FieldChartOCR

Extraction of Handwritten Charts and Tables

Introduction

This presentation summarizes the FieldChartOCR project proposal, focusing on extracting and digitizing handwritten charts and tables. It addresses the challenges faced in manual chart digitization and outlines a novel solution. The goal is to provide a clear overview for supervisory-level stakeholders on project motivation, approach, and evaluation plans.

01

Project Introduction

Project Name, Presenter, Date

FieldChartOCR: Extraction of Handwritten Charts and Tables

Presented by Chris Brechin

Date: November 18th, 2025

Motivation and Problem Statement

Digitizing handwritten charts is difficult due to varied handwriting styles, inconsistent layouts, and noise in scanned images. Manual transcription is slow and error-prone, causing significant pain points for data accessibility and analysis. **FieldChartOCR** aims to automate this process, improving accuracy and efficiency in converting handwritten visual data into machine-readable formats.

Key Challenges in Digitizing Handwritten Charts

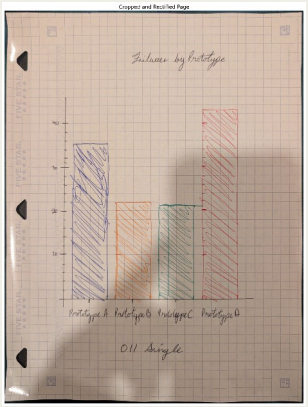
Handwritten charts are challenging to digitize due to **inconsistent handwriting**, diverse chart formats, and image noise. These factors lead to difficulties in automated recognition and interpretation. The lack of standardization increases error rates, making manual transcription time-consuming and costly. Addressing these challenges requires robust pre-processing, classification, and OCR techniques tailored specifically for handwritten data.

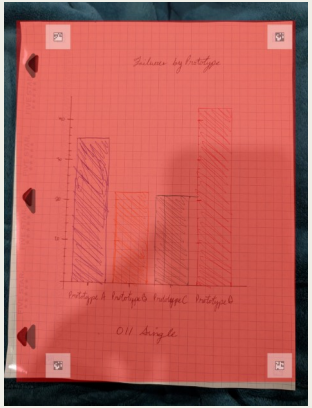
Examples:

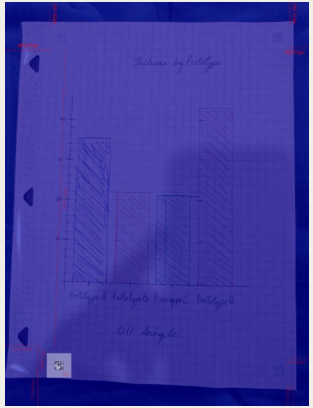
1. ~~Coded Timesheet~~
2. ~~Sparse, Coded Timesheet~~
3. ~~Table with mixed, float and integer values~~
4. ~~Table with mixed, string and integer values~~
5. ~~Checklist~~

Update

This project has been substantially more difficult than previously imagined. I have decided to limit the digitization efforts to only bar charts. This was done to allow me to build one pipeline for at least one kind of data.





