RenAIssance Test-1

Synthetic renaissance text generation with generative models

23.03.2025

—

Haleel Sada N P

Overview

This report explores methods for extracting the layout of text in an image of printed text. The task was to use AI for this purpose, but instead, I tried to use a statistical approach without AI. It was working for 75% of the time.

I have completed one task related to this project, and this is an additional task that I attempted as an extension of my work.

Description

This report explores methods for extracting the layout of text in an image of printed text. The task was to use AI for this purpose, but instead, I attempted a statistical approach without AI. My method involved analyzing the histogram of the greyscale image along the x-axis and y-axis. By applying a threshold, I identified points where the histogram value changed from below to above the threshold, marking the start of the layout, and from above to below, marking the end. By performing this analysis along both axes, I obtained a rectangular representation of the text layout. This approach demonstrates a structured yet non-AI-based method for text layout detection and provides insights into how statistical techniques can be used for image analysis. The report discusses the methodology, results, challenges, and potential improvements for more accurate layout extraction.

Evaluation Metrics

Since this is not an AI-based model, there was no labelled dataset with ground truth to be compared with. As a result, I had to evaluate the performance manually by inspecting a few sample images. I visually analyzed whether the detected layout boundaries aligned with the actual text regions in the images.

Pros:

- No need to train an ML model, eliminating the need for dataset creation.
- Performs well on homogeneous datasets.
- Works effectively with scanned images.

Cons:

- Not suitable for diverse datasets.
- Struggles with noisy images containing shadows and distortions.

Final Note

The algorithm is not fully optimized. Currently, it selects the largest column as the column size, even when a smaller column exists within it. A better approach would be to first detect row edges before applying column operations, then detect column edges inside the rows. Additionally, a single threshold value that works well for clear images may not perform well on noisy images. Each image may require a different threshold, which should be optimized either through trial and error or by analyzing the pixel distribution. Another challenge is distinguishing between images and text, which could potentially be improved only by using an AI model.