FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG

FACULTY OF ENGINEERING

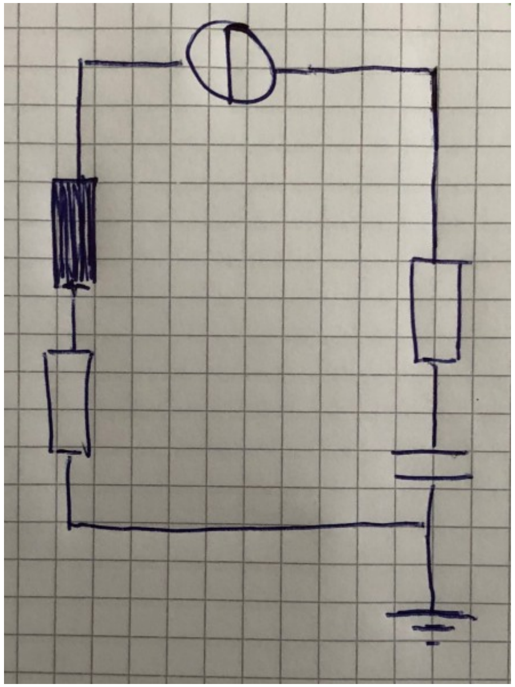
Detection of Hand Drawn Electrical Circuit Diagrams and their Components using Deep Learning Methods and Conversion into LTspice Format

Master Thesis: Intro Talk

Dmitrij Vinokour

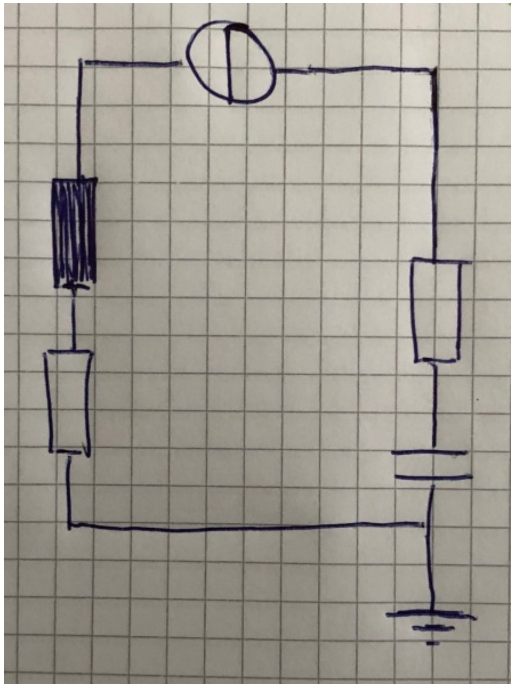
Supervised by Florian Thamm M. Sc, Felix Denzinger M. Sc., Prof. Dr. Andreas Maier
Pattern Recognition Lab, Friedrich-Alexander University Erlangen-Nuremberg

Introduction

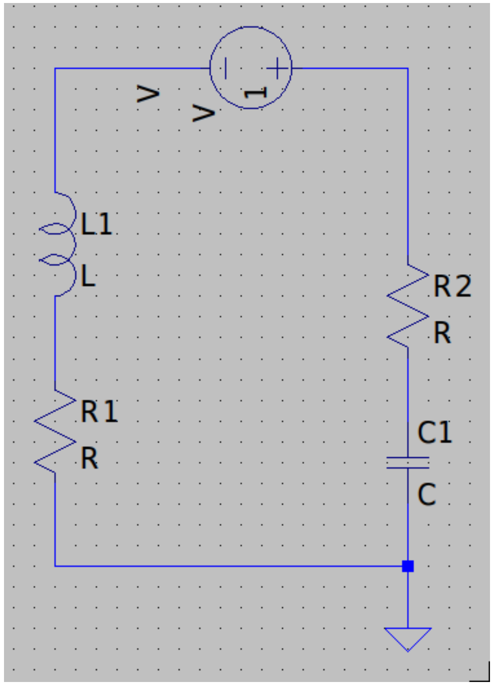


Hand-Drawn Electrical Circuit

Introduction



Hand-Drawn Electrical Circuit



Model in LTspice

Motivation

- Drawing speed of hand-drawn diagrams ~90% faster than drawing in a tool (UML, flow charts, Markov chains etc. vs. Microsoft Visio) [1]

	Real World	LTspice World
Drawing	Fast	Slow
Calculating	Slow	Fast

[1]: Khaled S. Refaat et al., A New Approach for Context-Independent Handwritten Offline Diagram Recognition using Support Vector Machines. 2008.

