EVENT PHOTO STREAM SEGMENTATION: CHAPTER-BASED PHOTO ORGANIZATION FOR PERSONAL DIGITAL PHOTO LIBRARIES

JESSE PRABAWA GOZALI

NATIONAL UNIVERSITY OF SINGAPORE

2013

EVENT PHOTO STREAM SEGMENTATION: CHAPTER-BASED PHOTO ORGANIZATION FOR PERSONAL DIGITAL PHOTO LIBRARIES

JESSE PRABAWA GOZALI
(B.Comp. (Comp.Eng.) (Hons.), NUS)

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF COMPUTER SCIENCE

NATIONAL UNIVERSITY OF SINGAPORE

2013

DECLARATION

I hereby declare that this thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

Jesse Prabawa Gozali
11 March 2013

Acknowledgements

I would like to thank my advisor, Dr. Kan Min-Yen for his constant support, help and guidance throughout the years. I would also like to thank my collaborators, Dr. Hari Sundaram and Dr. Ramesh Jain for their wisdom, feedback and guidance at various stages of the project. I am grateful for the opportunity and privilege of working under the best minds in the field.

To my parents, family, and closest friends, I dedicate this thesis to you. Thank you for helping me in this journey and for lending an ear or two when I needed them the most. Gwen, Ben, Rox, Jing, Justicia, Jennifer, and the most wonderful friends at LWMC, you are the best.

To my lab mates and WING group members past and present, thank you for enduring my presence (and absence) through the many years, for tolerating me in my ups and downs and in giving invaluable feedback to my research, my many paper submissions and research updates.

Most of all, I dedicate this thesis to God. I thank Him for His countless blessings and for His grace and mercy for allowing me to pursue this to completion, despite the many challenges. Without Him, this thesis and its entirety would not have been possible.

"Don't worry about anything; instead, pray about everything. Tell God what you need, and thank him for all he has done." — Phil 4:6 NLT

Table of Contents

Abstract

Most commercial photo browsers today have an automatic mechanism to help users group their photos by event. This automatic **event-based photo organization** has not always been available. In the early days, digital photo management was similar to its analog counterpart where users had to manually organize their photos into photo albums. This thesis is motivated by the same issues today, but for photos within an event. People now are more liberal with their photo taking and have even more photos to manage for each of their events.

To complement event-based photo organization and help users manage photos in each event, this thesis proposes a **chapter-based photo organization** where photos from each event are organized further, *i.e.* separated into smaller groups according to the moments in the event. We refer to this task as **event photo stream segmentation**. In this thesis, we developed a method to accomplish this exact task. Our method is based on a hidden Markov model with parameters learned from 1) a dataset of unlabelled, unsegmented event photo streams and 2) the event photo stream we want to segment. Our method is unsupervised, relies on features from temporal, camera parameters and visual information that are fast to compute. Our approach is based on our novel observation that an event's photo stream consists of alternating feature types: features of the photo and features between consecutive photos. In an experiment with over 5000 photos from 28 personal photo sets, our method outperforms baseline methods including the state-of-the-art with $p < 0.05$.

This thesis also describes results from the first user study on chapter-based photo organization. The findings reveal key insights on how people organize their event photos. For example, users value chapter consistency more than the chronological order of the photos. The study also reveals common criteria people use to group their events into chapters. Another novel contribution is the photo layout study findings where we found that users value the chronological order of the chapters more than maximizing screen space usage and that users like having chapter thumbnails, but not at the expense of screen space utilization.

Finally, the work we present culminates in CHAPTRS ver. 2, a publicly available, fully-implemented chapter-based photo browser that 1) complements event-based photo organization by working with users' existing digital photo libraries (iPhoto and Aperture), 2) automatically separates events into chapters, 3) presents the photos with a user interface design and photo layout based on the user study findings, and 4) allows easy drag-and-drop operations to fine-tune the photo arrangement with any criteria.

To further research in this area, we used CHAPTRS ver. 2 to build a large public dataset of anonymous photo features and describe how using the Mac App Store as a distribution channel allowed us to reach a large number of participants and their personal digital photo libraries, a feat that would be difficult to achieve with volunteers or other conventional means.

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Background

Most personal photos are commonly associated with an event: a holiday trip, birthday, wedding, gathering, picnic, walk in the park, etc. This is true for photos from both analog and digital cameras (Rodden, 1999; Rodden and Wood, 2003). With the former, film rolls must be developed in their entirety or not at all. As such, they are often developed whenever they become completely used and thus, produce photos from multiple events. These multi-event photos would then either all go into storage, *e.g.* a shoebox, or — sometimes — be painstakingly sorted through and placed into separate photo albums.

With digital cameras, people now have the freedom of importing their photos whenever they want, *e.g.* diligently after every event without having to wait for a full memory card. The less inclined may still import their photos as a batch, spanning over multiple events from one or more memory cards. Commercial photo browsers however, make this process easier by automatically placing the photos into separate digital photo albums, each corresponding to an event. This **automatic albuming** is a common feature among many popular commercial photo browsers like iPhoto[1], Picasa[2], and Windows Photo Gallery[3]. Research into automatic methods to enable such an **event-based photo organization** yielded many papers in 2003–2007, which we will review in Chapter 2. These automatic albuming methods are capable of producing very satisfactory results. In fact, some commercial photo browsers like iPhoto suffice today by using a simple time interval (1-day, 8-hour, or 4-hour) for its automatic albuming, *e.g.* photos spanning over two days will be grouped into two events if the 1-day time interval was selected by the user.

As compact cameras and film rolls have enabled people to acquire large photo collections that need to be grouped into separate albums, continuing advancements in digital photography have enabled people to freely capture every moment of their life events, yielding hundreds of photos for a single event. Photos in such events are as large as the analog era photo collections that needed to be grouped into albums.

Today, our digital cameras can take more than a thousand 14 megapixel photos with every 4GB of storage. With each new version, digital cameras take even less time to start up and to wait between shots. The Apple iPhone 4S, the most popular camera and most popular cameraphone on Flickr[4], starts up in 1.5 seconds and waits a mere 0.7 seconds in between shots[5]. The advent of such easy-to-use and portable photo capture devices with large memory stores have changed people's photo taking habits — people now are more liberal with their photo taking, as compared to the previous era of film rolls and analog cameras (Kirk et al., 2006).

