

Gallery Composer

Chong Zhu
997962940

Fangzai Hong
998030036

Anh Thu Nguyen
1002332665

FINAL REPORT

Professor Jonathan Rose | ECE 1778 Creative Applications for Mobile Devices | University of Toronto

April 7, 2016

Word Count: 2462

Table of Contents

1. Introduction	1
2. Overall Design	2
2.1. Block Diagram.....	2
2.2. Description	2
3. Statement of Functionality & Screen Shots	3
3.1. Gallery.....	3
3.2. Detailed Photo View	3
3.3. Photo Metadata Form	3
3.4. Favorites	4
3.5. Camera.....	4
3.6. Cloud.....	4
3.7. Settings.....	4
4. Lessons Learned	5
4.1. What did we learn?.....	5
4.1.1. Technical	5
4.1.2. Non-technical.....	5
4.2. What would we do differently?	6
5. Contribution by Group Members	7
5.1. Programmers	7
5.1.1. Chong Zhu.....	7
5.1.2. Fangzai Hong.....	7
5.2. Specialist.....	7
5.2.1. Anh Thu Nguyen.....	7
6. Specialist Context	8
6.1. Current State of Image Collections.....	8
6.2. Contribution to the Field.....	8
7. Future Work	9
8. Source Code	9
9. Availability of Content Online	9

1. Introduction

Scholars of art history primarily study material objects spread throughout different geographical regions and time periods. Because of their inability to constantly be in the museums where those objects are housed, they often must record what they see in the form of photographs and handwritten notes in order to revisit the objects at a later time. The typical art historian eventually transfers her photographs onto a computer and manually sorts them into different folders along with her notes. The process is time-consuming and may lead to errors as the information recorded during fieldwork might be incomplete or inaccurate. As the art historian cannot always return to those objects due to time and distance constraints, she must rely on other resources, if available, to complete her database of primary sources.

Gallery Composer proposes to alleviate this task by helping art historians take photographs more efficiently during their research trips. Building upon available photo apps on the market, we have introduced three innovative features specifically with the art historian in mind. First, photographs are taken in batches. If the many sides of one single object are photographed subsequently, those photos will remain within a same folder to facilitate with their classification. Second, each photograph may be annotated with metadata elements, audio recordings, and text descriptions. In particular, the metadata elements will remain inseparable from its respective photograph: when the latter is opened on a computer, its metadata may be viewed in the file's properties. Third, photographs and their attached annotations may be uploaded onto a cloud storage provider for further consultation and image sharing. Because of these in-built features, the note-taking process is integrated directly into the database as the art historian stands in front of her objects, thus minimizing errors that might occur.

Since personal digital image collections are at the foundation of any art historical research project and because they are difficult to manage, we hope that Gallery Composer can contribute to this aspect of *digitized* art history by making the process easier.