Motivation

- Drawing speed of hand-drawn diagrams ~90% faster than drawing in a tool (UML, flow charts, Markov chains etc. vs. Microsoft Visio) [1]

	Real World	LTspice World	This Thesis
Drawing	Fast	Slow	Fast
Calculating	Slow	Fast	Fast

[1]: Khaled S. Refaat et al., A New Approach for Context-Independent Handwritten Offline Diagram Recognition using Support Vector Machines. 2008.

Goals & Constraints

Goals:
- Create a dataset
- Detect circuit components and their annotations
- Detection should be possible on gridded paper
- Conversion into LTspice file
- Maintain appearance of the circuits after conversion to LTspice

Constraints:
- Only vertical / horizontal components are allowed
- Only subset of possible circuit components allowed
- No skip connections

Dataset

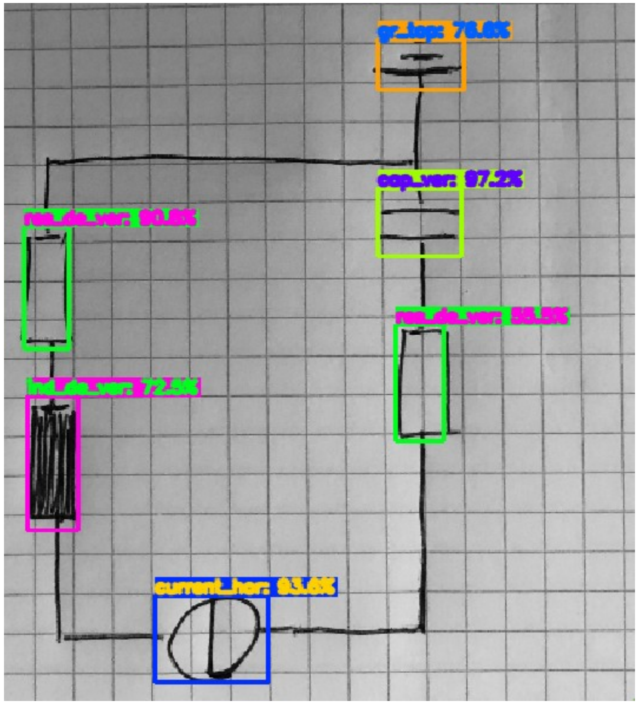
- 20 contributors, ~120 circuits
- Bounding boxes labeled with rotation
- Semi-auto generated labels for segmentation
- Very low amount of circuits on grid paper
- For training circuit masks are projected on grid paper

Capacitor	475
Inductor	363
Resistor	382
Diode	452
Voltage	241
Current	247
Ground	397

Bounding boxes per class

Current Pipeline

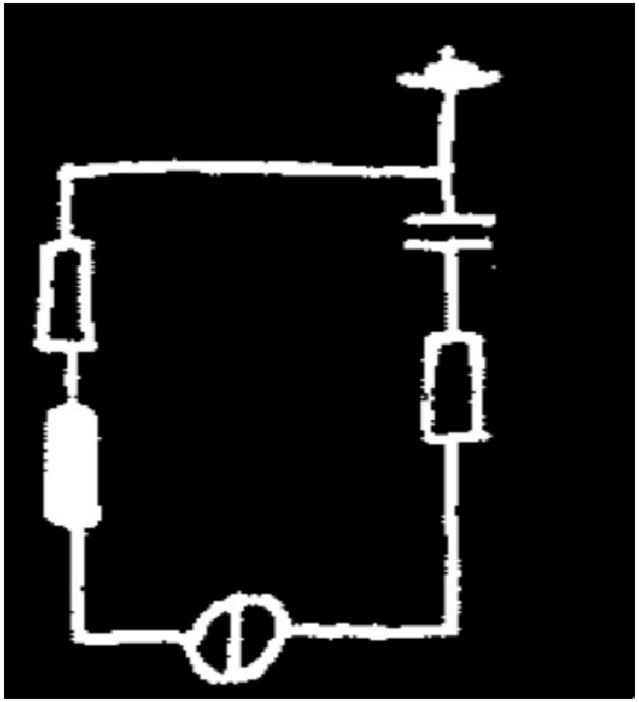
1. Detect components and annotations (tiny-YOLOV4)



