

단일 입출력 심볼로 구성된 순환형 전자 회로도 영상의 네트리스트 변환

시종욱*¹, 김문년*², 김성영**

Converting Close-Looped Electronic Circuit Image with Single I/O Symbol into Netlist

Jongwook Si*¹, Moonnyeon Kim*², and Sungyoung Kim**

이 연구는 금오공과대학교 학술연구비로 지원되었음 (202000990001)

요 약

전자 도면이나 관련 서적에는 회로도가 많이 포함되어 있고 결과를 보기 위해서는 회로 구성 작업이 필요하다. 그런데 실제로 구성하는 것은 비용과 노동력의 낭비가 크기 때문에 컴퓨터 시뮬레이션 프로그램을 많이 사용한다. 하지만 시뮬레이션 프로그램은 사용자가 회로를 배치하고 값을 입력하는 등의 수동적인 작업이 필요하다. 이에 본 논문에서는 전자 회로의 도면 영상을 입력하면 자동으로 PSPICE 네트리스트로 변환하는 기술을 개발하였다. 우선 회로 심볼에 대한 학습을 통해 회로 심볼을 인식하고 인식한 심볼들을 도면에서 제거한다. 그리고 심볼이 제거된 도면에서 영상처리 기술을 활용하여 회로 심볼들을 연결하는 전선 사이의 관계를 분석한다. 마지막으로 심볼 인식 및 전선 분석 정보를 종합하여 PSPICE 네트리스트를 생성한다. 본 논문에서 제안한 방법은 단일 입출력으로 구성된 순환형 전자 회로도에 대해 우수한 성능을 나타내는 것을 확인하였다.

Abstract

Electronic drawings and related books contain many circuit diagrams and require circuit configuration to see the results. However, it uses a lot of computer simulation programs because it is a waste of money and labor that is actually constructed. However, simulation programs require passive tasks such as placing circuits and entering values by the user. Therefore, in this paper, we have developed a new method to convert drawing images of electronic circuits into PSPICE Netlists. First, circuit symbols are recognized through learning about circuit symbols. We then use image processing to analyze the relationship between wires connecting the recognized circuit symbols based on the drawing where the symbols were removed. Finally, we construct Netlists by combining the symbol recognition and wire analysis information. The results of the proposed method show high performance on cyclic electronic circuit diagrams consisting of a single I/O.

Keywords

object detection, PSPICE netlist, electronic circuit, automatic programming, image analysis

* 금오공과대학교 컴퓨터공학과 석사과정
- ORCID¹: <http://orcid.org/0000-0003-2092-2769>
- ORCID²: <http://orcid.org/0000-0001-6147-7476>
** 금오공과대학교 컴퓨터공학과 교수(교신저자)
- ORCID: <http://orcid.org/0000-0002-7722-6759>

• Received: Jun. 08, 2021, Revised: Jul. 16, 2021, Accepted: Jul. 19, 2021
• Corresponding Author: Sungyoung Kim
Dept. of Computer Engineering, Kumoh National Institute of Technology, 61
Daehak-ro (yangho-dong), Gumi, Gyeongbuk, [39177] Korea
Tel.: +82-54-478-7530, Email: sykim@kumoh.ac.kr

1. 서 론

회로도란 수많은 전선과 전자 소자로 구성된 전기 회로를 그림으로 표시한 것이다. 집을 짓기 위해서는 사전에 설계 도면을 작성하는 것과 같이 전자 기기를 제작하기 위해서는 전자 회로도를 작성한다. 이 단계가 전자 기기에 대한 설계 단계인데 회로 구현은 전자공학의 관련 지식이 있는 전문 인력, 관련 장비 및 노동력이 필요하다. 전자공학의 기초가 되는 “회로 이론” 과목에서는 간단한 회로도를 이용해 회로망을 해석한다. 이 과정에서 기존에 존재하는 전자 회로도를 동일하게 입력하는 경우가 빈번하게 발생한다.

OrCAD의 PSPICE[1] 등과 같이 전자 회로도를 시뮬레이션하는 소프트웨어를 사용하면 회로망에 대한 분석과 동작 여부를 확인할 수 있다. 이 프로그램은 사용자가 회로상에 전자 소자(이하 심볼로 표기)를 위치시키고 전선을 사용하여 이들을 연결하여 회로도를 구성하면 동작 결과를 시뮬레이션할 수 있다. 따라서 보드를 설계하기 전 시뮬레이션을 통해 설계된 회로에 대한 검증할 수 있게 지원한다는 장점이 있다. 하지만, 이러한 프로그램들은 심볼에 대해 관련 지식이 부족하거나 사용법을 모르는 일반 사용자들의 경우에 사용성이 떨어진다. 또한, 기존의 작성된 회로도를 동일하게 다시 작성하는 작업은 매우 번거로운 과정이다. 따라서 기존

에 존재하는 회로도를 디지털화하여 간편하게 시뮬레이션할 수 있도록 구성하는 기술이 필요하다.

본 논문에서는 회로도 영상을 입력으로 하여 출력 결과를 볼 수 있는 시스템을 소개한다. 영상처리와 딥러닝 기술을 사용하여 PSPICE에 적용이 가능한 넷리스트로 변환하는 것에 목표를 둔다. 그림 1은 PSPICE를 처음 실행했을 때 출력되는 템플릿을 저장한 넷리스트 파일(.sch)이다. 이 템플릿을 기반으로 필요한 전선이나 심볼을 추가한다. 넷리스트 변환에서의 중점적인 부분은 @parts, @conn 및 @junction이다. @parts는 회로 심볼에 대한 정보, @conn은 선의 좌표가 어느 전선과 연결된 것인지에 대한 라벨링을 나타낸다. @junction은 전선이 어느 좌표에서 어떤 좌표로 연결되는지에 대한 정보를 포함한다. 이 세 부분만을 정확하게 구성하면 프로그램을 통해 시뮬레이션이 가능하다.

