IN SUPPORT OF NATURAL HISTORY DIGITIZATION:

ASSISTIVE TOOLS FOR THE MOBILIZATION OF

BIODIVERSITY DATA

By

Caleb Adam Powell

Joey Shaw
UC Foundation Professor of
Biology, Geology, and Environmental Science
(Chair)

Stylianos Chatzimanolis
Guerry Professor of
Biology, Geology, and Environmental Science
(Committee Member)

Hong Qin
Associate Professor of
Computer Science and Engineering
(Committee Member)

IN SUPPORT OF NATURAL HISTORY DIGITIZATION:

ASSISTIVE TOOLS FOR THE MOBILIZATION OF

BIODIVERSITY DATA

By

Caleb Adam Powell

A Thesis Submitted to the Faculty of the University of
Tennessee at Chattanooga in Partial
Fulfillment of the Requirements of the Degree
of Master of Science

The University of Tennessee at Chattanooga
Chattanooga, Tennessee

May 2020

ABSTRACT

Much of the primary biodiversity data supporting biology and environmental science are preserved in natural history collections (NHCs). Recently, these collections began to make those data digitally available through online portals thereby improving their accessibility and usability. The processes involved in creating digital representations of these physical objects are, as of now mostly manual and labor intensive. The goal of this work is to assist NHCs in making these data available. Pursuant to this goal, three assistive tools were produced based on experiences digitizing herbaria (i.e., NHCs focusing on the kingdom *Plantae*). The tools produced through this effort are: (1) a labor estimation model, (2) a program to assist in the capture and refinement of herbarium specimen images, and (3) a pair of programs which when used while collecting new specimens circumvents the most labor intensive step in the process of digitization.

ACKNOWLEDGEMENTS

I am eternally grateful to my advisor, mentor, and friend Dr. Joey Shaw for challenging me to enter the graduate program and encouraging me to pursue research with creativity. I would like to thank my committee members Dr. Chatzimanolis for his thoughtful instruction and straightforward approach and Dr. Qin for his persistent support and interdisciplinary perspective. I am grateful to Dr. Craddock for advocating my research and fostering a love of mycology. I would also like to acknowledge my peers in the graduate program who have continuously impressed and inspired me. I am especially thankful to my mother and father for their love and support along my circuitous route through academia.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

PART I : OVERVIEW AND THESIS OBJECTIVES

ABSTRACT

Concerted efforts to digitize history collections have mobilized (i.e., make digitally accessible through online repositories) millions of preserved biological specimen records (Nelson & Ellis, 2019). These digitization projects are making otherwise difficult to access biodiversity data freely accessible. In their physical form, these specimens stand as evidence of the presence and state of biological entities in space and time. Preserving original biological material enables future experts to reference, verify, or challenge research based on these data even as our environment and the organisms in it continue to change. Attentive curation and persistent scrutiny by specialists qualifies these specimens as the foundation from which we assert what is known about the natural history of our world.

At one time the physical organization of specimens within a natural history collection defined which research questions that collection could feasibly address. By mobilizing data associated with these specimens, an increasingly diverse range of researchers are benefiting from the expert attention invested into them (Soltis & Soltis, 2016; Losos et al., 2013; Souza & Hawkins, 2017; Willis et al., 2017; Lang et al., 2019). Unsurprisingly, improved usability and accessibility of these data has coincided with increases in the prevalence of their use in scientific literature (Lavoie, 2013; Nelson & Ellis, 2019). Currently, the process of digitizing these data is a massive effort which is not near completion (Ariño, 2018; Barkworth & Murrell, 2012; Vollmar et al., 2010).

In support of natural history digitization, this work communicates lessons learned and tools developed during the digitization of Tennessee's herbaria. Those collections (and many others) were digitized as a part of the larger National Science Foundation's Advancing Digitization of Biodiversity Collections (ADBC) program which funded the establishment of a consortium of southeastern herbaria:

the Southeast Regional Network of Expertise and Collections (SERNEC). This regionally focused consortium was organized hierarchically by state, where a designate from each state was tasked with the digitization of the collections housed within that state. Through this organizational structure, the Tennessee Herbaria Consortium (THC) formed to facilitate the sharing of equipment, knowledge, and technical services. The tools presented in this work were developed based on THC's technical needs.

The initial objective of this work was to synthesize the rates of task performance from each of the THC collections. This study was proposed to serve as a reference from which future digitization proposals might base labor estimates. Part II presents this synthesis as well as a series of task rate estimation formulas, and reference project labor estimations. Assessing task performance rates for multiple ongoing digitization projects brought to light processes which were either constraining throughput, or generally inefficient. Parts III, and IV present software created to improve two such processes. Part III introduces the development of the Herbarium Application for Specimen Auto Processing (HerbASAP), a real-time image processing program. HerbASAP was developed to reduce the steps necessary to prepare specimen images for internet dissemination. Most digitization efforts focus on mobilizing a backlog of historic specimens which were collected during 200+ years prior to programs like ADBC. These workflows are consequently reactive in nature causing the most demanding task, label data transcription, to be performed multiple times. Part IV documents the development of collNotes and collBook. Combined, these two programs provide a field-to-database solution which future researchers may use to organize their specimen label data in standardized, database ready formats. Although the perspective of these projects is limited to the experiences gained digitizing *Plantae* collections, considerations have been taken such that these programs may be readily adapted across some taxonomic barriers. In aggregate these efforts comprised training over 100 individuals on natural history digitization tasks, mobilizing over 273,000 herbarium specimen records with images, and writing more than 7,000 lines of original, free, and open-source code.

REFERENCES

Ariño, A. 2018. Putting your finger upon the simplest data. *Biodiversity Information Science and Standards*, *2*, e26300. https://doi.org/10.3897/biss.2.26300

Barkworth, M. E., & Murrell, Z. E. 2012. The US Virtual Herbarium: working with individual herbaria to build a national resource. *ZooKeys*, *209*, 55–73. https://doi.org/10.3897/zookeys.209.3205

Lang, P. L. M., Willems, F. M., Scheepens, J. F., Burbano, H. A., & Bossdorf, O. 2019. Using herbaria to study global environmental change. *New Phytologist*, *221*(1), 110–122. https://doi.org/10.1111/nph.15401

Lavoie, C. 2013. Biological collections in an ever changing world: Herbaria as tools for biogeographical and environmental studies. *Perspectives in Plant Ecology, Evolution and Systematics*, *15*(1), 68–76. https://doi.org/10.1016/j.ppees.2012.10.002

Losos, J. B., Arnold, S. J., Bejerano, G., Brodie, E. D., Hibbett, D., Hoekstra, H. E., Mindell, D. P., Monteiro, A., Moritz, C., Orr, H. A., Petrov, D. A., Renner, S. S., Ricklefs, R. E., Soltis, P. S., & Turner, T. L. 2013. Evolutionary Biology for the 21st Century. *PLoS Biology*, *11*(1). https://doi.org/10.1371/journal.pbio.1001466

Nelson, G., & Ellis, S. 2019. The history and impact of digitization and digital data mobilization on biodiversity research. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *374*(1763), 20170391. https://doi.org/10.1098/rstb.2017.0391

Soltis, D. E., & Soltis, P. S. 2016. Mobilizing and integrating big data in studies of spatial and phylogenetic patterns of biodiversity. *Plant Diversity*, *38*(6), 264–270. https://doi.org/10.1016/j.pld.2016.12.001

Souza, E. N. F., & Hawkins, J. A. 2017. Comparison of herbarium label data and published medicinal use: herbaria as an underutilized source of ethnobotanical information. *Economic Botany*, *71*(1), 1–12. https://doi.org/10.1007/s12231-017-9367-1

Vollmar, A., Macklin, J. A., & Ford, L. 2010. Natural history specimen digitization: challenges and concerns. *Biodiversity Informatics*, *7*(2). https://doi.org/10.17161/bi.v7i2.3992

Willis, C. G., Ellwood, E. R., Primack, R. B., Davis, C. C., Pearson, K. D., Gallinat, A. S., Yost, J. M., Nelson, G., Mazer, S. J., Rossington, N. L., Sparks, T. H., & Soltis, P. S. 2017. Old plants, new tricks: phenological research using herbarium specimens. *Trends in Ecology & Evolution*, *32*(7), 531–546. https://doi.org/10.1016/j.tree.2017.03.015

PART II: Estimation of Herbarium Specimen Digitization Rates

FORWARD

Part II has been organized in preparation for publication in collaboration with the following co-authors: Alaina Krakowiak, Rachel Fuller, Erica Rylander, Emily Gillespie, Shawn Krosnick, Brad Ruhfel, Ashley B. Morris, and Joey Shaw. This work possible thanks to the data and efforts contributed by these collaborators. The term "we" used in Part II refers to those collaborators, and myself. My primary contributions to this work included (1) project coordination, (2) data cleaning, (3) rate estimation formula derivation and labor estimation simulations, (4) setback analysis, and (5) most of the writing.

INTRODUCTION

During the digitization of nine SERNEC collections (Table 1), task-specific data were collected over 34 months (June 2016 to May 2019) spanning 7,808 hours across a workforce of 105 people producing over 273,000 digitized herbarium specimen records. Labor comprised the majority of these effort's budgets (>96%)(J. Shaw, personal communication, 2020). Here, an analysis of these data is presented as a reference from which future digitization efforts may benefit. Most of the technicians involved in these efforts were undergraduate students, many of whom (including four of the authors of this manuscript) were introduced to natural history collections through these initiatives. It is therefore presumed that the average participant of this study had little to no pre-existing expertise in natural history collections. In general, the rate of manual task performance is expected to improve with experience. Under the assumption that future digitization efforts will be similarly staffed, labor estimates may benefit from accounting for changing rates of task performance and consequent economies of scale associated with longer technician retention times (i.e., contract duration).

Factors such as location, space, specimen organization, and scope of digitization tasks make every digitization effort unique and difficult to generalize. Consequently, the pre-digitization curation necessary to prepare for such a project is highly specific to each collection such that the rate of this task is not discussed in detail. Tasks which ultimately follow pre-digitization curation, (i.e., "digitization tasks") include: indexing specimens usually referred to as "barcoding," imaging specimens, and transcribing either complete specimen label data which we refer to as "complete databasing" or a minimal subset of label data, which we refer to as "skeletal databasing."

Few works have published task-specific rates from which to reference and among those tasks may be combined, or in some instances omitted (Nelson et al., 2012; Tulig et al., 2016; Harris and Marsico, 2017; Sweeney et al., 2018). These issues of data availability and compatibility represent challenges for future digitization projects seeking reference task rates. Tulig et al. (2012) presented complete databasing rates of 0.167 specimens per minute (SPM), and skeletal rates of 2.083 SPM (Tulig et al., 2012). In both cases, these rates included barcode application. Additionally, Tulig et al. (2012) present an image capture rate of 1.417 exposures per minute (EPM), a metric distinct from SPM in that it accounts for the infrequent incidence of a single specimen occupying multiple herbarium sheets. Nelson et al. (2012) presented an imaging capture rate of approximately 1.667 herbarium sheets per minute, which is functionally equivalent to EPM (Nelson et al., 2012). Thiers et al. (2016) published an imaging rate of approximately 1.667 images per minute (Thiers et al., 2016). Harris & Marsico (2017) published an imaging rate of 2.417 SPM and complete databasing rates of 0.417 SPM using undergraduate students, and 0.783 SPM with a graduate student (Harris and Marsico, 2017). Sweeney et al. (2018) published digitization rates resulting from an automated conveyor system (Sweeney et al., 2018). This automated system combined imaging and data capture of a set of fields significantly exceeding skeletal databasing at a rate of 0.593 SPM when accounting for system minutes, and 0.375 SPM when accounting for combined operator minutes (i.e., "person minutes"). Additionally, Sweeney et al. (2018) present an imaging only rate of 2.19 SPM when accounting for system minutes. Although valuable references for planning a digitization effort, none of these works specifically account for the productivity increases associated with task mastery as a function of technician experience.

The data gathered and analyzed for this study is highly granular, containing individual technician rates per task per work session. Using this level of specificity, we assess the rate of task performance and provide guidance on the estimation of the labor required to digitize a herbarium. The objectives of this analysis were to: (1) determine the average rate of worker improvement as a function of experience, and (2) derive required labor estimates across a range of specimen counts for each task, specifically, barcoding, skeletal databasing, and specimen imaging. Additionally we attempt to characterize the nature and impact of the unexpected setbacks experienced throughout the project.

METHODS

Scope of digitization tasks

Technicians from each collection were trained on digitization tasks and provided step-wise workflows consistent with those proposed by Nelson et al. (2015). Generally, the skeletal data transcribed in this dataset included: a collection-unique barcode number called a catalog number, the specimen's scientific name, and the state and county in which the specimen was collected. Barcoding involved applying archival stickers with unique identifiers to herbarium sheets in a consistent fashion. Imaging at each collection was performed using identical or functionally similar sets of equipment, and in practice involved moving specimens from one stack into a rigid light box, capturing a photograph, and moving them to another stack. Skeletal databasing was performed directly in the SERNEC portal (sernecportal.org), built on the Symbiota platform (Gries et al., 2014), using the skeletal data entry tool and generic USB barcode scanners.

Data collection

Google Forms were used as a reporting tool whereby any workers on this project, be they volunteers, hourly workers, work study students paid by a university, or students working for academic credit on independent study, were required to report minutes spent, tasks performed, and any setbacks experienced during each work session. For consistency, pulldown or selection menus were created for student name, date, and herbarium associated with specimens. Additionally, data validation was enforced for numeric inputs (i.e., number of minutes performing a task and number of specimens on which the task was performed). Workers were asked never to round their time beyond the nearest five minutes and to report as precisely as possible the exact number of specimens barcoded, imaged, or skeletally databased. A free-entry field was provided to describe any setbacks or problems which may have occurred that were not representative of a typical workflow (e.g., training, technical difficulties, etc.).

Data cleaning

The task-specific reports were cleaned and analyzed in Python, using the Pandas module (McKinney, 2010). Cleaning the dataset omitted 2,649 hours across 885 entries based on the following criteria: non-representative reporters (337 entries), exceptionally non-representative workflow (3 entries), apparent entry errors (36 entries), indicated setbacks (453 entries), and finally extreme outliers (82 entries). Non-representative reporters were identified from one of three categories: ambiguous usernames (two usernames), unreliable reporters (eight users), and the primary author, whose tasks were not readily generalizable. The two ambiguous usernames in the dataset represented multiple students participating in digitization assignments integrated into course work at Tennessee Tech University (TTU) and the University of Tennessee at Chattanooga (UTC). Among the eight unreliable reporters, three were discovered to be submitting fraudulent reports, and five were strongly suspected of submitting unreliable reports. Known cases of fraudulent reporting were identified based on equipment and space availability (e.g., a technician reporting using an imaging station at a time in which it was being used elsewhere). Also omitted from the analyses were three entries from an exceptionally non-representative workflow, as well as 36 apparently erroneous entries, wherein task rates (specimens per minute) were suspiciously low such that the inverse ratio (minutes per specimens) resembled usual task averages. The free-entry field used for setback reporting lacked data validation or controlled vocabulary and thus required explicit cleaning. All text in the setback fields had punctuation removed and were converted to lowercase. The most frequent words and phrases included in the setbacks field were assessed manually to derive a set of acceptable phrases which indicate that no significant setbacks occurred (e.g., “no,” “none,” “no setbacks”). All instances of setbacks were evaluated for equivalency to any of the acceptable phrases whereby each setback entry was converted to lowercase, had punctuation removed and then compared to each acceptable phrase for equality. Any entry with nonequivalent (i.e., unacceptable) setback data was identified as a non-typical measurement resulting from a significant setback. These setback entries were stored for separate evaluation yet omitted from subsequent task rate analyses. Phrases indicating “training” were not among the acceptable phrases, therefore the time invested during initial training of new technicians is not accounted for in the task rate analyses. Extreme outliers were defined as entries with any numeric values exceeding 5 standard deviations within that field. A total of 5,158 hours across 2,475 entries remained in the dataset following cleaning operations.