2. Overall Design

2.1. Block Diagram

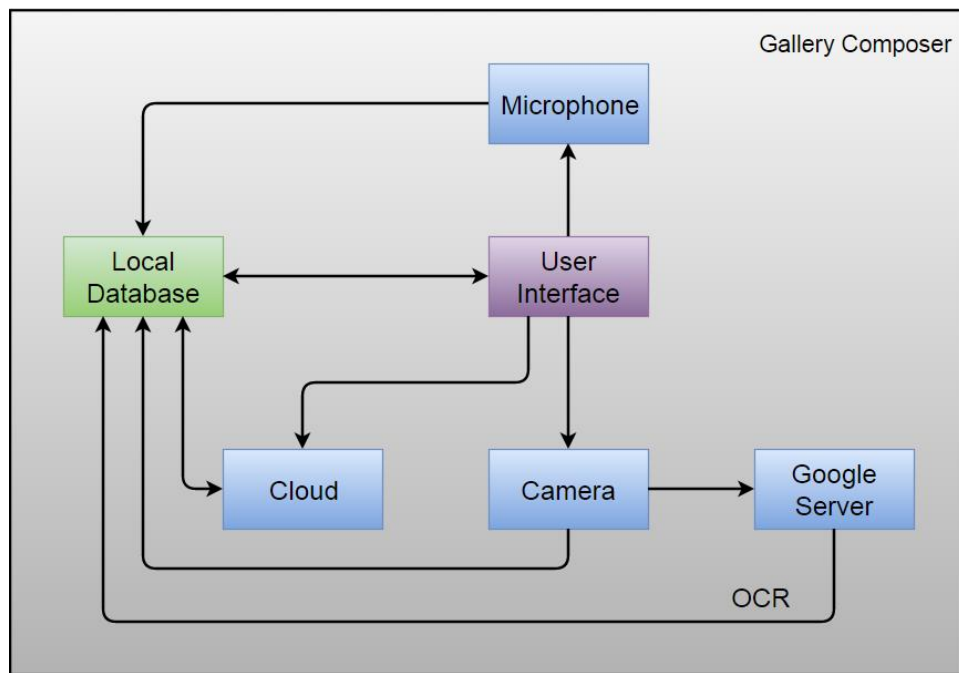


Figure 1. Block Diagram

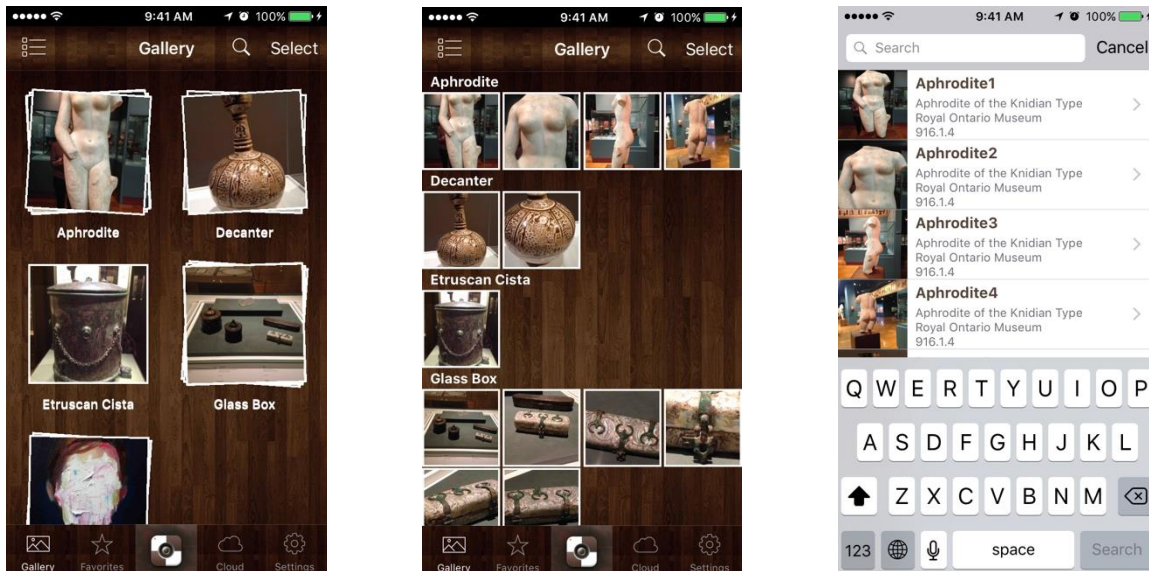
2.2. Description

As shown in Figure 1, Gallery Composer consists of five main parts: a local database, a cloud, a microphone, a camera, a user interface (UI), and the Cloud Vision service from Google. The core function of our app is to take photographs. The camera is activated through the UI according to user input. Data is saved onto the disk with correlation created in the database. With the camera functionality, a user may modify the metadata description of each photograph by first taking pictures of object labels and subsequently sending them to the Cloud Vision server. These label images are then translated into a text entry via Optical Character Recognition (OCR). Moreover, users may record audios as notes by using the microphone, or type in notes directly from the UI using the screen keyboard. All this information is stored and preserved in the metadata section of each photograph. Finally, in order to share their work, users may connect to the cloud and upload or download resources. In our case, we are using Dropbox as our cloud storage provider.

3. Statement of Functionality & Screen Shots

3.1. Gallery

The gallery nicely lays out all the photos stored in the local database as albums or lists. The user is also able to search through the photos via keywords.

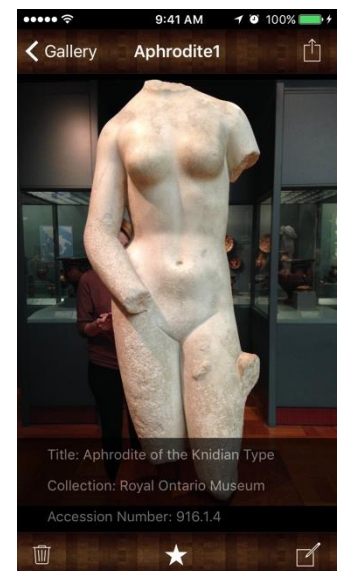


3.2. Detailed Photo View

By clicking into one of the photos from the gallery, the user will be taken to a page showing a detailed view of it. On this page, the user can pinch to zoom into the photo, swipe left/right to go to the previous/next photo. The user is also given the options to star the photo for later reference, delete it from the database, upload it to the cloud as well as edit the metadata associated with the photo.