While today's photo browsers automatically group imported photos into separate albums by event, the resulting albums — especially those corresponding to holiday trips or other important life events — contain hundreds of photos spanning over multiple moments throughout the event. For example in Figure 1.1, in a family trip to the zoo, photographed moments may include arriving at the zoo,

[1] `http://www.apple.com/ilife/iphoto`

[2] `http://picasa.google.com`

[3] `http://windows.microsoft.com/en-US/windows-live/photo-gallery-get-started`

[4] `http://www.flickr.com/cameras`

[5] `http://www.smartdevicecentral.com/article/289761_1.aspx`

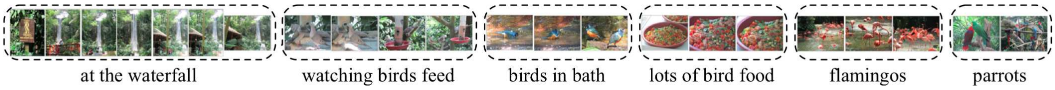


Figure 1.1: Part of a family photo album of a trip to the zoo, shown consisting of multiple chronological moments

at the waterfall, watching birds feed, birds in a bath, seeing lots of bird food, visiting flamingos, looking at parrots, petting baby animals, picnic lunch at the park, etc. Having all these photos grouped into a single album is appreciated, but sifting through all these photos and not able to easily perceive and appreciate the constituent moments is still cumbersome.

1.1.1 Problem Statement

In this thesis, we propose a complementary goal to event-based photo organization we call **chapter-based photo organization** in which photos from a single event are separated into smaller groups according to moments in the event.

Hypothesis: *Chapter-based photo organization provides a better user experience than event-based photo organization in a photo browser for a personal digital photo library.*

To investigate our hypothesis, we developed an automatic method to achieve this organization that outperforms all our baselines with statistical significance. We conducted a user study to observe how people organize their event photos in a chapter-based photo organization setting and also measured their preference in several photo-related tasks with and without chapters to organize their event photos. In a photo layout study, we explored orthogonal photo layout aspects, *e.g.* chronological ordering and screen-space utilization, to best visualize chapters of the event. Our proposed method, photo organization study, and photo layout study are the central topics of this thesis. Together, our work informs the development of our publicly available chapter-based photo browser we call CHAPTRS ver. 2.

Through our investigation, this thesis presents four main contributions: the event photo stream segmentation algorithm, the photo organization study, the photo layout study, and our photo browser CHAPTRS ver. 2. We elaborate on these in the following sections.

1.2 Event Photo Stream Segmentation

We refer to the chapter-based photo organization task as **event photo stream segmentation**, *i.e.* the process of finding contiguous groups of photos from an event photo stream, each group corresponding to a photo-worthy moment in the event (see Figure 1.2). **An event photo stream** is a chronological sequence of photos from a single event.

We distinguish between an event photo stream and **a photo stream**, which is a more general term that refers to a chronological sequence of photos that may span over multiple events, consisting of many days or even months of photos. Many segmentation methods have been proposed for such photo streams to produce groups of photos where each group corresponds to an event. To distinguish between their task and ours, we shall refer to their task as **automatic albuming**. For example, in Figure 1.2, the sequence of photos referred to as "My Photos (2011 - 2012)" is a photo stream that spans multiple events. On the other hand, the sequence of photos referred to as "Dad's 62nd Birthday" is an event photo stream because it is a photo stream of one particular event.

While both tasks segment photo streams, automatic albuming methods may not be suitable for event photo stream segmentation due to issues of data sparsity, indistinct time gaps, and visual similarities:

1. *Data sparsity* — Each group of photos produced through event photo stream segmentation has only a handful of photos as each corresponds to a photo-worthy moment in the event. In contrast, each group produced through automatic albuming corresponds to an event and has many more photos. A photo stream of multiple events also has many more photos than an event photo stream, which is of just one event. The increased sparsity associated with

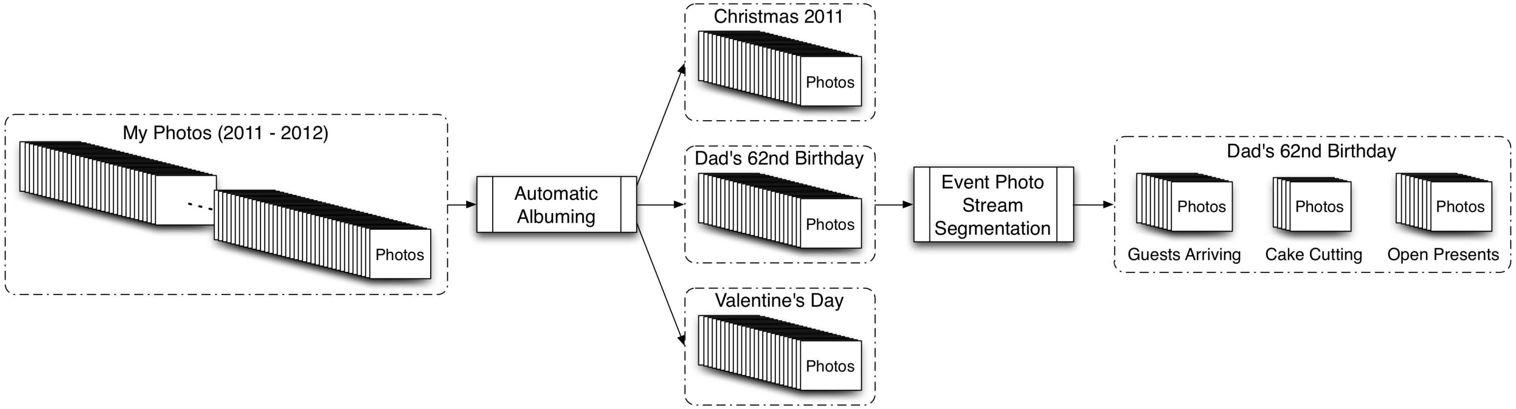


Figure 1.2: Event photo stream segmentation is the process of finding contiguous groups of photos from an event photo stream. In contrast, automatic albuming is the process of grouping photos from a collection into separate events.

event photo stream segmentation makes it harder to develop computational models.

2. *Indistinct time gaps* — In a photo stream, **time gap** is the time difference between the capture times of two consecutive photos. While the time gap between two photos of different events is in hours or even days, the time gap between photos of the same event is typically in seconds or minutes. This time scale difference is useful to identify event boundaries for automatic albuming. In contrast for event photo stream segmentation, the time gap between two consecutive photos belonging to different photo-worthy moments in the event is also in seconds or minutes. Indistinct time gaps at segment boundaries in an event photo stream makes the segment boundaries difficult to identify using simple heuristics.

3. *Visual similarities* — Photos in an event are often visually similar because they share aspects such as participants, location, and scene. With photos of other events, however, they are often visually distinct because these aspects are different. The visual difference between photos of different events is useful for automatic albuming, but the visual similarities among photos of an event make event photo stream segmentation more difficult.

