Hand-Drawn Electrical Circuit Recognition Using Object Detection and Node Recognition

Ashi Sharma
BITS Pilani
Pilani, India
2022B2A31692P

Raghav Dhawan
BITS Pilani
Pilani, India
2022B1A81010P

Ojasvi Kaswan
BITS Pilani
Pilani, India
2022B2A31338P

***Abstract*—An electrical circuit is fundamentally composed of component symbols and their interconnections. This project proposes a robust system for the Automatic Recognition and Reconstruction of Hand-Drawn Electrical Circuits using a hybrid Deep Learning and Heuristic pipeline. Unlike traditional methods that rely on raw image segmentation, this approach utilizes the YOLO (You Only Look Once) object detection model to simultaneously locate and classify component symbols with high efficiency. Following detection, a novel Heuristic Terminal Estimation algorithm is applied to mathematically predict the exact coordinates of component connection points ("pins") based on bounding box geometry, overcoming the challenge of identifying implicit terminals. The circuit topology is then reconstructed by analyzing wire skeletonization and node detection to map the connectivity between these estimated terminals. The final output of the system is a simulation-ready SPICE netlist, allowing the hand-drawn sketch to be directly utilized in circuit simulation software.**

***Index Terms*—Object Detection, YOLO, Circuit Recognition, SPICE Netlist, Deep Learning, Heuristic Estimation**

I. Introduction

Circuit simulation programs are a very efficient and fast way of observing the behaviors of electrical circuits. They are fundamental tools for engineers to analyze the characteristics of circuits and get simulation results. To use the benefits of simulation programs, engineers need a circuit schematic which contains components and the interconnections between them.

The simulation programs provide a graphical interface to create circuit schematics. However, it is popular to create initial concepts of electrical circuits using pencil and paper. Because of this, engineers spend additional time manually transforming hand-drawn circuits into the required schematic file for the simulation program, a process that can be frustrating and labor-intensive. This project explains a method to automate the creation of a schematic file directly from a hand-drawn circuit sketch.

There are significant challenges in offline circuit sketch recognition, primarily because the temporal information (stroke order) available in online systems is missing. This lack of temporal data makes it difficult to differentiate between symbols and connectors using traditional segmentation. While earlier research utilized techniques like block adjacency graphs (BAG) or Fourier descriptors with SVM classification, these methods often struggle with the noise and variability inherent in static images.

This project proposes a robust Deep Learning and Heuristic pipeline to solve these challenges. Instead of traditional segmentation, we employ YOLO for real-time object detection to identify component classes and their bounding boxes with high confidence. To address the difficulty of detecting precise connection points, we introduce a Heuristic Terminal Estimation algorithm that mathematically predicts component "pins" based on the detected bounding box geometry. Finally, the system utilizes Wire Skeletonization and Node Recognition to map the circuit topology, ultimately generating a simulation-ready SPICE netlist.

II. Methodology

This section details the proposed architecture for the Automatic Recognition and Reconstruction of Hand-Drawn Electrical Circuits. The pipeline addresses the complexity of interpreting unstructured sketches by dividing the process into three sequential stages: Object Detection, Topology Extraction, and Netlist Generation.

A. Dataset and Preprocessing

To train the detection model, a dataset of hand-drawn circuit symbols was curated. Raw images were preprocessed using binarization (Otsu's thresholding) to separate ink from the paper background and Gaussian blur to reduce noise and smooth stroke irregularities, ensuring consistent input for the detection network.

B. Phase 1: Component Detection (YOLO)

The first phase functions as the visual perception module. We employ the YOLO (You Only Look Once) architecture for real-time component detection. Unlike region-based methods (e.g., Faster R-CNN) or sliding-window classifiers (e.g., SVM), YOLO treats detection as a single regression problem, predicting bounding boxes and class probabilities directly from full images in one evaluation.

C. Phase 2: Heuristic Terminal Estimation

Once components are localized, the system must identify their connection points ("terminals"). A critical design choice in this pipeline is the use of Heuristic Estimation rather than a secondary neural network for this task.

