 맺는다.

2. 도면 인식 방법

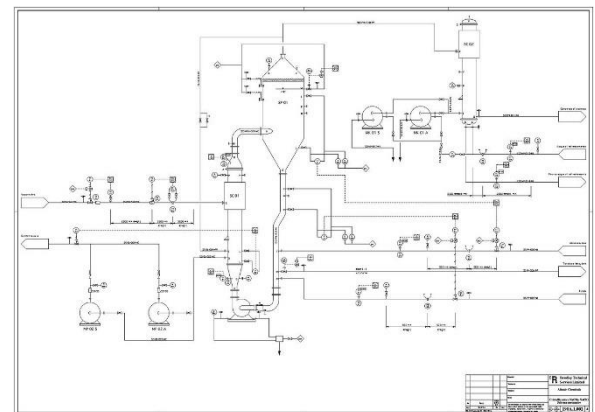


그림 1. P&ID 도면[4]

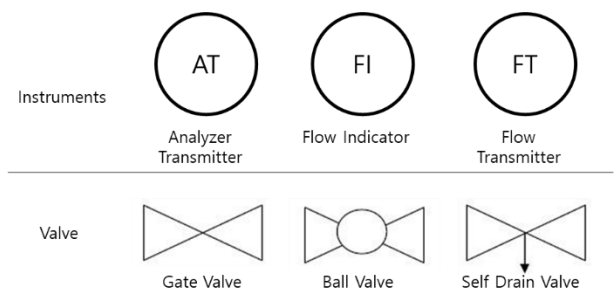


그림 2. P&ID 기호 클래스 내 기호 예시

그림 1. 은 P&ID 도면의 예시로, 설비들이 도면 상에 기호로 표현되어 있다. 도면 상의 기호들을 인식하고 분류하기

위해 딥러닝을 이용하였다. 딥러닝 네트워크의 입력을 도면 이미지로 하고, 이에 대한 출력을 Tensor 구조받는다.

유사한 역할을 수행하는 설비의 경우, 그림 2. 와 같이 유사한 형태의 기호로 도면 상에 표현된다. 앞선 출력 Tensor 에서 도면에서 기호 이미지를 경계 박스에 맞추어 잘라낸 후, 잘라낸 기호를 기호 클래스의 특징에 맞게 OCR 과 Shape Matching 을 이용해 기호를 분류하였다.

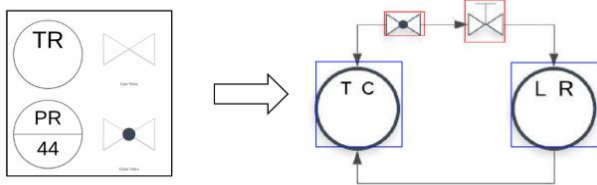


그림 3. 기호 클래스 간 분류

먼저, 도면 인식을 수행하기 위해 기호 클래스 간 분류를 수행한다. ANSI/ISA-S5.1 [5] 규격에 따라 각 클래스에 속하는 설비가 그림 3 과 같이 유사한 형태의 기호로 표시되는 것을 이용하여 전체 기호 중 일부 클래스(Instrument, Valve, Equipment, Pump)의 기호를 먼저 딥러닝을 이용해 인식한다.