Data analysis

After cleaning, session report data were grouped by technician name and sorted by ascending date of work session. Additional fields were calculated to track individuals’ cumulative time performing each task at the time the report was submitted. All technicians’ cumulative times were then grouped into shared two-hour bins. The mean rate of performance (i.e., number of specimens over minutes) for each task, was calculated among all entries present in each two hour bin. This method allowed non-contemporary technicians to be compared at times when they had achieved approximately equal experience. In a few cases work sessions spanned periods of time longer than the task bins (i.e., over two hour sessions), omitting those individuals from that bin's mean rate. The task rates per bin were then fit to a regression scatterplot using the Python Seaborn library (Waskom et al., 2018) and visually assessed for fitness. Each scatterplot was visually inspected for a fit, and cumulative-hour cutoff threshold. A 64 cumulative-hour threshold was selected for all tasks, as the number of participants informing the mean of each two-hour task-rate bin becomes greatly diminished beyond that point. Imaging and skeletal databasing were fit to a simple linear function which was calculated using the linregress function available from the Python library Scipy (Virtanen et al., 2019). Barcoding was fit to a second order polynomial calculated using the polyfit function from the Python library Numpy (van der Walt et al., 2011).

Using these models, a series of simulations were performed using Python to generate labor estimations for digitization tasks over specific numbers of specimens, as a function of technician turnover rate (Table 2). Each simulation accounted for technician turnover by resetting the task rate following the completion of a quantity of labor hours which represented the simulated technician's contract duration (i.e., total time performing a single task) (Fig. 4). Contract durations simulated under these methods were: 15, 45, and 135 hours performing each task. Since 135 cumulative hours for a single individual on a single task exceeds the 64 hour model limits, arbitrary rate limits were set based on reasonable extremes so that no rate estimations were extrapolated beyond the limits of the dataset. Maximum rates of 4.00 SPM and 6.50 SPM were set for imaging and databasing, respectively, while a minimum rate of 3.00 SPM was set for barcode application. During simulations, rate estimates exceeding these limits were instead held at the respective limit.

RESULTS

Labor requirements - by task

Across all tasks and collections, 7,808 hours were recorded, after cleaning 5,023 total hours remained. Of the post-cleaning hours, 3,493 were spent on the primary digitization tasks (i.e., imaging, skeletal databasing, barcode application) with the remaining 1,530 hours spread across other tasks (e.g., pre-digitization curation). Across all collections, post-cleaning: 229,333 specimens were imaged at 2.30 per minute over 1,660 hours, while 231,307 specimens were skeletally databased at 3.14 per minute over 1,228 hours, and 180,949 barcodes were applied at 4.07 per minute over 740 hours.

Project wide digitization rates depend on how one defines a specimen as being digitized and which hours are included in that effort. Using the mean total pre-cleaning specimens from each task as total specimens digitized and the combined pre-cleaning hours reported: all collections combined documented digitizing 306,069 specimens at 0.653 per minute over a total of 7,808 hours. Using the mean total post-cleaning specimens from each task as total specimens and the total post-cleaning hours reported: all collections combined documented digitizing 213,863 specimens at 0.691 per minute over a total of 5,158 hours. Finally, using the mean total post-cleaning specimens from each task as total specimens and the post-cleaning hours spent on primary digitization tasks: all collections combined documented digitizing 213,863 specimens at 0.983 per minute over a total of 3,628 hours.

Average technician imaging and data entry rates were fit to a linear function and plotted (Fig. 1, Fig. 2). Imaging rate as a function of cumulative time is estimated as $y=1.95170+0.02118x$. Skeletal data entry rate as a function of cumulative time is estimated as $y=2.55659+0.02760x$. Barcode application rates were fit to a second order polynomial and plotted (Fig. 3). Barcode application rate as a function of cumulative time is estimated as $y=3.7216+0.09928x-0.00175x^2$. In each formula, *y* represents the estimated specimens per minute, while x represents the technician's cumulative hours performing the specific task. The labor estimations informed by these models are reported in Table 2 .

The 453 entries identified as significant setbacks totaled 946 hours or 12% of total pre-cleaning hours documented in this dataset. The mean duration of all sessions with setbacks was 133 minutes. By task, the mean duration of sessions with setbacks was 142 minutes for barcode application, 143 minutes for imaging, and 121 minutes for skeletal data entry. Conversely, the mean duration of all sessions without setbacks was 101 minutes. By task, the mean duration of sessions without setbacks was 88 minutes for barcode application, 112 minutes for imaging, and 97 minutes for skeletal data entry. Many of the most frequent words (and their frequencies) among setback descriptions were as follows: “specimen(s)” (68), “training” or “blitz” (59), “sernec” (43), “skeletal” (31), “slow” (25), “barcode(s)” (17), “imaging” (15), “folders” (14), “computer” (14), “label” (14), “missing” (13), “camera” (11), “collector” (11), “setting” (11), ”repair” (11), “internet” (9), “species” (9).

DISCUSSION & CONCLUSION

A priori planning and organization of a natural history digitization project, as well as individual technicians, will impact project efficiency. Over the course of this work technician imaging and databasing rates were shown to improve with experience (Figs. 1,2). Barcode application rates on the other hand began to drop following 30 cumulative hours (Fig. 3). It is our assumption that this inflection point where barcoding rates stop improving and begin to deteriorate, reflects the ease of mastering the simple task. However, an alternative explanation to this aberration might be explained by technicians “graduating” to other tasks. Since barcode application was often the first task on which technicians were trained, it is possible that the degradation in mean barcode application rates beyond 30 hours is due to improving technicians progressing to more complex tasks, while those technicians exhibiting no improvements continued to barcode. The labor estimations (Table 2) suggests that high technician retention can reduce total labor requirements by up to 20%. While high retention times detrimentally influence barcode rates, those rate losses are mitigated by improvements in the other tasks.

Although there are technical and physical limitations to the maximum achievable rates, technician retention time clearly influences overall digitization efficiency. In general, the longer technicians are engaged on a project, the more efficient the project will be. Longer retention times naturally implies fewer overall technicians. This intuitive outcome represents an unfortunate trade-off between efficiency and the type of outreach which initially introduced the primary author to this field. Integrating biological specimen digitization, or perhaps better yet the use of digitized specimen data into undergraduate coursework, may recoup the outreach opportunity cost associated with longer technician retention periods. This strategy was tested at UTC, where students were assigned to work a few hours on the digitization project. Since the students needed training and oversight by experienced persons, this method did not efficiently produce digitize specimens. It did however expose students to natural history collections and helped identify proficient students for project recruitment. Another solution to abate the reduction in outreach could be to use the barcoding task as an outreach opportunity while still maximizing retention times for those technicians performing imaging and data entry. This solution would have the additional benefit of utilizing the most efficient portions of each task's performance curve (Figs. 1,2,3).

The dataset and analysis presented here are provided in hopes it may be a useful reference for future digitization project planning. We acknowledge two caveats which should be addressed when using these data to formulate a labor budget. The first caveat is that the very process of collecting these data certainly has affected the results we present. All participants were trained on, and therefore aware of the rate tracking associated with these digitization projects. During the 34 month period of this study, the form used to gather the data became an invaluable tool for planning, reporting, and labor management. Because of this, we believe that active management of the labor force based on these reported rates is integral to the rate improvements documented in imaging and databasing. High rates, along with other achievements were periodically praised in a public context, yet in order to strike a balance between efficiency and quality of work, no material bonuses were provided for specific rates. Additionally, the habit of calculating and reporting a task rate after each session certainly maintained awareness for the importance of efficiency. Subjectively, it was observed that this heightened awareness motivated improvement over time with participants attempting to beat previous rates. Also, we observed that this awareness influenced individuals' task selection with participants preferring the tasks in which they were most competitive. The second caveat is that by omitting extreme outliers, non-representative reporters, and significant setbacks (including training), the estimation formulas assume unrealistically ideal scenarios. For example, from the 7,808 total pre-cleaning hours documented, approximately 12% (946 hours) contained significant setbacks. The mean session duration (for all tasks combined) increased by 32% when a setback was present. Subjectively, the majority of the setbacks documented were of a technical nature (e.g., internet connectivity, or camera settings). In addition to the setbacks, nearly 12% of pre-cleaning hours (918 hours) were identified as either unreliable, or in rare cases outright fraudulent. Labor predictions derived from these formulas, using 45 hour contract durations underestimate real world labor expenditures of the participating collections by 25%. In light of these caveats, we recommend efforts citing these data for labor estimations should implement and actively use a similar session rate tracking system and include an appropriately sized buffer (e.g., 20-30%) to the labor budget.

Labor estimations based on Table 2 or the formulas presented here are influenced by total specimen count, as well as expanding technician experience by way of contract duration. These estimates also assume a relatively similar workflow. It is therefore difficult to formulate an equitable comparison among the published rates discussed here. We are not aware of any works which account for technician experience, and very few include the number of specimens informing their task rates. Only the recent works from Sweeney et al. (2018), and Harris and Marsico (2017) include the specific quantity of specimens from which their rates were derived (Harris and Marsico, 2017; Sweeney et al., 2018). Among these two works, only the imaging process reported by Harrs & Marsico (2017) followed remotely similar workflows. In their work, Harris and Marsico (2017) estimate based on their findings that a single person could image 20,000 specimens in 13, 10-hour weeks or 130 hours. Our method assumes technician turnover following a specified contract duration, using an arbitrarily large contract duration therefore implies a single technician performing the task. In this way we estimated one technician could image 20,000 specimens in 114 hours (Table 2). Our estimate of 114 hours, falls 14.0% below that of Harris and Marsico's (2017) estimated 130 hours and 20.1% below their observed rate of 2.417 SPM when extrapolated over 20,000 specimens (135 hours). Within the context of the caveats discussed above, these underestimates are anticipated. Tulig et al. (2012) present rates of 2.083 SPM for the combined tasks we define as skeletal databasing, and barcoding and 1.417 EPM for imaging (Tulig et al., 2012). Although they do not provide a specific specimen count informing that rate they do state their rates exclude technical training and troubleshooting. Extrapolating from their rate of 2.083 SPM suggests it would require 800 hours to barcode and skeletally database 100,000 specimens. Our estimates for performing these two tasks across 100,000 specimens range from 1,388 hours using 15 hour-contracts, 874 hours using 45-hour contracts and 872 hours using 135-hour contracts. Given the scope of the effort presented in their work, we believe it is reasonable to assume that longer contract periods are more representative. Using only those longer contract periods, our methods overestimated the hours necessary by 9%. Extrapolating the imaging rate of Tulig eg al. (2012), 1.417 EPM suggests it would require 1,176 hours to image 100,000 specimens yet imaging estimates range from 787 using 15-hour contracts, 685 using 45-hour contracts and 561 using 135-hour contracts. Assuming longer contract periods and equivalency between SPM and EPM our estimates are significantly smaller than those resulting from the 1.417 EPM rate presented Tulig et al. (2012), by as much as 52%. Increases in computational power, differences in workflow, and labor management tools are all possible contributing factors in this disparity.

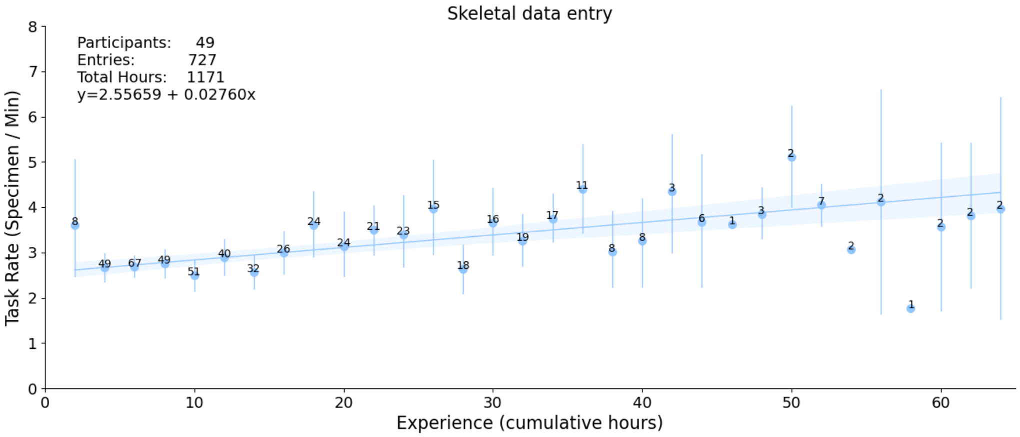
We evaluated 7,808 hours of herbaria digitizing activities spanning 34 months across a workforce of 105 people. These data were assessed to determine the average rate of worker improvement as a function of cumulative experience and provide labor estimates for common digitization tasks. We believe the estimations presented are achievable when using similar workflows and incorporating individual session rate tracking tools into the digitization effort. Since these estimates represent the rates possible when no unforeseen delays are present, we recommend an additional 20-30% of labor funding be included to account for setbacks such as those discussed here. Supplemental information is available which includes: an anonymized form of the pre-cleaning data and task-specific labor estimation line graphs for multiple contract durations (github.com/CapPow/digitization_rates_si).

Table 1: The collections which contributed task rate data, and their contributions. Factors which may contribute to unequal specimen counts are: incomplete reporting, data cleaning, or workflow differences such as existing progress from previous efforts.

