CHAPTER 3

HARDWARE AND SOFTWARE DESCRIPTION

3.1 HARDWARE DESCRIPTION

3.1.1 STEP-UP TRANSFORMER

A device that converts low-voltage input to high-voltage output is essential for applications requiring amplified electrical power. It achieves this by using components like transformers, voltage multipliers, or inverters, depending on the design. Such devices are widely used in plasma generation, medical equipment, and power systems. They ensure efficient energy transfer while maintaining safety and minimizing energy losses. Proper insulation and regulation mechanisms are critical to handling the high-voltage output effectively.

Figure 3.1: High Voltage testing Transformer

3.1.2 RECTIFIER

A rectifier is used to convert AC voltage to DC, providing a stable and consistent power supply for plasma formation. It ensures that the electrical input is smoothed and regulated, preventing fluctuations that could disrupt plasma stability. This conversion is achieved using diodes or other rectifying components, often paired with filters to minimize ripple. The resulting DC voltage is essential for maintaining controlled plasma conditions in various applications. Reliable rectification enhances the efficiency and precision of plasma generation systems.

Figure 3.2: High Voltage oil cooled rectifier

3.1.3 CAPACITOR/FILTER

A capacitor stores electrical energy and releases it rapidly when the Rotating Spark Gap (RSG) triggers a discharge. This sudden release generates a high-energy spark, which serves as the ignition source for plasma formation. The capacitor's ability to store and discharge energy efficiently is critical for sustaining the high-voltage pulses needed in plasma systems. Properly selected capacitance values ensure consistent and powerful energy release. This mechanism is fundamental in applications where precise plasma ignition is required.

Figure 3.3: High voltage capacitor filter

3.1.4 VOLUME DISCHARGE REACTOR

The plasma treatment chamber is the main unit where seeds are exposed to plasma for quality enhancement. It provides a controlled environment to ensure uniform plasma interaction with the seeds. This exposure improves germination rates, growth parameters, and overall seed vitality. In the volume discharge reactor, plasma is generated from the sharp pins connected to high voltage. The glass sheet acts as a dielectric barrier, and seeds are placed on it to receive the plasma. The pin mesh above ensures even plasma distribution over the seed surface.

Figure 3.4: Volume Discharge Reactor

3.2 SOFTWARE DESCRIPTION

3.2.1 JUPYTER NOTEBOOK

Jupyter Notebook is an interactive development environment widely used in data science and machine learning. In this project, it was employed for developing and testing code related to image preprocessing, feature extraction, and model training.

3.2.2 PYTHON PROGRAMMING

Python served as the backbone of the entire project. Its readability, simplicity, and rich support for scientific computing made it ideal for implementing tasks ranging from loading and processing images to training and deploying machine learning models.

3.2.3 IDLE (INTEGRATED DEVELOPMENT AND LEARNING ENVIRONMENT)

IDLE, the default editor and shell for Python, was used during the early phases of development for testing individual functions and small segments of code. It provided a lightweight interface to quickly prototype logic, especially for basic calculations and preprocessing routines, before integrating them into the larger Jupyter Notebook environment.

3.2.4 SCIKIT-LEARN

Scikit-learn is a popular machine learning library built on top of NumPy and SciPy. It was used in this project for implementing classical machine learning algorithms such as Support Vector Machines (SVM) and K-Nearest Neighbors (KNN).

3.2.5 FLASK SOFTWARE

Flask is a micro web framework written in Python. It was used in this project to deploy the machine learning model as a web application. With Flask, we developed a simple and interactive interface where users can upload images of plant leaves and receive instant feedback on whether the leaf is healthy or diseased.

3.2.6 NUMPY

NumPy is a fundamental package for scientific computing in Python. It was extensively used in this project for handling multi-dimensional arrays and performing mathematical operations on image data.

3.2.7 OPENCV (OPEN SOURCE COM PUTER VISION LIBRARY)

OpenCV was the primary tool used for image processing in the project. It provided a wide range of functions to read images, convert them between color spaces (RGB, grayscale, HSV), apply filters to reduce noise, detect edges and contours, and perform segmentation

3.2.8 MATPLOTLIB

Matplotlib is a visualization library that was used to graphically represent the results of image processing and model training. In this project, it was utilized to display input and processed leaf images, plot color histograms to analyze RGB features, and visualize performance metrics such as model accuracy and loss over training epochs.

3.2.9 TENSORFLOW

TensorFlow is a powerful deep learning framework developed by Google. In this project, it was used to build and train Convolutional Neural Networks (CNNs) for classifying plant leaves. CNNs, with their ability to automatically extract spatial features from images, significantly improved the accuracy of disease detection.

3.2.10 PANDAS

Pandas is a high-level data manipulation library that was used to manage structured data in this project. After extracting features like area, perimeter, RGB values, and DPI from leaf images, Pandas was used to organize them into tabular formats such as data frames.