논문의 구성은 다음과 같다. 2장에서는 심볼 및 라인(전선)을 검출하는 기존 연구와 프로그래밍 자동화 관련 연구를 소개한다. 그리고 객체 검출 연구 및 활용 분야들에 살펴본다. 3장에서는 제안하는 전자 회로 영상을 넷리스트로 변환하는 방법을 자세하게 설명한다. 4장에서는 각 회로 소자의 정확도, 변환 성공률 및 결과를 제시하고 분석하고 마지막으로, 5장에서는 제안 방법의 결론과 향후 과제에 대하여 제시한다.

```

*version 9.1 1592269696
u 2
@libraries
@analysis
@targets
@attributes
@translators
a 0 u 13 0 0 0 hln 100 PCBOARDS=PCB
a 0 u 13 0 0 0 hln 100 PSPICE=PSPICE
a 0 u 13 0 0 0 hln 100 XILINX=XILINX
@setup
unconnectedPins 0
connectViaLabel 0
connectViaLocalLabels 0
NoStim4ExtIFPortsWarnings 1
AutoGenStim4ExtIFPorts 1
@index
pageloc 1 0 389

@status
c 120:02:20:21:43:19;1584708199
*page 1 0 970 720 iA
@ports
@parts
part 1 titleblk 970 720 h
a 1 s 13 0 350 10 hcn 100 PAGESIZE=A
a 1 s 13 0 180 60 hcn 100 PAGETITLE=
a 1 s 13 0 300 95 hrn 100 PAGENO=1
a 1 s 13 0 340 95 hrn 100 PAGECOUNT=1
@conn
@junction
@attributes
a 0 s 0:13 0 0 0 hln 100 PAGETITLE=
a 0 s 0:13 0 0 0 hln 100 PAGENO=1
a 0 s 0:13 0 0 0 hln 100 PAGESIZE=A
a 0 s 0:13 0 0 0 hln 100 PAGECOUNT=1
@graphics
    
```

그림 1. PSPICE의 템플릿 넷리스트

Fig. 1. Template netlist in PSPICE

II. 관련 연구

2.1 심볼 및 전선 검출

회로도에서 심볼 및 전선 검출을 시도했던 기존의 연구들이 존재한다. 본 논문의 예비 연구[2]에서는 회로도 영상에 대하여 영상처리 기술을 사용하여 선을 분리하고 YOLO를 이용해 심볼을 검출하여 아이디어의 타당성을 검증하였다. 차재민 연구팀은 도면의 핵심 구성 요소인 심볼, 텍스트, 선 정보를 자동으로 추출하는 방법을 제시하고 기하학적 특징 정보를 이용하여 인식하는 기술을 제시하였다[3]. 하지만, 다양한 심볼에 대한 인식은 고려되지 않았다. 백영목 연구팀은 기존의 방대한 분량의 도면을 CAD 시스템에 적합하게 바꾸어 주는 것과 전자 회로 도면을 대상으로 하여 패턴 인식을 기반으로 문자와 그래픽 영역을 분리, 심볼 분리, 벡터화 및 인식하는 법을 제안하였다[4]. 하지만, 검출한 심볼과 전선에 대한 활용 방법을 확인할 수 없다. Vasudevan 연구팀은 머신러닝 및 영상처리 기술을 사용하여 많은 회로의 구성 요소들을 인식하여 프로그램에 입력을 가능하게 하였다[5]. 하지만, 전선의 연결성을 판단하지 않고 장애물로 판단하여 도면에서 지우므로 본 연구의 방향과는 큰 차이가 있다.

2.2 선 검출

영상에서 선을 검출하는 방법과 활용은 다양하게 개발되어 사용되고 있다. 강대현 연구팀은 선의 개수를 알고 있는 바둑판에 대하여 Canny, 허프 공간, 결정 모델을 사용하여 후보를 탐색하고, 선을 검출하는 방법을 제시하였다[6]. 김창곤 연구팀은 복사간에 열화가 되는 가는 선을 검출하기 위해 후보 화소를 도출하고 형태를 비교하는 방법을 제안하였다[7]. Wang 연구팀은 항공 영상의 전력선의 특징을 기반으로, 방향 필터링을 기반으로 실시간으로 전력선을 감지하는 방법을 제안하였다[8]. Xu 연구팀은 에지가 없는 변환에 대하여 DETR 등의 기술을 포함한 엔드투엔드 라인 세그먼트 알고리즘(LETR)을 제시하였다[9]. 이승목 연구팀은 불명확한 경계를 지닌 디스플레이 패널의 에지를 검출하기 위하

여 측면에 대하여 선을 검출하는 알고리즘을 제시하였다[10].

2.3 프로그래밍 자동화

Sketch2Code[11]은 AI를 기반으로 손으로 그린 HTML 내부의 그림을 프로토타입으로 변환하는 시스템을 제공한다. 윤영선 연구팀은 GUI 프로그래밍의 번거로운 해결에 초점을 맞추어, 캡처 영상 혹은 스케치 영상에 대해 코드를 자동화하여 생성한 결과를 보여주었다[12].