3.3. Photo Metadata Form

One of the essential features of the app is the ability to annotate photos with persistent metadata. The metadata form enables the user to enter relevant information about the depicted art object such as its title, collection, accession number as well as description. The user can also leverage the OCR feature and audio recording support to suit his/her needs when necessary.

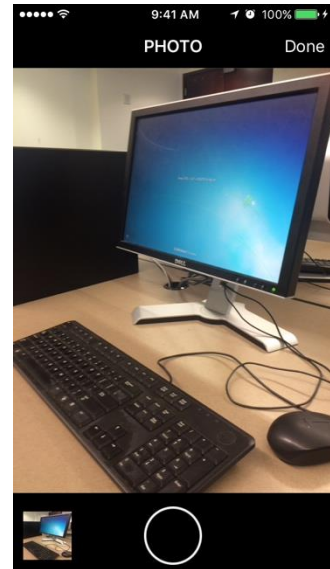


3.4. Favorites

The Favorites tab provides a glance of all the photos that the user has previously starred so that he/she can refer back to them quickly.

3.5. Camera

The app also empowers the user to take a batch of new photos and annotate them on the fly. To do so, the user simply launches the camera and starts shooting, and the photos taken in the same batch will automatically be grouped together. When the user is done, he/she will be presented with a summary page showing all the photos just taken, which serves as a central place for the user to populate each individual photo with metadata.



3.6. Cloud

When the user chooses to connect to his/her Dropbox account from our app, a folder named “Gallery Composer” will be automatically created on Dropbox to manage any photos exported from our app. The user can also download photos from the “Gallery Composer” folder by selecting photos from the Cloud view.

3.7. Settings

The Settings tab provides a few preferences for the user to customize such as changing the app theme.



4. Lessons Learned

4.1. What did we learn?

4.1.1. Technical

Chong Zhu:

I got exposed to the advanced and customized UICollectionViewLayout to give the gallery view a fresh look. I also self-learned how to improve the UI performance (e.g. smoothness of scrolling) when loading UITableView and UICollectionView data. In addition, I learned how to implement a search feature to allow the user to get down to whatever he/she wants quickly. I got to know how to bring OCR (Optical Character Recognition) into our app by incorporating third-party services such as Google Cloud Vision. Lastly, I familiarized myself with SQLite 3.

Fangzai Hong:

I had a better understanding of how multiple MVCs work. By using Dropbox APIs, I learned how to import frameworks from other libraries. I gained better understanding of how to use protocols and delegation method. I also did a lot of research to learn about the life cycle of iOS applications. In order to solve the problem of infinite segue caused by subfolder browsing, I learned how to execute self-instantiation and coincidentally how to pass data to view controllers embedded in a container with segues. This contributes a lot to my audio function development. At last, by writing/reading the data from/to SQLite, I accumulated experience on relational database operations.

4.1.2. Non-technical

Programmers (Fangzai Hong & Chong Zhu)

The most important lesson we learned from this project is the balance between planning and execution. Due to lack of planning, we could not anticipate the unexpected issues that we would run into. On the flip side, the overall development progress could be hindered by excessive planning. Being able to find a perfect balance between planning and execution is in fact a profound engineering problem. The second takeaway from the course is that presenters should deliver the ideas from the audience's perspective instead of from the developer's perspective. Not all the audience understands technical details, presentations should start from a general background in a more intuitive way.

Anh Thu Nguyen:

The most difficult aspect of this project had been to convey my ideas clearly to both my teammates and members of the Art Department. Throughout the many meetings we have had, I was able to develop new ways of describing what I had in mind in more succinct ways. It was also a learning experience to develop a coherent narrative when giving a demo presentation to a general audience.

4.2. What would we do differently?

Programmers (Fangzai Hong & Chong Zhu):

What would we do differently basically lies in how we prioritize the program features to implement. One important feature we left out is enabling the user to upload, delete or edit multiple photos in one shot, because in real life scenarios, art historians could be editing a large amount of photos at the same time. In addition, we could've allowed simultaneous file downloading and uploading in the background. The aforementioned two problems might be solved in an easier way if we opted to adopt the latest version of Dropbox API in Swift instead of the out-dated Objective-C API that we currently used for the cloud feature.

Anh Thu Nguyen:

Concurrently with the development of Gallery Composer, I performed a small-scale ethnographic study on the information behaviour of three art history students in order to understand the ways in which they used photographs and wrote notes when doing fieldwork in the museum. It had been my intention to use the findings of that study to support the conceptualization and development of this application. Unfortunately, the timeline of both these projects did not concord as I obtained my findings later than expected. If I were to rethink this iOS application, I would redesign it so as to include different features of interest to art historians such as the ability to overlay and juxtapose photographs and the possibility to creating presentation slides from within Gallery Composer.