Collection	Collection Code	Participants	Total Hours	Barcode Application (Specimens)	Skeletal Data Entry (Specimens)	Skeletal Data Entry (Specimens)
Berea College, Ralph L. Thompson Herbarium	BEREA	7	237	51	22,105	22,105
East Tennessee State University	ETSU	7	127	9,861	4,355	4,355
Middle Tennessee State University	MTSU	17	875	3,348	73,639	73,639
Rhodes College	SWMT	5	174	4,881	15,789	15,789
Tennessee Technological University	HTTU	25	520	18,053	15,950	15,950
University of Tennessee Chattanooga	UCHT	42	1,004	51,351	34,478	34,478
University of Tennessee Knoxville	TENN	36	2,960	174,118	175,966	175,966
University of Tennessee Martin	UT-M	2	49	1,810	2,736	2,736
University of the South Sewanee	UOS	3	8	525	661	661

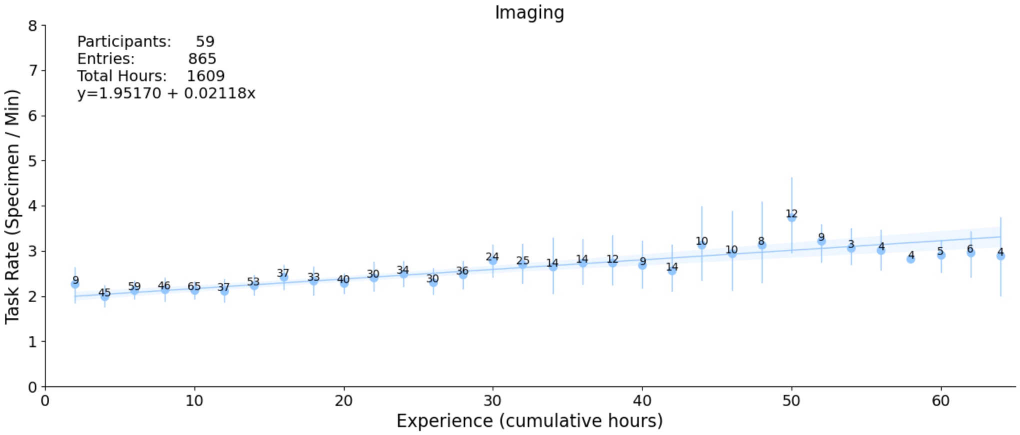
Table 2: Labor estimates for various digitization tasks for multiple collection sizes across multiple contract durations.

Contract Duration	**Task**	Specimen Count												
		10,000	**20,000**	**30,000**	**40,000**	**50,000**	**75,000**	**100,000**	**125,000**	**150,000**	**200,000**	**250,000**	**300,000**	**500,000**
15	Barcoding	39	77	115	153	191	287	382	477	573	763	954	1,145	1,907
	Databasing	61	121	181	241	301	451	601	751	901	1,201	1,501	1,801	3,001
	Imaging	79	158	236	315	394	590	787	983	1,179	1,572	1,965	2,358	3,929
	Combined	179	356	532	709	886	1,328	1,770	2,211	2,653	3,536	4,420	5,304	8,837
45	Barcoding	35	71	106	141	175	262	349	436	524	698	873	1,047	1,743
	Databasing	54	107	159	211	263	394	525	655	786	1,047	1,306	1,568	2,612
	Imaging	71	138	208	275	344	515	685	855	1,027	1,369	1,709	2,053	3,418
	Combined	160	316	473	627	782	1,171	1,559	1,946	2,337	3,114	3,888	4,668	7,773
135	Barcoding	35	80	135	170	214	323	440	558	673	887	1,114	1,346	2,233
	Databasing	51	90	129	179	218	327	432	532	641	858	1,064	1,282	2,127
	Imaging	64	114	175	227	281	422	561	701	840	1,119	1,397	1,674	2,782
	Combined	150	284	439	576	713	1,072	1,433	1,791	2,154	2,864	3,575	4,302	7,142



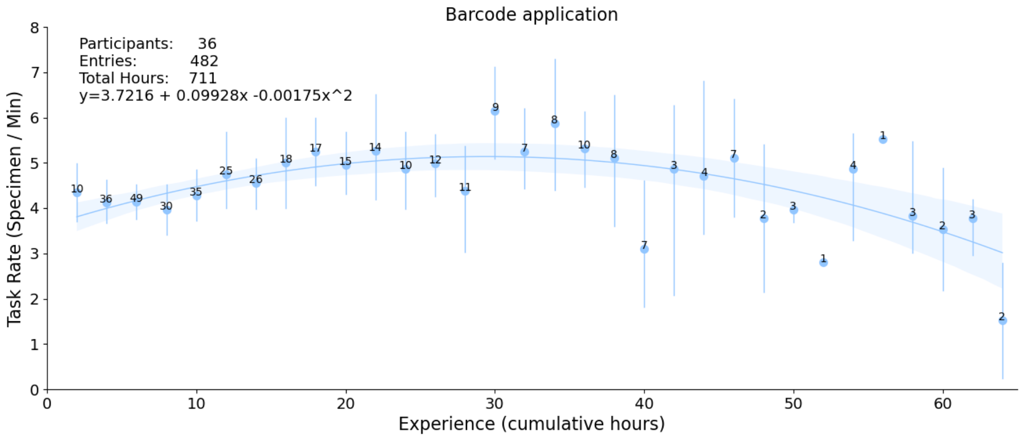
The average technician skeletal data entry rate (specimen/minute) as a function of cumulative hours performing skeletal data entry. The mean rate at each 2-hour bin is indicated by the blue point, and the number of data points informing the mean is annotated over each point. The range of values at each bin is indicated by vertical bars.

Figure 1 Average skeletal data entry rates as a function of experience



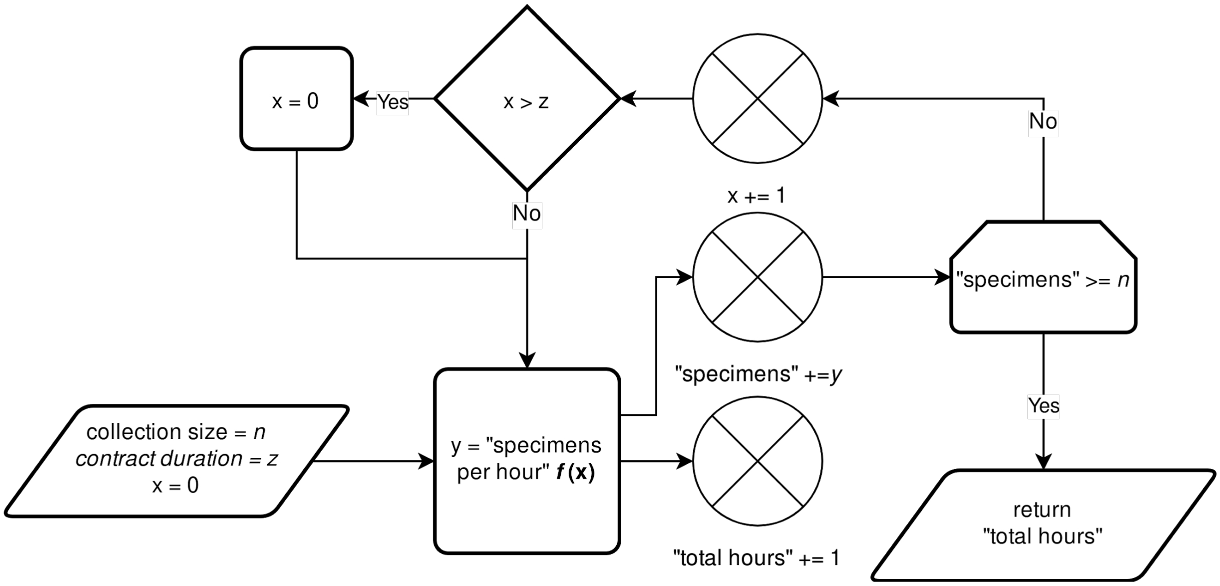
The average technician imaging rate (specimen/minute) as a function of cumulative hours imaging. The mean rate at each 2-hour bin is indicated by the blue point, and the number of data points informing the mean is annotated over each point. The range of values at each bin are indicated by vertical bars.

Figure 2 Average imaging rates as a function of experience



The average technician barcode application rate (specimen/minute) as a function of cumulative hours applying barcodes. The mean rate at each 2-hour bin is indicated by the blue point, and the number of data points informing the mean is annotated over each point. The range of values at each bin is indicated by vertical bars.

Figure 3 Average barcode application rates as a function of experience



Flowchart illustrating the process of estimating the total hours necessary to perform a digitization task for a specific number of specimens. Given the collection size to be digitized ("n"), a maximum contract duration ("z"), the estimated specimens processed per hour is determined for each hour until "n" specimens are processed. The rate is determined as a function of technician experience ("x") proxied by cumulative hours performing the task. If "x" exceeds the contract duration "z", it is reset to 0, simulating technician turnover. Once total specimens processed is greater than or equal to "n," the simulation halts returning the total accumulated hours.

Figure 4 Labor estimation process flowchart

REFERENCES

Gries, C., E. E. Gilbert, and N. M. Franz. 2014. Symbiota – a virtual platform for creating voucher-based biodiversity information communities. *Biodiversity Data Journal*.

Harris, K. M., and T. D. Marsico. 2017. Digitizing specimens in a small herbarium: a viable workflow for collections working with limited resources. *Applications in Plant Sciences* 5: 1600125.

McKinney, W. 2010. Data structures for statistical computing in Python. 51–56.

Michael Waskom, Olga Botvinnik, Drew O'Kane, Paul Hobson, Joel Ostblom, Saulius Lukauskas, David C Gemperline, et al. 2018. mwaskom/seaborn: v0.9.0 (July 2018). Zenodo.

Nelson, G., D. Paul, G. Riccardi, and A. Mast. 2012. Five task clusters that enable efficient and effective digitization of biological collections. *ZooKeys* 209: 19–45.

Nelson, G., P. Sweeney, L. E. Wallace, R. K. Rabeler, D. Allard, H. Brown, J. R. Carter, et al. 2015. Digitization workflows for flat sheets and packets of plants, algae, and fungi. *Applications in Plant Sciences* 3: 1500065.

Sweeney, P. W., B. Starly, P. J. Morris, Y. Xu, A. Jones, S. Radhakrishnan, C. J. Grassa, and C. C. Davis. 2018. Large–scale digitization of herbarium specimens: development and usage of an automated, high–throughput conveyor system. *TAXON* 67: 165–178.

Thiers, B. M., M. C. Tulig, and K. A. Watson. 2016. Digitization of The New York Botanical Garden Herbarium. *Brittonia* 68: 324–333.

Tulig, M., N. Tarnowsky, M. Bevans, Anthony Kirchgessner, and B. M. Thiers. 2012. Increasing the efficiency of digitization workflows for herbarium specimens. *ZooKeys*: 103–113.

Virtanen, P., R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, et al. 2019. SciPy 1.0--Fundamental algorithms for scientific computing in Python. *ArXiv:1907.10121*.

van der Walt, S., S. C. Colbert, and G. Varoquaux. 2011. The NumPy Array: a structure for efficient numerical computation. *Computing in Science Engineering* 13: 22–30.

PART III : Automating Specimen Image Post-Processing

FORWARD

Part III has been organized in preparation for publication in collaboration with the following co-authors: Dakila Ledesma, Jacob Motley, Jason Best, Hong Qin, and Joey Shaw. This work was the product of that collaboration and consequently the term “we” used in Part III refers to those co-authors, and myself. Portions of this work which could safely be considered as having been primarily my effort include (1) project conceptualization, (2) project coordination (3) user interface design excluding the icon, (4) settings profile management system, (5) the scaleRead module, (6) the bcRead module, and (7) most of the writing.

INTRODUCTION

One of the most important tasks involved in digitizing herbaria is the generation of high quality, high resolution specimen images. Variations on this fundamental digitization task have been described in published workflows, some of which address the computational operations performed on image files after they are captured, so called “image post-processing” (Nelson et al., 2012, 2015; Tulig et al., 2012). Such operations may include: demosaicing, metadata refinement, rotation, white balancing, and image file renaming. These operations are performed to produce image derivatives which are accessible, and accurate representations of the specimens useful to future research. Among these tasks, demosaicing is the most ubiquitous process.

Images produced by the digital cameras commonly used for natural history digitization store unprocessed pixel values, often in a proprietary file formats (e.g., “CR2”, “NEF”), so called RAW images. These formats are appealing for archival purposes because they retain the highest level of detail through what is fundamentally a matrix of unaltered sensor readings taken at the time of exposure. Each discrete sensor reading, represented as a pixel, only measures one color making RAW images unsuitable for direct observation. The specific color each pixel is able to sense is distributed in a pattern across an array, the most common of which is a Bayer array where individual pixels sense either Red, Green, or Blue (RGB)(Figs. 5A, 5B). For digital images captured in this manner the missing colors at each pixel must be inferred from neighboring pixels in a process called demosaicing (Stamatopoulos et al., 2012). Numerous demosaicing algorithms exist, with varying degrees of performance respecting speed and color accuracy. It is important to note that the need to demosaic is not circumvented when digital cameras are set to produce formats other than RAW (e.g., JPEG). In those instances, demosaicing is performed using the camera's onboard processor some which often balances speed with accuracy and does not necessarily produce the most accurate representation of the object (Stamatopoulos et al., 2012). We are not aware of any works which explicitly examine the demosaicing process as it relates to herbaria digitization, yet all digitization projects utilizing DSLR cameras are performing the process. For many such projects, this is being automatically performed through proprietary image processing software recommended in published workflows (Nelson et al., 2012, 2015; Tulig et al., 2012).

The necessity of other image post-processing operations (i.e., metadata refinement, rotation, white balancing, and image file renaming) is determined by curator preference and digitization workflow. Metadata are stored in specimen images to document ancillary data concerning the specimen, the collection in which it is stored, and any pertinent digitization steps. Image rotation is the verification and, if necessary, correction of image orientation with respect to the specimen label text. Many DSLRs contain an orientation sensor calibrated to determine if the operator is holding the camera in landscape or portrait mode. Because digitization workflows often involve mounting the camera face down and angled on an axis the orientation sensor is not calibrated for, subtle movements in the imaging apparatus may result in fluctuations of image orientation.

White balance is performed to mitigate unrealistic color representations caused by the color temperatures of lighting equipment. In herbaria digitization workflows, this is frequently performed by referencing white points within the white patches on a standardized color reference chart (CRC). In some cases, the Red, Green and Blue (RGB) values of a reference white point are set using camera settings and stored in the image metadata. Most image processing software will automatically perform white balance based on these metadata making this method relatively simple to set up and use. Alternatively, many workflows perform white balance on batches of specimen images which were captured within a similar time period. In this method, image processing software is used to select a white point on the CRC of a representative image and the RGB values at that point are used to white balance the entire batch. In both methods, a degree of lighting condition continuity is assumed across many specimens. Accurate white balance requires periodic white point updates to account for changes in ambient lighting conditions such as those caused by exposed windows or lighting equipment wear.

Common herbaria digitization workflows recommend naming image files based on values encoded in barcodes present on the specimen (Nelson et al., 2015). This process is used to facilitate linking specimen images with their associated label data. Sequential naming schemes are the most direct method to achieve this. Yet, sequential naming assumes images are captured in the same order which barcodes were applied and is therefore vulnerable to synchronization errors. One alternative to sequential naming is to combine a custom dialog box with a handheld barcode scanner, adding an additional albeit brief, step for the operator. Due to the potential synchronization errors or delays involved with these approaches, some collections have developed programmatic solutions which are applied during post-processing (Barber et al., 2013; Sweeney et al., 2018).

Post-processing represents an additional step in the digitization workflow, requiring time and resources. Decisions made during this step will influence the quality and consistency of the research objects produced during the digitization process. In some cases, quality control measures are taken after specimens have been imaged, making it a reactive process. Here we present the development and use of the Herbarium Application for Specimen Auto Processing, or HerbASAP. HerbASAP uses parallelized processing and artificial intelligences to automate specimen image post-processing in real time concurrent with the imaging task. Beyond the procedures typically associated with post-processing, HerbASAP incorporates additional features we believe assistive to both the imaging process, and research which uses those images. These features include novel quality controls (e.g., blur detection), efficiency tools (e.g., real-time imaging rate calculator), and supplemental image analysis (e.g., pixel-to-mm scale). For example, we are not aware of any collections including a pixel-to-mm scale in the specimen images they distribute yet we believe this information may be valuable for future researchers. Numerous works have applied machine learning and computer vision techniques to these emerging herbarium specimen image datasets (Unger et al., 2016; Wilf et al., 2016; Carranza-Rojas et al., 2017; Gehan et al., 2017; Weaver et al., 2018; Younis et al., 2018; Lorieul et al., 2019). A pixel-to-mm ratio combined with such efforts could automate the measurement of morphological traits such as petiole length or leaf area vastly increasing the feasible sample size for related studies.