2.4 YOLO를 이용한 연구

YOLO v3[13]는 클러스터링을 통해 앵커 박스를 생성하여 바운딩 박스를 객체를 예측하는 객체검출 기술이다. 이 기술은 다양한 분야로 확장되어 활발한 연구들이 진행되고 있다. 오세량 연구팀은 시각 장애인들에게 유용한 의미를 전달하는 유도 블록을 인식하기 위해 YOLO를 기반으로 낮은 연산량의 알고리즘을 제시하였다[14]. 조국한 연구팀은 Lidar 센서와 360도 카메라를 결합하여 클라우드 환경에서 동작하는 영상인식 시스템을 구현하고, 클라우드에 저장된 영상 데이터들을 YOLO를 활용해 분석하였다[15]. 김청화 연구팀은 드론을 활용한 교통상황 모니터링 및 교통관제 분야 연구에서 촬영 위치와 조명 등의 취약점을 해결하기 위해 YOLO를 활용해 차량들을 검출하여 높은 성능을 보였다[16]. 김재정 연구팀은 환경 변화에 영향을 미치지 않는 자동차 번호판 실시간 인식 시스템을 구현하기 위해 YOLO를 이용하여 관심 영역의 번호판 영역을 추출하였다[17]. 시종욱 연구팀은 RGB 카메라를 기반으로 한 기차 예매 어플리케이션 음성 안내 시스템의 첫 번째 과정에서 YOLO를 사용하여 휴대폰 영역을 검출하고, 이 영역을 이용해 물체와의 원근을 인식하였다[18]. 최태민 연구팀은 기존 YOLO 모델의 성능을 향상시키기 위해 어텐션 YOLO를 제안하고 이를 이용해 질병과 관련된 특징이 존재하는 관심 영역의 중요도를 강화하여 성능을 개선시켰다[19]. 김인선 연구팀은 YOLO 기술을 활용해 관광지역을 축소해 제작한 미니어처 모형을 스캔하면

해당 관광지의 정보를 보여주는 기술을 제안하였다 [20]. 장재영 연구팀은 CCTV 환경에서 입력된 영상 정보를 기반으로 특정 인물에 대하여 빠르게 검색하고 실시간으로 전송하는 기술을 보였다[21].

III. 제안 방법

본 논문에서 제안하는 방법은 영상처리와 딥러닝을 융합한 영상분석 기술을 사용한다. 그림 2는 전체적인 과정을 나타낸다. 먼저 전자 회로도의 영상에서 심볼을 인식하여 파일로 저장한다(3.1절 참조). 그런데 심볼을 연결하는 전선과 이들의 끝점을 인식하기 위해 전자 회로도의 영상을 그대로 사용하면 전선과 심볼 내부의 선과 점을 혼동하는 확률이 매우 높다. 따라서 검출된 심볼을 배경색으로 지운 후에 전선과 이들의 끝점을 찾는다. 캐니, 모폴로지, 허프변환 및 세션화의 영상처리 기법을 사용하여 전선의 위치를 찾는다(3.2절 참조). 끝점의 경우 해리스 코너를 사용하여 검출한 후, 중복 제거 및 전선들의 군집화를 적용한다(3.3절 참조). 앞의 모든 정보들을 통합하여 PSPICE의 프로그램으로 실행 가능하도록 네트리스트로 변환한다(3.4절 참조).

3.1 심볼 인식

심볼을 검출하고 인식하기 위하여 YOLO v3[13]의 기술을 활용한다. 본 논문에서는 전자 회로도에서 상대적으로 빈번하게 사용되는 심볼을 선별하여

학습에 사용한다. 표 1은 선별한 13가지의 심볼의 종류를 나타낸다. 심볼이 검출되면 심볼의 종류(0~12)와 회로도에서의 상하좌우의 좌표값을 텍스트 파일에 차례대로 저장한다.

3.2 전선 검출

심볼을 인식한 결과의 좌표는 선 검출에 사용된다. 선 검출의 정확도를 높이기 위해 검출한 모든 심볼들은 원본 영상에서 삭제하여 심볼을 연결하는 전선만 존재하는 영상을 생성한다.

영상의 크기를 (400, 400)으로 고정후에 캐니를 수행하여 영상의 에지를 검출한다(그림 3(a)). 이어서 모폴로지의 침식, 팽창 연산을 통해 윤곽선을 병합한다(그림 3(b)). 마지막으로 세션화를 이용하여 최소한의 픽셀만 남기고 나머지를 제거하고(그림 3(c)), 허프변환으로 전선의 좌표들을 찾는다(그림 3(d)). 검출이 완료되면 입력과 동일한 크기의 영상으로 변환한다.

표 1. 사용한 심볼의 종류와 클래스 번호

Table 1. Symbol lists and corresponding class number

Num	Symbol name	Num	Symbol name
0	NOT	7	COIL
1	BUFFER	8	LED
2	DIOAD	9	CAPACITOR
3	ALTERNAT	10	SWITCH
4	0V COMMON	11	BATTERY
5	REGISTER(ALT)	12	DIRECT
6	REGISTER(IEC)		

