# Narrative

Printed artifacts communicate historical information through three different kinds of features:

- *bibliographic features*: the paper, binding, typeface, size, quality, price, and organization of the physical book;
- *visual or graphic features*: the arrangement of text, images and white space on a page;
- *linguistic features:* the syntactic and semantic aspects of the words presented on the book's pages.

The researcher working directly with the historical artifact frequently considers its bibliographical and visual features alongside its linguistic content, either explicitly or implicitly. A researcher who discovers, for example, that a book contains an illustration, may thumb through its pages to find others, and then look through other books for related illustrations. But researchers working in large scale digital libraries like the HathiTrust will frequently only have access to digital surrogates and not the material artifacts.

Visual features of printed books that are of interest to humanities researchers are captured in the scanned page images, but may not be recorded in the cataloging record or other metadata associated with the digital file. Our prototype software application uses the visual characteristics of digitized printed pages to identify documents that contain three types of visually distinctive materials of interest to humanities researchers: illustrations, music, and poetry.

Project objectives during this initial grant period include:

- designing and building core system architecture and image analysis pre-processing components;
- development of segment classifiers for illustrations, music, and poetry;
- selection and preparation of a data set for testing
- deployment of the application in the HTRC testing environment
- initial evaluation

## Significant Changes to Personnel

There were no significant changes to personnel or management during the course of the project.

## Project Progress

The four main technical achievements for this project are:

- an extensible framework for configuring and executing image analysis workflows
- implementations of several algorithms from the research literature for image cleaning, manipulation and segmentation

- a library that encapsulates access to HathiTrust APIs and data structures for ease of use within Java applications
- an application to run on HTRC servers that reads a list of items, loads the corresponding page images and executes an image analysis workflow

Document image analysis is a mature research field with significant ongoing work and a number of commercial products. Indeed, many of these technologies have been deployed to support the analysis of a range of cultural heritage material. Despite these advances, there are no robust tools that allow scholars to approach the visually encoded information contained in document page images analogous to the widely used text analysis tools. This restricts research aimed at understanding visually constructed meaning to datasets small enough to be studied closely or to projects with sufficient funding and resources to invest in custom image analysis software. Given the quantity and variety of digitized page images in the HathiTrust digital library, developing image analysis tools that can be configured by scholars and used at scale complements the exiting text analysis services provided by HTRC.

The core technical component of our system is DataTrax, a framework for executing user-configurable image analysis workflows. These workflows are comprised of discrete image analysis tasks that take one or more well-defined inputs such as a color image or a collection of segmented glyphs and transform them into an output value such as a black and white image or the glyphs grouped into lines. The specific tasks that can be executed within a workflow are defined externally to the DataTrax framework and registered by an application. This software architecture, shown in figure 1, promotes easy reuse of existing code.

