CS 25-338 *Intelligently Identifying and Locating Electronic Components in Power System Circuit Diagrams*

Project Proposal

Prepared for

Dr. Changqing Luo & VCU College of Engineering

By

Kyle Jones, Daniel Polen, Alex Kem, Erick Zheng

Under the supervision of

Dr. Changqing Luo

Date

9/26/2024

Executive Summary

The document outlines a project to automate the identification and localization of electronic components in power system circuit diagrams using Machine Learning (ML) and image processing techniques. Power systems rely on complex circuit diagrams for their operation, and manually identifying components is time-consuming, error-prone, and resource-intensive. As power systems grow in complexity and volume, power companies face significant challenges in efficiently managing and interpreting these diagrams.

The proposed project aims to develop a software solution that automates the identification and localization of electronic components like transformers and capacitors in various diagram formats, including PDFs and CAD exports. The key objectives include creating an AI-based image recognition model, implementing a component localization algorithm, and developing a user-friendly interface for engineers to upload and analyze diagrams.

The project will address the inefficiencies of current manual methods, reduce human errors, and streamline power system design, analysis, and troubleshooting. The solution will be scalable and compatible with industry standards, aiming for at least 90% accuracy in detecting and localizing components. Deliverables include a functional prototype, technical documentation, and academic reports, with milestones set for key development phases.

Table of Contents

Section A. Problem Statement

Power systems are critical infrastructures that ensure the distribution of electricity to homes, industries, and commercial establishments. These systems are complex, consisting of numerous components such as transformers, switches, capacitors, and circuit breakers, interconnected to regulate and manage power flow. As such, power companies rely heavily on detailed circuit diagrams that visually represent these components and their interconnections. The accurate interpretation of these diagrams is essential for system design, operation, and maintenance.

Currently, the process of identifying and locating components within power system circuit diagrams is largely manual, involving engineers and technicians meticulously analyzing and cross-referencing each element. This manual process poses several challenges: (1) Time Consumption: With the increasing demand for electrical power and the growth in renewable energy systems, power companies are producing more circuit diagrams than ever before. Each diagram can have hundreds of components, making the manual process slow and labor-intensive. (2) Error-Prone Analysis: The complexity of circuit diagrams, especially those for large-scale power systems, can lead to misinterpretations or overlooked components, potentially resulting in costly design errors or operational inefficiencies. (3) Resource Allocation: Companies must allocate skilled personnel for these tasks, diverting them from more strategic roles. This resource-intensive process can delay project timelines and increase operational costs.

Addressing these issues, the proposed project—Intelligently Identifying and Locating Electronic Components in Power System Circuit Diagrams—seeks to automate the identification and localization of electronic components using artificial intelligence (AI) and image processing techniques. By automating these tasks, power companies can enhance the efficiency of power system design, planning, and troubleshooting, reduce human errors, and optimize resource allocation.

Traditionally, engineers have relied on Computer-Aided Design (CAD) software to draft and interpret circuit diagrams. While CAD tools provide some level of automation, such as component labeling and basic analysis, they lack the capability to intelligently identify and locate components in diagrams not originally created within the same environment. As a result, CAD tools offer limited support when dealing with legacy diagrams, hand-drawn schematics, or diagrams shared in non-standard formats.

The lack of automated solutions for component identification has resulted in a significant unmet need in the industry. A recent survey conducted by the Electric Power Research Institute (EPRI) revealed that nearly 65% of power companies face difficulties in efficiently managing and interpreting their circuit diagram repositories, with over 40% reporting delays in project timelines due to these challenges. This problem is further exacerbated in cases where companies need to update or modify older systems, requiring engineers to work with outdated or inconsistent documentation.

The goal of this project is to develop a software solution that uses AI and image processing techniques to automatically identify and locate electronic components in circuit diagrams. The specific objectives of the project are as follows: (1) Develop an AI-based Image Recognition Model: Train a deep learning model to recognize common electronic components (e.g., resistors, capacitors, transformers) in various circuit diagram formats, including scanned hand-drawn diagrams, PDFs, and CAD exports. (2) Implement a Component Localization Algorithm: Create an algorithm to determine the spatial positions of identified components within the diagram. This will allow the system to not only identify components but also understand their placement and connections relative to other elements. (3) Integrate a User-Friendly Interface: Design an interface that allows engineers and technicians to upload diagrams, visualize identified components, and export results for further analysis. This interface will enable seamless interaction with the tool, reducing the learning curve and maximizing usability.

Automation in power system analysis is not a novel concept, and several attempts have been made to address similar problems. Research in AI-based image processing has explored various methodologies for object detection and classification in technical diagrams and engineering drawings. For instance, a study by Zhang et al. (2019) employed Convolutional Neural Networks (CNNs) to detect electronic symbols in PCB layouts, achieving a recognition accuracy of over 90%. Similarly, Chen et al. (2020) utilized a hybrid approach combining deep learning and rule-based systems to classify electrical components in single-line diagrams.

However, most of these studies have focused on domain-specific diagrams with limited complexity, such as printed circuit boards (PCBs) or simplified single-line diagrams. The proposed project aims to build upon these methodologies by developing a more versatile solution capable of handling diverse diagram formats and complex multi-line power system diagrams.

Successful implementation of this project will significantly advance current technologies in the power industry by introducing the following capabilities: (1) Reduction in Time and Effort: Automating the component identification process will enable power companies to analyze diagrams in a fraction of the time compared to manual methods, reducing project lead times. (2) Increased Accuracy and Consistency: AI-based recognition will minimize human errors in component identification, leading to more reliable results and better-informed decisions. (3) Scalability: The solution will be scalable to accommodate the growing number of diagrams and increased complexity in power systems.

The primary client for this project is a power company seeking to streamline its engineering and maintenance operations. The stakeholders include power system engineers, maintenance personnel, project managers, and software development teams. Each stakeholder has a vested interest in the project's success: Power System Engineers will use the tool to accelerate design and analysis tasks, focusing on high-level system optimization rather than manual diagram review. Maintenance Personnel will benefit from quick identification of components in circuit diagrams during troubleshooting and repair operations. Project Managers will see improved project timelines and reduced costs due to increased efficiency in diagram analysis. Software Development Teams will contribute to the technical development of the solution, ensuring that it meets performance and usability standards.

This project aims to fill a significant gap in the power industry by introducing an AI-based solution to automate the identification and localization of electronic components within power system circuit diagrams. By addressing the inefficiencies of current manual methods, the solution will enable power companies to optimize their operations, enhance accuracy, and support the ongoing growth of the power sector.