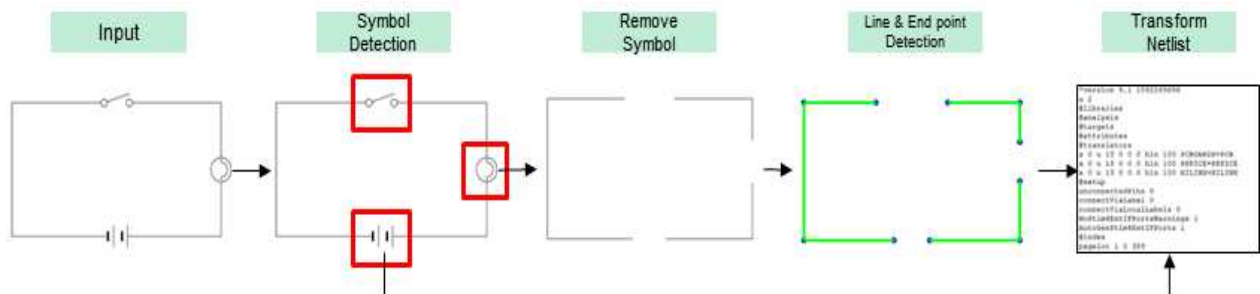


그림 2. 전체 프로세스
Fig. 2. Overall process

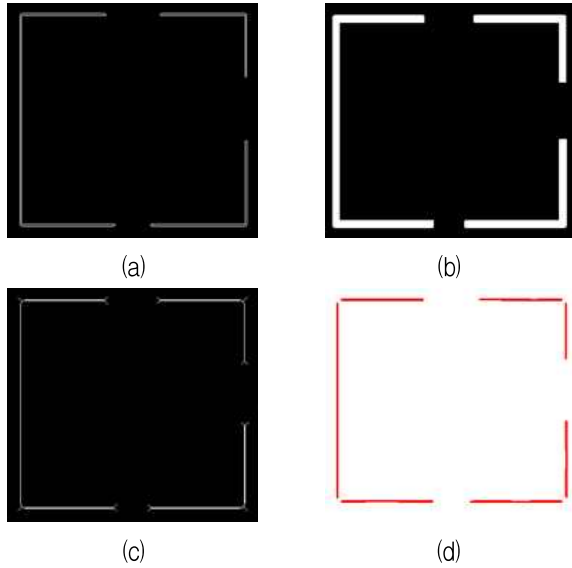


그림 3. 선 검출 과정
Fig. 3. Line detection process

3.3 끝 점 검출

전선 검출 결과 영상에 해리스코너를 적용하여 코너점의 위치를 검출한다(그림 4(a)). 하지만, 제약 조건 없이 수행하면 같은 코너에 여러 개의 코너점이 검출되는 문제점이 발생한다. 따라서 본 논문에서는 코너점을 찾을 때 이미 찾았던 코너점과 거리가 10이하인 새로 발견한 코너점은 무시한다(그림 4(b)). 또한, 코너점들의 좌표 값들로 연결성을 찾기 위해, 검출된 좌표들을 후처리 과정을 거쳐 10의 배수로 좌표들을 수정한다(그림 4(c)). 수정된 좌표들이 같은 행 또는 열에 위치하면 x 나 y 의 좌표는 동일하게 된다. 이후 수정된 좌표들의 값들을 이용하여 연결성을 찾는다. 먼저 가로 방향의 연결성을 찾기 위해 x 좌표가 같고 y 좌표가 다른 점들을 찾는다. 이어서 세로 방향을 고려하여 x 좌표가 같고 y 좌표가 다른 점들을 찾는다. 좌표와의 거리가 가장 작은 좌표 사이에 심볼의 영역이 없다면, 두 좌표 사이에 선을 그리고 그 선의 값을 저장한다. 위의 과정을 모든 좌표들에 대해 완료될 때까지 진행한다. 저장된 선들의 정보들을 활용해, 선의 시작점과 끝 점이 같은 선들을 하나의 선으로 인식하고 연결성을 포함한 내용들을 저장한다(그림 4(d)).

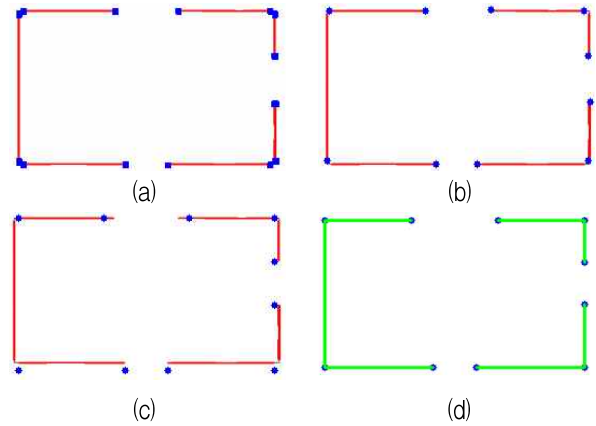


그림 4. 끝 점 검출 과정
Fig. 4. End point detection process

3.4 네트리스트 변환

3.4.1 PSPICE에서의 심볼

심볼의 방향은 수평 및 수직의 두 가지 경우로 가능하며 네트리스트에서는 h , v 로 표현한다. 심볼에 대한 최소 경계 사각형의 가로 길이가 더 긴 경우는 수평, 세로 길이가 더 긴 경우 수직으로 인식한다. 심볼의 고정된 크기는 종류별로 상이하며 검출된 심볼의 좌표와는 무관하게 중간값을 사용한다. 따라서 수평의 경우, 좌표는 [좌측, 상하평균], [좌측+크기, 상하평균]이 된다. 그리고 수직의 경우는 [좌우평균, 하측-크기], [좌우평균, 하측]로 표시한다. 심볼이 다르더라도 목적이 같은 경우에 PSPICE에서는 동일한 Class로 분류한다. Class에 따른 심볼 개수를 네트리스트 파일 상단에 기록하며 이는 C, D, V, R 등의 다양한 식별자를 통해 분류된다. 표 2는 PSPICE에서 사용되는 심볼의 이름, 크기 및 클래스를 나타낸다.