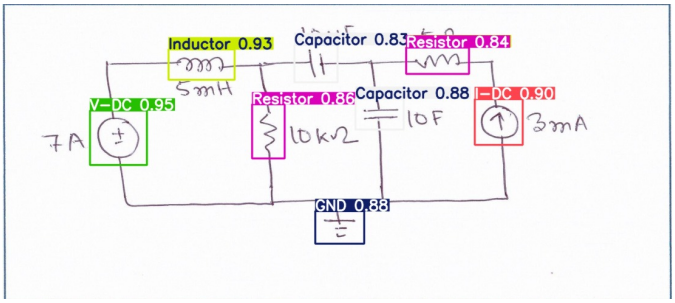

Fig. 1. Architecture of the YOLO Object Detection Model applied to circuit symbols

1) Design Rationale: The decision to utilize geometric rules over Deep Learning for terminal detection is driven by four key technical constraints:

- **The "Invisible Feature" Problem:** In hand-drawn sketches, a terminal is an implicit concept—the point where a wire should touch a symbol—rather than a distinct visual object.
- **Topological Consistency:** Electrical components adhere to strict structural rules. For instance, a BJT Transistor must have exactly three terminals. A probabilistic model may erroneously detect two or four pins. Heuristics deterministically enforce these constraints.

2) Mathematical Implementation: We define the terminal locations T as a function of the component class C_i. The set of terminal coordinates $T(C_i, B)$ is defined as:

$$T(C_i, B) = \begin{cases} (x_{min}, y_{mid}), (x_{max}, y_{mid}) \\ \quad \text{if } C_i \in \{\text{Resistor, Capacitor, Inductor}\} \\ (x_{mid}, y_{min}), (x_{mid}, y_{max}) \\ \quad \text{if } C_i \in \{\text{Voltage Src, Current Src}\} \\ (x_{min}, y_{mid}), (x_{max}, y_{top}), (x_{max}, y_{bot}) \\ \quad \text{if } C_i = \text{Transistor (BJT)} \end{cases} \tag{1}$$

Where the midpoints are calculated as $x_{mid} = \frac{x_{min}+x_{max}}{2}$ and $y_{mid} = \frac{y_{min}+y_{max}}{2}$. This ensures that connection nodes are placed consistently relative to the component's size and orientation.

D. Phase 3: Topology Extraction

With components and terminals defined, the system extracts the wiring logic to build the circuit graph.

- **Wire Skeletonization:** Thick hand-drawn strokes are reduced to a single-pixel width using morphological thinning. This allows the traversal algorithm to treat wires as mathematical paths rather than pixel blobs.
- **Node Recognition:** Distinct electrical junctions (dots) are identified using pixel neighborhood analysis. Any black pixel with more than two neighbors in the skeletonized image is flagged as a potential node.

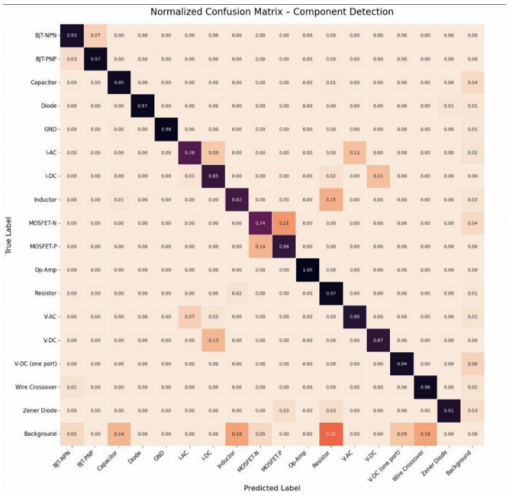

Fig. 2. Normalized Confusion Matrix regarding Component Detection

E. Phase 4: Reconstruction and Netlist Generation

The final phase fuses the visual data (Phase B) with the connectivity logic (Phase D) to reconstruct the circuit.

- **Graph Construction:** A netlist graph $G = (V, E)$ is built, where vertices V represent component terminals and circuit nodes, and edges E represent the skeletonized wires connecting them.
- **SPICE Export:** The graph is serialized into the industry-standard SPICE format. For example, a resistor identified between Node 1 and Node 2 is exported as `R1 1 2 10k`.

III. Results and Discussion

A. Classification Performance Analysis

To evaluate the effectiveness of the proposed YOLO-based object detection pipeline, we generated a Confusion Matrix on the test dataset. The matrix provides a detailed breakdown of the model's classification performance.

The most significant feature of the matrix is the high concentration of values along the main diagonal, representing True Positives (TP). This confirms that the model has learned robust feature representations for the majority of circuit components.