METHODS & RESULTS

HerbASAP development philosophy

HerbASAP was developed as an open-source solution for the automation of herbarium image processing, designed to run concurrently with imaging operations. There are several benefits to real time post-processing. For example, error corrections can be performed before specimens are filed back into the collection, thereby reducing handling. Import and processing times required for post-processing are offloaded to operate within the brief window in which the imaging station is idle while the operator transitions to the next specimen to be imaged. Additionally, by avoiding batch processing when performing white balance it is no longer necessary to assume lighting condition continuity.

Two development goals were set for this project: (1) HerbASAP should be highly accessible to collections despite funding limitations, and (2) HerbASAP's operation should not impede the rate at which specimens are imaged. Pursuant to the first goal, HerbASAP is being released with open-source licenses on all major operating systems (Linux, macOS, & Windows) with a focus on supporting low cost, highly accessible computer equipment. The second goal was addressed by targeting a perceived runtime (i.e., "wall time") for processing high resolution images (3,840px by 5,700px) which is faster than nearly all imaging technicians documented in previous work (in prep, Part II of this thesis). The summation of these goals meant processing images in under 15 seconds on affordable equipment (e.g., Dell Latitude E7440 notebook).

The program was written in Python 3.7 using Qt5 for the graphical user interface (UI). The interface layout was designed using Qt designer 5.12. The prioritization of wall time means the majority of operations are performed through libraries which leverage faster codebases. RAW images are demosaiced using the python library Rawpy, (github.com/letmaik/rawpy) which is a compatibility layer (i.e., "wrapper") for the C++ library LibRaw (github.com/LibRaw/LibRaw). Image manipulations are performed using the Python Imaging Library (wiredfool et al., 2016), Numpy (van der Walt et al., 2011), as well as the Python implementation of OpenCV (Bradski, 2000).

As frequently as possible, operations are parallelized through PyQt's QThread class using the "TimeCriticalPriority" scheduling policy which requests the operating system preferentially perform them before others. These optimizations make HerbASAP a multithreaded program, able to leverage multicore processors. A consequence of multithreaded programming is the added complexity of managing asynchronous returns from parallel operations. These challenges were addressed by dividing post-processing operations into two phases. Phase one focuses on gathering information from the source image while phase two uses what was learned to produce derivative images suitable for distribution. Phase one begins by loading the RAW image file using Rawpy parameters which explicitly disable or minimize pixel manipulation while optimizing wall time through fast yet less accurate demosaicing methods (Figs. 5C, 5D). Phase two loads the same RAW image in Rawpy but using white balance and rotation parameters determined during phase one while optimizing image quality by using adaptive homogeneity-directed demosaicing (Hirakawa and Parks, 2005)(Figs. 5E, 5F). Equipment corrections and metadata are applied to the phase two image object which is then passed to openCV's "imwrite" function to create a web ready jpeg image file. A generalization of this process is illustrated in Fig. 6.

Using HerbASAP

Designed to be setup by curatorial staff, yet operated by potentially inexperienced operators (e.g., students, or volunteers), HerbASAP uses a profile system to select and manage settings configurations. A setup wizard is used to walk curators through the process of creating the settings profile(s). If no profiles are found at initialization, or if the user selects to create or modify a settings profile, the setup wizard is initiated. Each step of the setup wizard is used to explain to users HerbASAP's optional features, and save input based on relevant preferences. For example, if a collection opts to utilize CRC based features (i.e., white balance, orientation correction, scale determination) the settings wizard requests that the user draw a box over the CRC of an example image. This task is used to determine an appropriate partition size used to divide the image into distinct sections for analysis.

Menus in HerbASAP are navigated using horizontal tabs on the top of the UI, one for processing images, and another for settings. The Processing images screen is organized into two vertical parts: (1) the information, and activity panes (Fig. 7A), (2) a processed image preview window (Fig. 7B). The information pane displays information about the image processing following phase one, such as the cropped region of the image identified as the CRC and the white patch used to sample white balance RGB values. The activity pane contains buttons and user entry boxes relevant for image processing. The image preview window displays a preview of the image following phase two. Parameters selected during the image resizing necessary to prepare the preview window's image do not affect the image's derivatives, and were selected to optimize wall time. Therefore, the preview window provided for spot checking the results of phase two processing displays images at a lower quality than the exported derivatives.

There are two mutually exclusive options to initiate image processing, which are selected using tabs in the image processing initiation options (Fig. 7A). To support backlog and one-off processing, image processing may be initiated by explicitly selecting image(s), or entire folders. However, HerbASAP is primarily intended to run concurrent with specimen imaging by monitoring a source folder during an imaging session. When new RAW images are detected during that session, HerbASAP immediately begins processing them and exporting the derivatives. HerbASAP keeps a running queue of images which need to be processed, so it is not necessary to wait for image processing to complete before adding new images to the source folder. In many digitization workflows specimens are imaged following a logical organization resulting in select features being shared across multiple, sequentially captured images. The UI has optional fields which may adventitiously leverage this organization by recording shared features in metadata. Two typical use cases are supported by these optional fields: (1) when imaged specimens are organized taxonomically, a taxon name field is available and (2) when a single researcher has accessioned multiple specimens to a collection, a collector name field is available. These two use cases were selected based on the infrequency of updates they would require from the operator. The operator's name may also be stored in the metadata to document their contribution to the effort and facilitate the collection of digitization project metrics. While a folder is being monitored, metrics relevant to that session are automatically tracked and displayed in the UI but not included in the metadata of derivatives. These metrics are included to simplify performance tracking similar to the recommendations described in Part II (in prep, Part II of this thesis). Metrics include: duration of the session, the number of images processed, and the rate at which derivatives are produced which is intended as a proxy for specimen image capture rate. In addition to session metrics, a series of graphical badges are presented to the operator based on the most recent calculated rate, to preserve wall time these badges are only updated every 12 images. Entertainment value and performance motivation are the only function of these badges.

Currently, HerbASAP is provided with a graphical user interface. Due to an interest in exposing specific functionality to command line interfaces in potential future server applications, HerbASAP was written with a modular approach. Core functions are organized into importable classes from separate python modules. Modular organization, in combination with the open-source makes HerbASAP features readily adaptable into other works. The most relevant of these modules, are named: ccRead, scaleRead, eqRead, blurDetect, bcRead, and metaRead.

The ccRead module

The ccRead module is used for CRC detection and is vital for other HerbASAP features, specifically white balance, image orientation, and scale determination. Detection is performed using either a modified Faster R-CNN, or the ColorNet artificial neural network both presented by Ledesma et. al (2020). Both models were trained and tested in tandem with HerbASAP, and as a result are similarly optimized to achieve low wall time on readily accessible equipment (Ledesma et. al, 2020). Relatively large CRCs (24ColorCard-2x3, CameraTrax™, Kodak® Q-13, X-Rite Colorchecker Classic, and the X-Rite Colorchecker Passport) are detected using the Modified Faster R-CNN while “Image Science Associates ColorGauge Nano” (ISA Nano) CRCs are detected using ColorNet. In addition to implementing the detection models, the ccRead module contains high precision CRC cropping and white point detection methods necessary for image specific white balancing.

<u>The scaleRead module</u>

The scaleRead module calculates a pixel-to-mm ratio based on internal CRC color patch dimensions. Consequently, ccRead is a dependency of this module. Multiple methods were explored to automate the determination of a pixel-to-mm scale using the scale bars usually included in herbarium specimen images. In general these methods were either not robust to the diversity of potential scale bars, or too computationally expensive to be feasible for this project. For example, the mode distance between parallel lines was able to quantify the pixels between tick marks on a scale bar but identifying which units those ticks represented was inconsistent. Since in many use cases wall time is already invested to determine the CRC location and there is a limited diversity of CRC types, we chose to determine scale using predefined internal CRC color patch dimensions.

Scale is determined as the mean square root of all qualifying patch pixel areas (measured as px^2) divided by a known internal patch area (measured as mm^2) not exceeding one standard deviation among those qualifying samples. Potentially qualifying patches are identified from within a cropped CRC using openCV's "flood fill" method and a series of well distributed seed points. This method assumes a strong contrast between patches which is not present in the gradient of adjacent grey patches on in the interior on the ISA Nano. To overcome this, seed points for the ISA Nano are selected using a series of 12 staggered points near the border of the cropped CRC. Seed points for all other CRCs are determined using 24 equidistant points across the cropped CRC. Potential patches are omitted if their area measures less than 1/75th or greater than 1/10th of the total cropped CRC area. Potential patches are also omitted if they express an aspect ratio exceeding 2:1. Potential patches with any pixels present along the edge of the cropped CRC are also omitted. These conditions ensure the remaining qualifying patches are relatively square, appropriately proportioned, and not partially cut-off during the CRC crop method. If fewer than 6 patches qualify for measurement then no scale determination is attempted. Whenever possible, the predefined internal patch dimensions were obtained from the CRC manufacturers. Since no documentation was available for the Kodak® Q-13, or either X-Rite CRCs and the manufacturers did not reply to queries, internal patch dimensions for those CRCs were determined by averaging multiple manual measurements. In addition to determining the pixel-to-mm ratio, the scaleRead module also calculates a 95% confidence interval by multiplying 1.96 by the standard deviation of all qualifying pixel-to-mm ratios divided by the square root of the qualifying sample size.

The eqRead module

The eqRead module corrects equipment based image distortions using the lensfunpy library (github.com/letmaik/lensfunpy) which is a wrapper for the C++ library Lensfun (lensfun.github.io) to correct geometric distortions, and chromatic aberrations. Geometric distortions are the result of unequal magnifications across a focal plane causing sensed points to be disproportionately distributed across an image. Chromatic aberrations are the result of various wavelengths being differentially refracted when passing through a lens, causing colors such as reds and blues to separate at the image edges (Marimont and Wandell, 1994). Lensfunpy uses a packaged database of known correction coefficients to generate a correction matrix for a given RAW image array. The correction matrix is generated based on a combination of image variables: resolution, lens model, and camera body. Although the application of these corrections is computationally expensive, some wall time is preserved by retaining the correction matrix across successive images which share the same image variables.

The blurDetect module

The blurDetect module uses the variance of the Laplacian operator to quantify image blurriness (Pech-Pacheco et al., 2000). This method is sensitive to overall image variance, meaning “busier” images with more contours are determined more blurry. To mitigate this, the laplacian variance is normalized over the image variance. This compensation reduces the impact of image “busyness,” yet has been subjectively observed to disproportionately penalize low contour images with relatively mild blur. A warning dialog box is presented to the operator should the normalized laplacian value exceed a threshold value stored in the settings profile. This allows curators to customize the level of blur detection sensitivity.

The bcRead module

The bcRead module decodes data matrices and barcodes which may be present in the specimen image. The decoded values may then be compared to a pattern saved in the settings profile to ensure extraneous, or partial codes are retained. Data matrices are decoded using the Natural History Museum of London's python library pylibdmtx (Hudson et al., 2015) which wraps the C library libdmtx (github.com/dmtx). Barcodes are decoded using the PyZbar library (Hudson et al., 2015), also maintained by the Natural History Museum of London. PyZbar is a python wrapper which exposes the C library "Zbar" (zbar.sourceforge.net). Zbar decodes barcodes by line scanning i.e., evaluating pixels of an image both row-wise, and column-wise to determine if they correspond to a barcode value. There are two limitations to this line scanning method. The primary limitation is its inability to decode barcodes which are sufficiently skewed from horizontal or vertical orientations. One work around to this problem is to perform a series of rotations on the image until a barcode is detected but this incurs significant wall time. The second limitation to line scanning is the wall time associated with evaluating each pixel twice, once horizontally, and again vertically. Compounding these limitations is the tendency for these barcodes to be small relative to the high resolution specimen image (e.g., 3,840px by 5,760px). To overcome these limitations, a novel vector assisted region proposals (VARP) method was developed.

The VARP method is rotationally invariant and was observed to be faster, and more accurate than traditional methods at decoding barcodes in herbarium specimen images. Single dimension barcodes (e.g., Code 39, and Code 128 formats), which are common in natural history collections, are simply a series of black rectangles with varying widths. VARP uses OpenCV to detect rectangles within the image. This process is not rotationally sensitive and simultaneously detects the rectangles within barcodes oriented at any angle. Next, a vector is calculated which intersects the longest pair of each rectangles' parallel lines. The vectors are then extended in both directions by 1/6th of the image's smallest dimension. Post extension, it is assumed some of these vectors represent coordinates which intersect a complete barcode value (Fig. 8). Pixel values are then sampled from the image along each vector, producing multiple series of single pixel values. These series are then horizontally concatenated into a matrix, forming a vertically organized composite image of all pixels falling along the vector coordinates (Fig. Error: Reference source not found). The size of this composite image varies depending on the quantity of vectors identified, yet can be expected to contain significantly fewer pixels than the original. This lower resolution makes the decoding process more efficient, for example the composite image presented in Fig. Error: Reference source not found is 209px by 1,272px while the source image from which it was extracted was 3,840px by 5,760px. Additionally, since pixel values from each vector are vertically oriented there is no longer meaningful information on the horizontal axes, rendering row-wise line scanning unnecessary. To take advantage of this organization the PyZbar library was modified to enable selectively decoding along a singular axes. This modified version of PyZbar is bundled with HerbASAP, and available through Github (github.com/CapPow/pyzbar).

To evaluate the VARP method, a test dataset was generated containing 500 herbarium specimen images. Images were acquired by querying iDigBio for *Plantae* records and then using the Pandas library (McKinney, 2010) to reduce query results to a maximum of 25 records from each unique collection code producing 7,828 specimen records from 166 unique collection codes. From those records, image containing URLS were randomly selected and accessed in python until 500 unique images were retrieved. These test images were not visually inspected, so it is unknown if 100% of the test dataset contains decodable barcodes. Three barcode decoding methods were tested on a grayscale conversion of each image in the dataset. The first method tested, paired PyZbar with a custom rotation algorithm previously developed and deployed by the primary author in the digitization of Tennessee's Herbaria (github.com/CapPow/bcAudit). This method decoded barcodes from 72.5% of the test images at an average rate of 1.92 seconds per image. The second method tested was from the "Gouda" library which is included with The Natural History Museum of London's program "Inselect" (Hudson et al., 2015). Inselect was developed to automate cropping individual specimen images from whole-drawer type images typical for some natural history collections (e.g., entomology collections). Gouda combines PyZbar with a series of optimization strategies, from which the "resize" strategy was selected as it was found most effective on the test dataset. The third method tested was the VARP method described here, which decoded barcodes from 94.0% of the test images at an average rate of 0.59 seconds per image.