Prediction with YOLO

Current Pipeline

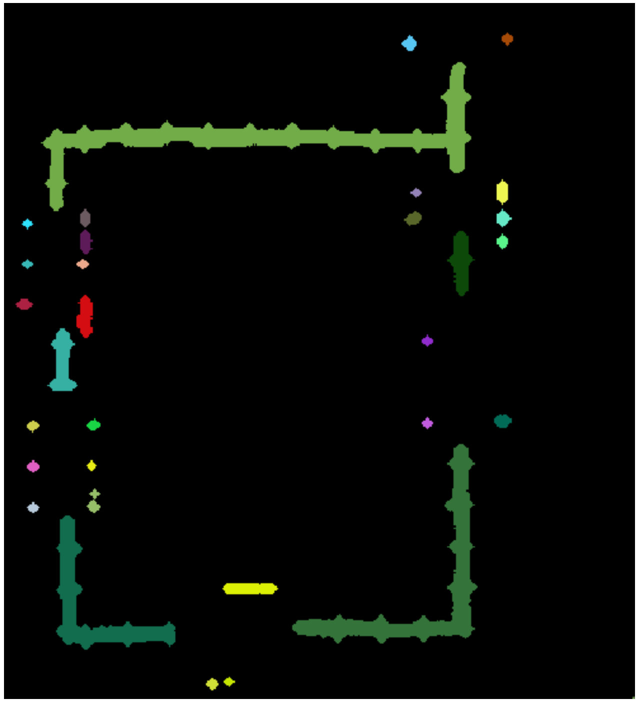
1. Detect components and annotations (tiny-YOLOV4)
2. Segment the circuit from the background (Mobile-UNet)



Prediction with Mobile-UNet

Current Pipeline

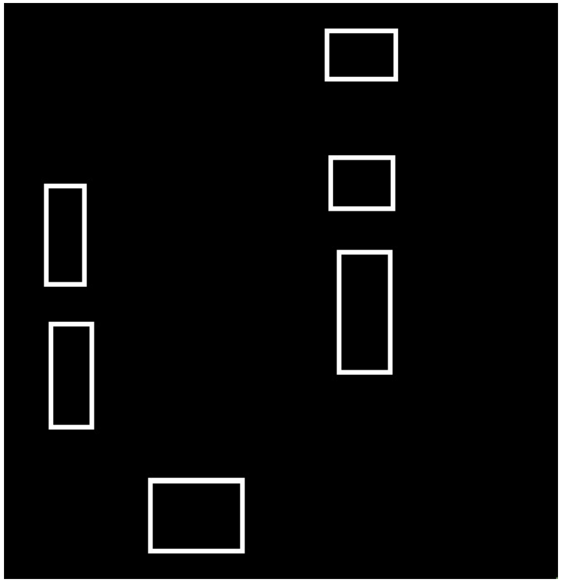
1. Detect components and annotations (tiny-YOLOV4)
2. Segment the circuit from the background (Mobile-UNet)
3. Remove bboxes and apply connected components analysis



