Woolworm:
A Python Package for Digitizing Historical Documents and Archives

Aerith Netzer
aerith.netzer@northwestern.edu
Northwestern University Libraries

September 19, 2025

Abstract

Digitization is a core practice in librarianship, particularly in academic universities with very large, often physical archives. Traditional software packages built for this task make the assumption that most steps can be completed by hand. Northwestern University Libraries recently received a grant to digitize approximately 3.5 million pages of Environmental Impact Statements in a two-year timeline. This time compression created a new need in the library: end-to-end automation of image processing and optical character recognition workflows. With the recent rise of transformer-based architectures, we created a high-level Python library called *Woolworm*, a user-friendly library for conditional image binarization, de-skewing, and text extraction. Further, this architecture was built for use with SLURM on high-performance computing clusters. Increasing data throughput and ensuring high availability.

Keywords OCR · Digital Libraries · High-Performance Computing

1 Image Processing Pipeline

There are three main steps in the image processing pipeline where it concerns digitization of historical documents: de-skewing, conditional binarization, and border removal. To meet these technical needs, we adopted OpenCV[1] for its wide compatibility and long history in the computer vision community.

1.1 Image DeSkewing

In automated scanning machines, individual pages in the film-based medium are often rotated. We correct for this by first applying a bitwise operation, inverting the colors to black/white. We then instantiate a 30×1 rectangular kernel and apply a closing operation on the image. This returns a text line mask, highlighting candidate text lines. We then compute the Shannon entropy of the mask:

$$H = -\sum p(x) \log_2 p(x) \qquad (1)$$

If H is a relatively high value, we assume that the content of the page is mostly text. If H is a relatively low value, we assume it is the contrary case.

$$A(H) := \begin{cases} 0 \text{ if } H \leq T \\ 1 \text{else } H > T \end{cases} \tag{2}$$

Bibliography

[1] Bradski, G, “The OpenCV Library.” 2000.