3.4.2 전선의 끝 점 검출

선의 연결성을 알기 위해서는 선의 시작과 코너, 끝을 알아야 한다. 이는 연결된 전선의 모든 좌표들을 모두 리스트에 넣은 후 중복을 제거한다. 좌표들의 집합에서 같은 좌표의 개수가 1인 경우에는 이어지는 좌표가 아니므로 끝 점이라 판단한다.

표 2. PSPICE에서의 심볼의 세부사항
Table 2. Symbol's detail in PSPICE

Num	Symbol name	Size	Class
0	7404	70	
1	7407	70	
2	d	10	D
3	vpwl	40	V
4	EGND	0	
5	r	40	R
6	r	40	R
7	L	60	
8	D1N750	30	D
9	c	30	C
10	Sw_tOpen	40	U
11	vdc	40	V
12	vdc	40	V

3.4.3 끝 점 좌표 수정

끝점의 좌표와 수정된 심볼의 좌표가 정확하게 일치한다고 보장할 수 없다. 따라서 끝점의 좌표를 심볼의 수정된 좌표로 연결시켜야 한다. 이는 끝점을 하나 선택하여 유클리디안 거리를 계산하여 검출된 심볼들 중 가장 가까운 심볼의 좌표를 찾는

후 선택한 끝 점의 좌표를 심볼의 좌표로 변경한다. 수정된 좌표를 이용해 @part와 @conn에 기입한다.

3.4.4 심볼간의 전선 연결

좌표만 같다고 연결이 되는 것이 아니라 심볼 자체에 전선을 연결하는 기능이 있어야 한다. 이는 네트리스트에서 이러한 부분을 @junction에 기입한다. 해당 junction의 약자인 'j'와 심볼에 연결될 좌표를 기입하고 그 아래에 '+'라는 문자와 Part의 약자인 'p'를 기입한다. 해당 좌표가 좌측이나 하측인 경우 1, 우측이나 상측인 경우 2를 기입한다. 그리고 그 아래에 해당 연결되는 선을 표시하기 위해 그룹화된 선을 기입한다.

IV. 성능 평가 및 결과

4.1 심볼 검출

심볼을 검출하기 위해 본 논문에서는 YOLO v3[13] 모델을 사용하여 진행하였다. 그림 5는 학습에 사용한 심볼 당 레이블의 개수를 나타낸다.