To address these challenges, we propose a hidden Markov model (HMM) - based approach that uses a combination of time, Exif[6] metadata, and visual information to determine the segment boundaries (*i.e.* chapter boundaries) in an event photo stream. Parameters of the HMM are learned from 1) a set of unlabelled, unsegmented event photo streams and 2) the event photo stream we want to segment. Our model supposes that an event photo stream is the result of a stochastic process that generates feature vectors from a set of foreground and background models. The foreground models generate feature vectors corresponding to segment boundaries while the background models generate feature vectors that do not.

[6] JEITA Exchangeable image file format for digital still cameras

This generative model follows from our observation that photos taken in events are often the result of several **photo taking sessions** — each session corresponds to a photo-worthy moment. At such a moment, we take several photos. Then, our camera idles until the next moment arises and invites us for another photo taking session. In each session, photos would likely be similar in terms of visual appearance, photo metadata and timing. The photographer, for example, could choose to adjust the focal length and aperture settings to suit the scene of the moment. These camera parameter values would be similar for photos within the same session. If we look at photo timestamps, each session would appear to be a burst of photo activity (Graham et al., 2002).

1.3 Photo Organization Study

While there have been several user studies on personal photography in the past decade — which we will cover in more detail in Chapter 2 — to our knowledge there has not been a user study for photo organization within an event, *i.e.* at the chapter level.

In this study, we want to answer the following questions: *How do people organize their photos in each event and how does it affect typical photo-related tasks such as storytelling, searching and interpretation tasks?* In exploring these questions, we explore our hypothesis that organizing photos in each event into chapters provides a better user experience. Additionally, we draw contrast and similarities with findings from previous studies done at the event level.

To facilitate this study, we developed the first version of our chapter-based photo browser called CHAPTRS. CHAPTRS helps users organize their event photos by automatically grouping photos in each event into smaller groups of photos we call chapters. CHAPTRS builds upon our method for automatic event photo stream segmentation. CHAPTRS also affords users with a drag-and-drop interface to refine the chapter groupings. In Chapter 5, we describe how our work in this thesis culminates in CHAPTRS ver. 2 which was inspired by the findings of the user study.

By designing tasks where user behavior and performance can be observed and measured, we were able to compile novel insights into how the participants organize their photos in each event and how the organization affects the tasks.

1.4 Photo Layout Study

The photo layout study was done in conjunction with the photo organization study described in the previous section, in a two-week exploratory user study involving 23 college students with a total of 8096 personal photos from 92 events.

In CHAPTRS ver. 1, we presented users with four photo layouts which can be seen in Chapter 4 in Figures 4.1, 4.2, 4.3, and 4.4. The first is our baseline, a plain grid layout that offers no chapter-based photo organization. The other three layouts present chapter-based photo organizations but each emphasizes on a different key photo layout aspect. The **bi-level layout** emphasizes an overview of the event photos afforded by presenting chapter thumbnails. The **grid-stacking layout** emphasizes the chronological order of the chapters. Lastly, the **space-filling layout** maximizes screen space usage.

The three chapter-based photo layouts were chosen because they emphasize and represent distinct key photo layout aspects. As such, they facilitated our study to explore which key photo layout aspects are important for chapter-based photo organization. To our knowledge, our study is the first to explore chapter-based photo organization and its photo layouts.

1.5 CHAPTRS Photo Browser

From our method and our findings in the photo organization study and the photo layout study, we iterated on CHAPTRS ver. 1 and developed a fully-implemented, publicly available photo browser, which we will refer to as CHAPTRS ver. 2. Like its previous version, it complements event-based photo organization by reading

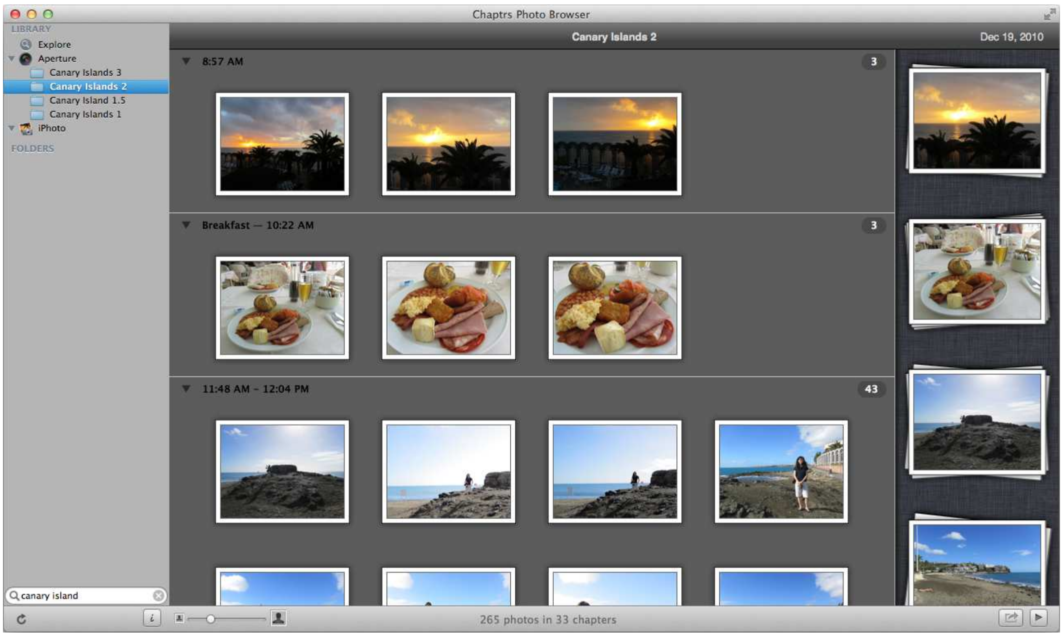


Figure 1.3: Screenshot of our photo browser, CHAPTRS ver. 2

existing events and albums from the user's computer (*i.e.* in iPhoto and Aperture) and automatically organizing them into chapters. The results are then presented to the user as shown in Figure 1.3.

CHAPTRS ver. 2 provides users with an easy drag-and-drop user interface for fine-tuning the arrangement. Photos and/or chapters can then be selected for sharing to various services and social networks like Flickr, Twitter, Facebook, etc. We will go into more details in Chapter 5.