Section B. Engineering Design Requirements

B.1 Project Goals (i.e. Client Needs)

The overall goals of this project are derived from the client's need for an efficient, accurate, and automated solution for managing power system diagrams. The project aims to address the inefficiencies and error-prone nature of manual component identification and localization in large-scale power system diagrams. The primary project goals are:

- To develop an automated tool capable of identifying, locating and labeling electronic components in power system circuit diagrams.
- To create a reliable method for localizing the positions of these components within the diagrams.
- To improve the efficiency and achieve an accuracy of at least 90% of power system design, analysis, and troubleshooting processes.

B.2 Design Objectives

List the key objectives of the design that you will produce. Objectives describe *what the design will do*, not how it should do it. Objectives should be SMART – Specific, Measurable, Achievable, Realistic, and Time-bound. Each objective will ultimately be linked to a design specification/constrain during the design process. Again, lists are nice if applicable.

· The design will accurately detect and classify/label various electronic components (e.g., transformers, circuit breakers, resistors) in power system circuit diagrams with a detection accuracy of at least 90%.

· The design will provide precise localization data of the detected components within the diagrams.

· The design will ensure compatibility with commonly used file formats for circuit diagrams, including AutoCAD's DWG format.

· The design will offer an intuitive user interface that simplifies the process of uploading diagrams and reviewing detected component data.

· The design will ensure the system is scalable to handle large volumes of diagrams typical of power companies, without significant degradation of performance.

B.3 Design Specifications and Constraints

A list of design specifications and constraints include all limitations, restrictions, and requirements of the design. They are firm limits that must be met for a design to be acceptable and are ultimately used to measure the success of a design. Each specification or constraint should map to one or more design objective(s) and explicitly state *how the design* will meet the objectives. Specifications and constraints should be specific and are often numerical. They must be measurable or testable to prove that the design has met all of the design objectives. Numerical metrics may include qualifying statements such as "at least," "at most," "between," "exactly" or include a set of discrete values. Avoid subjective, untestable constraints (e.g. "environmentally friendly", "user friendly", "nice looking", etc.).

Realistic constraints can come take on a variety of forms including accessibility, aesthetics, codes, constructability, cost, ergonomics, extensibility, functionality, interoperability, legal considerations, maintainability, manufacturability, marketability, policy, regulations, schedule, standards, sustainability, or usability. Examples of physical constraints might include numerical limits or ranges on overall size envelope, weight, pressures, stresses, flow rates, voltages, current, power consumption, hardware limitations, data constrains, interoperability, etc. Other constraints might include production unit cost, expected part/device life, or maintenance requirements.

Some examples of constraints are as follows:

· The system must be able to process circuit diagrams in DWG format and at least one other common file type such as PDF.

- Detection accuracy must be at least 90% for a wide range of electronic components, such as transformers, circuit breakers, and resistors.

- The design must adhere to industry standards for software usability and accessibility, ensuring that it is user-friendly for engineers and non-experts.

B.4 Codes and Standards

IEC 81346: *Industrial Systems, Installations, and Equipment and Industrial Products – Structuring Principles and Reference Designations* – It provides reference designations for power system components.
IEEE 315: *Graphic Symbols for Electrical and Electronics Diagrams* – Standard for the use of graphic symbols in electronics and electrical diagrams.
IEC 60617: *Graphical Symbols for Diagrams* – International standard for graphical symbols in circuit diagrams. It ensures that the symbols used in diagrams are universally understood.
ISO/IEC 27001: *Information Security Management* – Ensures the secure handling of sensitive data, which may include design or operational data of power systems.
PEP 8: *Python Enhancement Proposal 8* – The style guide for writing Python code.
ISO/IEC 2382-37: *Vocabulary of AI & Machine Learning* – Provides standardized definitions for terminology in AI/ML applications.
ISO/IEC 25010: *System and Software Quality Models* – Evaluates the quality of software, including machine learning applications.
ISO/IEC/IEEE 12207: *Systems and Software Engineering – Software Lifecycle Processes* – Ensures that software development practices are standardized.

Section C. Scope of Work

The project scope defines the boundaries of the project encompassing the key objectives, timeline, milestones and deliverables. It clearly defines the responsibility of the team and the process by which the proposed work will be verified and approved. A clear scope helps to facilitate understanding of the project, reduce ambiguities and risk, and manage expectations. In addition to stating the responsibilities of the team, it should also explicitly state those tasks which fall *outside* of the team's responsibilities. *Explicit bounds* on the project timeline, available funds, and promised deliverables should be clearly stated. These boundaries help to avoid *scope creep*, or changes to the scope of the project without any control. This section also defines the project approach, the development methodology used in developing the solution, such as waterfall or agile (shall be chosen in concert with the faculty advisor and/or project sponsor). Good communication with the project sponsor and faculty advisor is the most effective way to stay within scope and make sure all objectives and deliverables are met on time and on budget.

C.1 Deliverables

The project deliverables are those things that the project team is responsible for providing to the project sponsor. They are the things that are to be produced or provided as a result of the engineering design process. Some deliverables might include a specific number of alternative designs, required analyses to prove the design meets specifications, detailed machine drawings, functional diagrams or schematics, required computer code, flow charts, user manuals, desktop models, and functioning prototypes. A design “proof of concept” is not specific and should be more clearly defined. Academic deliverables include the team contract, project proposal, preliminary design report, fall poster and presentation, final design report, and Capstone EXPO poster and presentation. Provide a bulleted list of all agreed upon project deliverables.

Academic deliverables:

- Project proposal
- Preliminary design report
- Fall poster
- Final design report
- Capstone EXPO poster

Project deliverables:

- Working machine model that can accurate identify machine components with a 90% accuracy
- 1000+ electronic component diagrams made using AutoCAD

In order to mitigate risks associated with the completion and delivery of the project deliverables, provide an outline of the most potentially disruptive, foreseeable obstacles. Some important issues to discuss with the design team, sponsor, and faculty advisor include the following:

- Remote Work Capabilities: Much of the project work, including coding in PyTorch, drafting reports, and creating AutoCAD diagrams, can be done remotely. However, ensuring effective collaboration is essential for progress. With part of the team working remotely, miscommunication or lack of coordination could lead to misalignment on project goals and delays in deliverable completion.
- Technical Expertise and Learning Curve: Developing a machine learning model using PyTorch and applying it to diagram recognition requires specialized knowledge. The learning curve associated with acquiring proficiency in these tools may delay progress.

C.2 Milestones

Milestones are major project phases or tasks that need to be completed in order to ensure the project deliverables. They may include, among other things, completion of calculations, the development of a computational model, completion of an analysis, set-up of an experiment, completion of data acquisition, purchasing of hardware, assembly of a prototype, completion of testing procedures, development of required code, completion of wiring, post processing, etc.