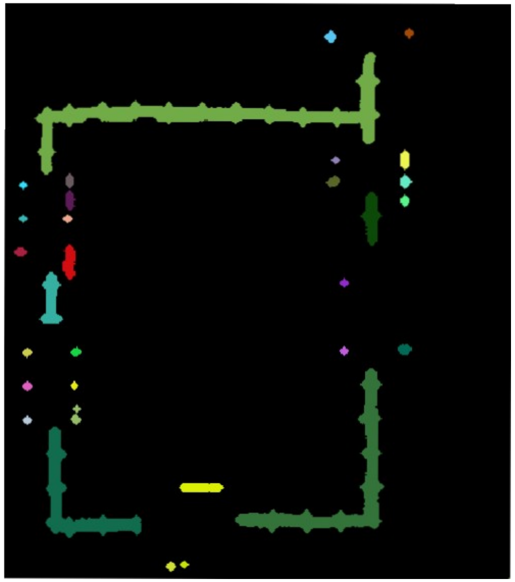
Connected components

Current Pipeline

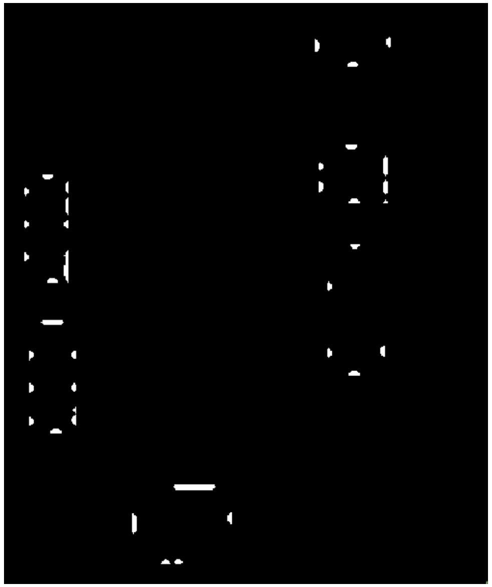
1. Detect components and annotations (tiny-YOLOV4)
2. Segment the circuit from the background (Mobile-UNet)
3. Remove bboxes and apply connected components analysis
4. Create bbox mask and find intersections with connected components



Bounding box mask



Connected components

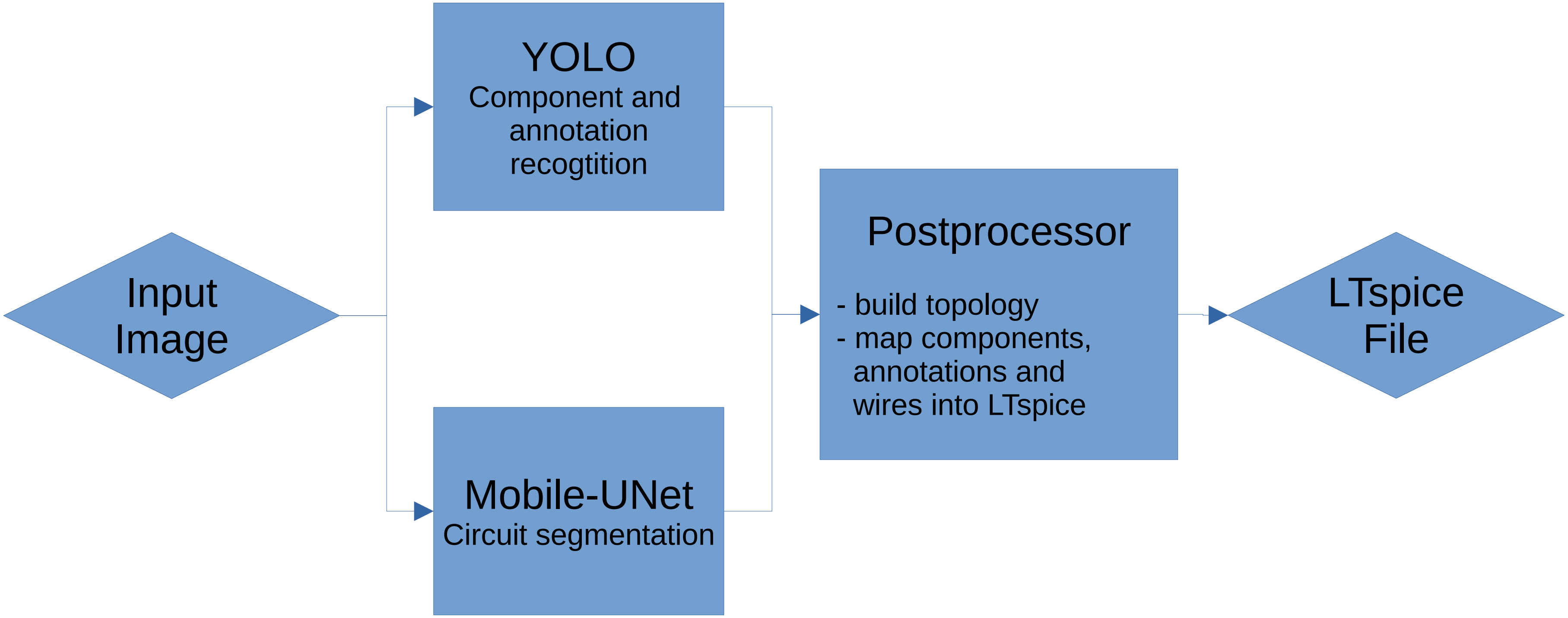


Intersections

Current Pipeline

1. Detect components and annotations (tiny-YOLOV4)
2. Segment the circuit from the background (Mobile-UNet)
3. Remove bboxes and apply connected components analysis
4. Create bbox mask and find intersections with connected components
5. Embed gathered data into a LTspice file
    a) Map detected components and annotations to LTspice
    b) Reflect wires to LTspice