The metaRead module

The metaRead module reads metadata from source images, appends additional details, then writes the refined metadata to HerbASAP's derivatives. This module is used to embed into image derivatives information which future researchers may find useful. The metadata is formatted in exchangeable image file format (EXIF) (JEITA, 2002) using the python library Piexif (github.com/hMatoba/Piexif). EXIF data stored in RAW images varies depending on the camera manufacturer, but often includes information such as: capture date and equipment details such as camera, and lens models. RAW image EXIF data may also include capture conditions such as: aperture, exposure time, focal length, ISO speed, and shutter speed. All EXIF data read from RAW images is included in the derivatives except for those stored as "Maker notes" (EXIF code 37500). Maker notes are stored in proprietary formats with unequal byte lengths which caused inconsistent performance when writing to derivatives.

Raw EXIF data is updated with additional details then embedded into image derivatives. HerbASAP's name and version number are stored under the "Software" EXIF tag (EXIF code 305). The collection's name is stored under the "Artist" EXIF tag (EXIF code 315). Image copyright information is stored in the "Copyright" EXIF tag (EXIF code 33432). Additional details which we considered relevant yet do not fit into specific EXIF fields are encoded as a JSON string, and added to the "Image Description" EXIF tag (EXIF code 270). Some of those additional fields (and the keys under which they are stored) are: barcode/datamatrix values ("barcodeValues") , the CRC quadrant ("ccQuadrant"), CRC bounding box coordinates ("ccLocation"), RGB Values used for white balance ("avgWhiteRGB"), pixels per mm ratio ("pixelsPerMM"), confidence interval of that ratio ("pixelsPerMMConfidence"), settings profile name ("settingProfile"), collection name ("collectionName"), collection URL ("collectionURL"), contact name ("contactName"), contact E-mail ("contactEmail"), and, if present, the imaging operator's name ("imagedBy").

HerbASAP evaluation

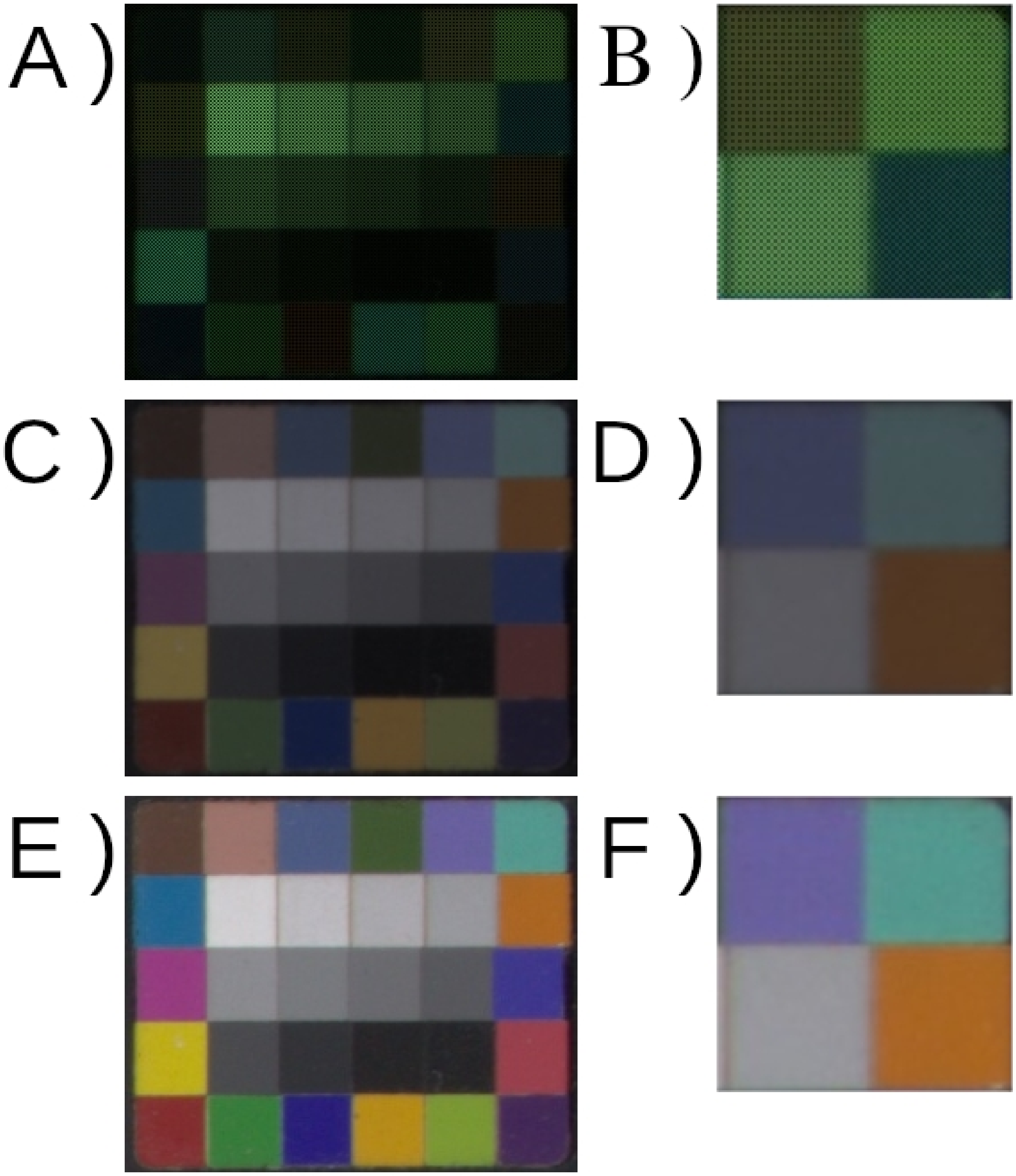
HerbASAP was evaluated against the initial development goal of processing images in under 15 seconds on affordable equipment by timing the runtime for processing a test set of RAW images on affordable, available equipment. The test set of 1,000 specimen images was compiled by randomly selecting 250 sequentially captured RAW images from among four collections which participated in the study explained in Part II. This test set is intended to represent 4 distinct imaging sessions at different institutions using similar equipment. All images had a resolution of either 3,840px by 5,760px or the inverse orientation of 5,760px by 3,840px. A 2014 Dell Latitude E7440 with an Intel i7-4600U processor was chosen for this test due to its availability to researchers and affordability on the second hand market. While processing images, Canon's EOS remote software was open and displaying a live view of an object in motion. This precaution was taken to simulate the additional processor load related to the live shooting software used while imaging. For testing, a settings profile was created which enabled all post-processing features and stored JPEG derivatives. The blur detection settings and barcode pattern matching settings were set to be excessively forgiving in order to omit user input times from the evaluation. These settings ensured all optional procedures were performed yet warnings which they may produce were suppressed (e.g., “a blurry image warning”). The average runtime for all 1,000 images was 11.07 seconds per image or 5.421 specimens per minute. From those images eight (0.8%) halted after phase one due to failure to locate the CRC, in normal operation, these would have required operator interaction. All 992 produced derivatives were properly oriented and their barcodes were successfully decoded. To evaluate performance consistency a pooled standard deviation (s_p) was calculated from numeric metadata embedded by HerbASAP in the 992 correctly produced image derivatives. The pools used were based on the four imaging sessions the test set simulated (i.e., one session per collection). For pixel-to-mm scale, s_p was 0.738 pixels, and the associated confidence intervals had an s_p of 0.110. The average red, green, and blue (RGB) values used for white balance were evaluated individually where appropriate numbers range from 0-255. The red s_p was 7.210, green s_p was 7.443, and the blue s_p was 8.108.

CONCLUSION

HerbASAP uses parallelized processing and artificial intelligences to enable real time specimen image post-processing concurrent with the imaging task. In addition to the typical post-processing procedures applied to herbarium specimen images (e.g., white balance, orientation correction), HerbASAP also includes quality controls, efficiency tools, and supplemental image analysis while maintaining acceptable run times. Using the methods described here, HerbASAP processes high resolution (3,840px by 5,760px) herbarium images at an average rate of 11.07 seconds per image using affordable equipment.

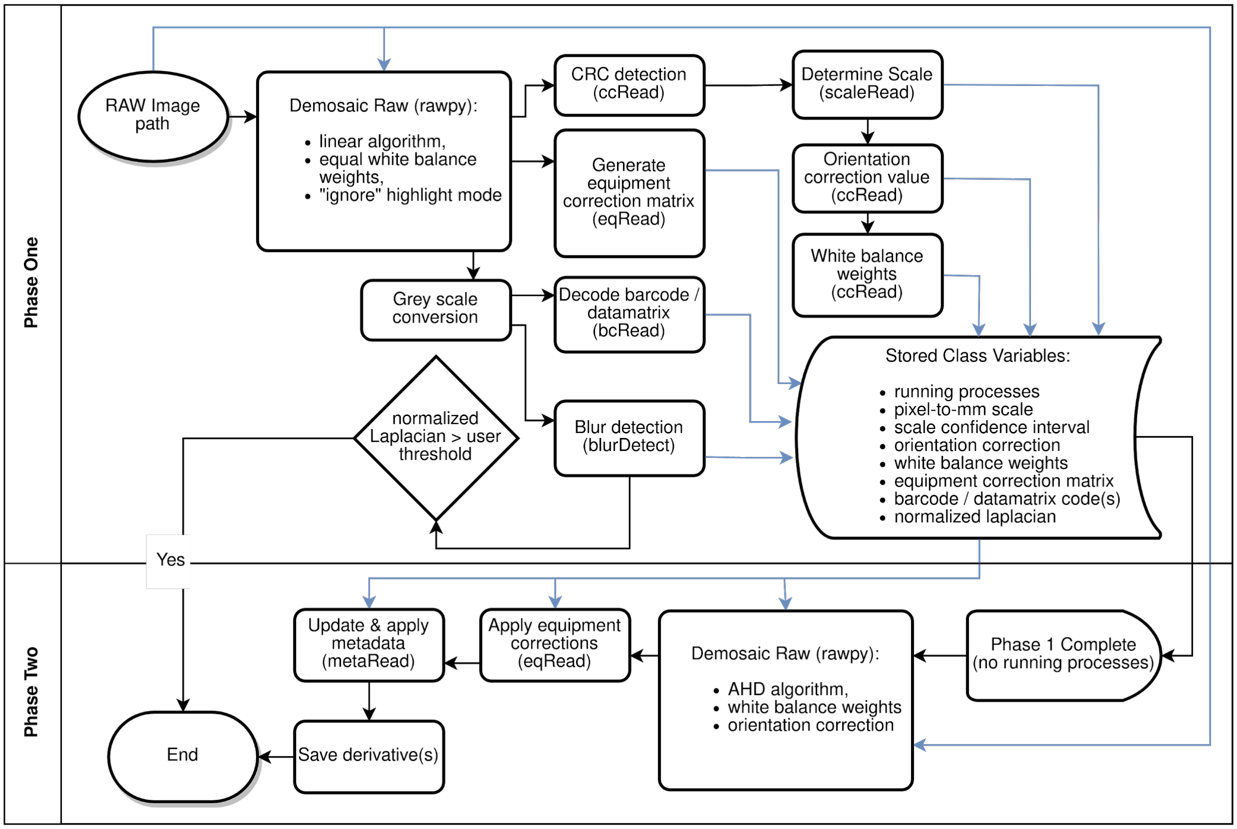
Although, it was our intent for HerbASAP to be fully automatic quality control warnings, such as those from blurry images or non-conforming barcode values periodically require operator attention making it a semi-automated solution. Undoubtedly these prompts will to some degree slow the operator. It is our hope that the overall efficiencies and quality improvements afforded by real-time post-processing will overcome the delays associated with an occasional dialog box. One challenge which requires additional improvement is the specificity of the appropriate partition size. While testing HerbASAP it was observed that reliability of results is strongly influenced by the precision of the partition size stored in the settings profile. In the future, we would like to reevaluate the method used to determine this value. Additionally, we would like to test HerbASAP on a wider array of equipment. HerbASAP is capable of processing RAW images from multiple DSLR cameras, correcting distortions from hundreds of lenses, and identifying all CRC types we are aware of being used in natural history digitization. However, limited access to equipment has prohibited rigorously testing speed and reliability across the many possible combinations of these variables. We hope engagement from the collections community can assist us in this effort and have released a beta version of the software for open testing.

HerbASAP is distributed as an open-source project, available for Linux, Mac OS, and Windows. While HerbASAP can be run as a python script, precompiled binaries are available for Mac OS and Windows at version releases (github.com/CapPow/HerbASAP). The accessibility of HerbASAP's source code is an invitation to the community to adapt, innovate, or incorporate portions of it into other works. We welcome feature requests, bug reports, and suggestions. When possible, we suggest such feedback be in the form of pull requests made on HerbASAP's Github repository.



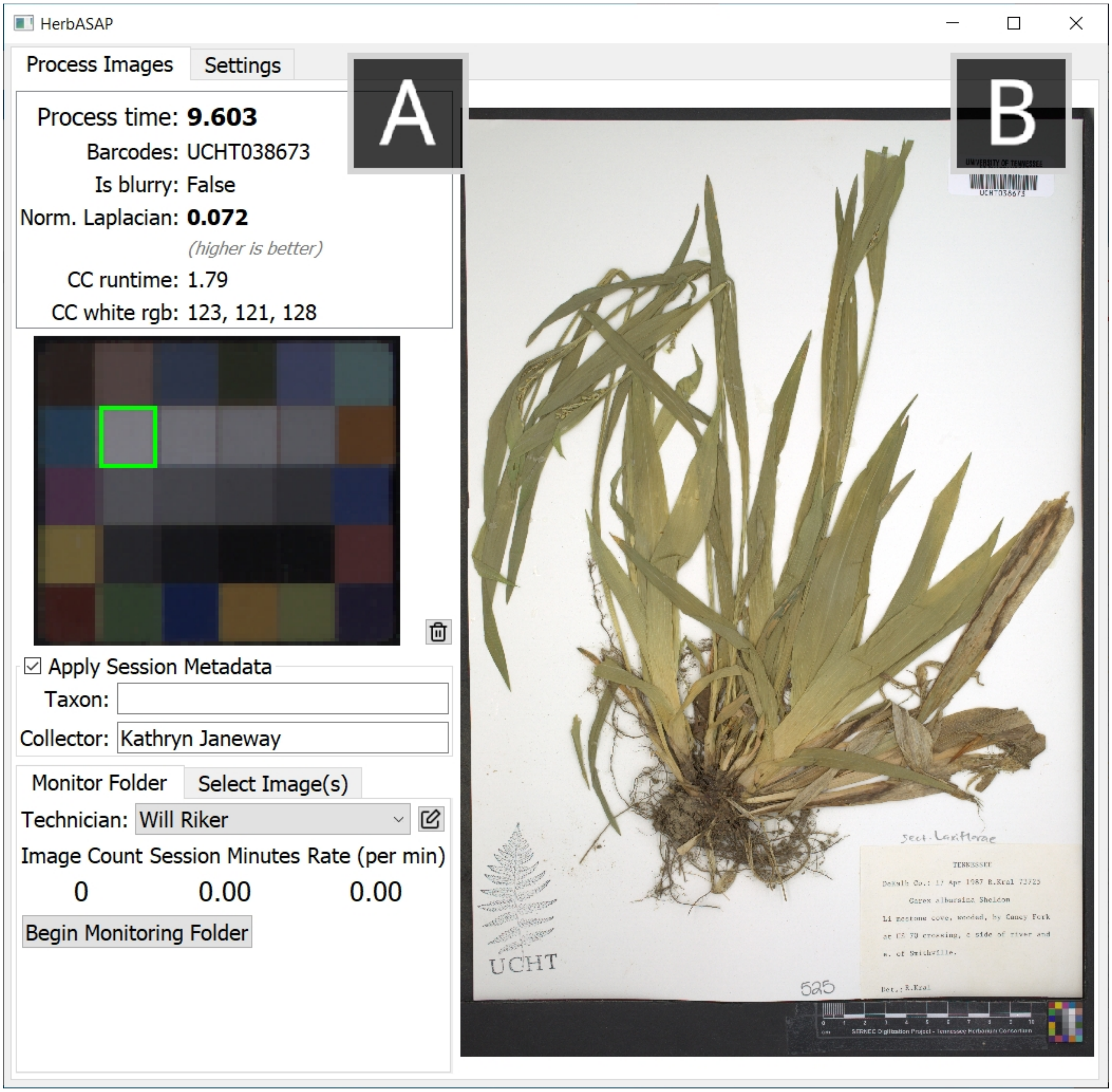
A color reference chart from a herbarium specimen image at varying states of processing. Subfig “A” shows a RAW image’s bayer pattern before demosaicing. Subfig “B” is a portion of “A” zoomed to 200%. Subfig “C” has been demosaiced using a linear algorithm and given equal white balance weights. Subfig “D” is a portion of “C” zoomed to 200%. Both “C” and “D” are indicative of demosaic settings used in phase one of HerbASAP image processing. Subfig “E” has been demosaiced using the adaptive homogeneity-directed demosaicing algorithm with white balance weights calculated from subfig “C”. Subfig”F” is a portion of “E” zoomed to 200%. Both “E” and “F” are indicative of demosaic settings used in phase two of image processing in HerbASAP.