1.6 Contributions

The three main challenges in this thesis is the development of an unsupervised method for automatic event photo stream segmentation, the exploration of user behavior in chapter-based photo organization, and the study of photo layout aspects to support effective chapter-based photo organization. In tackling these three challenges, this thesis makes four main contributions to the field of personal digital photo libraries:

- **Unsupervised method** — We developed an unsupervised method for event photo stream segmentation, finding contiguous groups of photos from an event photo stream, each group corresponding to a photo taking session in the event. Our method uses a hidden Markov model with alternating observation types to embody our novel observation that event photo streams exhibit alternating feature types (photo features and photo gap features) that cannot be captured effectively with a single observation type. Our method outperforms all baseline methods including the state-of-the-art with statistical significance, $p < 0.05$.

- **Photo organization study** — We conducted a user study with 23 college students of various photography backgrounds to ascertain how they organize photos within an event and how a chapter-based photo organization affects photo-related tasks such as storytelling, searching, and interpretation tasks. Our study is the first study to explore and draw insights from a chapter-based photo organization.

- **Photo layout study** — In the same user study, we conducted a photo layout study to explore a set of orthogonal features for presenting a chapter-based photo organization: timeline visualization, screen space usage, and view hierarchy. Similarly, our study is the first study to ascertain the relative importance of these layout features for chapter-based photo organization.

- **CHAPTRS Photo Browser** — We developed a fully-implemented publicly available chapter-based photo browser, CHAPTRS ver. 2. With the browser, we then built a large dataset of anonymous photo features that we are releasing to the research community. We also report on our experience building the dataset, using the Mac App Store as a distribution channel to alleviate issues with scalability, cost and reaching a large number of potential study participants and their personal digital libraries. Our experience and results shows that the Mac App Store provides a fruitful and viable alternative for large-scale data collection especially for reaching out to personal digital libraries.

1.7 Thesis Outline

In the next chapter, Chapter 2, we review related work for the three main challenges of this thesis: event photo stream segmentation, user studies on personal photography, and photo layouts in personal digital photo libraries.

In Chapter 3, we elaborate on our event photo stream segmentation method. We start by formally defining an event photo stream and what it means to produce its segmentation. We outline the information that we can derive from a given event photo stream and proceed to mathematically define the task of event photo stream segmentation. We then propose the concept of photo taking sessions which we use as a basis for our method. We detail how we model the event photo stream using a generative process and describe how we can use the Baum-Welch and Viterbi algorithms of the hidden Markov model to efficiently find the segment boundaries in our event photo stream. After our analysis of features and hidden Markov model structures, we describe our method pipeline, evaluate its performance and discuss the results.

In Chapter 4, we report on our user study on user behavior and photo layouts for chapter-based photo organization. Here, we report on novel insights on how users group their event photos into chapters. We also report statistically significant results on how chapter-based photo organization affects three photo-related tasks: storytelling, searching, and interpretation. Additionally, we gathered key insights on photo layout aspects for chapter-based photo organization.

In Chapter 5, we describe version 2 of our CHAPTRS photo browser. We describe how our work and findings from the previous chapters manifest themselves in this end-user application. In particular, we describe practical considerations in integrating our event photo stream segmentation method in CHAPTRS ver. 2 and how the user study and photo layout findings affected the user interface design.

Using CHAPTRS ver. 2, we constructed a dataset and report on our experience in using the Mac App Store in Chapter 6. Here we discuss how using the Mac App Store as a distribution channel allowed us to reach a large pool of potential study participants and thus build a large dataset of anonymous photo features.

Finally, we conclude in Chapter 7 on our work on event photo stream segmentation for a chapter-based photo organization, where we comment on the main issues in this topic going forward.

Chapter 2

Related Work

In this thesis, we identify three main areas of related work. The first is **photo stream segmentation**. This thesis explores photo stream segmentation where the photo stream consists of photos from *a single event*. While this problem has not been explicitly addressed in existing literature, we review related works where the photo stream consists of photos from a collection, comprising of *multiple events*. These works seek to identify events or albums within the photo collection. In our case, we seek to identify moments within the single event. Our research problem can be seen as a more fine-grain and data-sparse version of the problem addressed by these existing works.

The second area is **personal photography user studies**: from how people manage their printed or digital photo collections to the entire process that people go through from capturing to sharing of photos. To our knowledge, our user study is the first to explore chapter-based photo organization. Lastly, we explore the area of **photo layouts in personal digital photo libraries**. We identify issues addressed in photo layouts for event-based photo organization and discuss how they apply to a photo layout catered for chapter-based photo organization.

2.1 Photo Stream Segmentation

To our knowledge, the closest work to ours is by Graham *et al.* (2002). They posit that people tend to take photos in bursts and these bursts can be identified by looking at time gaps that are statistical outliers and not part of any burst. Their event photo stream segmentation method finds segments corresponding to bursts of photo taking activity. This method was used iteratively to form a hierarchy of segmentations, which was used to select 25 photos to summarize photos at various temporal levels (year, month, etc) in their proposed calendar photo browser.

Other photo stream segmentation methods were devised for automatic albuming. Most of these methods rely on time information. The simplest method to find segment boundaries is to check for time gaps that are greater than a fixed threshold (*e.g.* average time gap). Loui and Savakis (2003) used a time scaling function and K-means clustering with K=2 to determine this fixed threshold. Platt *et al.* (2003) proposed a method where the threshold becomes adaptive, computed over a sliding window. Some methods are similarly adaptive, although based on keen observations instead of thresholding; Zhao *et al.* (2006) observed that the probability of an event ending increases as more photos are taken and as the time span increases; Gargi (2003) observed that a long interval with no photo taking usually marks the end of an event and that a sharp upward change in the frequency of capture usually marks the start of a new event. Pigeau and Gelgon (2003) proposed a model-based incremental unsupervised classification where distinct classifications are built from both temporal and location information.

Few methods have utilized Exif metadata. Gong and Jain (2007) proposed a segmentation method based on changes in scene brightness. Mei *et al.* (2006) proposed a clustering approach using Exif metadata like aperture diameter, exposure time, and focal length. Their method also used time, location and visual features such as color histogram, and Tamura descriptor (texture). There are only few others that have utilized visual information. Platt *et al.* (2003) proposed a best-first model merging method based on color histograms. Cooper *et al.* (2003) proposed an approach based on scale-space analysis of both color and time information.

Most automatic albuming methods utilize time gap information. Because the time gaps at event boundaries are typically much larger than the time gaps between photos in an event, these methods work effectively to segment a photo stream by event. For event photo stream segmentation, where segments are more fine-grained, the segment boundaries may not be distinguishable with time information alone. Other information based on Exif metadata and visual information should be utilized. The data-sparsity of the task however, provides a challenge for the selection of viable features. We will revisit this issue on features in Chapter 3.