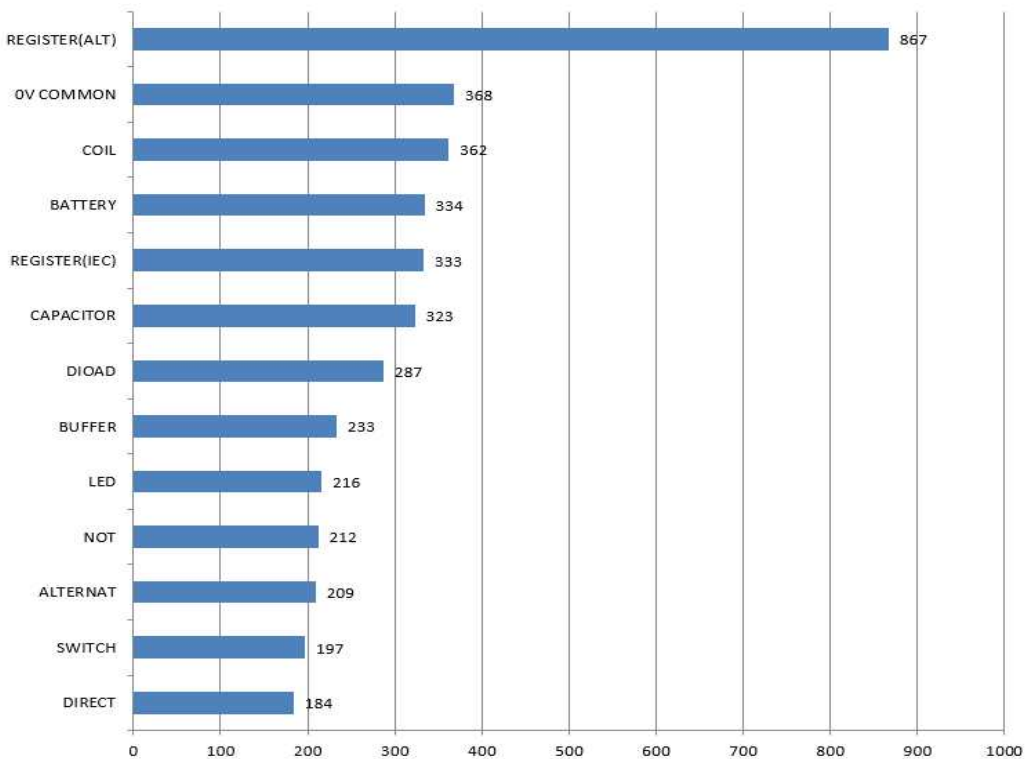


그림 5. 각 심볼의 레이블 개수
Fig. 5. Number of label about each symbols

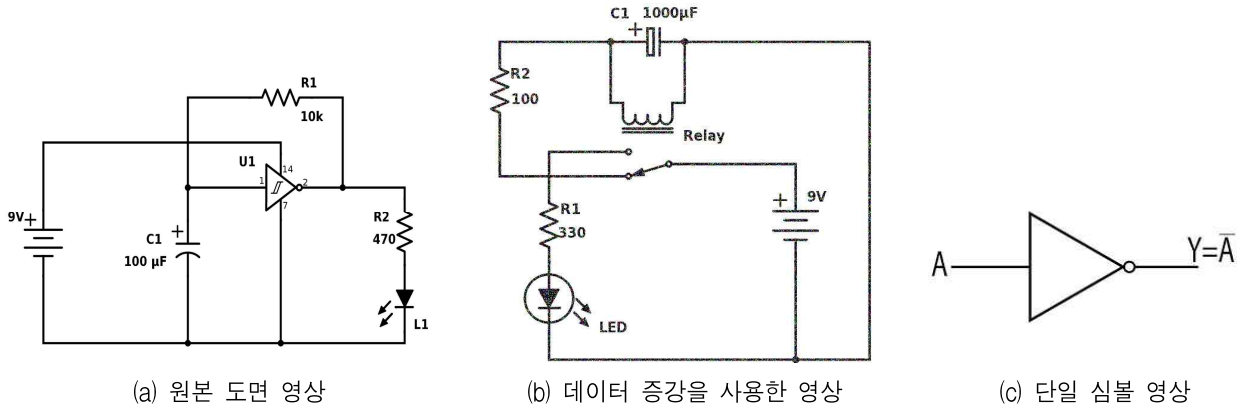


그림 6. 학습에 사용한 영상 샘플

Fig. 6. Sample images for training (a) Original circuit image, (b) image using data augmentation, (c) Single symbol image

Register(Alt)가 867개로 가장 많았으며 총 4,125개의 심볼 데이터를 이용하여 학습을 진행하였다. 학습에 사용한 회로도 영상은 127장이며(그림 6(a)), 데이터 증강을 통해 추가로 70장을 생성했다(그림 6(b)). 또한, 회로 없이 심볼만 포함된 영상 119장을 포함하여 총 316장을 학습에 사용하였다(그림 6(c)).

Ubuntu 18.04 LTS 환경에서 GTX TITAN Z를 사용하여 학습을 진행하였다. 학습에는 20시간 30분이 소요되었고 최종 Loss는 0.4151이다. 그림 7은 객체 검출 성능이 낮은 CAPACTOR(a)와 DIRECT(b)의 심볼인데 모양이 매우 비슷하여 서로 오인식하는 경우가 많았다. 표 3은 각 심볼 검출의 정확도를 나타낸다. 0V COMMON, 레지스터 및 배터리에서 90% 이상의 높은 성능을 보인다.

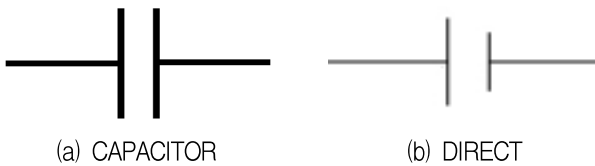


그림 7. 모양이 비슷한 심볼의 예시
Fig. 7. An example of two symbols with similar shape

4.2 PSPICE 실행 결과

본 논문에서는 제안한 방법을 통해 변환된 넷리스트를 PSPICE(ver. 9.1)으로 직접 실행하여 성능 평가를 진행한다. 본 절에서는 실행 결과에 대한 정확도를 측정하므로 넷리스트로의 변환(T)뿐만 아니라, 객체 검출이 모두 성공한 상황(O) 및 끝점이

정확하게 검출된 상황(E)에 대한 조건부 확률을 이용해 평가한다. 표 4는 단일 입출력 심볼로만 구성된 순환적인 회로에서 심볼의 개수(N)에 따라 측정된 결과이다. 성능 평가에는 36장의 회로 영상을 사용하며, 전체적인 성능은 N 이 작은 경우에 높게 평가된다.

표 3. 심볼 검출 정확도

Table 3. Detection accuracy of symbols

Symbol	Accuracy
NOT	86.7%
BUFFER	80.0%
DIOAD	87.8%
ALTERNAT	63.6%
0V COMMON	97.1%
REGISTER(ALT)	93.9%
REGISTER(IEC)	90.9%
COIL	75.0%
LED	66.7%
CAPACITOR	57.1%
SWITCH	62.5%
BATTERY	94.0%
DIRECT	55.6%

표 4. PSPICE를 실행을 통한 넷리스트 변환 성능

Table 4. Improvement of transformation to netlist using PSPICE

N	$P(T)$	$P(T O)$	$P(T E)$
1	90.0%	90.0%	100%
2	72.7%	80.0%	88.9%
3	66.7%	75.0%	85.7%
4	50.0%	75.0%	75.0%

$P(T|O)$ 은 심볼을 정확히 검출했을 때, 변환 성공률을 나타낸 것이며 검출하지 못한 심볼이 있는 4장의 영상이 제외된다. 그리고 $P(T|E)$ 은 끝점까지 모두 정확히 검출했을 때의 변환 성공률을 나타낸 것이며, 검출 오류가 있는 3장의 사진이 추가로 제외된다. 실험 결과를 살펴보면 N 이 작을수록 성능이 높다. 그런데 단계마다 성능 차이가 큰 것을 보아, 심볼 검출이나 끝점을 제대로 찾는 것이 변환 성능이 큰 영향을 주는 것으로 보인다. 이는 영상마다 선의 두께 및 검출된 점의 수가 상이하기 때문이다. 전체 변환 과정에서 오류 없이 성공하면, PSPICE 프로그램 내부에서 회로를 출력한다. 만약 변환에 성공하지 못하면 아무런 출력을 내지 않는다. 변환에 실패한 이유는 매우 다양하며, 대부분은 객체 검출의 오류로 선과 심볼이 연결되지 않았기 때문이다. 그림 8은 넷리스트 변환을 성공한 PSPICE 결과를 나타낸다. 그림 8(a)는 객체 검출이 모두 정확히 이루어진 결과를 나타내며, 이를 최종 변환에 성공할 시 그림 8(b)와 같이 시뮬레이션 회로의 결과가 나타난다.