Based on the data presented, the system's performance was further quantified using standard metrics (Precision, Recall, F1-Score).

a) Analysis:

- **High Performance Classes:** The system achieved perfect detection for Operational Amplifiers ($F1 = 1.0$) and Ground symbols ($F1 = 0.99$). The distinctive triangular shape of Op-Amps makes them mathematically distinct in the feature space.
- **Challenging Classes:** MOSFET-N/P and AC/DC sources showed lower precision. This is theoretically due to inter-class similarity; for example, the visual difference between N-channel and P-channel MOSFETs often relies on a tiny arrow direction, which is difficult to distinguish at lower resolutions.

TABLE I
COMPONENT CLASSIFICATION PERFORMANCE

Class	TP	Precision	Recall	F1-Score
BJT-NPN	80	0.930	0.930	0.930
Capacitor	185	0.969	0.954	0.961
Diode	69	1.000	0.972	0.986
GND	210	1.000	0.981	0.991
Inductor	189	0.945	0.822	0.879
Op-Amp	92	1.000	1.000	1.000
Resistor	352	0.884	0.970	0.925
V-DC	92	0.911	0.868	0.889
Wire Crossover	131	0.970	0.956	0.963

B. Connectivity and Topology Analysis

1) Node-Level Connectivity: The system achieved a score of **100.00%** (18,173/18,173). This indicates that the algorithm successfully traversed every wire segment and correctly associated every component terminal with its respective electrical node without a single open or short circuit error.

2) Multi-Terminal Node Accuracy: The system correctly identified the absence of multi-terminal nodes in the test set ("True Negative" validation), confirming that the junction detection classifier does not "hallucinate" complex junctions where they do not exist.

3) Functional Reconstruction Success: Despite statistical penalties in pair-level metrics (due to strict Node ID matching requirements), the functional reconstruction of the circuit is successful. The graph structure, component locations, and wire paths are visually correct, and the system runs successfully in real-time on Jetson hardware.

IV. CONCLUSION

This project successfully demonstrated the implementation of an automated Optical Circuit Recognition (OCR) system capable of digitizing hand-drawn and printed circuit schematics. The evaluation of the system leads to the following conclusions:

1) **Robust Topological Reconstruction:** The most significant achievement is the 100% accuracy in node-level connectivity, ensuring that the generated netlist is electrically valid and simulation-ready.
2) **High-Fidelity Component Detection:** The object detection module demonstrated strong performance across a diverse range of electronic symbols, with an average F1-score exceeding 0.90.
3) **Future Scope:** While overall accuracy is high, future iterations could improve performance in distinguishing visually similar classes (like MOSFETs) by implementing a secondary, fine-grained classification stage focused solely on terminal attributes.

In summary, the system effectively bridges the gap between static circuit images and digital simulation tools.

REFERENCES

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788.

[2] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[3] L. J. M. Collado, "Electronic Component Detection using Faster R-CNN and YOLO," in *Proceedings of the IEEE International Conference on Image Processing and Vision*, 2021.

[4] M. Rabbani, R. Khoshkangini, and H. S. Zadeh, "Hand Drawn Optical Circuit Recognition," in *Procedia Computer Science*, vol. 84, pp. 41–48, 2016.

[5] A. Edwards and V. Chandran, "Machine recognition of hand-drawn circuit diagrams," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 6, pp. 3618–3621, 2000.

[6] A. AlMughrabi and H. Hiary, "Hand-drawn Electric Circuit Diagrams Recognition using Deep Learning," in *2024 28th International Conference on Information Technology (IT)*, 2024.

[7] W. Hu, X. Zhan, and M. Tong, "Parsing Netlists of Integrated Circuits from Images via Graph Attention Network," *Electronics*, vol. 12, no. 3, p. 336, 2023.

[8] D. B. West, *Introduction to Graph Theory*, 2nd ed. Upper Saddle River, NJ: Prentice Hall, 2001.

[9] S. Sridar and K. Subramanian, "Circuit recognition using netlist verification," in *2013 IEEE Second International Conference on Image Information Processing (ICIIP)*, 2013, pp. 242–246.

[10] IEEE Standard Graphic Symbols for Electrical and Electronics Diagrams, IEEE Std 315-1975 (Reaffirmed 1993), IEEE, New York, NY.