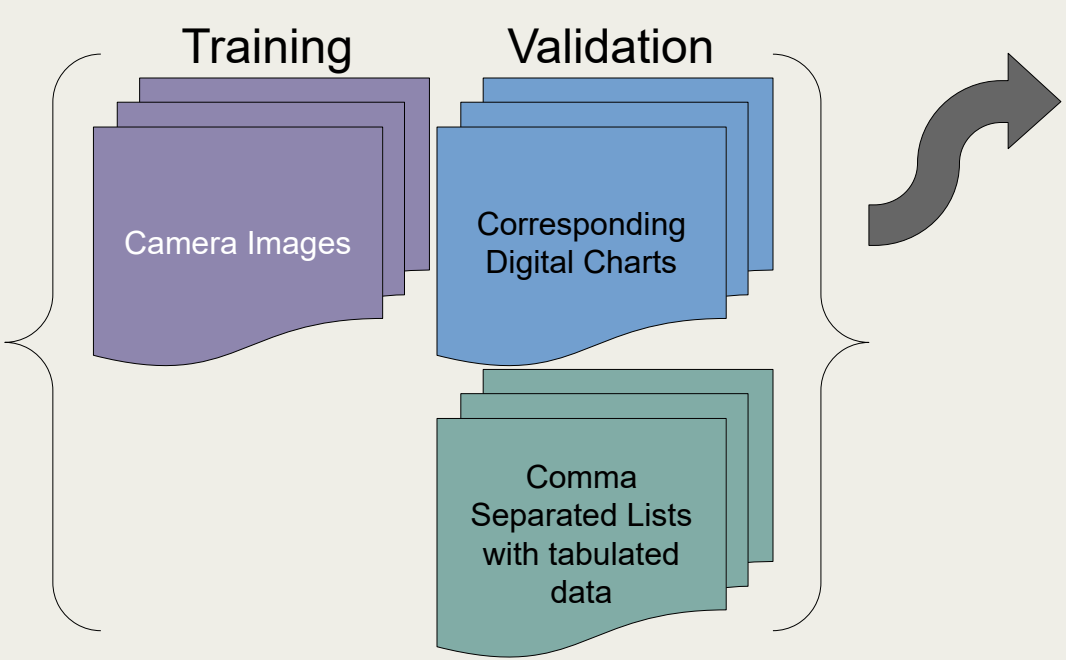


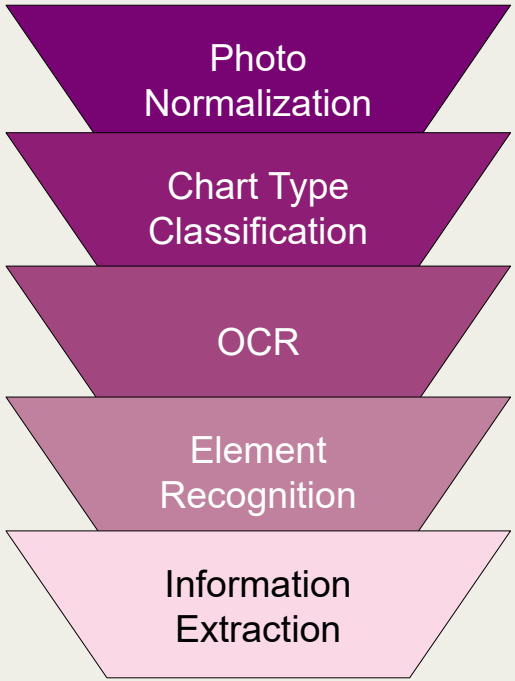
02

Technical Approach and Planning

Overview of FieldChartOCR Solution

FieldChartOCR employs a modular approach combining preprocessing, classification, and detection models. It will integrate OCR tailored for handwriting, alongside semantic assembly modules that reconstruct charts and tables accurately. I aim to automate the extraction pipeline, significantly reducing manual effort while maintaining high accuracy in digitizing complex handwritten visual data.