V. 결론 및 향후 과제

본 논문에서는 기존에 존재하지 않았던 방식의 전자 회로도 영상 변환 시스템을 제안하였다. YOLO v3를 사용하여 심볼을 검출하고 각종 영상 분석 기술을 활용하여 넷리스트로 변환하는 시스템을 제안하였다. 실험 결과로 간단한 회로의 환경 그리고 입출력이 하나인 경우에 변환이 잘 이루어지며, 회로 심볼의 개수에 따라 성능 차이가 있음을 확인하였다. 하지만, 회로 영상을 순환형으로 고정할 것이 아니라, 다양한 형태의 변환도 필요하다.

또한, 심볼 옆에 기입되어 있는 각 심볼의 값을 인식하고 다중 입력 게이트에 대해서도 변환이 이루어져야 한다. 향후에는 다중 입력에 대한 심볼 및 조금 더 복잡한 회로 영상에 대한 변환을 하기 위해 선을 연결하는 알고리즘에 대하여 연구해 볼 필요가 있다. 심볼의 검출 정확도를 높이기 위하여 두께와 크기를 동일하게 표현하여 심볼의 일관성을 높여야 한다.

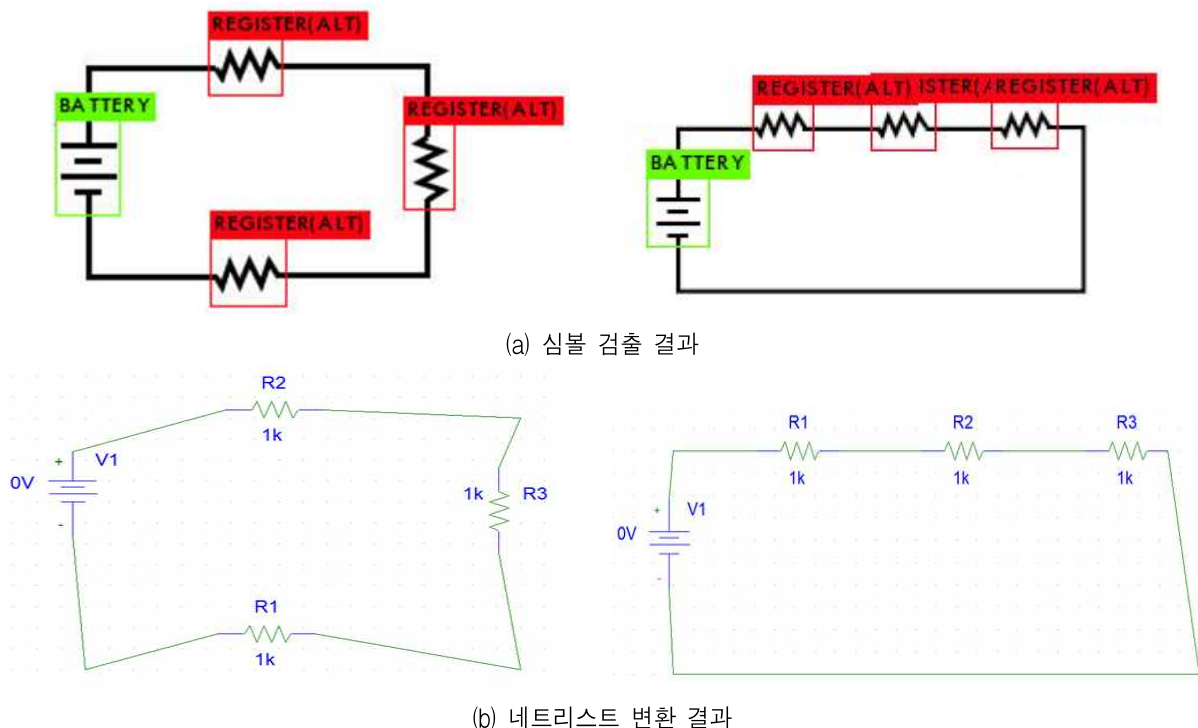


그림 8. 객체 검출 및 넷리스트 변환 결과

Fig. 8. Results of symbol detection and Netlist conversion, (a) Results of symbol detection, (b) Results of Netlist conversion