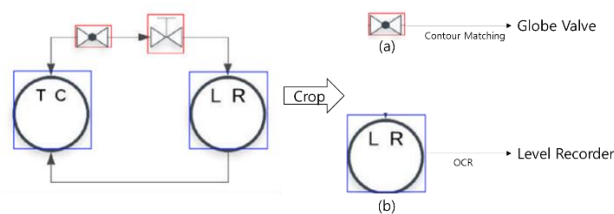


그림 4. 기호 클래스 내 분류

이후, 기호 클래스 내 분류를 수행한다. 앞서 인식된 기호의 경계박스와 좌표를 이용해 도면에서 인식된 기호를 그림 4 의 (a), (b)와 같이 이미지를 잘라 낸다. 잘라낸 이미지 기호는 구분 가능한 특징에 따라 Contour Matching 과 OCR 을 이용해 상세히 분류한다. (a)와 같이 기호의 형태가 기호를 구분하는데 사용될 경우, Contour Matching 은 잘라낸 이미지의 Contour 와 기준 이미지들의 Contour 를 구하고, Hu Moments 를 이용해 Contour 를 비교하여 유사성을 판단한다. [6] 모든 기준 이미지들에 대해 유사성을 판단하고, 이미지들 중 가장 유사하다고 판단되는 기호로 분류한다. 이를 수행하기 위해 기준 이미지를 CAD 프로그램을 이용해 작성하였다. (b)와 같이 기호에 표시된, 원 내의 글자가 기호를 구분하는데 사용될 경우, OCR 을 이용해 기호를 상세히 분류한다. Instrument 의 경우, 원으로 표시된 기호 내부의 글자가 설비를 구분하는데 사용된다. 이를 위해 OCR 을 이용해 기호 내부의 글자를 인식하여 구분한다.

3. 실험 결과

위 방법이 도면에서 기호를 인식할 수 있는지 확인하기 위해 YOLO-v3[7] 네트워크 모델 학습을 수행하였다. YOLO-v3[7] 모델의 학습을 위해 수집한 도면을 모델이 학습할 수 있도록 도면 상의 기호를 경계 박스로 나누어 레이블링을 수행하였다. 그리고 YOLO 모델을 4 개의 클래스에 대해 인식할 수 있도록 변경하였다. Google API 인 Tensorflow [8]를 사용하여

구현했다. 학습과 인식의 사용된 PC의 사양은 표 1. 에 나타나 있다. 학습용 데이터셋으로 총 130 장의 도면을 사용하였으며, 테스트를 위한 데이터셋은 20 장의 도면을 사용하였다. 딥러닝 모델 학습은 총 2200 Epoch 동안 진행되었고, Batch size 는 32 로 학습하였다.

표 1. 수행 환경

| | |
|-----|------------------|
| CPU | I5-8400 |
| GPU | GTX1080TI |
| RAM | 16GB |
| OS | Ubuntu LTS 16.04 |

네트워크 모델 학습을 위해 수집한 P&ID 도면에 대해 레이블링을 수행했다. 각 클래스기호 당 약 200 개의 데이터를 사용했다. 별개로 모델을 평가하기 위해 20 개의 도면을 제작하여 사용하였고, 전체 도면은 클래스당 약 50 개의 기호를 포함하고 있다.

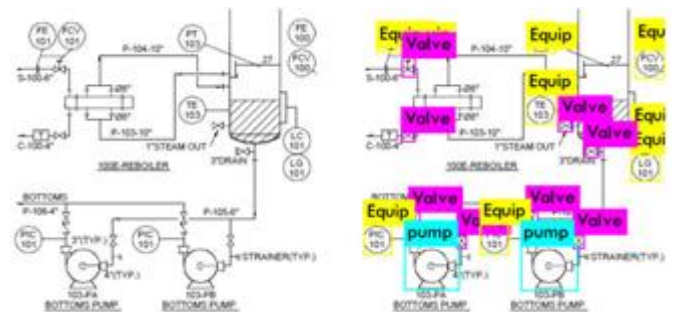


그림 5. YOLO 를 이용한 기호 분류 수행 결과

딥러닝을 이용하여 도면 상의 기호의 위치와 결과는 그림 5 와 같다. 인식된 기호의 경우 도면에 Box 로 표시하였다. 20 개의 도면에서 딥러닝을 이용한 분류를 수행하였을 때, 도면에 존재하는 기호 중 약 85%의 기호가 올바르게 분류되었다. 20 개의 도면의 기호를 모두 분류하는데 걸린 실행 시간은 약 25 초로 도면 당 1.25 초가 소요되었다.

