WSUAG-Arduinoapp

Project Solution Approach

AgriData Intelligence

Members: Aidan Bachart, Forest Cook, Shreyas Jagalur

Oct. 5, 2023

TABLE OF CONTENTS

I. Introduction

The purpose of this part of the document is to provide an overarching description of the design of the Arduino Data Processing Project. This portion will be split into the following sections/design elements: Radiometric Correction, and Plot Segmentation.

Radiometric Correction - In this section, we will delve into the techniques and methods used for radiometric correction, which involves adjusting image data to ensure consistency and accuracy. We'll explore how the system corrects for variations in light and color to provide reliable data for further analysis.

Plot Segmentation - This component will discuss the techniques used to identify and segment individual plots within the crops. It will address how the system distinguishes different pre-labeled breeds of crops and enables monitoring of plant development over time.

This section will serve as a guiding framework for the development team, ensuring that radiometric correction and plot segmentation align with project goals. Additionally, it will act as a reference point for the client to track progress and verify that the prototype aligns with the original project objectives.

Our team's objective is to provide comprehensive and well-documented information in each of these sections to give a thorough understanding of the Arduino Data Processing Project's radiometric correction and plot segmentation.

II. System Overview

The system overview is a crucial section that serves as the gateway to understanding the intricacies of the Arduino Data Processing Project. It provides a high-level introduction to the project's core functionality, design principles, and objectives.

Radiometric Correction - Radiometric correction is a pivotal aspect of the project, ensuring that images captured by sensors are adjusted for variations in lighting and color to provide accurate and consistent data. This correction process is fundamental to the project's success as it directly impacts the quality of the data that is further analyzed. By explaining the radiometric correction process, this section sets the foundation for the subsequent discussions on image processing and data analysis.

Plot Segmentation - Plot segmentation is another critical element of the project. It involves the identification and separation of individual plots within the captured images, enabling the monitoring of plant development over time and distinguishing different plant breeds. This process is essential for precision agriculture and provides valuable insights to local farmers for better decision-making.

Design Considerations - The design considerations emphasize the guiding principles that have shaped the project. These considerations include scalability, as the system must be capable of accommodating a large volume of data generated by multiple sensors. Real-time data processing is another vital element, ensuring that farmers have access to up-to-date information. These considerations have steered the project towards efficient automation and robust data processing.

Goals - The overarching goal of the project is to democratize precision agriculture. By providing local farmers with advanced tools for crop monitoring, the project aims to enhance crop productivity, reduce labor-intensive tasks, and improve agricultural practices. The system's ability to perform radiometric correction and plot segmentation plays a pivotal role in achieving these goals.

In summary, this system overview provides a brief but informative introduction to the Arduino Data Processing Project, highlighting the significance of radiometric correction and plot segmentation in the context of precision agriculture. It sets the stage for the subsequent sections, where the technical details of these processes will be explored in depth.

III. Architecture Design

III.1. Overview

Our Arduino Data Processing Project is structured around a "Rule-Based System." This system is essentially a set of predefined instructions that guide how different parts of our project work together. These instructions make sure tasks are carried out consistently and accurately.

Why We Use the Rule-Based System

We have chosen the Rule-Based System because it offers a structured approach to managing complex tasks. It's like having a well-documented recipe for each task. This ensures that everything is done correctly and consistently. Additionally, it provides us with the flexibility to adapt to future project requirements by simply updating the rules.

Architecture Components

- Rule Engine: The Rule Engine acts as the central decision-making component. It interprets and executes predefined rules, facilitating the automation of tasks. Rules are essential for the data processing of our task. The Rule Engine ensures that the defined rules are executed in a systematic and reliable manner.

- Data Processing Rules: This set of rules guides the data processing workflow. Each rule corresponds to a specific data processing task, such as data extraction, radiometric correction, and plot segmentation. These rules ensure that data processing is consistent and follows predefined criteria, leading to accurate and reliable results.