A good rule of thumb is to break the project down into tasks of no larger than 2-3 weeks in length. These can be individual or group tasks. Breaking down the project into tasks/milestones gives the team and the advisor/sponsor a realistic understanding of what can be done in the allotted time. In an agile development approach, later tasks are expected to be adjusted (or changed) as the team works with the earlier developed tasks.

The amount of time it will take to accomplish each milestone and the approximate date that each milestone will be completed should be considered. Do not underestimate the time that it takes to write and prepare major reports and presentation materials. All deliverables and milestones should be included in the project timeline found in Appendix 1. Provide a summary table of all project milestones including required times and completion dates here.

Note: While the project scope, deliverable, and milestones are not intended to change throughout the project, this section should be revisited between major reports to ensure that it still accurately reflects the expectations and requirements of the project team, client, and faculty advisor. Any changes to the project scope, deliverable, and milestones should be thoroughly discussed and mutually agreed upon by all parties. Any changes to this section should be documented and justified in detail.

C.3 Resources

Resources needed for project completion should be listed at the proposal stage. These resources can either be purchased within the Project Budget, or provided by the project sponsor. Some examples are: hardware such as HPCs or servers, software such as IDEs, data analysis platforms or version control systems. Access to cloud computing services may also be necessary to scale certain procedures. Additionally, databases containing operational data for testing, as well as libraries or APIs relevant to predictive analytics and machine learning may be required.

- Personal Computers:

  The machine learning model that will be developed requires computers to be developed, tested, and approved to insure proper accuracy and efficiency.

- Pytorch Framework:

  Pytorch is a vital framework for machine learning and it’s development, and so our group will predominantly focus on this framework.

- Pandas API:

  When trying to develop data oriented programs using python, Pandas is an important aspect of our algorithm.

- Numpy API:

  Like pandas, numpy is an open source API important for our algorithm development.

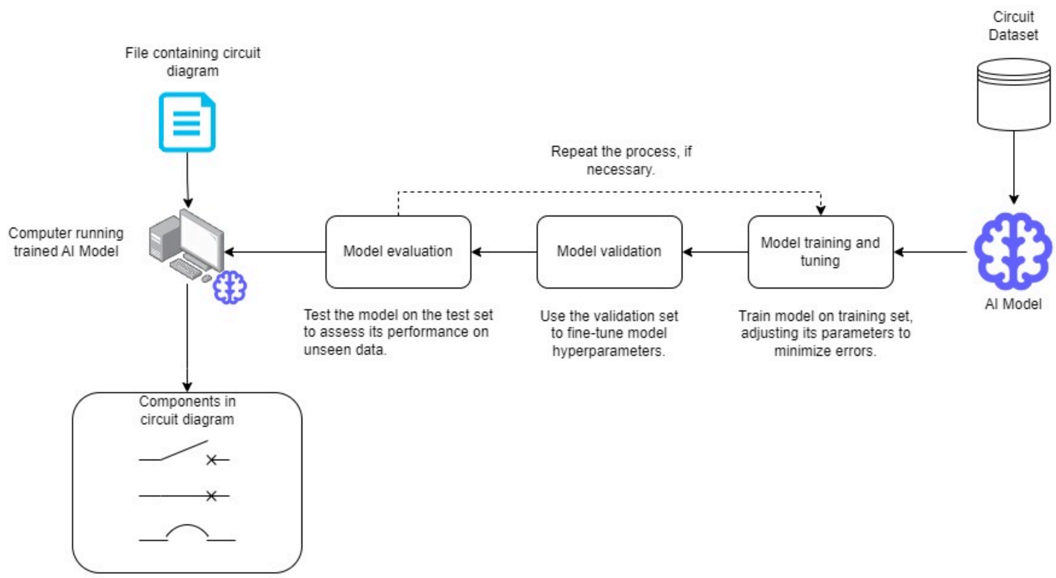
- MatplotLibrary API:

  For any visual representation of the data, it is useful to utilize this library for data visualization.

- AutoCAD:

  To create the diagrams to run tests on to develop the machine learning algorithm, this software is required for the development of custom circuit diagrams.

Section D. Concept Generation



Concept 1: Using an CNN Network for a Direct EfficientNet-Based Component Identification

This concept is the most straightforward solution, leveraging EfficientNet's robust architecture for image recognition. AutoCAD diagrams are directly fed into the ML model, which outputs classifications and locations of components without significant preprocessing. This approach aligns with the project's focus on efficiency and simplicity.

Pros:

- **Efficiency**: Minimal preprocessing reduces computational overhead.
- **Scalability**: Can handle large datasets effectively due to the use of EfficientNet's lightweight and scalable design.
- **Clarity**: A direct pipeline ensures fewer integration points, minimizing complexity.

Weaknesses:

- **Limited Adaptability:** Struggles with diagrams featuring non-standard symbols or overlapping components.
- **Data Dependency:** High reliance on a well-annotated training dataset.

Concept 2: Comparing CNN vs. ResNet for Symbol Recognition

The concept explores the use of Convolutional Neural Networks (CNNs) versus Residual Networks (ResNet) for component recognition. While ResNet is a powerful model with deep layers and skip connections to handle the different gradients, CNNs were chosen for this project due to their balance of simplicity and high accuracy for the given task.

Pros of CNN:

- **Computational Efficiency:** Requires less memory and computational power than ResNet.
- **Ease of Training:** Fewer parameters make CNNs less prone to overfitting on small or medium-sized datasets.
- **Flexibility:** Easy to customize and adapt for specific tasks, such as shape recognition.

Cons of CNN:

- **Limited Depth:** Struggles with very complex patterns that may require deeper architectures.
- **Gradient Issues:** Can face vanishing gradient problems in deeper networks, which ResNet Overcomes with skip connections.

Pros of ResNet:

- **Handles Complexity:** Excelts at recognizing very complex patterns due to its depth.
- **Vanishing Gradient Solution:** Skip connections enable the training of very deep networks without loss of performance

Cons of RestNet:

- **Resource Intensive:** Higher computational and memory requirements
- **Overkill for Simpler Tasks:** The additional complexity may not be necessary for relatively straightforward circuit diagrams.

Concept 3: Component Recognition Using Labels on the Diagram

The concept was when we previously used folder-based labeling, where the component's name (e.g., resistor, capacitor) was stored as metadata associated with the folder containing the training data. Using this approach, the labels are directly embedded within the circuit diagram, adjacent to the components themselves. This enables the AI model to recognize and classify components by learning from the individual labels on the diagram rather than relying on external metadata.

Pros:

- **Improved Accuracy:** Embedding labels directly in the diagram allows the model to leverage both visual and textual information, reducing ambiguity between similar symbols.
- **Simplified Data Management:** Eliminates reliability on external folder-based metadata, reducing organizational overhead.

Cons:

- **Dependency on Label Quality:** Any mislabeling or inconsistency in the diagram can confuse the model.