Figure 5 A color reference chart at varying states of HerbASAP image processing


Phase One
RAW Image path
Demosaic Raw (rawpy):
• linear algorithm,
• equal white balance weights,
• "ignore" highlight mode
CRC detection (ccRead)
Generate equipment correction matrix (eqRead)
Determine Scale (scaleRead)
Orientation correction value (ccRead)
White balance weights (ccRead)
Grey scale conversion
Decode barcode / datamatrix (bcRead)
Blur detection (blurDetect)
normalized Laplacian > user threshold
Stored Class Variables:
• running processes
• pixel-to-mm scale
• scale confidence interval
• orientation correction
• white balance weights
• equipment correction matrix
• barcode / datamatrix code(s)
• normalized laplacian
Yes
Phase Two
Update & apply metadata (metaRead)
Apply equipment corrections (eqRead)
Demosaic Raw (rawpy):
• AHD algorithm,
• white balance weights
• orientation correction
Phase 1 Complete (no running processes)
End
Save derivative(s)

An overview of the operation order, and information flow executed in HerbASAP when processing an image. Black arrows represent the order operations are initialized. Blue arrows represent the flow of information. Phase one focuses on asynchronous information gathering, and begins by demosaicing the source image using parameters which emphasise speed and minimal pixel manipulation. Phase two uses what was learned during phase one to produce derivative images suitable for distribution.

Figure 6 HerbASAP image processing flowchart



Process time: **9.603**

Barcodes: UCHT038673

Is blurry: False

Norm. Laplacian: **0.072**

CC runtime: 1.79

CC white rgb: 123, 121, 128

☑ Apply Session Metadata

Taxon:

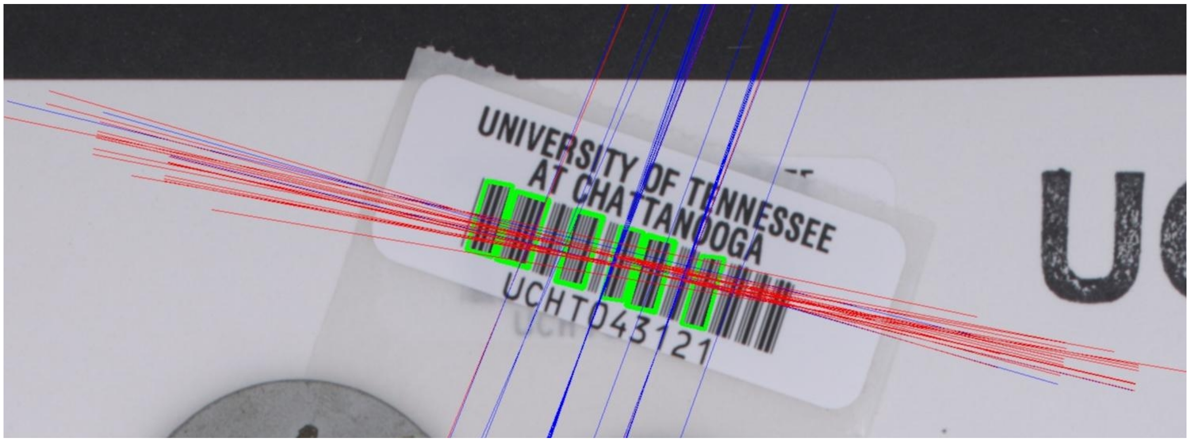
Collector: Kathryn Janeway

Technician: Will Riker

Image Count	Session Minutes	Rate (per min)
0	0.00	0.00

The HerbASAP user interface. Subfig ”A” contains an information pane, color reference chart preview, session statistics, as well as the image processing initiation options. Subfig “B” displays a preview quality version of the image following the final post-processing steps

Figure 7 The HerbASAP user interface



A visualization of the rectangles, and vectors identified through the vectorized barcode decoding (VARP) method.

Figure 8 Vector assisted region proposal barcode decoding composite image

REFERENCES

Barber, A., D. Lafferty, and L. R. Landrum. 2013. The SALIX method: a semi-automated workflow for herbarium specimen digitization. *Taxon* 62: 581–590.

Bradski, G. 2000. The OpenCV library. *Dr Dobb's Journal* 25: 120–125.

Carranza-Rojas, J., H. Goeau, P. Bonnet, E. Mata-Montero, and A. Joly. 2017. Going deeper in the automated identification of Herbarium specimens. *BMC Evolutionary Biology* 17: 181.

Gehan, M. A., N. Fahlgren, A. Abbasi, J. C. Berry, S. T. Callen, L. Chavez, A. N. Doust, et al. 2017. PlantCV v2: Image analysis software for high-throughput plant phenotyping. *PeerJ* 5: e4088.

Hirakawa, K., and T. W. Parks. 2005. Adaptive homogeneity-directed demosaicing algorithm. *IEEE Transactions on Image Processing* 14: 360–369.

Hudson, L. N., V. Blagoderov, A. Heaton, P. Holtzhausen, L. Livermore, B. W. Price, S. van der Walt, and V. S. Smith. 2015. Inselect: automating the digitization of natural history collections. *PLOS ONE* 10: e0143402.

JEITA. 2002. Exchangeable image file format for digital still cameras: Exif Version 2.2.

Ledesma, D. A., Powell, C. A., Shaw, J., & Qin, H. (2020). Enabling automated herbarium sheet image post-processing using neural network models for color reference chart detection. Applications in Plant Sciences, e11331.

Lorieul, T., K. D. Pearson, E. R. Ellwood, H. Goëau, J.-F. Molino, P. W. Sweeney, J. M. Yost, et al. 2019. Toward a large-scale and deep phenological stage annotation of herbarium specimens: case studies from temperate, tropical, and equatorial floras. *Applications in Plant Sciences* 7: e01233.

Marimont, D. H., and B. A. Wandell. 1994. Matching color images: the effects of axial chromatic aberration. *Journal of the Optical Society of America A* 11: 3113.

McKinney, W. 2010. Data structures for statistical computing in Python. 51–56.

Nelson, G., D. Paul, G. Riccardi, and A. Mast. 2012. Five task clusters that enable efficient and effective digitization of biological collections. *ZooKeys* 209: 19–45.

Nelson, G., P. Sweeney, L. E. Wallace, R. K. Rabeler, D. Allard, H. Brown, J. R. Carter, et al. 2015. Digitization workflows for flat sheets and packets of plants, algae, and fungi. *Applications in Plant Sciences* 3: 1500065.

Pech-Pacheco, J. L., G. Cristobal, J. Chamorro-Martinez, and J. Fernandez-Valdivia. 2000. Diatom autofocusing in brightfield microscopy: a comparative study. Proceedings 15th International Conference on Pattern Recognition. ICPR-2000, 314–317 vol.3.

Stamatopoulos, C., C. S. Fraser, and S. Cronk. 2012. Accuracy aspects of utilizing raw imagery in photogrammetric measurement. ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 387–392. Copernicus GmbH.

Sweeney, P. W., B. Starly, P. J. Morris, Y. Xu, A. Jones, S. Radhakrishnan, C. J. Grassa, and C. C. Davis. 2018. Large–scale digitization of herbarium specimens: Development and usage of an automated, high–throughput conveyor system. *TAXON* 67: 165–178.

Tulig, M., N. Tarnowsky, M. Bevans, Anthony Kirchgessner, and B. M. Thiers. 2012. Increasing the efficiency of digitization workflows for herbarium specimens. *ZooKeys*: 103–113.

Unger, J., D. Merhof, and S. Renner. 2016. Computer vision applied to herbarium specimens of German trees: testing the future utility of the millions of herbarium specimen images for automated identification. *BMC Evolutionary Biology* 16: 248.

van der Walt, S., S. C. Colbert, and G. Varoquaux. 2011. The NumPy Array: A structure for efficient numerical computation. *Computing in Science Engineering* 13: 22–30.

Weaver, W. N., J. Ng, and R. G. Laport. 2018. LeafMachine: using machine learning to automate phenotypic trait extraction from herbarium vouchers. 2018 ESA Annual Meeting (August 5--10), ESA.

Wilf, P., S. Zhang, S. Chikkerur, S. A. Little, S. L. Wing, and T. Serre. 2016. Computer vision cracks the leaf code. *Proceedings of the National Academy of Sciences* 113: 3305–3310.

wiredfool, Alex Clark, Hugo, Andrew Murray, Alexander Karpinsky, Christoph Gohlke, Brian Crowell, et al. 2016. Pillow: 3.1.0. Zenodo.

Younis, S., C. Weiland, R. Hoehndorf, S. Dressler, T. Hickler, B. Seeger, and M. Schmidt. 2018. Taxon and trait recognition from digitized herbarium specimens using deep convolutional neural networks. *Botany Letters* 165: 377–383.

PART IV: A born-digital field-to-database solution for collections-based research using collNotes and collBook

FORWARD

Part IV was previously published with slight differences in formatting by the same title in Applications in Plant Sciences by Caleb Powell, Jacob Motley, Hong Qin, and Joey Shaw (Powell et al., 2019). This work would not have been possible without the hard work of multiple collaborators. The term “we” in this part refers to my co-authors and myself. My primary contributions to this paper include (1) project conceptualization coordination and management, (2) the majority of the source code included in collBook, (3) organization of testers, and (4) most of the writing.

INTRODUCTION

Biologists conducting field research, such as floristic studies, accession thousands of specimens into natural history collections. Many of these specimens’ digital records are now becoming available through online portals such as iDigbio (iDigBio, 2019), Global Biodiversity Information Facility (GBIF) (GBIF: The Global Biodiversity Information Facility, 2018), Symbiota (Gries et al., 2014), and regional consortia (e.g. SERNEC) (SERNEC, 2019). One major challenge in digitizing these specimens is the accurate transcription of their label data. Citizen science platforms, such as Notes from Nature (Hill et al., 2012) have been helping to overcome this challenge. In 2018 the platform generated just over 438,000 classifications which include identifying phenology, or transcribing label data (Zooniverse, 2019). To account for human error, Notes from Nature requires classifications be gathered in triplicates, bringing the platform’s total 2018 transcriptions to just over 146,000 records. Citizen science initiatives are proving to be a feasible option for transcribing the backlog of historic records, but they are not keeping pace with the rate at which new accessions are added. Among the *Plantae* records available on iDigbio, 77% have date of collection data and of those there is an average rate of 348,000 specimens collected per year (2006 - 2015). Attention to this issue has highlighted the need for “born digital” records, i.e., field data that are initially gathered in digital formats and simultaneously ready for online data portals as well as printing labels (Deborah Paul et al., 2015; James et al., 2018).

Here we introduce the public releases of collNotes and collBook, two open-source programs that, when combined, are a field-to-database solution for collections based research. Together these programs were developed with the goal of initially collecting biological specimen data in a digital format which would not contribute to the backlog records in need of transcription. To achieve this, collNotes, a mobile application, was developed to supplement the traditional field journal. A companion desktop application, collBook, enables users to refine field notes and produce a comma separated value (CSV) file formatted in the Darwin Core (DwC) (Wieczorek et al., 2012) data standard. The DwC is used by many biodiversity data portals such as those built on the Symbiota framework. Adherence to this data standard makes collBook’s output directly importable by any DwC capable portal or collection management system.

The success of migrating collections to a born digital workflow will depend on the adoption of these new methods by field biologists. Thus, we prioritized user experience and efficiency in the field. One time-intensive and generally tedious step in the traditional collection process is label preparation, requiring the collector to organize and digitally transcribe field notes. To encourage adoption, a PDF file containing formatted, ready to print labels is the second output of this solution. Both collNotes, and collBook are open-source projects, released under GNU General Public License v3.0. It is our hope that these programs will improve the efficiency, accuracy, and accessibility of collections based research.

METHODS AND RESULTS

collNotes development

A mobile application, collNotes, was developed using Microsoft's Xamarin development kit, and is available on Android and iOS devices. collNotes was developed to supplement a traditional field journal. Although it is a mobile application, collNotes does not require cellular service to record field notes. It was designed with a minimalistic interface, prioritizing time saving features, especially where location data are concerned. The locality entered by the user is expected to be limited to the highest resolution portion of a locality string (e.g., "50m northeast of the Illinois Monument."). There are no state, county, or municipality entry fields in collNotes. These location data are inferred later in collBook based on global positioning system (GPS) coordinates captured in the field, by collNotes. Most entry fields in collNotes are optional, and some, such as eventDate and primary collector, can be automatically populated. In the case of "reproductive condition," where the DwC recommends controlled vocabulary, a list of terms is provided. Data from collNotes are stored in the mobile device's local storage as a SQLite database file. Exporting records produces a UTF-8 encoded CSV file, organized (with few exceptions) under DwC terms. This resulting output may then be refined into labels and database ready records using collBook. Features included in collNotes are: structured data, field number generation, and coordinate capture.

Structured data

Records from a collection event often contain redundant information. To avoid repetitive manual entries, we designed a hierarchy of data categories (i.e., classes) similar to DwC classes. In collNotes, we used three classes: "trip", "site", and "specimen." These classes identify which data are appropriate to duplicate across records. For example, a collection trip for a project named "Flora of Risa," might be associated with or multiple sites, all of which inherit "Flora of Risa" as the project name.