2.2 Personal Photography User Studies

Over the past decade, there have been a number of studies on how people manage—including organization and sharing—their personal photo collections. Rodden (1999; 2003) has studied how people manage their photo collections, printed or otherwise. Some findings from his study include: printed photo albums are mostly classified by event, with one album for each event. Searching a printed photo collection is typically done for a photo album of a specific event. Even if the search was for a specific photo, people will try to locate the album containing the photo first before starting the search. For personal digital photo libraries, people regard the ability to organize photos into folders as very useful and would arrange them according to events in a chronological order. People prefer to browse their photos by event rather than querying. Similar findings were also found by Cunningham and Masoodian (2007). They conclude that browsing, rather than searching, is a more practical tool for locating photos.

Other studies go beyond how the photos are organized. Kirk *et al.* (2006) coined the term "photowork", *i.e.* activities done after photo capture but before sharing. These include reviewing, downloading, organizing, editing, sorting, as well as filing of photos. Frohlich *et al.* (2002) conducted a study to establish requirements for photo sharing technologies. A recent article by Sandhaus and Boll (2011) presents a good overview of research in this field of personal photo collections, including many works that we review in this chapter.

To our knowledge, our work is the first to explore chapter-based photo organization. In Chapter 4, we report on novel insights on how users group their event photos into chapters and how chapter-based photo organization affects photo-related tasks such as storytelling, photo search and event photos interpretation.

2.3 Photo Layouts in Personal Digital Photo Libraries

An effective photo layout is one that presents photos in a way that supports users in one or more photo-related tasks. Here, we review existing works on photo layouts for personal digital photo libraries to gather the key aspects they emphasize and the tasks they support effectively.

While there has been prior work to study layouts for event-based photo organization, the absence of prior work on photo layouts for chapter-based photo organization, *i.e.* layouts to present groups of photos with all groups belonging to *the same event* is notable. In event-based photo organization, the groups of photos belong to *different events*. The closest work we found was by Graham *et al.* (2002). They proposed a hierarchical calendar photo browser to better support search tasks by presenting a 25 photo summary at various levels of hierarchy of the user's photo collection: year, month, event, and also for groups of photos within an event. The user navigates through the view hierarchy using a tree view in the sidebar.

For event-based photo organization, the most common photo layout is a 2D grid: photos are ordered chronologically row by row on a grid. Many photo browsers (Kuchinsky et al., 1999; Mills et al., 2000; Drucker et al., 2004; Mei et al., 2006) including commercial ones like Picasa and iPhoto adopt this layout to display photos of an event. A plain grid layout is a simple layout that maximizes use of the available screen space. Having many photos visible at once allows users familiar with the photos to scan them very quickly (Rodden and Wood, 2003).

Photo browsers typically display one event (one grid) at a time, but some photo browsers relieve users from having to select individual events from the view hierarchy by displaying all the events at once: the grids are stacked on top of each other in chronological order, *e.g.* Picasa. The layout remains uniform as the grids have the same number of columns. With this layout, users can browse their events by simply scrolling. To demarcate the events, each grid has a title bar on top with the event information. Alternatively, in the timeline view of one photo browser (Mills et al., 2000), each grid is labeled hierarchically on its left margin by month and year. In another (Chen et al., 2006), all the photos in the collection are displayed as one massive grid and event titles are displayed as grid elements to demarcate the events.

Time Quilt (Huynh et al., 2005), a zoomable photo browser designed to enhance search tasks, also displays photos from all events at once. Its layout trades-off screen space usage for better presentation of the chronological order of the photos. Photos from each event are displayed in their own grid. The grids are then displayed chronologically column by column. The number of rows and columns of each grid follows the aspect ratio of the corresponding thumbnail of the event. Each grid is replaced with the event thumbnail of the same size and the grid only becomes visible when the user zooms in.

Some photo browsers do not use a grid layout. TreeBrowser (Chen et al., 2010) is a photo browser for multiple photo collections. The collections are displayed chronologically at the top of the photo browser as a single scrollable row of thumbnails. The main part of the photo browser displays events from the selected collection as a tree of depth one. The tree root is the collection thumbnail. Each leaf corresponds to an event in the collection and is displayed as a single row of photos.

The works we have reviewed so far have weaved the chronological order of the photos into two dimensions (*e.g.* row-by-row) to make better use of screen space.

However, in interfaces where visualizing the timeline is more important, chronological order is commonly conveyed as a single dimension in the layout (Plaisant et al., 1996; Fertig et al., 1996; André et al., 2007). Photo storytelling interfaces exhibit similar linear structures in their layouts. Here, we highlight three notable interfaces: the first two are well-cited and the third is a recent contribution to the field. First is the story-editing environment in FotoFile (Kuchinsky et al., 1999). Here, users can select photos from an *Image Tape* at the top of the photo browser and place them into one of the row of *Scraplets* in the main part of the photo browser. Each scraplet displays its photos as a single column. Balabanović *et al.* (2000) developed a portable device for sharing and authoring stories. In its interface, the navigation area consists of rows of photo thumbnails. Photos in the rows are shown in groups of alternating backgrounds to distinguish separate photo rolls. Recently, Raconteur (Chi and Lieberman, 2010) is a story editing system that helps users assemble stories from annotated media files. The media files are arranged in chronological order in a single row.

Some photo browsers were designed to emphasize inter-photo similarity, *e.g.* in terms of visual appearance, location, or tag. These photo browsers generally present more visually interesting and novel layouts. However, the chronological order of the photos often suffers as a result. For example, PhotoMesa (Bederson, 2001) employs quantum treemaps and bubblemaps to display labelled photo clusters in a grid layout to maximize screen space usage. More recently, MediaGlow (Girgensohn et al., 2010) uses a spring layout algorithm to help users stack and retrieve similar photos. PHOTOLAND (Ryu et al., 2010) presents a layout that places photos on a 2D grid based on an inter-photo similarity measure computed from temporal and spatial information. The result is a layout that presents photos from an event as an island of thumbnails.

The works we have reviewed have layouts that emphasize one or more of the following key aspects: use of view hierarchy, chronological order of event photos, and maximization of screen space usage. In Chapter 4, we emphasize similar key aspects in the three layouts used in our user study.

2.4 Conclusion

In this chapter, we have reviewed work on event photo stream segmentation from three main areas: photo stream segmentation, personal photography user studies, and photo layouts in personal digital photo libraries. While we only discuss works in these three areas, our work on a chapter-based photo organization has applications in other areas where such an organization is a helpful, if not necessary, pre-processing step to their tasks.

For example, in the area of automatic photo book creation, some works (Gao et al., 2009; Xiao et al., 2010) employ a selection process as part of the photo book creation which could benefit from a chapter-based photo organization. Another work describes the CeWe Color photo book software (Sandhaus et al., 2008) which actually employs a time clustering method as part of its process.