5. Contribution by Group Members

5.1. Programmers

5.1.1. Chong Zhu

I engineered the whole app in a modular fashion so that work items could be distributed independently among programmers. Specifically, I designed and implemented the gallery view, which included the ability to switch between album and list layouts. In addition, I developed the favorites view, which allowed users to refer back to what they starred. I incorporated the search function into the gallery view for quicker access to targeted photos. Furthermore, I revamped the app camera logic by allowing users to take a batch of photos and filling in the metadata information in the end. I also took the responsibility to implement the metadata form, which included input fields for entering title, collection, accession number of a particular art object. I also enabled OCR capability to translate words embedded in photos into editable text, eliminating the need for entering art object label description by hand. Last but not least, I optimized app overall performance in terms of RAM usage and UI responsiveness.

5.1.2. Fangzai Hong

In cloud tab, I designed the layout and realized the self-instantiation of view controller for infinite subfolder browsing. I also implemented upload and download function by using Dropbox API with progress bar displayed and photo thumbnail. In the audio part, I implemented recording and playing function with creation and deletion of data entry in relational database and actual file on disk. In the settings tab, I designed the layout and provide location access option and photo library storage calculation. I also contributed to UI design by enabling background change function in setting tab. In the last place, I helped develop the camera function by trying out the advanced camera library (AVfoundation).

5.2. Specialist

5.2.1. Anh Thu Nguyen

I was able to prepare metadata sets by consulting auction/exhibition catalogues as well as literature on “digital art history”. I also performed an ethnographic study on the information behaviour of three art history students so as to understand the ways in which they used photographs and wrote notes when doing fieldwork in the museum. Moreover, I tested different features of our app at the Royal Ontario Museum to determine how well it worked. Finally, the most important part

of my work has been to brainstorm ideas directly with members of the Art Department in order to ensure that our app would be intuitive and accessible from the art historian's point of view.

6. Specialist Context

6.1. Current State of Image Collections

To possess an app that can quickly organize, store, and share photographs of museum objects has great value to art historians as this convenient tool alleviates the weight they must carry during fieldwork. Indeed, while museums are gradually allowing photographs to be taken in their spaces, DSLR cameras may be cumbersome, especially when carried along with notebooks and laptops (large bags remain forbidden in most museums). Since mobile devices are capable of capturing photographs with good image quality, art historians are now resorting to this equipment, leaving behind the heavier cameras. Consequently, having a small and portable platform on which both photos and notes may be recorded concurrently constitutes a great advantage to art historians who may then focus on the study of their objects rather than the organization of their photos and notes. Furthermore, since art historians must juggle with different types of information, i.e. visual, textual, and aural, they must be reminded of the different things that need to be recorded: Gallery Composer provides these reminders via its prompted metadata form.

6.2. Contribution to the Field

With the advent of the digital humanities, many humanistic disciplines are turning towards methodologies that incorporate digital tools in order to enhance research practices and findings. While art history had striving ambitious projects in the late twentieth century¹, the current state of “digital art history” remains stagnant as no new tools are being developed despite numerous publications on the changing research practices of art historians². I see Gallery Composer as contributing to the new *digital* and *digitized* research methodologies of art history³ by turning previous abstract recommendations into a concrete tool simple to use and easily available. With the making of this app, I also hope to influence other art historians by inspiring them to build their own digital tools, in whatever format these may be.

¹ Benjamin Zweig, “Forgotten Genealogies: Brief Reflections on the History of Digital Art History,” *International Journal for Digital Art History* 1 (2015): 39-49.

² Matthew P. Long and Roger C. Schonfeld, *Supporting the Changing Research Practices of Art Historians* (New York: Ithaca S + R, 2014), 192-205.

³ Johanna Drucker, “Is There a Digital Art History?” *Visual Resources: An International Journal of Documentation* 29, no. 1 (2013): 7.

7. Future Work

An important feature of our application is to allow metadata to be incorporated within the properties of its respective photograph. Because this metadata reveals the geographic and temporal distribution of museum objects, they may be used for quantitative analyses. We therefore envision a way in which this information could be extracted onto a spreadsheet for such research projects. Furthermore, in order to open up Gallery Composer to a wider audience, we would need to support for various device form factors (i.e. tablet) as well as create a desktop version (Mac OS) to facilitate viewing of image collections across multiple platforms.

8. Source Code

The source code for the project is submitted via BlackBoard.

9. Availability of Content Online

Please only make our report and video available and keep the source code private, since we may continue developing the app and hopefully commercialize it.