The classes, and the data fields they are associated with are listed in Table 3. One advantage of this kind of structured data can be illustrated if numerous voucher specimens are collected from a single location. In this scenario, locality and habitat information can be entered once and propagated to pertinent records. The geographic range of a single site is left to the researcher's discretion. However, since localities, and coordinates are inherited by the site class, and mobile device GPS is generally accurate to about 20m (Tomaštík et al., 2017), a site range between 5 to 30 meters is recommended.

Field number generation

When creating a new site in collNotes, a site number is automatically generated which is used by collNotes and collBook to link associated specimen records. When collections are made indiscriminately, i.e., multiple taxa from a single site are placed in the same container, the site number should be used to label the container. This allows users the option to forgo documenting specimen specific data in the field, and instead generate it while refining the records in collBook. Similarly, when creating a specimen record in collNotes, a specimen number is generated which is synonymous to a traditional field number. A specimen number is formatted as two values separated by a dash, with the first value being a site number and the second, a unique value for that specimen. The starting number for specimen collection can be user defined to accommodate workers keeping lifetime numbers, although it will prepend a site number. To avoid duplicate specimen numbers, the starting site or specimen numbers can be altered in collNote's settings. For proper specimen container labeling, both site and specimen numbers are prominently displayed when generating a new record of either class (i.e., site or specimen).

Coordinate capture

GPS coordinates are the most useful data the user can capture in collNotes. Coordinates are captured in collNotes using the "SET GPS" button available when creating (or editing) a site level record.

When the user selects this feature, their phone makes a location request using the GeoLocator plugin (Montemagno, 2016/2019). This location request includes the altitude, coordinates, and accuracy in the form of uncertainty in meters (e.g., “20,” meaning “±20 meters”). On screen text notifies the user of a successful location request. When uncertainty is high, successive location requests may improve it, so the font of this notification is color-coded to reflect coordinate uncertainty, less than 20m is green, 21m - 30m is yellow, and an uncertainty over 30m is red.

collBook development

A desktop application, collBook was written in Python 3.7 using the Qt5 framework for the graphical user interface. Qt designer 5.12 was used to design the interface layouts. Multiple custom python modules are present in collBook’s source code. A list of those modules, and a brief description of their function is provided in Table 4. Available for Linux, OS X, and Windows, collBook is designed to use at the same time as specimen identification for refining field notes into database ready files, and specimen labels. Performing data refinements in collBook, as opposed to collNotes, permits web service dependent features without cell service dependency.

Designed to be feature rich, the user interface contains four prominent panes: a label preview, form view, site navigator, and table view (Fig. 9). The label preview (Fig. 9A) presents a dynamically generated label, which is updated as edits are made. The form view is the primary method of editing or adding new records (Fig. 9B). Many of the form view’s fields (e.g., date, latitude, and longitude) impose DwC recommended formatting. The site navigator is used to select which records are to be edited, refined, or exported (Fig. 9C). All edits made to parent classes (i.e., those of a higher class) are automatically propagated to their associated children records (i.e., lower class records). For example, in Fig. 9C, selecting “Site 1” sets the scope of records to be acted upon as all those which were collected at that site. To avoid confusion caused by changing scopes, reminder text was placed in the status bar along the bottom of the interface informing the user of the current selection type (i.e., “All records”, “Site view”, or “Specimen view”). The table view presents spreadsheet style access to all selected records (Fig. 9D). Contrary to the rest of the interface, the table view imposes no formatting, validation, or inheritance, providing a method to override many of the functions discussed above. Data entered using the table view may not always be visible in the form view, yet will be reflected on the label preview and in the exported data. Features included in collBook are: reverse geocoding localities, taxonomic alignments, inferred associated taxa, and creation of customizable labels which can optionally include catalog number barcodes.

Reverse geocoding

In collBook, location data not recorded in collNotes (i.e., “state”, “county”, “municipality”) are inferred from the GPS coordinates, and prepended to the user entered locality string, supplementing the minimal locality data recorded in collNotes. Inference from coordinates is performed using Google’s reverse geocoding web service (Google, 2019) For example, the locality string: “50m northeast of the Illinois Monument.” would become: “US, Tennessee, Hamilton County, Chattanooga, Orchard Knob Reservation, near East 4th Street, 50m North East of the Illinois Monument.” One flaw inherent to this feature is that the user entered locality and the generated preamble may contain redundant terms. While testing these programs, familiarity with this feature when using collNotes was found to reduce the occurrences of such redundant terms. Nevertheless, it must remain the user’s responsibility to verify labels in collBook for accuracy and redundancy.

Taxonomic alignments

When refining records, the status of the taxonomy, and the associated authority, are verified. To accommodate user preference, several sources for these alignments were included. The most recently available version of the Integrated Taxonomic Information System (ITIS) (Interagency Taxonomy Steering Committee, 2007) is bundled with the program, while Catalog of Life (Roskov et al., 2013), and the Taxonomic Name Resolution Service (TNRS) (Boyle et al., 2013) are made available through their web services. So as not to overload web services, a one-second delay is imposed on web service requests, making alignments through ITIS much faster. Since ITIS was packaged with collBook, it is also used to inform autofill suggestions when entering scientific names. TNRS has the capability of performing partial matches, correcting minor spelling discrepancies when verifying taxonomies. In those cases; TNRS returns a score of the match's accuracy, a minimum threshold for this score can be modified at the user's preference. Alignments from these sources are applied based on user defined policies which delegate how recommendations should be made, and if ever they should be automatically applied. While not discussed in detail here, collNotes and collBook are being evaluated for groups beyond *Plantae*. *Fungi*, for example, is currently supported with a locally bundled Mycobank (Robert et al., 2013) taxonomy, and Catalog of Life support.

Inferred associated taxa

An additional benefit to structured data inheritance is the ability to document associations among sibling specimen records. Associated taxa information is frequently overlooked by field researchers yet may be informative for community composition, habitat, or ecosystem studies. At site level, collNotes offers an associated taxa entry field. In collBook, during record refinement, but after taxonomic alignments, the user is optionally presented with a checklist dialog box and may select some, all, or none of the taxa contemporaneously collected at the parent site. Once the taxon list is finalized, those taxa are included as comma separated associated taxa in the pertinent records. Associated taxa are appended following any existing user entries which may have been recorded in the field using collNotes. The determined name of each record is omitted from the inferred associated taxa for that record so that no record names itself as an associated taxon. One potential flaw in this feature occurs when a user alters a determination after the refinement steps, thereby leaving an inappropriate associated taxon in the sibling records of the altered specimen. To avoid this issue, users are encouraged to perform record refinements only after they are confident in initial determinations.

Customizable labels

Label containing PDF files are produced in collBook using the Python library, Reportlab (ReportLab, 2019). There are numerous user settings for label customization, such as font type, base font size, label dimensions, and optional label elements such as: associated taxa, verified by, collection name, and collection logo. Label dimensions determine not only the resulting PDF's size but consequently, space available on each label. It is assumed no label should exceed the label dimensions (i.e., there should be no multi-page labels) and since collBook often produces information rich labels, space availability can become an issue. User preferences, in conjunction with dynamic placement, and sizing aid in reducing this issue. The associated taxa, for example may be omitted or restricted in item length on the label without impacting the electronic record data. By default, some label elements will share a line, but when label width is insufficient those elements may be split into separate lines. The font size of some label elements are scaled relative to the base font size. For example, GPS coordinate size is always the base font size reduced by 20%, while the scientific name is usually increased somewhat. Altering the base font size therefore impacts all fonts, but does not necessarily reflect the actual font size of all elements. Another customization option we've included is the ability to load an image as a background logo, or watermark. This logo may be anchored to set locations, and scaled down in either size, or opacity. For best results, users should select cropped images larger than the target labels' dimensions.

Catalog number barcodes

Assigning catalog numbers, usually by applying a barcode sticker is an additional step of the digitization process. Optionally, in collBook, catalog numbers may be sequentially assigned and included on the labels as barcodes with human readable subtext. These barcodes are generated in the “code 39” format, using Reportlab. The catalog numbers assigned are based on a series of user inputs available in the preferences menu. By providing a prefix (e.g., “UCHT”), a digit count (e.g., “6”), and a starting value (e.g., 12345), catalog numbers are assigned sequentially to each record (e.g., “UCHT012345”, “UCHT012346”,...). This feature avoids the costs of procuring and the time of applying barcode stickers, a significant portion of the digitization process. Nevertheless, this feature should be used with caution. Assigning non-unique catalog numbers, or over provisioning catalog numbers to specimens which are eventually not accessioned into collections is possible. To reduce over provisioning, catalog numbers are only assigned during the final export process; a dummy value is displayed in the preview window until those assignments are made (Fig. 9A). As these concerns do not impede the core function of the software, this feature was cautiously included. A catalog number management system which can overcome these issues remains a priority.

Interoperability with existing alternatives

We are not aware of any other solution for a complete field-to-database workflow, so alternatives were evaluated for collNotes and collBook independently. An alternative to collBook’s label printing feature is Symbiota’s, in-browser label printing option. This provides basic label formatting with some of the same features in collBook, including barcode preparation. Since records on Symbiota may already have catalog numbers, barcode printing is less problematic. Because collNotes and collBook use Symbiota friendly data formats, these two platforms are not mutually exclusive. For example, a user may prepare records with collBook, upload them to Symbiota, and use Symbiota’s inbrowser label services.

There are a few alternatives to collNotes for gathering field notes directly into digital formats. Notably, the android application: ColectoR (Maya-Lastra, 2016) was developed for quick and efficient data capture in the field. ColectoR features taxonomic and location refinements and can be integrated into a Microsoft Excel template for label production. However, ColectoR does not export to a standardized format and requires mobile data service for many of its features. A clever solution for gathering digital field notes was documented in the workflow presented by Heberling and Isaac (Heberling & Isaac, 2018) where the citizen science platform, iNaturalist (iNaturalist, 2019) is used to record field observations. One advantage to Heberling and Isaac's workflow is the association of iNaturalist hosted data with the voucher specimen's record. For example, in this process *in situ* photos captured using the iNaturalist mobile application are documented with the record, and hosted through iNaturalist's servers. Designed for citizen science observations, the iNaturalist mobile application lacks entry fields for many field notes recommended for voucher specimen labels, such as habitat or relative abundance estimates (Bridson & Forman, 1998). As noted by Heberling and Isaac, custom observation fields may be added to records made using the iNaturalist app; however, these additional data must be entered through iNaturalist's web interface, and not in the mobile application. This limitation makes either field web browser access, or the eventual transcription of ancillary field notes necessary.

Preferences and priorities of individuals vary according to research needs. So, while collBook has seamless integration with collNotes we have also included functions in collBook to parse data produced by either ColectoR, or iNaturalist. Additionally, a user may attempt to import into collBook any DwC formatted CSV. The degree of success from such an attempt, however, will vary depending on formatting. None of these alternatives incorporate a hierarchical data structure and so lack a method to link specimens collected from the same site or during the same event. For these reasons, we chose to develop both collNotes and collBook to address the challenge of "born digital" collections based research.

CONCLUSION

Combined, collNotes and collBook provide a solution for field researchers, and natural history collections to transition to a "born digital" process for new accessions. We believe, if adopted, this can mitigate the continued growth of backlogged natural history data which need to be transcribed while improving the efficiency, accuracy, and accessibility of collections based research. The source code for both of these works are available as Github repositories (Motley, 2018/2019; Powell, 2019/2019), under GNU General Public License v3.0 licenses. The mobile application, collNotes may be downloaded for free on the Google Play store as well as the iOS App store. The desktop program, collBook is distributed for free through the Github repository: https://github.com/CapPow/collBook#Installation. The accessibility of these works is an invitation to the community to improve, modify, or incorporate portions of them into other projects. We welcome suggestions, bug reports, and feature requests. When possible, we encourage that feedback to be in the form of pull requests to our Github repositories.

ACKNOWLEDGEMENTS

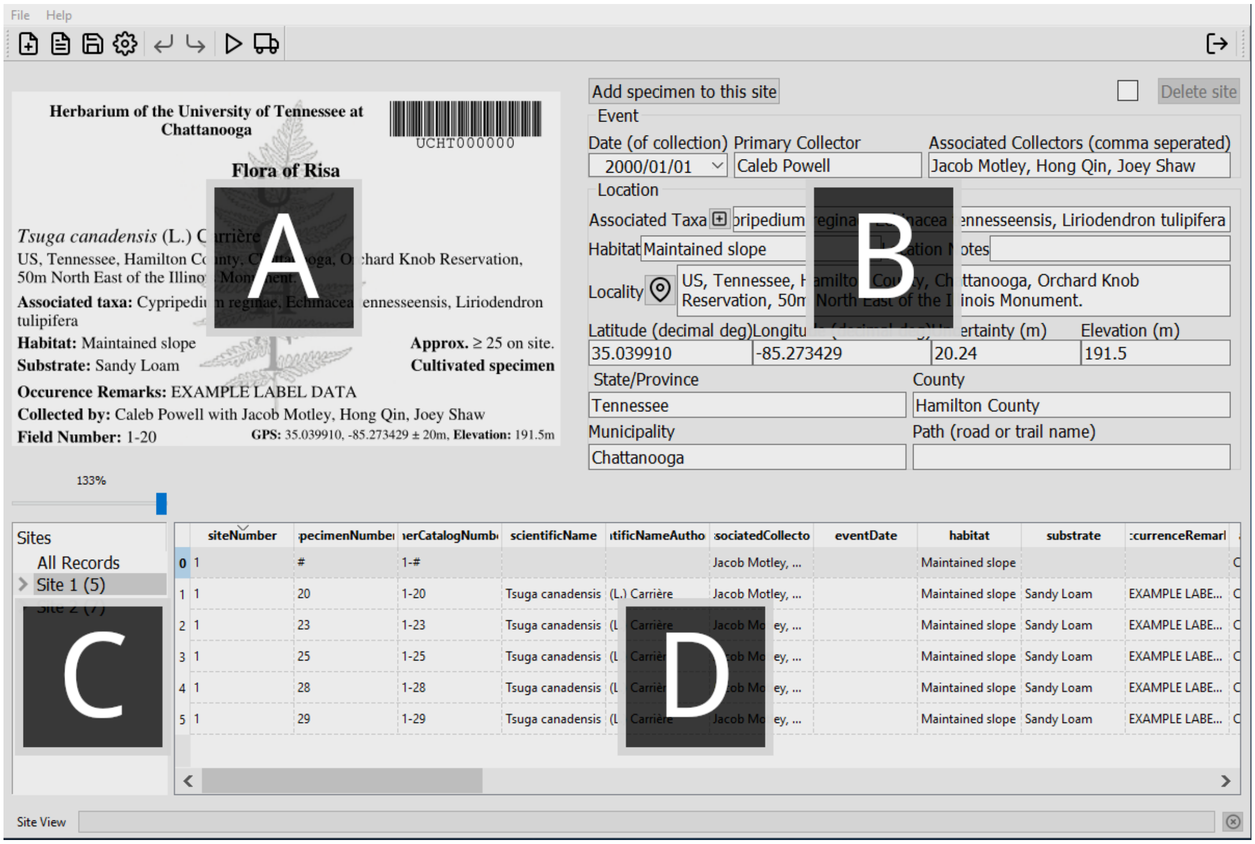
We thank Courtney Alley, Hill Craddock, Alaina Krakowiak, John Shelton and the 2017 - 2019 Field Botany, Mycology, and Plant Taxonomy students at the University of Tennessee at Chattanooga for their patient testing and constructive feedback. Additionally, we would like to thank Github user "Zularizal," for the thoughtful logo design. This work benefited from the critical feedback of 3 anonymous reviewers, we appreciate their time and service. We are also thankful for internal funding from the University of Tennessee at Chattanooga, support from the Department of Biology, Geology, and Environmental Science as well as NSF awards #1761839 and #1410069.