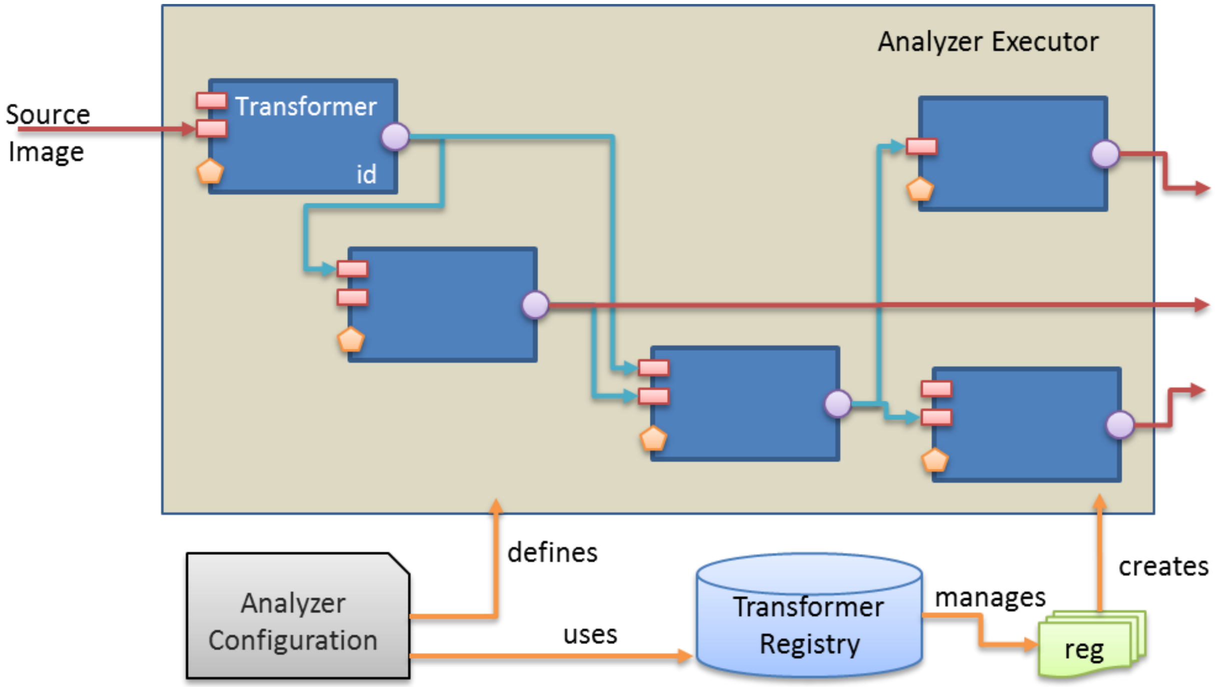


*Figure 1: Overview of DataTrax Architecture*

We have implemented an initial set of image analysis tasks from the research literature. These include preprocessing steps such as converting color images to black and white, image segmentation tasks such as finding all connected components (roughly glyphs on the page) and image transformations such as down sampling to create a smaller version of an image with specific characteristics. We have also created adapters that we use to integrate the widely used OpenCV library for computer vision into DataTrax. This is particularly important since it demonstrates the relative ease with which existing third-part software components can be incorporated into our application.

In addition to the core image-analysis system, the prototype development effort required that we connect this technology to the existing infrastructure and resources provided by HathiTrust. To achieve this, we have created a general-purpose software development kit (SDK) for interacting with the existing APIs and data objects provided HathiTrust and the HathiTrust Research Center. This includes tasks like communicating with the bibliographic and data REST APIs (including authentication and account credential management), resolving item data records stored in a Pairtree directory structure based on HathiTrust identifiers, and reading image data directly from the zipped item data records. This SDK meets the immediate needs of our application while providing a codebase for future projects that make use HTRC data resources without re-implementing the details of negotiating REST requests or parsing data records.

The three components, the DataTrax framework, the library of document image analysis algorithms and the HathiTrust SDK are integral to the efforts of the prototype grant but are implemented as separate libraries that can be used (and are being used) independently. This library-oriented development process will maximize the impact of our work beyond the scope of the WCSA project as we, and eventually others, use them in a range of projects.

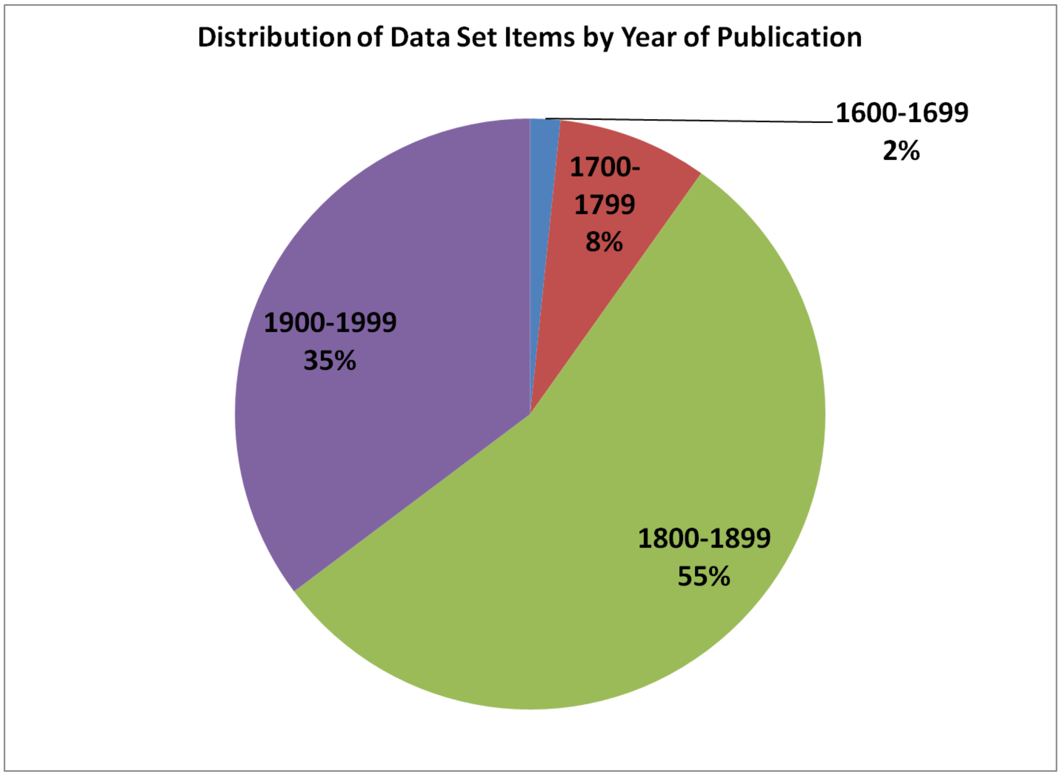
The final technical contribution of the project is the WCSA prototype application itself. This application is the component that ties together the three libraries discussed above and implements the control logic for reading in the list of items to be processed, creating the image analysis workflow using DataTrax and interpreting the results to identify specific features.