Current Pipeline



Input Image

YOLO
Component and annotation recogtition

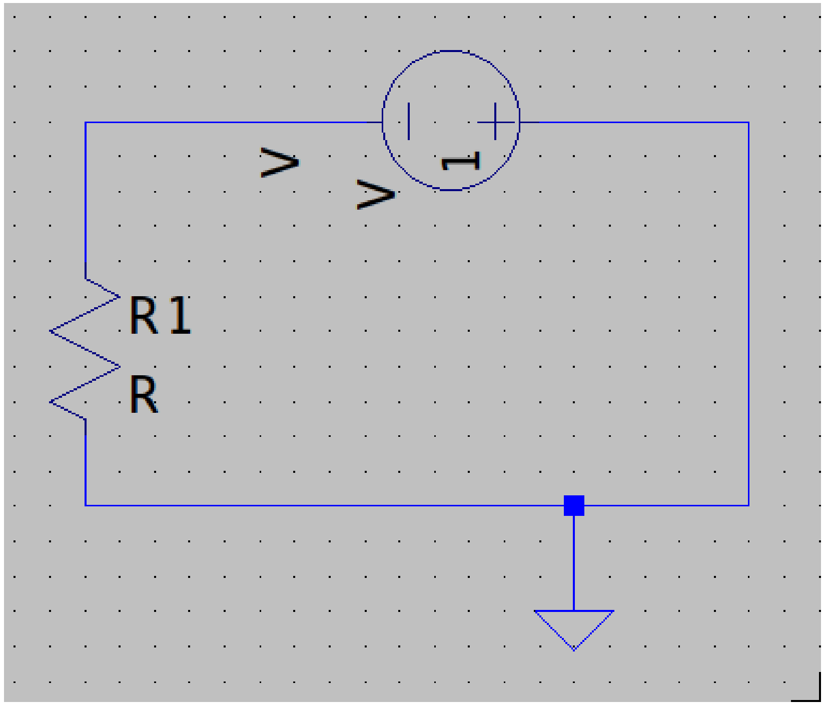
Mobile-UNet
Circuit segmentation

Postprocessor

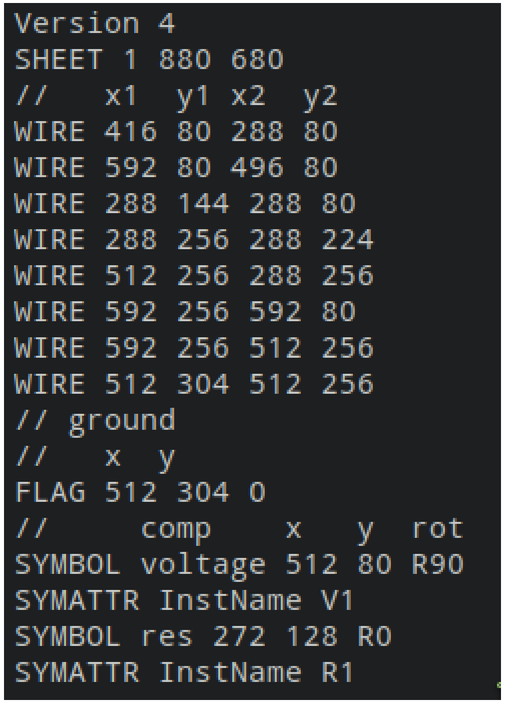
- build topology
- map components, annotations and wires into LTspice