We will elaborate on the contributions in each area (photo stream segmentation, personal photography user studies, and photo layouts) in Chapters 3 and 4. But first, we will formally define the task of event photo stream segmentation in the next chapter.

Chapter 3

Event Photo Stream Segmentation

Given an event photo stream, we want to find groups of photos in the stream such that each group corresponds to a photo taking session. The groups should also form a partition over all the event photos (see Figure 3.1).

We start by formally defining an event photo stream and what it means to produce its segmentation. In the absence of semantic information, we propose the concept of photo taking sessions as a basis for automatic event photo stream segmentation.

We then describe how an event photo stream can be modelled by a generative process and show that in this model, the segmentation solution can be efficiently found with the Baum-Welch algorithm of a hidden Markov model (Baum et al., 1970). We then report results from our feature and structure analysis and subsequently, describe further enhancements using probability smoothing and spuri-

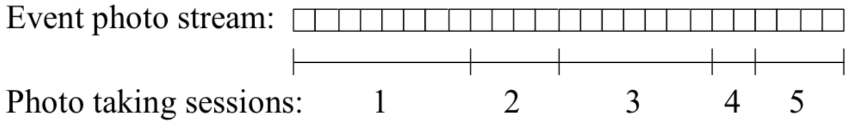


Figure 3.1: Photo taking sessions form a partition over the event photo stream.

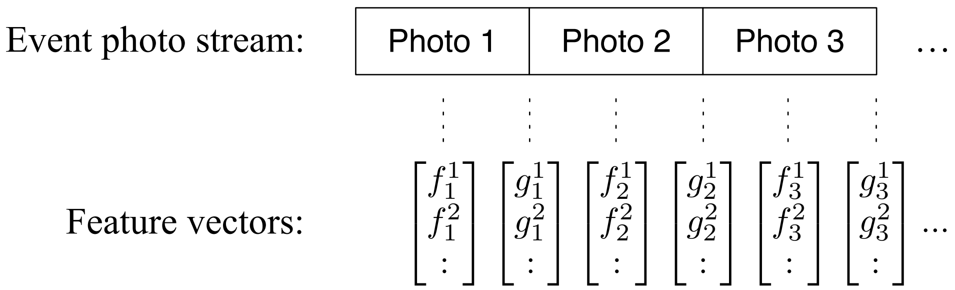

Event photo stream:
Photo 1
Photo 2
Photo 3
...
Feature vectors:
$\begin{bmatrix} f_1^1 \\ f_1^2 \\ \vdots \end{bmatrix} \begin{bmatrix} g_1^1 \\ g_1^2 \\ \vdots \end{bmatrix} \begin{bmatrix} f_2^1 \\ f_2^2 \\ \vdots \end{bmatrix} \begin{bmatrix} g_2^1 \\ g_2^2 \\ \vdots \end{bmatrix} \begin{bmatrix} f_3^1 \\ f_3^2 \\ \vdots \end{bmatrix} \begin{bmatrix} g_3^1 \\ g_3^2 \\ \vdots \end{bmatrix} \cdots$

Figure 3.2: Given an event photo stream, we can derive two types of features: 1) Photo Feature, *i.e.* features about the photos (f_i^j), and 2) Photo Gap Feature, *i.e.* features about the gap between consecutive photos (g_i^j), where j is a feature index and i is a photo or photo gap index. The extracted photo and photo gap features from the event photo stream form a sequence of alternating feature types.

ous solution filtering techniques before concluding with the final pipeline of our method.

3.1 Alternating Feature Types: Photo and Photo Gap

In our literature review in Chapter 2.1, most photo stream segmentation methods rely on time information alone. Some incorporate visual features and very few use features derived from Exif metadata. In this thesis, we organize the different features that can be extracted from an event photo stream using the following schema: Given a sequence of photos, for example in Figure 3.2, we can derive two types of features[1]:

1. **Photo Feature** — *i.e.* feature about the photo. For example, the visual information contained in the pixels of the photos, the camera parameters that tell us how the photos were captured using the camera, as encoded in the photos' Exif metadata.

2. **Photo Gap Feature** — *i.e.* feature about the gap between consecutive photos, *i.e.* the difference between consecutive photo feature values. For exam-

[1] We evaluated both types of features for our method; See Section 3.9.

ple, time gap, which is the time difference between capture times of consecutive photos.

This observation that the event photo stream features belong to two alternating types — photo feature and photo gap feature — is novel and forms the basis of how we formally define the problem and proposed solution to the event photo stream segmentation task.

3.2 Problem Definition

With the features we extract from the event photo stream, we end up with a sequence of vectors with alternating types (see Figure 3.2). From an event photo stream of N photos, we get a sequence of $2N - 1$ vectors, of which $N - 1$ are photo gap features whose locations correspond to *potential segment boundaries* in the event photo stream segmentation.

We define an event photo stream segmentation X as a sequence of Boolean variables $\langle X_1, X_2, ..., X_{N-1} \rangle$ corresponding to these potential segment boundaries, such that $X_k = 1$ if there is a segment boundary between photos k and $k+1$, and 0 otherwise. Given a sequence of feature vectors S, our task is to find which gaps between consecutive photos correspond to segment boundaries and which do not:

$$f(X_k|S) = \begin{cases} 1 & \text{if the gap between photos } k \text{ and} \\ & k+1 \text{ is a segment boundary,} \\ 0 & \text{otherwise.} \end{cases} \tag{3.1}$$

3.3 Photo Taking Sessions

The goal of event photo stream segmentation is to find groups of photos corresponding to moments in the event. In Chapter 1, we illustrate this with an example where moments in a zoo visit event may entail: arriving at the zoo, at the waterfall, watching birds feed, birds in a bath, seeing lots of bird food, visiting flamingos, looking at parrots, petting baby animals, picnic lunch at the park, etc. In the absence of semantic information however, how do we find these moments in such an event?

When we view photos from an event, we often make inferences about how each photo relates to its surrounding photos and how different groups of photos in the stream fit together to capture different moments in the event. Without semantic knowledge of the event, *i.e.* we are unfamiliar with the event, we make such inferences based on the visual appearance and timestamp of the photos.

We refer to a group of photos found through this manual inference process as a **photo taking session**, *i.e.* a period of time devoted to photo taking, producing photos with similarities in visual appearance, Exif metadata, and timing. We observe that photo taking sessions correlate well with moments in the event because whenever a photoworthy moment arises, we raise our camera, capture some photos in succession, possibly with slight variations in camera settings. Then we wait for the next moment to arise and repeat the process as part of another photo taking session.