- System Integration Rules: These rules manage system behavior and user interactions. They include:

    - User Interface Rules: Dictate how the user interface should respond to user inputs and system events. These rules provide a user-friendly experience and ensure that the user interface aligns with project objectives. While this set of rules would typically be part of this architecture pattern, they are not relevant to our system and will be omitted.

    - Configuration Rules: Govern system configuration settings and changes. These rules facilitate system adaptability, allowing for efficient adjustments to configuration parameters.

- Feedback Rules: Define how system feedback and user notifications are generated based on specific events. These rules ensure that users are informed of the system's status and any noteworthy events. While this set of rules would typically be part of this architecture pattern, they are not relevant to our system and will be omitted.

III.2. Subsystem Decomposition

There will be three subsystems for the radiometric correction portion and two for the plot segmentation. The subsystems shall be numbered 1-5.

Radiometric Correction Components
1. Reference Panel Identification Model
2. Pixel Information Retrieval Algorithm .
3. Image Adjustment Algorithm

Plot Segmentation Components
4. Plot Segmentation Model
5. Plot Normalizer

I.1.1. **Reference Panel Identification Model**

a) Description
This subsystem will be a TensorFlow model trained to identify the radiometric correction reference panels in the crop images.

b) Concepts and Algorithms Generated
The primary concept of this is a deep learning model generated by TensorFlow.

c) Interface Description
In place of typical interfacing, this will simply output the data for the Pixel Information Retrieval Algorithm.

I.1.2. **Pixel Information Retrieval Algorithm**

a) Description
This will output the pixel information for the identified reference panels, more specifically the pixel information for the data contained within the generated boxes identifying the panels.

b) Concepts and Algorithms Generated
This algorithm has yet to be developed.

c) Interface Description
This will provide the data needed for the Image Adjustment Algorithm.

I.1.3. **Image Adjustment Algorithm**

a) Description

This algorithm will adjust the image with the pixel information found by the Pixel Information Retrieval Algorithm component in order to match the other pictures and provide the intended output data.

b) Concepts and Algorithms Generated

This algorithm has yet to be developed.

c) Interface Description

This will adjust the images to be of a standard where further data can be taken from them.

I.1.4. **Plot Segmentation Model**

a) Description

This subsystem will be a TensorFlow model trained to identify the separate plots within the images of crops.

b) Concepts and Algorithms Generated

The primary concept of this is a deep learning model generated by TensorFlow.

c) Interface Description

In place of typical interfacing, this will simply output the data for the Plot Normalizer.

I.1.5. **Plot Normalizer**

a) Description

The plot normalizer will use the same image adjustment algorithm to adjust just the individual plot and output the results.

b) Concepts and Algorithms Generated

This algorithm has yet to be developed.

c) Interface Description

This will finalize the processing of the images and split the images into plot segments that further data can be extrapolated from.

IV. Data design

Currently our project doesn't require any data manipulation/storage. Here are a few things we may consider in the future if we do ever require data structures:

Raw Data Storage: Temporarily store the raw data collected from Arduino sensors. These structures can be arrays, lists, or other suitable data containers to hold the initial data before processing.

Processed Data Storage: After data processing steps such as radiometric correction, plot segmentation, and feature extraction, there may be a need for structured data storage for the processed information.

Event Logs: To maintain a history of system events and user interactions, consider using data structures like logs. These logs can be simple text files or structured data formats to track system activities and changes.

V. User Interface Design

Since the Arduino Data Processing Project adheres to a rule-based architecture, there is no graphical user interface (GUI) incorporated into the system. User interactions are exclusively facilitated through command-line interactions and configuration files.

The project's architecture prioritizes rule-driven system behavior and data processing, making graphical user interfaces unnecessary for its core functionality. However, it's important to remain open to future considerations for user interface enhancements, should the need arise.

VI. Glossary

Define technical terms used in the document.

VII. References

`(Dutoit, 2010)`, 3rd Edition, by Bernd Bruegge and Allen H. Dutoit, Prentice Hall, 2010.

VIII. Appendices