The overall system that we have developed through this work provides a framework for working with the large and diverse collection of page images housed at the HathiTrust digital library. As a result, we are now well positioned to take the next steps including partnering with experts in document image analysis to develop more refined algorithms, working on machine learning and pattern recognition tools to provide sophisticated computational analysis, and designing user interfaces that allow scholars to customize workflows, connect with an analyze workset and explore the results of their analysis.

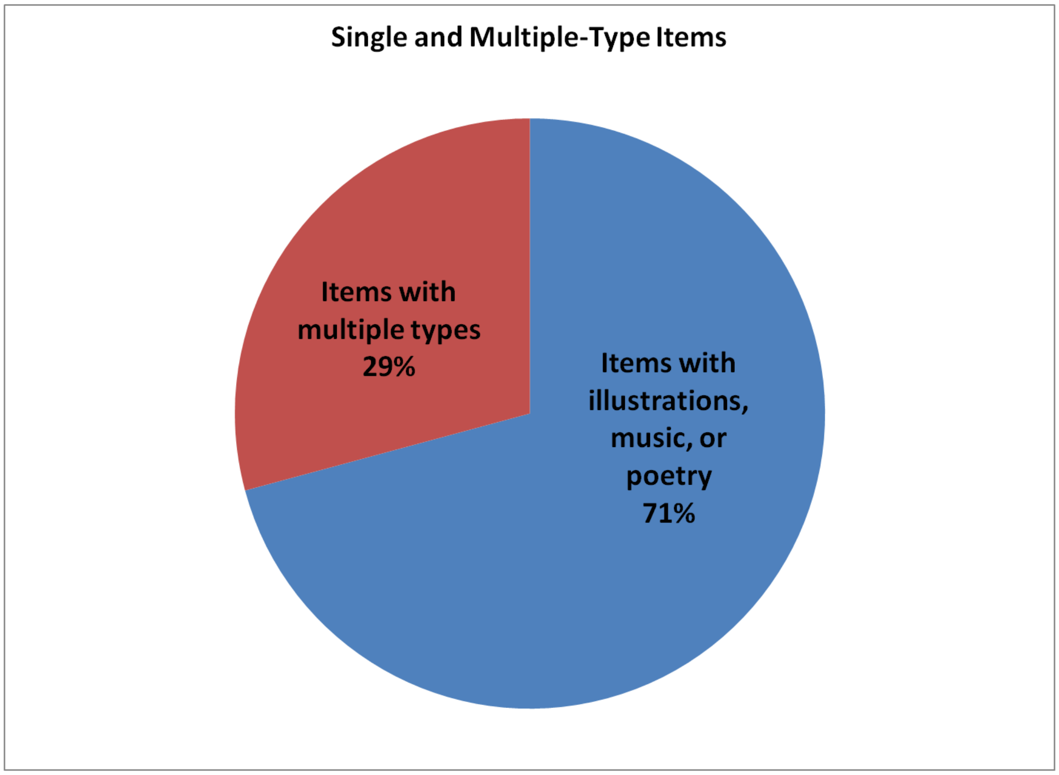
## Dataset Preparation

A hand-curated set of 250 volume ids were selected from a broad range of LC subject headings, including subjects in literature, history, music, the arts, architecture, science, and popular culture. These subjects were selected in order to provide the most varied kinds of books for testing.

Items were selected for the dataset with publication dates distributed over four centuries. The distribution of publication dates was weighted most heavily towards the nineteenth century (55%), followed by twentieth century items (35%). This distribution approximates that of the HathiTrust Digital Library holdings.



This curated dataset includes 122 volumes with illustrations (in sizes ranging from very small to partial page and full page), 58 volumes with music (small extracts, partial page and full page), and 128 volumes with poetry (extracts, partial page, and full page). Some volumes contain two or three of these types of materials, sometimes on the same page.