Thus, while we cannot find moments in the event photo stream using the *unavailable* semantic information, we can find the photo taking sessions that correlate with the moments. This is the basis for our event photo stream segmentation method.

3.4 Modeling Event Photo Streams With a Generative Process

Consider the event photo stream, E, shown in Figure 3.3. E consists of a sequence of N photos, *i.e.* $\langle p_1, p_2, ..., p_N \rangle$. Let us assume that E consists of a sequence of M photo taking sessions, *i.e.* $\langle \mathrm{PTS}_1, \mathrm{PTS}_2, ..., \mathrm{PTS}_M \rangle$. Unlike N, M is unknown to us.

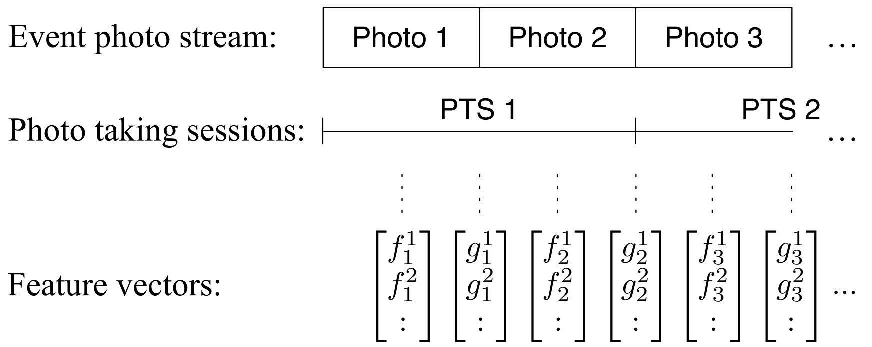


Figure 3.3: An event photo stream consists of a sequence of photos, each belonging to exactly one photo taking session (PTS). From the photos, we can extract photo features (f_i^j) and photo gap features (g_i^j), where j is a feature index and i is a photo or photo gap index.

Let every photo in E belong to exactly one PTS in E, *i.e.* PTS_k contains a sequence of N_k photos, $1 \leq k \leq M$, such that $\sum_k N_k = N$, and the set $\{\mathrm{PTS}_k\}$ forms a partition over the set of photos $\{p_i\}$, $1 \leq i \leq N$. Like M, the set $\{N_k\}$ is also unknown to us because we do not know the alignment between the photos $\{p_i\}$ and photo taking sessions $\{PTS_k\}$.

From the N photos, we can extract N photo feature vectors and $N - 1$ photo gap feature vectors. More specifically, each PTS_k — if N_k is known — would consist of N_k photo feature vectors and $N_k - 1$ photo gap feature vectors. Let v represent a feature vector of either type (photo feature or photo gap feature). Thus, the feature vectors in PTS_k form the set $\{v_l\}, |\{v_l\}| = 2N_k - 1, 1 \leq l \leq N$.

From our definition of a photo taking session in the previous chapter, photos belonging to the same PTS exhibit feature similarities. In our approach, we model these similarities with a multivariate Gaussian distribution, parameterised by a multidimensional mean μ and a diagonal covariance matrix Σ, *i.e.* $P_k(v) = \mathcal{N}(v; \mu, \Sigma)$. With this model, we are able to capture nuances of the feature similarities in terms of the mean and covariance. This model is generative because given these two parameters, it can generate feature vectors corresponding to the PTS:

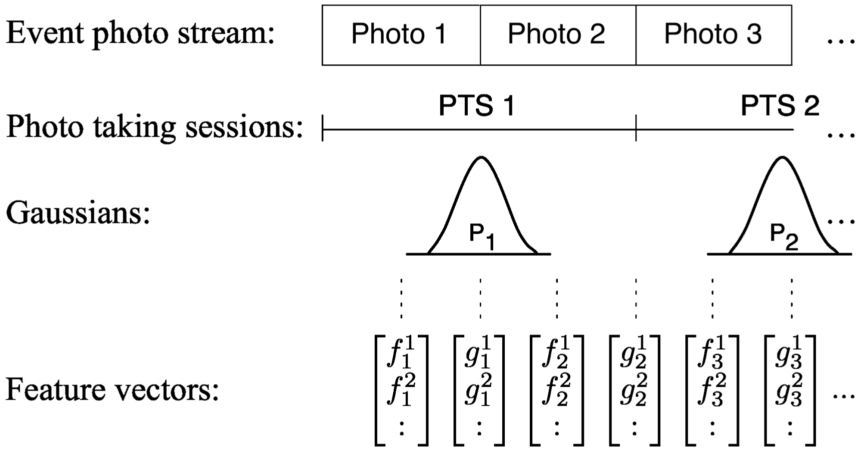


Figure 3.4: The event photo stream and its constituent photo taking sessions, can be modelled as a sequence of multivariate Gaussian distributions (P_k). The feature vectors shown consists of photo features (f_i^j) and photo gap features (g_i^j), where j is a feature index and i is a photo or photo gap index.

$$P_k(v) = \mathcal{N}(v; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} e^{\left\{-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)\right\}} \tag{3.2}$$

The event photo stream E, consisting of a sequence of M photo taking sessions, can then be modelled as a sequence of M multivariate Gaussian distributions: $\langle P_1, ..., P_k, ..., P_M \rangle$ (see Figure 3.4).

With this framework, the problem of finding the $M - 1$ segment boundaries in E is reduced to finding $\{P_k | \forall k, 1 \leq k \leq M\}$, that would best generate the sequence of feature vectors $\{v_l\}, |\{v_l\}| = 2N - 1$. In other words, we need to find:

1. The alignment between the sequence of P_k and the photos in E,
   i.e. $\{N_k | \forall k, 1 \leq k \leq M\}$

2. The parameters of P_k that would best generate the feature vectors in E,
   i.e. $\{P_k = \mathcal{N}(v; \mu, \Sigma) | \forall k, 1 \leq k \leq M\}$

The parameters for a multivariate Gaussian distribution, $P_k(v; \mu, \Sigma)$ can be estimated with:

$$\mu = \frac{1}{|2N_k - 1|} \sum_{l}^{|2N_k-1|} v_l \tag{3.3}$$

$$\Sigma = \frac{1}{|2N_k - 1|} \sum_{l}^{|2N_k-1|} (v_l - \mu)(v_l - \mu)^T \tag{3.4}$$

However, because we have M sets of parameters to estimate and we also need to find the best alignment for the M probability distributions, an expectation-maximization (EM) algorithm is required.