LTspice File

LTspice file



Model in LTspice



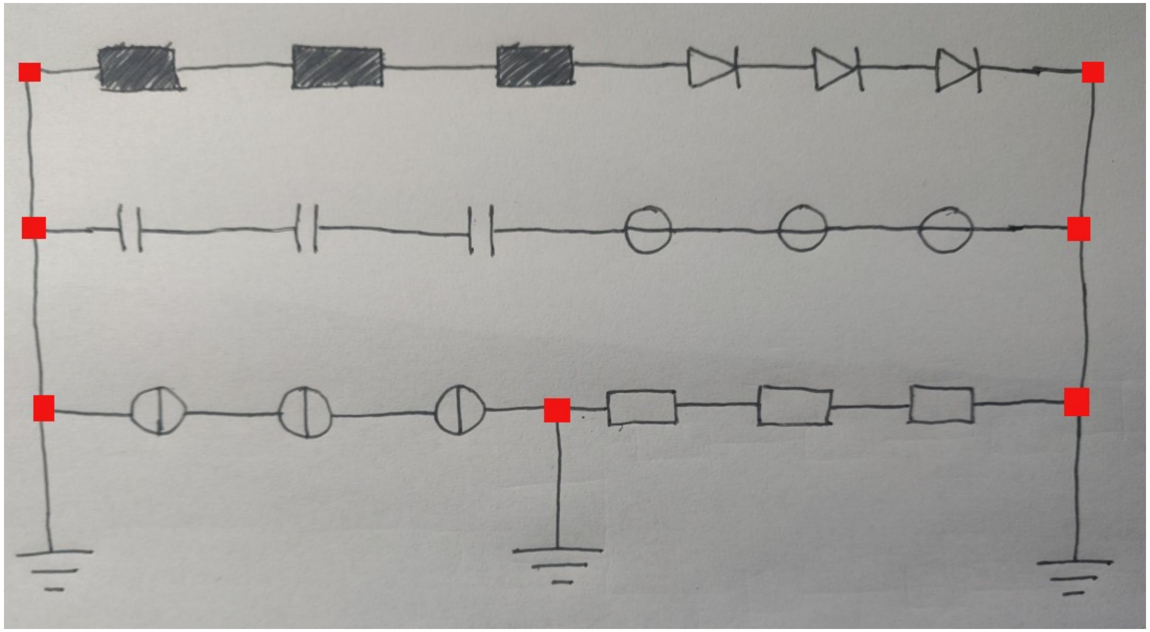
```
Version 4
SHEET 1 880 680
//    x1   y1  x2   y2
WIRE 416 80 288 80
WIRE 592 80 496 80
WIRE 288 144 288 80
WIRE 288 256 288 224
WIRE 512 256 288 256
WIRE 592 256 592 80
WIRE 592 256 512 256
WIRE 512 304 512 256
// ground
//    x   y
FLAG 512 304 0
//       comp     x    y   rot
SYMBOL voltage 512 80 R90
SYMATTR InstName V1
SYMBOL res 272 128 R0
SYMATTR InstName R1
```

Corresponding .asc

Problems

1. Reflect appearance is hard
   - Component size in the real world not the same as in LTspice
   - Wires have to be interpolated to be be reflectable

   Idea:
   - Split wire into horizontal / vertical sections
2. Assigning annotations can be ambiguous



Connections of interest (components > 2)

Results

- Performed on validation data (22% of the dataset, 2 unseen persons)

Class	AP@0.5	Class	AP@0.5
Diode Left	1.0	Ground Left	0.93
Diode Top	1.0	Ground Top	1.0
Diode Right	1.0	Ground Right	1.0
Diode Bottom	1.0	Ground Bottom	1.0
Resistor Horizontal	1.0	Inductor Horizontal	1.0
Resistor Vertical	1.0	Inductor Vertical	1.0
Capacitor Horizontal	1.0	Voltage Horizontal	1.0
Capacitor Vertical	0.97	Voltage Vertical	1.0
Current Horizontal	1.0		
Current Vertical	1.0		

Average precision @0.5IoU (YOLO)

mAP@0.5	0.993
mAP@0.5:0.05:0.95	0.636

Mean average precision (YOLO)

mIoU	0.910

Mean IoU (Mobile-UNet)

Summary & Outlook

Done:

- Dataset created
- Component detection
- Circuit segmentation
- Preprocessing to generate circuit topology
- Module to generate LTspice files

Next:

- OCR for annotations
- Replace YOLO with Mobile-UNet
- Optimize training (hyperparameters, augmentations)
- Fix the appearance after conversion

Maybe:

- Provide mobile application
- Release horizontal / vertical restriction

Questions?