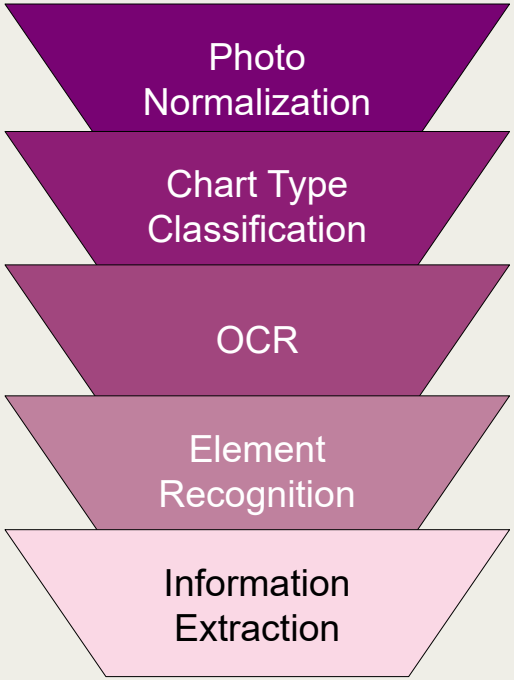




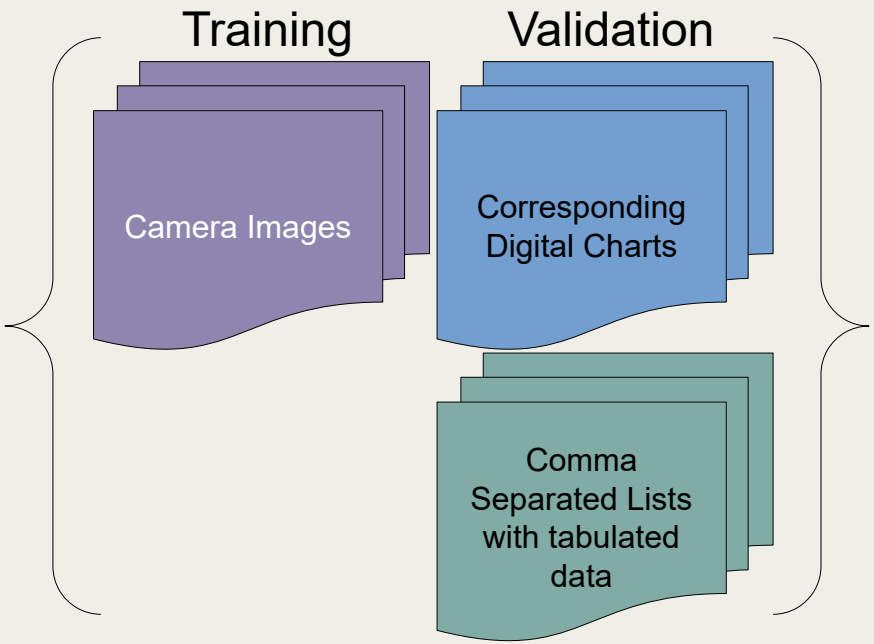
System Architecture and Processing Pipeline

The pipeline consists of four key stages:
1. preprocessing to enhance image quality;
2. classification to identify chart types;
3. detection and OCR to extract handwritten text and graphical elements; and
4. semantic assembly to organize extracted data into structured formats.

Each stage leverages machine learning models for the specific purposeses



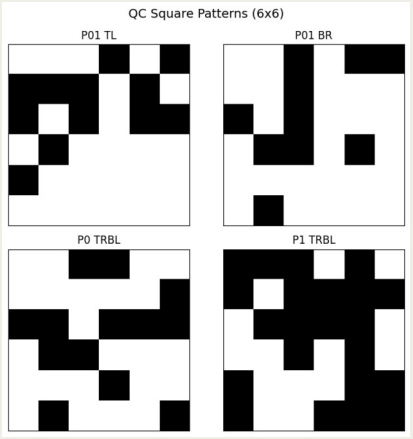
Data Sources and Evaluation Metrics



Data sources include annotated datasets of handwritten charts and synthetic data generated to augment training diversity. Success will be measured using classification F1 scores, detection accuracy, Character Error Rate (CER) for OCR, and runtime performance. These metrics will guide iterative improvements, ensuring the system meets practical usability and reliability standards.

Page Corner Search Metric

Training



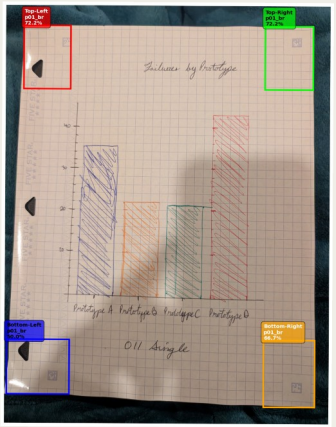
In the current step, I am trying to detect the edge of the page using quick calibration squares in the specialized paper I purchased.

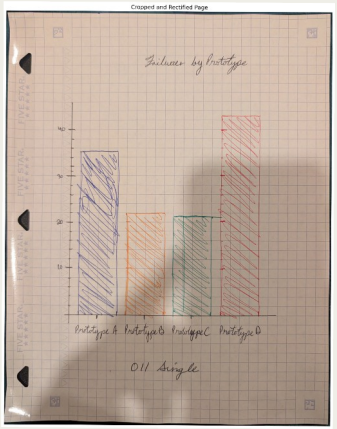
The QC squares are featured left:

- Front and Back Top Left are the same
- - Front and Back Bottom Right are the same
- - Front Page Top Right and Bottom Left are the same
- - Back Page Top Right and Bottom Left are the same

Page Corner Search Metric

Example





The measurement of accuracy is defined as:

```
# Sort by area (largest first) and prefer
more square-like shapes
square_candidates.sort(key=lambda x:
x['area'] * x['solidity'], reverse=True)
```

Note: you can find this on 111 and 112 of **Page Dimensions**

Conclusions

I am still working on the calibration of page detection so that I can normalize the image and crop to the content. The normalization step is important because it helps the next step detect the sizing of bar charts.

Thanks!

Do you have any questions?
chris@brechin.services

https://brechin.services