## Setbacks or Challenges

The proposed scope of work was ambitious. While we accomplished our primary goal of creating a framework for analyzing the visually salient features of printed material and deploying that framework for use on HathiTrust data sets, we were not able to analyze books to detect presence of poetry and our attempt to recognize musical scores were unsuccessful. These setbacks are largely due to the fact that recognizing pages with illustrations proved to be significantly more challenging than we anticipated.

We chose to focus on detecting illustrations at the expense of dedicating more time to music and poetry for two reasons. First, the core image analysis tools required for this task are broadly applicable and provide a solid base for future work. Second, our anecdotal experience indicates that there is a widespread interest within the community for better tools to mine large document collections for illustrations. Additionally, because we were interested in evaluating the feasibility of visual page analytics, we elected not to take shortcuts such as analyzing the generated OCR text to find illustrations.

With respect to detecting musical notation, we implemented the run length algorithm proposed by Bainbridge. Our initial implementation did not yield promising results. Rather than working to debug and revise this current work we elected to leave this task as an opportunity to collaborate with members of the optical music recognition community.

Our work on visually analyzing features of printed poetry in an earlier project informed our approach to visual text analytics for this project, which included the development of a feature ontology for printed books and for poetry in particular (see Appendix). One of the significant challenges in detecting poetry in a diverse digital library such as the HathiTrust is the range of print conventions and styles found over works from different locations and time periods.

# List of Project Disseminations and Final Deliverables

As part of the WCSA prototype effort, TCAT has developed and released four main software components. Three of these provide core library capacities that we use extensively within the WCSA prototype application and have broader applications for the DH a library community. The fourth is the main prototype application that is responsible for leveraging and configuring the libraries in order to implement the specific image analysis tasks set forth for our project. These components are briefly summarized below.

## DataTrax Framework

DataTrax provides an extensible framework for configuring and executing image analysis workflows. We anticipate that enhancements to this framework will be ongoing as we pursue additional work on VisualPage and other related projects.

Available at: https://github.com/tcat-tamu/DataTrax

License: Apache 2.0

## HathiTrust SDK

The HathiTrust SDK provides a Java-based API for interacting with the HathiTrust REST APIs and content material. This SDK currently and early level prototype being developed in conjunction with the WCSA prototype grant and other projects at TCAT.

Available at: https://github.com/tcat-tamu/HathiTrust-SDK

License: Apache 2.0

## Document Image Analysis Algorithms

We have implemented a number of low-level document image analysis algorithms to perform basic tasks that can be combined either programmatically or using the DataTrax framework to create complex image analysis workflows. These algorithms provide a starting point for creating open source image analysis tools for use within the DH and library communities.

Available at: https://github.com/tcat-tamu/Document-Image-Analysis

License: Apache 2.0

## WCSA Prototype Application

The WCSA Prototype application is the main driver that reads images from the HTRC server testbed and executed the image analysis work flows and post processing. This component is tailored to use within the scope of the WCSA project and would, with additional funding, be the starting point for developing a more general purpose application for deployment.

Available at: https://github.com/tcat-tamu/visualpage.wcsa

License: Apache 2.0

## Appendix A: Visual Features of Printed Books

Printed books contain a variety of visual features related to typography, page design, and structured information. Some of these features contribute to the discovery of the targeted types of visual material and other features distract from that discovery by producing false positive results or by preventing the optimal functioning of page recognition algorithms, particularly in older books with less regularized typeface and layout.

Visual features of print layout include:

- One column text block
- Two column text block
- Three column text block
- Indentation of text
- Running heads
- Page numbers
- Footnotes
- Titles
- Subtitles
- Margin consistency
- Margin size

Visual features related to illustrated materials include:

- Percentage of the page occupied by illustration (full page, half page, quarter page, or smaller)
- Placement of the illustration on the page
- Arrangement of text next to or around the illustration
- Decorative borders around text and/or image
- Small ornamental designs used to separate or mark portions of the text
- Maps
- Charts
- Tables

Visual features related to music include:

- Percentage of the page occupied by musical notation (full page, half page, quarter page, or smaller)
- Placement of musical notation on the page
- Arrangement of text next to or around musical notation
- Placement of text within the musical notation (to indicate notes, scales, performance instructions, or lyrics)
- Hand drawn or typeset staves
- Multi-part musical scores

Visual features related to poetry include:

- Percentage of the page occupied by poetry (full page, half page, quarter page, or smaller)
- Placement of poetry on the page
- Arrangement of other text next to or around poetry
- Capitalization of lines of poetry
- Indentation of lines of poetry
- Separation of poetry into stanzas (groups of lines) separated by white space