References

- [1] <https://www.orcad.com/pspice-free-trial> [accessed: Apr. 31, 2021]
- [2] J. Si, M. Kim, and S. Kim, "Wire Removal and Recognition on Circuit Elements on Electronic Schematics", Proc. Of International Workshop on Future Technology, pp. 68-70, Jan. 2020.
- [3] J. Cha, B. Ryu, S. Salim, and T. Lee, "A Preliminary Study on Recognition of Equipment Symbols from Image P&IDs based on Deep Learning", Proc. Of Korean Society of Mechanical Engineers, pp. 1761-1765, Dec. 2018.
- [4] Y. Baek, J. Seok, S. Chien, and C. Hwang, "The Vectorization and Recognition of Circuit Symbols for Electronic Circuit Drawing Management", Journal of Institute of Electronics and Information Engineers, Vol. 33, No. 3, pp. 602-611, Mar. 1996.
- [5] S. Vasudevan, K. Venkatachalam, S. Anandaram, and A. Menonm, "A Novel Method for Circuit Recognition Through Image Processing Techniques", Asian Journal of Information Technology, Vol. 15, No. 7, pp. 1146-1150, Jul. 2016. <http://dx.doi.org/10.36478/ajit.2016.1146.1150>.
- [6] D. Kang and Y. Lee, "Algorithm of detecting Go board lines", Proc. Of Institute of Electronics and Information Engineers, pp. 803-806, Jun. 2016.
- [7] C. Kim, J. Bae, C. Kim, H. Park, and K. Kang, "Thin Line Detection and Restoration Method for Digital Copy Machine", Proc. Of Korean Institute of Electrical Engineers, pp. 312-313, Oct. 2015.
- [8] D. Wang, T. Zhu, A. Xie, N. Fang, and H. Liu, "Research on Image Recognition Algorithm Technology for Power Line Detection", Journal of Physics Conference Series, Vol. 1732, pp. 1-6, Jan. 2021, <http://doi.org/10.1088/1742-6596/1732/1/012082>.
- [9] Y. Xu, W. Xu, D. Cheung, and Z. Tu, "Line Segment Detection Using Transformers without Edges", IEEE International Conference on Computer Vision & Pattern Recognition, pp. 4257-4266, Jun. 2021, <https://arxiv.org/abs/2101.01909>.
- [10] S. Lee, Y. Kim, and J. Eem, "A Method of Edge Line Detection for Noisy Panel Module Images", Journal of Korean Institute of Information Technology, Vol. 18, No. 7, pp. 75-80, Jul. 2020, <http://dx.doi.org/10.14801/jkiit.2020.18.7.75>.
- [11] Sketch2Code, <https://www.microsoft.com/en-us/ai/ai-lab-sketch2code> [Accessed: Apr. 31, 2021]
- [12] Y. Yun, J. Park, J. Jung, S. Eun, S. Cha, and S. So, "Automatic Mobile Screen Translation Using Object Detection Approach Based on Deep Neural Networks", Journal of Korea Multimedia Society, Vol. 21, No. 11, pp. 1305-1316, Nov. 2018, <https://doi.org/10.9717/kmms.2018.21.11.1305>.
- [13] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, RealTime Object Detection", IEEE International Conference on Computer Vision & Pattern Recognition, pp. 779-788, May 2016, <https://arxiv.org/abs/1506.02640>.
- [14] S. Oh and Y. Bae, "Braille Block Recognition Algorithm for the Visually Impaired Based on YOLO V3", Journal of Korean Institute of Intelligent Systems, Vol. 31, No. 1, pp. 60-67, Feb. 2021, <https://doi.org/10.5391/JKIIS.2021.31.1.060>.
- [15] G. Jo, H. Cho, and Y. Song, "Implementation of 360-degree Image Recognition System Using Lidar Sensors and Yolo v3 Libraries in Cloud Server Environment", Journal of Korean Institute of Information Technology, Vol. 18, No. 2, pp. 1-7, Feb. 2020, <https://doi.org/10.14801/jkiit.2020.18.2.1>.
- [16] C. Kim and G. Park, "A Study of Traffic Detection and Classification using Yolo", Proc. Of Korean Institute of Broadcast and Media Engineers, pp. 80-82. Nov, 2019.
- [17] J. Kim and C. Kim, "Implementation of Robust

License Plate Recognition System using YOLO and CNN", Journal of Korean Institute of Information Technology, Vol. 19, No. 4, pp. 1-9, Apr. 2021, <https://doi.org/10.14801/jkiit.2021.19.4.1>.

[18] J. Si, C. Lee, and S. Kim, "Development of a Train Reservation Application Voice Guide System Using RGB Camera", Proc. Of KIIT summer Conference, pp. 29-31. Jun. 2021.

[19] T. Choi, C. Son, and D. Lee, "Attention-Enhanced YOLO Model via Region of Interest Feature Extraction for Leaf Diseases Detection", Journal of Korean Institute of Information Technology, Vol. 19, No. 4, pp. 83-93, Apr. 2021, <https://doi.org/10.14801/jkiit.2021.19.4.83>.

[20] I. Kim, C. Jeong, and K. Jung, "AR Tourism Service Framework Using YOLOv3 Object Detection", Journal of The Institute of Internet, Broadcasting and Communication, Vol. 21, No. 1, pp. 195-200, Feb. 2021, <https://doi.org/10.7236/JIIBC.2021.21.1.195>.

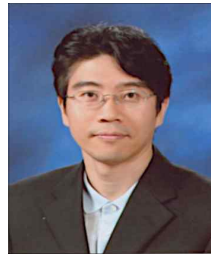
[21] J. Chang, C. Kang, J. Yoon, J. Cho, J. Jung, and J. Chun, "Development of Real-time Video Search System Using the Intelligent Object Recognition Technology", Journal of The Institute of Internet, Broadcasting and Communication, Vol. 20, No. 6, pp. 85-91, Dec. 2020, <https://doi.org/10.7236/JIIBC.2020.20.6.85>.

김 문 년 (Moonnyeon Kim)



2021년 2월 : 금오공과대학교
컴퓨터공학과(공학사)
2021년 3월 ~ 현재 :
금오공과대학교 컴퓨터공학과
대학원(석사과정)
관심분야 : 컴퓨터비전, 객체검출

김 성 영 (Sungyoung Kim)



1994년 2월 : 부산대학교
컴퓨터공학과(공학사)
1996년 2월 : 부산대학교
컴퓨터공학과(공학석사)
2003년 8월 : 부산대학교
컴퓨터공학과(공학박사)
2004년 ~ 현재 : 금오공과대학교

컴퓨터공학과 교수
관심분야 : 영상처리, 컴퓨터비전, 기계학습, 딥러닝

저자소개

시 종 욱 (Jongwook Si)



2020년 8월 : 금오공과대학교
컴퓨터공학과(공학사)
2020년 9월 ~ 현재 :
금오공과대학교 컴퓨터공학과
대학원(석사과정)
관심분야 : 스타일변환, 영상분석,
영상생성, 인공지능