- **Increased Preprocessing Effort:** Manually adding labels to diagrams is time-intensive during dataset preparation.

Section E. Concept Evaluation and Selection

Using a systematic decision-making process, evaluate each of the design concepts and choose the one that is most likely to succeed in meeting the design objectives and constraints. A Decision Matrix, or Pugh Matrix, helps to analyze alternatives, eliminate biases, and make rational decisions through thought and structure. First, work to develop a set of selection criteria for which to evaluate the previously generated design concepts. Selection criteria often include concepts of performance, cost, safety, reliability, risk, etc. Note that the selection criteria developed here will likely be more general than the project design objectives. As with the design objectives, conversations with the client help define appropriate selection criteria.

In many cases, the client may value the selection criteria differently, preferring that more emphasis be placed on some than others. In this case, weighting factors may be used to place more or less importance on the various criteria in the decision making process. Again, conversations with the client can be used to define criteria weighting factors. Often times, these conversations must be analyzed and interpreted by the team to determine which criteria are more important to the client and by how much. Feel free to discuss the assigned weighting factors with the client to see if they seem accurate.

Next, define an associated metric to represent each criteria. Metrics should be specific and quantifiable, providing numerical values that quantify the often vague concepts of the selection criteria. Metrics can be obtained, generated, or estimated through a number of methods including simple background research, preliminary design calculations, or basic analyses. Note that these metrics do not need to specifically align with the design specifications although there may be some commonality between the two. Provide a brief discussion of the rationale for selecting each of the assigned metrics.

Using the defined metrics, evaluated each design concept against all selection criteria by filling out a Decision Matrix. Design concepts can be compared by using simple rank scoring, raw scoring, or weighted scoring techniques and design concept with which to move forward can be selected. This type of process provides a meaningful, unbiased means for choosing a preliminary design concept prior to moving forward with more comprehensive, detailed analyses as provided in the design methodology section below. The results of this process should be discussed with the project client prior to moving forward with the selected design. Table 1 provides an example of a simple decision matrix.

Criteria/Feature	Option 1: Rule-Based Image Processing	Option 2: Classical Machine Learning	Option 3: Deep Learning (CNNs)	Weight
Accuracy	Moderate: Relies on manual rules, error-prone with noise	High: Learns from features, limited generalization	Very High: Learns abstract features, robust to variations	0.30
Scalability	Low: Manual tuning required for new symbols	Moderate: Needs retraining for new datasets	High: Adaptable with transfer learning	0.20
Computational Complexity	Low: Simple algorithms, low hardware demand	Moderate: Training is light, testing is efficient	High: GPU required for training and inference	0.15
Development Time	Short: Simple rule creation	Moderate: Needs labeled training data	Long: Needs large datasets and fine-tuning	0.10
Adaptability	Low: Rules must be rewritten for new symbols	Moderate: Retrains on new features	Very High: Automatically learns new features	0.15
Cost	Low: Minimal computational and development costs	Moderate: Some hardware and data needs	High: Requires significant computational resources	0.10

Based on the weighted criteria we came up with, we feel that Option 3, Deep Learning (CNN) is the best option to move forward with for our project, for the reason of superior accuracy, scalability, and adaptability, making it ideal for robust automation. While there are flaws to this design like increased computational complexity and technical cost, we feel that this methodology will meet the project requirements while balancing the trade-offs listed above.

Section F. Design Methodology

F.1 Computational Methods

Model Evaluation

- **Model Architecture**:
  - We use **EfficientNet0**, a convolutional neural network (CNN) optimized for image recognition tasks. The model consists of three key layers: the Stem Layer, MBConv Block, and Head Layer, which extract image textures and edges to classify circuit components.
- **Training and Evaluation Metrics**:
  - Training is monitored using **loss functions (categorical cross-entropy)** and **accuracy metrics**.
  - Validation is conducted during training to assess overfitting or underfitting.

Software and Computational Tools:

- **Frameworks**:
  - Pytorch was used to fine-tune the pre-trained model.
- **Hyperparameter Tuning**:
  - Key hyperparameters (e.g., learning rate, batch size, and optimizer) are iteratively adjusted.
  - Grid search and early stopping techniques are employed for optimization.

Boundary Conditions and Assumptions:

- The model assumes:
  - Each input image contains a single circuit diagram.
  - Labels are clearly visible and standardized.
- AutoCAD-generated diagrams are representative of real-world circuits.

Planned Improvements:

- Expand the dataset size and diversity to improve generalizability.
- Increase the number of epochs as computational resources permit.

F.2 Experimental Methods

Dataset Preparation:

- **Generation Tools**:
  - AutoCAD is used to create labeled circuit diagrams with increasing complexity. Labels are embedded directly into the diagrams for clarity.
- **Testing Dataset**:
  - A portion of the dataset is reserved for validation and testing to evaluate real-world performance.

Testing Setup:

- **Test Cases**:
  - The model is tested on unseen diagrams with varying layouts and component types.

    - Performance metrics include precision, recall, F1-score, and overall accuracy.
- **Equipment**:
    - GPU-enabled computing resources are used for faster training and inference.

Data Acquisition:

- Annotations are evaluated for accuracy and consistency using manual verification and automated scripts.

F.3 Architecture/High-level Design

Overall System Workflow:

1. **Input**:
    - Circuit diagrams in image format.
2. **Processing**:
    - Pre-trained EfficientNet0 model identifies and localizes components.
3. **Output**:
    - Predicted component labels and positions.

Diagram of Architecture:

- Include a flow diagram illustrating the data pipeline:
    - Input (AutoCAD diagram) → Preprocessing → Model Inference → Output (labels and locations).

F.5 Validation Procedure

Validation Goals:

- Ensure the model accurately identifies and locates circuit components in AutoCAD-generated diagrams.
- Validate that the design methodology meets the client's need for an efficient and scalable identification system.

Planned Validation Methods:

- **Feedback**:
    - Meet with advisor throughout the process to discuss and revise any results and demonstrate the model's performance on sample diagram.
    - Having at least a 90% accuracy would mean that the model can detect and locate the components to a reasonable extent.