표 2. 클래스 내 분류 평가

| | |
|------------|------|
| 정확도(mAP) | 85% |
| 소요 시간(sec) | 1.25 |

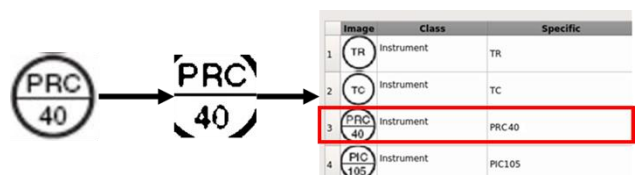


그림 6. Instrument 기호의 OCR 수행 결과

앞선 딥러닝을 통한 분류에서 세분화를 위해 인식된 기호를 OCR, Contour Matching 를 이용해 클래스 내 분류를 수행한 결과는 그림 6 과 같다. OCR 의 경우 Tesseract[3]를 사용해 수행하였다. Contour Matching 의 경우 OpenCV 를 이용해서 수행하였다. Instrument 클래스에 속하는 기호에 대해 OCR 을 수행한 결과를 Table 형식으로 출력하였다. 도면 상에 존재하는 200 개의 기호에 대해 기호 클래스에 맞게 분류를 수행하였을 때, 약 61%의 기호가 올바르게 분류되었다. 200 개의 기호를 분류하기 위해 걸린 실행 시간은 16 초로, 1 개의 기호를 인식하는데 0.8 초가 소요되었다.

표 3. 클래스 간 분류 평가

| | |
|------------|-----|
| 정확도(mAP) | 61% |
| 소요 시간(sec) | 0.8 |

4. 결론

본 논문에서는 딥러닝 기반 P&ID 인식을 수행하기 위해 YOLO-v3[7] 딥러닝 모델과 OCR 과 Contour Matching 을 이용하여 도면 상의 기호 인식을 기호 클래스 별 인식, 기호 클래스 내 인식으로 나누어 수행하여 정확도와 실행 속도를 향상시키는 방법을 제시하였다.

(본 논문에서는 딥러닝 기반 P&ID 인식을 수행하기 위해 YOLO-v3[7] 딥러닝 모델과 OCR 과 Contour Matching 을 이용하였다. 도면 상의 기호 인식은 기호 클래스 별 인식, 기호 클래스 내 인식으로 나누어 수행하여 정확도와 실행 속도를 향상시키는 방법을 제시하였다.

본 논문에서는 데이터 셋의 부족으로 인하여 딥러닝 모델 학습 및 실험 과정에서 인식되지 않는 기호들이 있지만, 충분한 데이터가 확보된다면 인식률을 높일 수 있을 것으로 보인다. 해당 알고리즘을 사용하여 도면 인식을 수행하면, 수기로 도면을 확인하는 과정에서 발생하는 인력 낭비 등의 문제를 해결할 수 있다.

감사의글

본 논문은 2019 년도 교육부의 대학혁신지원사업 사업비를 지원받아 수행된 연구임

참고문헌

- [1] AutoCAD[Computer Software], San Rafael, CA, US: Autodesk
- [2] Joseph Redmon, "You Only Look Once: Unified, Real-Time Object Detection", Proc. the IEEE conference on computer vision and pattern recognition, pp. 779-788, 2016
- [3] Tesseract Open Source OCR Engine[Website]. (2019, Nov 12). , <https://github.com/tesseract-ocr/tesseract>
- [4] Piping and instrumentation diagram[Website]. (2019, Nov 12).

https://en.wikipedia.org/wiki/Piping_and_instrumentation_diagram

[5] Instrumentation Symbols and Identification, American National Standard ISA5.1, 1984

[6] Shape Matching using Hu Moments (C++/Python)[Website], (2019. Nov. 12). <https://www.learnopencv.com/shape-matching-using-hu-moments-c-python/>

[7] J. Redmon, A. Farhadi, "Yolov3: An incremental improvement", arXiv preprint arXiv:1804.02767, 2018.

[8] Tensorflow[Website]. (2019, Nov 12).

<https://www.tensorflow.org>