Final Project Proposal
Reed Rogers
CS 437
University of Illinois Urbana-Champaign

1. Motivation

The purpose of this project is to create an Internet of Things (IoT) system capable of translating handwritten notes into digitized formats. In a world where digital record-keeping is vital for productivity and organization, handwritten notes are still commonly used, especially for creative brainstorming, quick thoughts, and meeting outlines. I personally find that I cannot properly take notes in meetings or for a class if it is not handwritten, as I like to express my thoughts in formats that are not always conducive to typing (sketches of diagrams, quick tables, etc.). The gap between handwritten and digital documentation often creates inefficiencies in managing, storing, and searching through physical notes. On this note, I am not even able to utilize a digitizer to take “handwritten” notes directly to my computer, as I am unable to download the exterior software required to make it work. Besides, what is better than taking notes on pen and paper? By implementing this system, handwritten notes can be effortlessly transformed into digital documents, improving accessibility, organization, and productivity. Moreover, the project addresses specific challenges related to how information is structured in handwritten notes, such as titles, subtopics, and version control. These additional features will make the system more powerful and customizable, particularly useful for students, researchers, or professionals who rely on both digital and physical mediums for their work. I know that I personally will be using this system every day if I can implement it!

2. Project Scope and Features

The IoT system will consist of a Raspberry Pi (4B) mounted on a desk, equipped with a camera that scans and digitizes handwritten notes when a button is pressed, or a command is run. As I would be utilizing my work computer, maybe I could just ssh into the Pi to initiate the process. The processing will either occur on the Raspberry Pi itself or be outsourced to an old desktop computer I have in my office running Linux Mint (a more powerful computer to run computations on if needed). Key features of this system include:

- Handwritten Note Recognition: The system will utilize image recognition algorithms to detect and digitize handwritten content, translating it into editable text. The camera will look down at a piece of paper and capture the written text. While I won’t be storing the images as new versions where the written words are now text, I will surely be storing the contents of each image to allow for searching capabilities. I will need to be able to recognize written words as text because it is crucial that my code can use the words written in specific coordinates of the paper, as that is what will be used to classify and name the images.
- Data Transfer: The digitized content will be automatically transferred to a designated location on my work computer. This will be a locally stored folder that contains all

images, but the folder structure is yet to be decided. Or, maybe I want this to be sent to a GitHub repository, such that I can access the documents on any computer? I assume that depending on the topic, I will have folders for each subtopic I write about. This is still to be decided, but the categorization will be done automatically based on what I write (not the content, but what I write in specific locations of the page to help categorize).

- Document Structure Recognition: As mentioned, the system will recognize and organize specific features of the notes based on predefined formats. For instance, it will detect that the title of a note will always be in the top-right corner, while subtopics will be in the bottom-left corner.
- Version Control: If the system detects that a note with the same topic and subtopic already exists, it will replace the note with a new version rather than creating a duplicate. At the very least, a new version will be added to the name so I can go back and see older versions.

3. Potential Enhancements

- Optical Character Recognition (OCR) Improvements: Implement machine learning algorithms to improve accuracy in recognizing varied handwriting styles.
- Summarization of Content: Can my code somehow generate a summary of my notes? Can this be included within the folder for the file? This would be extremely helpful to provide context to what a user is looking at. I have access through my work to ChatGPT API keys to pursue this.
- Customization of Note Templates: Allow users to define their own layout rules, such as where specific content types (like dates or action items) appear on the page.
- Voice Activation: Integrate voice commands to trigger note capturing, reducing the need for manual input.
- Tagging and Categorization: Automatically categorize notes based on content, making it easier to search and filter them in the future.
- Integration with Cloud Services: Provide the option to save notes directly to cloud platforms such as Google Drive, OneDrive, or Dropbox for greater accessibility.

4. Timeline

Week 1-2 (Sept 30 – Oct 13, 2024): Initial Setup Objective: Set up the hardware and camera system. Assemble and mount the Raspberry Pi and camera on the desk. Verify the camera functionality, ensuring that it captures high-quality images of the handwritten notes from the correct angle. Test basic connectivity between the Raspberry Pi and your desktop computer. Deliverable: A functioning hardware system capable of capturing clear images. I should also be able to send these images to my work computer.

Week 3 (Oct 14 – Oct 20, 2024): Basic OCR Implementation Objective: Implement basic OCR for handwritten note digitization. Install and configure OCR libraries on the Raspberry Pi or desktop. Test the system with handwritten notes, ensuring text is properly recognized. Begin exporting digitized content to your desktop in a simple text format.

Deliverable: Successful conversion of handwritten notes to text. The result should be a .txt file generated with each image that contains all the text in it.

Week 4-5 (Oct 21 – Nov 3, 2024): Document Structure Recognition Objective: Develop the system's ability to recognize specific structures in the notes. Implement algorithms to detect predefined locations (e.g., title in the top-right, subtopic in the bottom-left). Ensure the system can organize content based on these locations. Test with different note formats to refine accuracy.
Deliverable: The system recognizes and organizes notes based on the title, subtopic, and layout.

Week 6 (Nov 4 – Nov 10, 2024): Version Control Integration Objective: Add functionality to recognize existing notes and manage versioning. Develop logic for the system to check if a note with the same topic and subtopic exists. Implement the feature where the system replaces the note or creates a new version if needed.
Deliverable: Version control feature functional, with notes being replaced or updated based on existing topics.

Week 7 (Nov 11 – Nov 17, 2024): Data Transfer and Folder Organization Objective: Ensure that digitized notes are transferred to the correct folders on your desktop. Set up automatic file transfers from the Raspberry Pi to your Linux Mint desktop. Implement a file organization system to store notes based on topic, date, or other metadata.
Deliverable: Automated transfer and organization of digitized notes in the correct location on the desktop.

Week 8 (Nov 18 – Nov 24, 2024): Testing and Debugging Objective: Test the complete system and fix any bugs. Run the system end-to-end, ensuring that all components work together. Identify and address any issues in the note recognition, version control, or data transfer processes.
Deliverable: A fully functioning and reliable system, with most bugs resolved.

Week 9 (Nov 25 – Dec 1, 2024): Final Enhancements Objective: Implement and refine any additional features. Add customization options, such as allowing users to define their own note layouts. Fine-tune the accuracy of the OCR and document recognition systems. Explore optional features, such as tagging and voice-activated commands if time permits.
Deliverable: A polished, feature-complete system with any desired enhancements.

Week 10 (Dec 2 – Dec 10, 2024): Final Testing and Submission Objective: Conduct final testing and prepare for project completion. Ensure all aspects of the system are working reliably. Final documentation and presentation preparation, including instructions for use.
Deliverable: Final system ready for submission.