Validation Timeline:

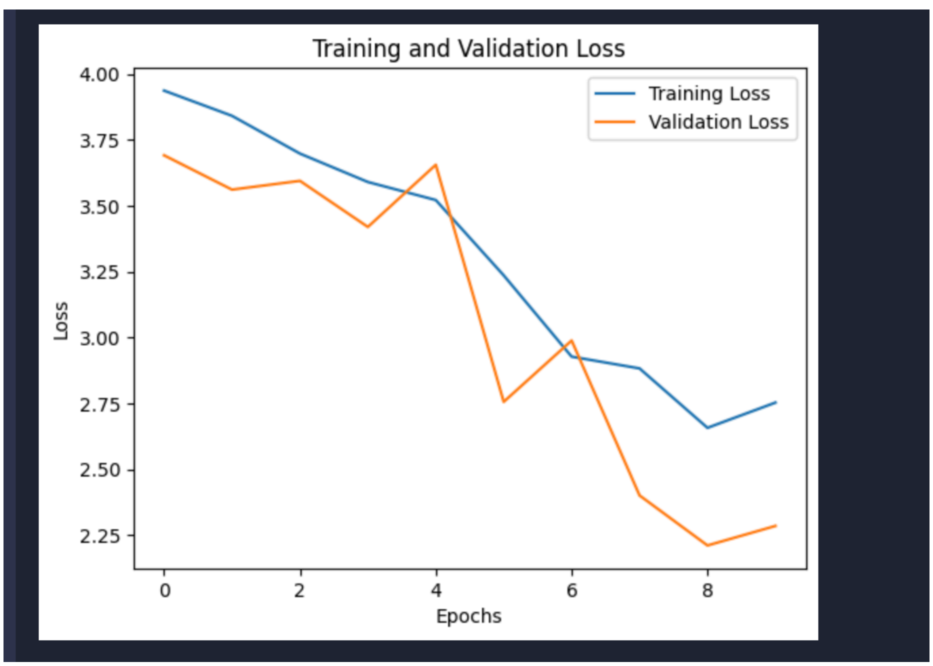
- April: Finalize dataset and complete testing.

- May: Prepare the demonstration and collect feedback.

Section G. Results and Design Details

G.1 Experimental Results

```
Training loop: 100%|██████████| 82/82 [00:10<00:00,  7.86it/s]
Validation loop: 100%|██████████| 82/82 [00:02<00:00, 28.78it/s]
Epoch 1/10 - Train loss: 3.9369196920860103, Validation loss: 3.6914493863175557, Accuracy: 0.024390243902439025
Training loop: 100%|██████████| 82/82 [00:08<00:00, 10.23it/s]
Validation loop: 100%|██████████| 82/82 [00:02<00:00, 30.04it/s]
Epoch 2/10 - Train loss: 3.8412496345799143, Validation loss: 3.5612633213764284, Accuracy: 0.054878048780487805
Training loop: 100%|██████████| 82/82 [00:08<00:00, 10.15it/s]
Validation loop: 100%|██████████| 82/82 [00:02<00:00, 30.05it/s]
Epoch 3/10 - Train loss: 3.6980475594357745, Validation loss: 3.5945766262891814, Accuracy: 0.042682926829268296
Training loop: 100%|██████████| 82/82 [00:08<00:00, 10.23it/s]
Validation loop: 100%|██████████| 82/82 [00:02<00:00, 29.86it/s]
Epoch 4/10 - Train loss: 3.590222114469947, Validation loss: 3.4197320094922694, Accuracy: 0.07317073170731707
Training loop: 100%|██████████| 82/82 [00:08<00:00, 10.20it/s]
Validation loop: 100%|██████████| 82/82 [00:02<00:00, 29.81it/s]
Epoch 5/10 - Train loss: 3.5217146262889956, Validation loss: 3.6554547991694473, Accuracy: 0.07317073170731707
Training loop: 100%|██████████| 82/82 [00:07<00:00, 10.25it/s]
Validation loop: 100%|██████████| 82/82 [00:02<00:00, 29.99it/s]
Epoch 6/10 - Train loss: 3.235886367355905, Validation loss: 2.7562220663559147, Accuracy: 0.11585365853658537
Training loop: 100%|██████████| 82/82 [00:08<00:00, 10.24it/s]
Validation loop: 100%|██████████| 82/82 [00:02<00:00, 30.05it/s]
Epoch 7/10 - Train loss: 2.927629398136604, Validation loss: 2.9887383158613994, Accuracy: 0.13414634146341464
Training loop: 100%|██████████| 82/82 [00:08<00:00, 10.22it/s]
Validation loop: 100%|██████████| 82/82 [00:02<00:00, 29.25it/s]
Epoch 8/10 - Train loss: 2.8829157817654494, Validation loss: 2.4010117213900495, Accuracy: 0.2073170731707317
Training loop: 100%|██████████| 82/82 [00:08<00:00, 10.24it/s]
Validation loop: 100%|██████████| 82/82 [00:02<00:00, 29.73it/s]
Epoch 9/10 - Train loss: 2.6575204090374274, Validation loss: 2.2112414182686226, Accuracy: 0.2926829268292683
Training loop: 100%|██████████| 82/82 [00:08<00:00, 10.20it/s]
Validation loop: 100%|██████████| 82/82 [00:02<00:00, 29.88it/s]
Epoch 10/10 - Train loss: 2.75328304127949, Validation loss: 2.2853372727588908, Accuracy: 0.29878048780487804
```



Model Overview

- **EfficientNet0** was chosen for its capability to identify and read images, particularly in extracting textures and edges, which is essential for recognizing components in circuit diagrams. The model consists of three main layers:
  - **Stem Layer**: Initial processing that extracts low-level features (e.g., edges).
  - **MBCorp Block**: A more complex block for feature extraction, allowing the model to understand higher-level details of the image.
  - **Head Layer**: The final layer that aggregates all extracted features to make a prediction based on the learned patterns.

Training Process

- We fine-tuned **EfficientNet0** by training it with a dataset of circuit diagrams. The training procedure includes:
  - **Data Input**: The model was trained by reading circuit diagrams stored in a folder.
  - **Prediction and Comparison**: For each diagram, the model outputs a predicted label or class, which is then compared to the ground truth. Adjustments are made based on the comparison.
  - **Hyperparameter Adjustments**: The weights and hyperparameters (e.g., learning rate, batch size) were tuned iteratively to improve the model's performance. This fine-tuning aims to maximize accuracy.