Table 3: The hierarchical classes used in collNotes and collBook and the Darwin Core data fields to which they are associated. [1] Not a Darwin Core field, used by symbiota. [2] Not a Darwin Core field, used by collBook to store road or trail name. [3] The only possible value for "establishmentMeans" is: "cultivated," set using the "cultivated" checkbox in either collNotes or collBook, otherwise the field is left blank.

Trip	**Site**	**Specimen**
additionalCollectors [1]	associatedTaxa	catalogNumber
eventDate	coordinateUncertaintyInMeters	establishmentMeans [3]
Label Project [1]	country	identificationReferences
recordedBy	county	identificationRemarks
samplingEffort	decimalLatitude	individualCount
	decimalLongitude	occurrenceRemarks
	habitat	recordNumber
	locality	reproductiveCondition
	locationNotes	scientificName
	minimumElevationInMeters	scientificNameAuthorship
	municipality	substrate
	path[2]	
	stateProvince	

Table 4: The Python modules written for collBook, and a brief description of their functions.

Module	Description of Function
associatedtaxa.py	A dialog for selecting which associatedTaxa to include for a site.
collBook.py	The "Main App," delegates commands to other modules.
formview.py	Manages the user entry fields in the main screen.
importindexdialog.py	A dialog to assist importing unrecognized data formats.
locality.py	Refines location related fields and calls geocoding API services (no GUI elements).
pandastablemodel.py	Displays the data in a table and handles the data manipulation functions.
pdfviewer.py	Converts pdf objects into images to display preview labels.
printlabels.py	Generates pdf objects, displayed as previews or output as label files (no GUI elements).
progressbar.py	A status bar replacement containing progress bar and "scope of view" notice.
scinameinputdialog.py	A dialog for requesting binomial names after a failed taxonomy check.
settingsdialog.py	A dialog containing the preferences window. Also used to manage stored persistent settings.
taxonomy.py	Verifies the status of binomial names and their authorities (no GUI elements).



Herbarium of the University of Tennessee at Chattanooga

UCHT000000

Flora of Risa

Tsuga canadensis (L.) C rrière

US, Tennessee, Hamilton Co nty, C oga, O chard Knob Reservation, 50m North East of the Illino Mon ent.

Associated taxa: Cypripedi n regi nae, Echinacea ennesseensis, Liriodendron tulipifera

Habitat: Maintained slope **Approx.** ≥ 25 on site.

Substrate: Sandy Loam **Cultivated specimen**

Occurence Remarks: EXAMPLE LABEL DATA

Collected by: Caleb Powell with Jacob Motley, Hong Qin, Joey Shaw

Field Number: 1-20 **GPS:** 35.039910, -85.273429 ± 20m, **Elevation:** 191.5m

Event

Date (of collection): 2000/01/01

Primary Collector: Caleb Powell

Associated Collectors (comma seperated): Jacob Motley, Hong Qin, Joey Shaw

Location

Associated Taxa: pripedium regina acea ennesseensis, Liriodendron tulipifera

Habitat: Maintained slope

ation otes:

Locality: US, Tennessee, amilto Co y, Ch ttanooga, Orchard Knob Reservation, 50m North East of the I inois Monument.

Latitude (decimal deg): 35.039910

Longitu: -85.273429

ertainty (m): 20.24

Elevation (m): 191.5

State/Province: Tennessee

County: Hamilton County

Municipality: Chattanooga

Path (road or trail name):

Sites

All Records

Site 1 (5)

	siteNumber	pecimenNumbe	erCatalogNumb	scientificName	tificNameAutho	sociatedCollecto	eventDate	habitat	substrate	:currenceRemarl
0	1	#	1-#			Jacob Motley, ...		Maintained slope		
1	1	20	1-20	Tsuga canadensis	(L.) Carrière	Jacob Motley, ...		Maintained slope	Sandy Loam	EXAMPLE LABE...
2	1	23	1-23	Tsuga canadensis	(L Carrière	Jacob Mo ey, ...		Maintained slope	Sandy Loam	EXAMPLE LABE...
3	1	25	1-25	Tsuga canadensis	(L Carrièr	ob Mo ey, ...		Maintained slope	Sandy Loam	EXAMPLE LABE...
4	1	28	1-28	Tsuga canadensis	(L Carrièr	ob Mo ey, ...		Maintained slope	Sandy Loam	EXAMPLE LABE...
5	1	29	1-29	Tsuga canadensis	(L Carrière	Jacob Mo ey, ...		Maintained slope	Sandy Loam	EXAMPLE LABE...

The collBook user interface. The section labeled A is the label preview, which presents an image of the label to be produced for a selected record. The section labeled B is the form view, where data pertinent to the selected class (e.g., "Site 1") may be edited. The section labeled C is the site navigator, which is used to select record(s) for editing or refining. The section labeled D is the table view, which provides an overview of the selected record(s), and allows unrestricted edits to the data.

Figure 9 The collBook user interface

REFERENCES

Boyle, B., N. Hopkins, Z. Lu, J. A. R. Garay, D. Mozzherin, T. Rees, N. Matasci, et al. 2013. The taxonomic name resolution service: an online tool for automated standardization of plant names. *BMC bioinformatics* 14: 16.

Bridson, D. M., and L. Forman. 1998. Herbarium Handbook. Royal Botanic Gardens, Kew.

Deborah Paul, Katja Seltmann, François Michonneau, Derek Masaki, Pam Soltis, Shari Ellis, and Kevin Love. 2015. iDigBio wiki- field to database. *iDigBio Wiki*. Website https://www.idigbio.org/wiki/index.php/Field_to_Database [accessed 5 March 2019].

GBIF: The Global Biodiversity Information Facility. 2018. What is GBIF? Website https://www.gbif.org/what-is-gbif [accessed 3 May 2019].

Google. 2019. Google Geocoding API. *Google Developers*. Website https://developers.google.com/maps/documentation/geocoding [accessed 8 March 2019].

Gries, C., E. E. Gilbert, and N. M. Franz. 2014. Symbiota – A virtual platform for creating voucher-based biodiversity information communities. *Biodiversity Data Journal*.

Heberling, J. M., and B. L. Isaac. 2018. iNaturalist as a tool to expand the research value of museum specimens. *Applications in Plant Sciences* 6: e01193.

Hill, A., R. Guralnick, A. Smith, A. Sallans, null Rosemary Gillespie, M. Denslow, J. Gross, et al. 2012. The notes from nature tool for unlocking biodiversity records from museum records through citizen science. *ZooKeys*: 219–233.

iDigBio. 2019. iDigBio Portal. Website https://www.idigbio.org/portal [accessed 12 July 2019].

iNaturalist. 2019. iNaturalist.org. *iNaturalist.org*. Website https://www.inaturalist.org/ [accessed 8 March 2019].

Interagency Taxonomy Steering Committee. 2007. Integrated Taxonomic Information System. Website https://www.itis.gov/ [accessed 8 March 2019].

James, S. A., P. S. Soltis, L. Belbin, A. D. Chapman, G. Nelson, D. L. Paul, and M. Collins. 2018. Herbarium data: Global biodiversity and societal botanical needs for novel research. *Applications in Plant Sciences* 6: e1024.

Maya-Lastra, C. A. 2016. ColectoR, a digital field notebook for voucher specimen collection for smartphones. *Applications in Plant Sciences* 4: 1600035.

Montemagno, J. 2019. Geolocation plugin for Xamarin and Windows. Contribute to jamesmontemagno/GeolocatorPlugin development by creating an account on GitHub.

Motley, J. 2019. collNotes: Xamarin Forms (Android & iOS) app for field biologists.

Powell, C. 2019. collBook: Refine biological field observations.

ReportLab. 2019. ReportLab. Website https://www.reportlab.com/documentation/ [accessed 14 March 2019].

Robert, V., D. Vu, A. B. Amor, N. van de Wiele, C. Brouwer, B. Jabas, S. Szoke, et al. 2013. MycoBank gearing up for new horizons. IMA Fungus 4: 371–379.

Roskov, Y., T. Kunze, L. Paglinawan, T. Orrell, D. Nicolson, A. Culham, N. Bailly, et al. 2013. Species 2000 & ITIS Catalogue of Life, 2013 Annual Checklist.

SERNEC. 2019. SERNEC. *SERNEC Portal*. Website http://sernecportal.org/portal/ [accessed 5 March 2019].

Tomaštík, J., J. Tomaštík, Š. Saloň, and R. Piroh. 2017. Horizontal accuracy and applicability of smartphone GNSS positioning in forests. Forestry: An International Journal of Forest Research 90: 187–198.

Wieczorek, J., D. Bloom, R. Guralnick, S. Blum, M. Döring, R. Giovanni, T. Robertson, and D. Vieglais. 2012. Darwin Core: An Evolving Community-Developed Biodiversity Data Standard. *PLOS ONE* 7: e29715.

Zooniverse. 2019. Zooniverse. Website https://www.zooniverse.org/projects/zooniverse/notes-from-nature/stats/?classification=month&classificationRange=1%2C971. [accessed 18 February 2019].Boyle, B., N. Hopkins, Z. Lu, J. A.

PART V: Process Adoption, and Dissemination Progress

PROCESS ADOPTION

Natural history collections archive more than just biological specimens, they preserve the collective efforts of those, past and present who have spent their lives documenting and categorizing the diversity of life on Earth. Mobilizing these data gives researchers access to that accumulated expertise, regardless of geographic distance, political barriers, available funding, or institutional prominence. Already many researchers are digitally consulting this expertise in efforts to better inform conservation and land management efforts (Alley et al., 2020; Deason, 2018; Krakowiak & Shaw, 2019; Miller, 2008; Rylander & Shaw, 2019). In support of digitization, this thesis was produced with the goal of enhancing the usability, and accessibility of that biodiversity expertise. For this goal to be realized the tools included need to be communicated to the collections community and iteratively improved upon following that engagement. To conclude, a brief report for each tool is appended concerning the progress of communication and outcomes, desired or realized, of community engagement.

DISSEMINATION PROGRESS

The formulas and tools presented in Part II may be a useful reference during the early planning stages of future digitization projects. In order to make them appropriate for citation in proposals, they are being prepared for publication in an academic journal. The accuracy of labor estimates made using the recommendations in Part II may remain to be seen for some time. However, if there exists extraneous task rate data from similar projects which are unpublished, or which we missed during the literature review, we believe they would serve as a useful comparison.

HerbASAP, and the related submodules discussed in Part III were developed to simplify the imaging task. This work is being presented at a series of conferences, and prepared for publication in an academic journal. As noted in Part III, HerbASAP performed well using a combination of equipment which is common among the Tennessee Herbarium Consortium (THC). We would like to validate support for collections both outside of THC and beyond the kingdom *Plantae*. To do so, we hope interested parties test or review the code which is available on the project's GitHub repository (github.com/CapPow/HerbASAP/releases). We also encourage contributions to that code, particularly any which improve performance, contribute documentation, or expand taxonomic support.

The programs discussed in Part IV (collNotes, and collBook) were produced to provide a proactive means by which future biological specimens might be "born digital" and enter collections with accompanying label data organized into standardized web ready formats. Communication of this work has already begun with a series of presentations and a publication describing the software (Powell et al., 2019). The communities feedback has produced numerous improvements and feature additions to both programs which are expected to be released during the summer of 2020. Among the improvements for collBook are: the inclusion of optional family names on labels, the addition of Tropicos for taxonomic alignments (tropicos.org), and the option to generate barcodes based on randomly generated globally unique identifiers. Similarly, collNote improvements resulting from community feedback include: a high contrast user interface update, and optional google maps integration for GPS point adjustments. We continue to request input from field researchers and invite user interface design, as well as code contributions to the project's GitHub repository (github.com/CapPow/collBook).

REFERENCES

Alley, C., Rylander, E., Dawson, J., Feely, M., Ledesma, D., Parrish, N., Powell, C., Shelton, J., Barger, W., Davidson, P., & Shaw, J. 2020. Saxifraga tridactylites (Saxifragaceae) Naturalized in the Southeastern and Northwestern United States. *Castanea*.

Deason, T. 2018. *Conservation and collection of Castanea dentata germplasm in the South* [Honors, The University of Tennessee at Chattanooga]. https://scholar.utc.edu/honors-theses/146

Krakowiak, A., & Shaw, J. 2019. The vascular flora of Orchard Knob Reservation, Chattanooga, Tennessee. *Castanea*, *84*(2), 161–178. https://doi.org/10.2179/0008-7475.84.2.161

Miller, R. J. 2008. *Herbarium infrastructure development and ecological applications of specimens using geographic information systems* [M.S., The University of Tennessee at Chattanooga]. https://search.proquest.com/docview/304398750/abstract/C69EBB31F7CB4F6BPQ/1

Powell, C., Motley, J., Qin, H., & Shaw, J. 2019. A born-digital field-to-database solution for collections-based research using collNotes and collBook. *Applications in Plant Sciences*, *0*(0), e11284. https://doi.org/10.1002/aps3.11284

Rylander, E., & Shaw, J. 2019. Using digitized herbarium specimens to predict the potential distributions of Tennessee’s historical plant species. *80th Annual Meeting of ASB*. Association of Southeastern Biologists, Memphis, Tennessee.

VITAE

Caleb Powell was born in Red Oak Iowa, the youngest of three to loving parents Randy and Rosemary Powell. Caleb spent his childhood exploring the wooded areas of Bradley county Tennessee, playing fantasy role-playing games, and learning woodcraft in his father’s workshop. At age 15, Caleb sought employment in food-service to support an emerging hobby operating a computer bulletin board system. In 2002, Caleb graduated from North Davidson High School (Welcome, NC) where he participated in electives spanning technology, science, as well as visual and performing arts. After graduation, Caleb managed a video game retail store while pursuing a degree in greenhouse management at Forsyth Tech Community College (Winston-Salem, NC). In 2012, Caleb began studying Environmental Engineering at the University of Tennessee at Chattanooga (UTC) (Chattanooga, TN), during which he served an internship performing on-site evaluations of geotextile applications in Bogotá, Colombia. Caleb also volunteered during this period as a youth camp councilor providing on-site water quality education in Tennessee, and Belize with a Chattanooga based non-profit organization now called “WaterWays.” In 2016, Caleb was introduced to natural history collections and their mobilization by herbarium curator and advisor Dr. Joey Shaw. In 2017, Caleb achieved a Bachelor of Science degree in Environmental Science with a concentration in Biology and a minor in Economics at UTC. Ongoing engagement in the digitization of Tennessee’s herbaria led to Caleb continuing studies at UTC and earning a Master of Science degree in Environmental Science in 2020. In the fall of 2020, Caleb will continue his education, pursuing a PhD in Biology at the Arizona State University’s School of Life Sciences (Tempe, AZ).