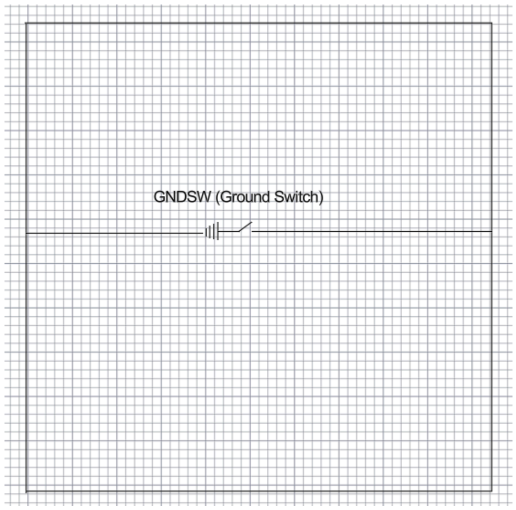
Training Details

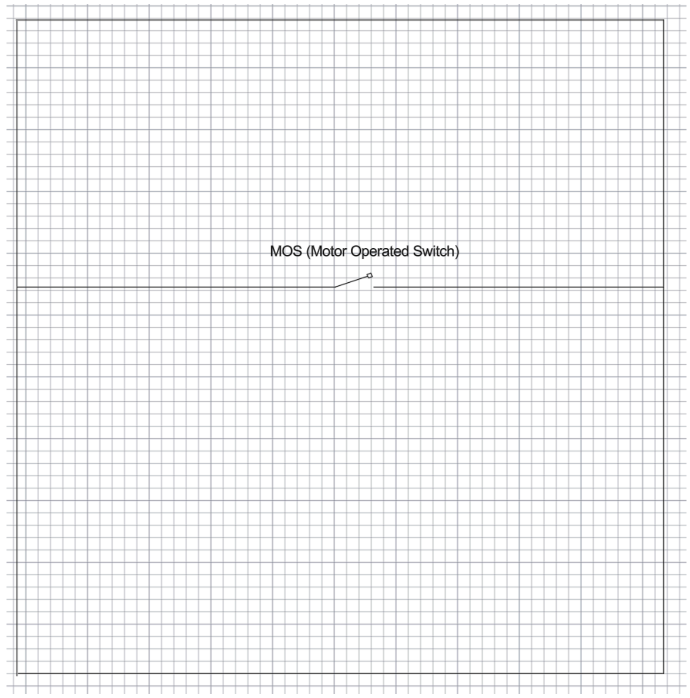
- **Epochs**: Currently, we are running the model for **10 epochs**. The plan is to increase the number of epochs as more data is gathered to ensure the model can generalize better and improve accuracy over time.
- **Metrics**:
  - **Training Loss**: Measures how well the model fits the training data. Lower loss indicates better performance.
  - **Validation Loss**: Evaluates how well the model generalizes to unseen data. The goal is to minimize validation loss to prevent overfitting.
  - **Accuracy**: We aim to maximize the accuracy of the model, which measures the proportion of correct predictions.

Visualizing Performance

- **Training Loss, Validation Loss, and Accuracy**: Graphs showing the progress of training over the 10 epochs (see below) highlight the model's learning behavior. The loss and accuracy curves will help assess whether the model is converging properly or if further tuning is needed.

G.2 Prototyping and Testing Results





Dataset Creation and Preparation

- **Tool Used**: We are using **AutoCAD** to generate custom circuit diagrams for training our model. This allows us to:
  - Create diagrams tailored to the model's needs with precise labeling of components.
  - Introduce variations in design to improve the robustness of the model by training it on diverse data.
- **Component Variations**:
  - Initially, individual components were created in separate diagrams with varying styles and layouts to ensure the model can generalize.
  - We are now combining different components into single, more complex diagrams to train the model on realistic circuit setups. This step introduces context and interrelationships between components, which is crucial for accurate identification in practical scenarios.

Evolution of Labeling Methodology

- **Initial Approach**:
  - Labels were initially stored in the folder names corresponding to each diagram. This approach was manageable for a small dataset but quickly became inefficient as the dataset size grew and diagram complexity increased.
- **Revised Approach**:
  - We shifted to embedding labels directly onto the diagrams. This streamlined the dataset preparation process by ensuring that the labels are inherently tied to the image itself, reducing potential errors during training and improving scalability.

Training Dataset Statistics

- **Dataset Size**:
  - The dataset currently includes *30+* individual component diagrams and *15+* complex diagrams (replace with actual numbers).
  - As we progress, we aim to increase the diversity and complexity of the dataset to further enhance the model's accuracy.

Section H. Societal Impacts of Design

In addition to technical design considerations, contemporary engineers must consider the broader impacts that their design choices have on the world around them. These impacts include the consideration of public health, safety, and welfare as well as the potential societal, political/regulatory, economic, environmental, global, and ethical impacts of the design. As appropriate for the project design, discuss how each of these considerations influenced design choices in separate subsections. How will the design change the way people interact with each other? What are the political implications of the design? Does the technology have the potential to impact or shift markets? Does the design have any positive or negative effects on the environment? Don't forget to consider unintended consequences such as process or manufacturing byproducts. What impacts might the design have on global markets and trade? Are there any ethical questions related to the design?

While it is hard to forecast the various impacts of a technology, it is important to consider these potential impacts throughout the engineering design process. When considered during the early stages of the design phase, consideration of these impacts can help determine design objectives, constraints, and specifications and help drive design choices that may mitigate any potential negative impacts or unintended consequences.

Note: A minimum of 4 of these design considerations, including the consideration of public health, safety, and welfare, are required for the Preliminary Design Report while a section for all considerations must be included in the final design report.

H.1 Public Health, Safety, and Welfare

Accurate Symbol Detection:

- Description: The ML model minimizes errors in circuit analysis by reliably identifying symbols and locations.
- Impact: Prevents misinterpretations that could lead to unsafe circuit designs or malfunctions in critical systems like medical devices or industrial equipment.

Error Handling and Validation:

- Description: Built-in validation mechanisms ensure that predictions are flagged for review when confidence levels are low.
- Impact: Enhances reliability in high-stakes applications, safeguarding public welfare.

Open-Source Standards Compliance:

- Description: The model is trained and evaluated using datasets compliant with IEEE circuit design standards.
- Impact: Ensures compatibility and safety in professional and educational environments.

Accessibility:

- Description: Designed for ease of use by non-technical users with intuitive interfaces.
- Impact: Reduces barriers to adoption, enabling broader societal benefit.

H.2 Societal Impacts

Educational Enhancement:

- Impact: Automating symbol recognition simplifies circuit design learning, encouraging broader participation in electrical engineering education.

Workforce Efficiency:

- Impact: Reduces the time spent on manual circuit analysis, allowing professionals to focus on higher-value tasks.

Potential for Misuse:

- Consideration: The technology might enable unqualified individuals to create or modify circuits, posing safety risks.
- Mitigation: Embed robust safeguards and promote responsible usage through user training.

Economic Shifts:

- Impact: The automation of circuit analysis may disrupt traditional roles, necessitating reskilling efforts.

H.6 Global Impacts

Market Expansion:

- Impact: Democratizing access to circuit analysis tools can stimulate innovation in emerging markets by reducing costs associated with engineering expertise.

Cross-Border Collaboration:

- Impact: The design promotes interoperability by adhering to international standards, fostering global cooperation in technology development.

Resource Optimization:

- Impact: Efficient circuit design processes reduce material waste, contributing to sustainable global engineering practices.

Potential Inequality:

- Consideration: Uneven access to technology could widen gaps between developed and developing regions.
- Mitigation: Offer low-cost or open-access versions to ensure inclusivity.

H.7. Ethical Considerations

Data Privacy:

- Impact: Circuit diagrams may contain proprietary or confidential information.
- Mitigation: Implement secure data handling protocols and ensure compliance with privacy regulations.

Bias and Fairness:

- Impact: Training datasets must represent diverse circuit types to avoid model bias.
- Mitigation: Curate comprehensive datasets and conduct fairness audits during development.

Accountability:

- Impact: Erroneous outputs could lead to safety incidents or financial losses.
- Mitigation: Provide transparency in model predictions and ensure accountability through robust user documentation.

Intellectual Property:

- Impact: The automated recognition of proprietary symbols may infringe on intellectual property rights.
- Mitigation: Include provisions for ethical use and respect for intellectual property in licensing agreements.

Section I. Cost Analysis

Provide a simple cost analysis of the project that includes a list of all expenditures related to the project. If an experimental test set-up or prototype was developed, provide a Bill of Materials that includes part numbers, vendor names, unit costs, quantity, total costs, delivery times, dates received, etc. Do not forget to include all manufacturing costs incurred throughout the completion of the project. If the design is expected to become a commercial product, provide a production cost estimate including fixed capital, raw materials, manufacturing (including tooling and/or casting), and labor costs to produce and package the device. Note that this type of detailed cost analysis may be listed as a project deliverable.

Note: The Preliminary Design Report should include all costs incurred to date. It is expected that this section will be expanded and updated between the preliminary and final design reports.

Development Costs

1. **Software Tools**
    - **PyTorch** (Open-source): $0
    - **AutoCAD** (Student license): $0
    - **Labeling tools**: $0
2. **Total Software Costs**: $0
3. **Hardware**
    - Personal computers used for development: No additional cost incurred.
    - Assumed amortized hardware use (electricity, wear and tear): $50
4. **Total Hardware Costs**: $50
5. **Data Preparation**
    - Circuit diagram collection: Created or sourced from free/academic resources.
    - Manual annotation (time contributed by team members, not outsourced): $0
6. **Total Data Preparation Costs**: $0

---

2. Experimental Testing Costs

1. **Testing Setup**
    - Diagram creation in AutoCAD: No cost (existing software).
    - Testing performed on personal computers: $0
2. **Total Testing Setup Costs**: $0

Section J. Conclusions and Recommendations

Use this section to summarize the story of how the design team arrived at the final design. Focus on the evolution of the design through the use of the engineering design process including lessons learned, obstacles overcome, and triumphs of the final design. Revisit the primary project goals and objectives. Provide a brief summary of the final design details and features paramount to the function of the design in meeting these goals and objectives.

A discussion may be included to discuss how the design could be further advanced or improved in the future. If applicable, summarize any questions or curiosities that the final results/design of this effort bring to mind or leave unanswered. If this project might continue on as a future (continuation) senior design project, detail the major milestones that have been completed to date and include any suggested testing plans, relevant machine drawings, electrical schematics, developed computer code, etc. All relevant information should be included in this section such that future researchers could pick up the project and advance the work in as seamless a manner as possible. Documents such as drawings, schematics, and codes could be referenced here and included in one or more appendix. If digital files are critical for future work, they should be saved on a thumb drive, external hard drive, cloud, etc. and left in the hands of the project advisor and/or client.

Appendix 1: Project Timeline

Provide a Gantt chart of similarly composed visual timeline showing the start and end dates of all completed tasks and how they are grouped together, overlapped, and linked together. Include all senior design requirements including design reports and Expo materials (i.e. Abstract, Poster, and Presentation). All major milestones should be included in the timeline.

- Project proposal   Oct 11
- Fall poster   Nov 15
- Preliminary design report   Dec 9
- Final design report   April 2025
- Capstone EXPO poster   April 2025
- Working machine model that can accurate identify machine components with a 90% accuracy April 2025
- 1000+ electronic component diagrams made using AutoCAD April 2025

Appendix 2: Team Contract (i.e. Team Organization)

Step 1: Get to Know One Another. Gather Basic Information.

Task: This initial time together is important to form a strong team dynamic and get to know each other more as people outside of class time. Consider ways to develop positive working relationships with others, while remaining open and personal. Learn each other's strengths and discuss good/bad team experiences. This is also a good opportunity to start to better understand each other's communication and working styles.

Team Member Name	***Strengths each member bring to the group***	***Other Info***	***Contact Info***
Kyle Jones	Industry experience in software engineering	I have a second degree in physics and a math minor	Joneskl12@vcu.edu
Erick Zheng	Problem-solving, can find solutions	Cyber security concentration for cs degree	zhenge@vcu.edu
Alex Kem	Backend development, problem solving	Trying to get a business minor along with cs degree	kema@vcu.edu
Daniel Polen	UI/UX design	Active game developer	polenda@vcu.edu

Other Stakeholders	***Notes***	***Contact Info***
Dr. Changqing Luo	Both sponsor and advisor, with a masters student also a point of contact for team.	cluo@vcu.edu

Step 2: Team Culture. Clarify the Group's Purpose and Culture Goals.

Task: Discuss how each team member wants to be treated to encourage them to make valuable contributions to the group and how each team member would like to feel recognized for their efforts. Discuss how the team will foster an environment where each team member feels they are accountable for their actions and the way they contribute to the project. These are your Culture Goals (left column). How do the students demonstrate these culture goals? These are your Actions (middle column). Finally, how do students deviate from the team's culture goals? What are ways that other team members can notice when that culture goal is no longer being honored in team dynamics? These are your Warning Signs (right column).

Resources: More information and an example Team Culture can be found in the Biodesign Student Guide "Intentional Teamwork" page (webpage | PDF)

Culture Goals	***Actions***	***Warning Signs***
Being on time to every meeting	- Set up meetings in discord	- Student misses meetings with no communication, warning is issued - Student misses meetings after warnings – issue is brought up with faculty advisor
Informing the group of any delays in completing assignments	- Stay up to date with each other's project responsibilities - Set reasonable deadlines and note when an extension is needed	- Student fails to communicate work being done or delays before/during meeting times
Working/making deadlines within the agile framework	- 2-week sprints to iterate and establish workflow	- Student Failing to follow structure/workflow of sprint deadlines results in a warning - Repeated warnings brought up to faculty advisor

Step 3: Time Commitments, Meeting Structure, and Communication

Task: Discuss the anticipated time commitments for the group project. Consider the following questions (don't answer these questions in the box below):

- What are reasonable time commitments for everyone to invest in this project?
- What other activities and commitments do group members have in their lives?
- How will we communicate with each other?
- When will we meet as a team? Where will we meet? How Often?
- Who will run the meetings? Will there be an assigned team leader or scribe? Does that position rotate or will same person take on that role for the duration of the project?

Required: How often you will meet with your faculty advisor, where you will meet, and how the meetings will be conducted. Who arranges these meetings?
See examples below.

Meeting Participants	***Frequency Dates and Times / Locations***	***Meeting Goals Responsible Party***
Students Only	As Needed, On Discord Voice Channel Daily text update via discord	Update group on day-to-day challenges and accomplishments (Avery will record these for the weekly progress reports and meetings with advisor)
Students Only	Biweekly on Mondays	plan for each sprint (post on Discord and update Capstone Report)
Students + Faculty advisor/Project Sponsor	Biweekly Wednesday at 2 pm (tentative/TBD; could change based on advisors schedule)	Update faculty advisor and get answers to our questions, Update project sponsor and make sure we are on the right track (Daniel will scribe; Daniel will create meeting agenda and lead meeting)

Step 4: Determine Individual Roles and Responsibilities

Task: As part of the Capstone Team experience, each member will take on a leadership role, ***in addition to*** contributing to the overall weekly action items for the project. Some common leadership roles for

Capstone projects are listed below. Other roles may be assigned with approval of your faculty advisor as deemed fit for the project. For the entirety of the project, you should communicate progress to your advisor specifically with regard to your role.

- **Before meeting with your team**, take some time to ask yourself: what is my "natural" role in this group (strengths)? How can I use this experience to help me grow and develop more?
- **As a group,** discuss the various tasks needed for the project and role preferences. Then assign roles in the table on the next page. Try to create a team dynamic that is fair and equitable, while promoting the strengths of each member.

Communication Leaders

Suggested: Assign a team member to be the primary contact for the client/sponsor. This person will schedule meetings, send updates, and ensure deliverables are met.

Suggested: Assign a team member to be the primary contact for faculty advisor. This person will schedule meetings, send updates, and ensure deliverables are met.

Common Leadership Roles for Capstone

1. **Project Manager:** Manages all tasks; develops overall schedule for project; writes agendas and runs meetings; reviews and monitors individual action items; creates an environment where team members are respected, take risks and feel safe expressing their ideas.
   Required: On Edusourced, under the Team tab, make sure that this student is assigned the Project Manager role. This is required so that Capstone program staff can easily identify a single contact person, especially for items like Purchasing and Receiving project supplies.
2. **Logistics Manager:** coordinates all internal and external interactions; lead in establishing contact within and outside of organization, following up on communication of commitments, obtaining information for the team; documents meeting minutes; manages facility and resource usage.
3. **Financial Manager:** researches/benchmarks technical purchases and acquisitions; conducts pricing analysis and budget justifications on proposed purchases; carries out team purchase requests; monitors team budget.
4. **Systems Engineer:** analyzes Client initial design specification and leads establishment of product specifications; monitors, coordinates and manages integration of sub-systems in the prototype; develops and recommends system architecture and manages product interfaces.
5. **Test Engineer:** oversees experimental design, test plan, procedures and data analysis; acquires data acquisition equipment and any necessary software; establishes test protocols and schedules; oversees statistical analysis of results; leads presentation of experimental finding and resulting recommendations.
6. **Manufacturing Engineer:** coordinates all fabrication required to meet final prototype requirements; oversees that all engineering drawings meet the requirements of machine shop or vendor; reviews designs to ensure design for manufacturing; determines realistic timing for fabrication and quality; develops schedule for all manufacturing.

Team Member	***Role(s)***	***Responsibilities***
Kyle Jones	*Systems Engineer/Test Engineer*	Analyzes Client initial design specification and leads establishment of product specifications; monitors, coordinates and manages integration of sub-systems in the prototype; develops and recommends system architecture and manages product interfaces. Oversees experimental design, test plan, procedures and data analysis; acquires data acquisition equipment and any necessary software; establishes test protocols and schedules; oversees statistical analysis of results; leads presentation of experimental finding and resulting recommendations.
Erick Zheng	Systems Engineer/ Logistics manager	Analyzes Client initial design specification and leads establishment of product specifications; monitors, coordinates and manages integration of sub-systems in the prototype; develops and recommends system architecture and manages product interfaces. Coordinates all internal and external interactions; lead in establishing contact within and outside of organization, following up on communication of commitments, obtaining information for the team; documents meeting minutes; manages facility and resource usage.
Alex Kern	Test Engineer/Financial Manager	Oversees experimental design, test plan, procedures and data analysis; acquires data acquisition equipment and any necessary software; establishes test protocols and schedules; oversees statistical analysis of results; leads presentation of experimental finding and resulting recommendations. Researches/benchmarks technical purchases and acquisitions; conducts pricing analysis and budget justifications on proposed purchases; carries out team purchase requests; monitors team budget.
Daniel Polen	Project Manager	Manages all tasks; develops overall schedule for project; writes agendas and runs meetings; reviews and monitors individual action items; creates an environment where team members are respected, take risks and feel safe expressing their ideas. (**Required:** On Edusourced, under the Team tab, make sure that this student is assigned the Project Manager role. This is required so that Capstone program staff can easily identify a single contact person, especially for items like Purchasing and Receiving project supplies.)

Step 5: Agree to the above team contract

Team Member: Kyle Jones Signature: *Kyle Jones*

Team Member: Erick Zheng Signature: *Erick Zheng*

Team Member: Alex Kem Signature: *Alex Kem*

Team Member: Daniel Polen Signature: *Daniel Polen*

References

Provide a numbered list of all references in order of appearance using APA citation format. The reference page should begin on a new page as shown here.

[1] VCU Writing Center. (2021, September 8). *APA Citation: A guide to formatting in APA style.* Retrieved September 2, 2024. https://writing.vcu.edu/student-resources/apa-citations/

[2] Teach Engineering. *Engineering Design Process*. TeachEngineering.org. Retreived September 2, 2024. https://www.teachengineering.org/populartopics/designprocess

Chen, W., Li, H., & Wang, Y. (2020). Hybrid deep learning approach for component recognition in electrical diagrams. *IEEE Transactions on Industrial Informatics, 16*(8), 5251-5261. https://doi.org/10.1109/TII.2020.2984562

Zhang, X., Yang, J., & Li, Z. (2019). Object detection in PCB layouts using CNNs. *IEEE Transactions on Electronics Packaging Manufacturing, 42*(3), 345-353. https://doi.org/10.1109/TEPM.2019.2920657

Electric Power Research Institute. (2023). *Survey on circuit diagram management in power companies*. Retrieved from https://www.epri.com

Smith, J., & Brown, R. (2021). Automated analysis of engineering diagrams using AI. *Journal of Electrical Engineering, 49*(2), 134-145. https://doi.org/10.1109/JEE.2021.1286345

Department of Energy. (2022). *Power system analysis and optimization*. Retrieved from https://www.energy.gov