In the next section, we show how our problem of parameter estimation and alignment is equivalent to the training of a hidden Markov model (HMM). As such, we can use the Baum-Welch algorithm (Baum et al., 1970) to effectively find N_k and $P_k, \forall k, 1 \leq k \leq M$ and thus find the $M - 1$ segment boundaries in E.

3.5 The Hidden Markov Model

A hidden Markov model (HMM) is a finite state automaton with stochastic state transitions and observation emissions (Rabiner, 1989). An HMM assumes the process to be Markovian[2] and as such, computations with HMMs are very efficient. Even though a simple probabilistic model, the HMM is a well-developed tool for modeling observation sequences and has been successfully applied to tasks in domains such as speech recognition (Rabiner, 1989); text segmentation and topic detection (Mulbregt et al., 1998); and information extraction (Freitag and Mccallum, 1999).

3.5.1 Parameters of an HMM

Consider the HMM shown in Figure 3.5. An HMM is fully defined by the following four parameters:

[2] This refers to the memoryless property of a stochastic process where the conditional probability distribution of its next state depends only on its present state and not on the sequence of states before it.

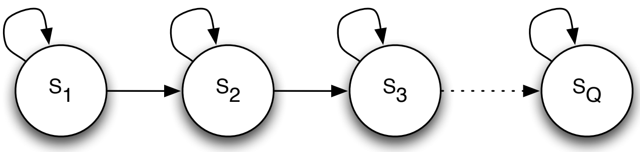


Figure 3.5: A hidden Markov model (HMM) with Q states

1. $Q = |S_i|$ — the number of states in the model

2. $A = \{a_{ij}\}$ — the state transition probability distribution,
   where $a_{ij} = P(q_{t+1} = S_j | q_t = S_i), 1 \leq i, j \leq Q$

3. $B = \{b_j(v_t)\}$ — the observation symbol probability distribution in state j,
   where $b_j(v_t) = P(v_t | q_t = S_j), 1 \leq j \leq Q$ and v_t refers to the feature vector (observation) at time t.

4. $\pi = \{\pi_i\}$ — the initial state distribution,
   where $\pi_i = P(q_1 = S_i), 1 \leq i \leq Q$

We shall use the standard compact notation $\lambda = (A, B, \pi)$ to represent the complete parameter set of an HMM, noting that Q can be derived from A, B, or π.

An HMM generates a sequence of observations, *e.g.* vectors of feature values, *i.e.* at time t, the HMM would generate v_t. The HMM generates the entire sequence of observations, $\langle v_1, v_2, ..., v_T \rangle$, by starting at one of its states according to its prior probability, π. In this state, an observation is generated according to the emission probabilities of the state, *i.e.* $b_j(v_1)$ for state S_j. The HMM then transitions to one of its states according to its state transition probabilities, A, which depends only on the current state[3]. After the transition, another observation is generated according to the emission probabilities of the new state. The process continues until all observations have been generated.

[3] This is true for a standard 1st order HMM with the Markov property.

3.5.2 The Three Basic HMM Problems

For any given HMM, there are three basic problems with known efficient solutions. We briefly review the three problems and their solutions here and show in the next section how solutions to the second and third problems are what we need to perform our parameter estimation and alignment in event photo stream segmentation.

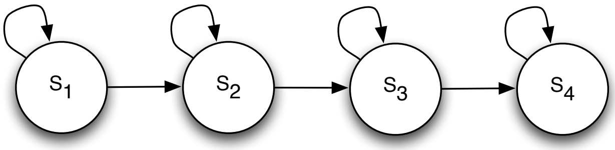
1. Given the observation sequence $O = \langle v_1, v_2, ..., v_T \rangle$ and HMM $\lambda = (A, B, \pi)$, what is the probability of the sequence given the HMM, $P(O|\lambda)$?
2. Given the observation sequence $O = \langle v_1, v_2, ..., v_T \rangle$ and HMM $\lambda = (A, B, \pi)$, what is the most probable state sequence $Q = \langle q_1, q_2, ..., q_T \rangle$ to generate O?
3. Given the observation sequence $O = \langle v_1, v_2, ..., v_T \rangle$, how do we adjust the model parameters $\lambda = (A, B, \pi)$ to maximize $P(O|\lambda)$?

For the first problem, since the alignment between state and observation is unknown, we compute the expected likelihood over all possible state sequences of length T. We thus find $P(O|\lambda)$ using the following marginalisation, which can be computed efficiently in $\mathcal{O}(Q^2T)$ using the forward-backward procedure (Rabiner, 1989):

$$\begin{aligned} P(O|\lambda) &= \sum_{Q} P(O|Q,\lambda)P(Q|\lambda) && (3.5) \\ &= \prod_{t=1}^{T} \sum_{i,j} P(v_t|q_t = S_j)P(q_t = S_j|q_{t-1} = S_i) && (3.6) \end{aligned}$$

The second problem is also known as the HMM decoding problem because we are trying to find the best (most probable) state sequence given the observation sequence and the HMM. This can be computed efficiently using a dynamic programming algorithm — the Viterbi algorithm (Rabiner, 1989).

The last problem is to adjust parameters of the HMM given the observation sequence. In other words, how do we train the HMM? For this, we have an efficient



		Destination State			
		S_1	S_2	S_3	S_4
Origin State	S_1	0.5	0.5	0	0
	S_2	0	0.5	0.5	0
	S_3	0	0	0.5	0.5
	S_4	0	0	0	1.0

Figure 3.6: An example of a Left-Right HMM with 4 states and its corresponding state transition matrix

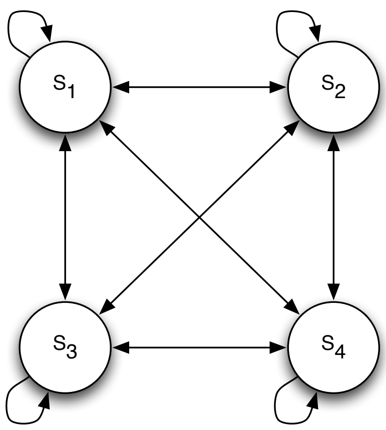
expectation-maximisation algorithm called the Baum-Welch algorithm (Rabiner, 1989).

3.5.3 HMM Structures

An HMM can have a variety of structures, depending on how many states it has, Q, and the transition matrix defined for those states, A. For example, Figures 3.6 and 3.7 show a Left-Right HMM and an Ergodic HMM respectively, along with their transition matrices.

A Left-Right HMM is an HMM where aside from self-transitions (transitions from a state to itself), all other transitions go from left to right. This structure has been used to model time series data where the state sequence of the generative process follows a particular order. For example in Figure 3.6, when the process is at State S_3, it will not transition to any of the lower-numbered states. This ordering is the reason for the name, Left-Right HMM. Variations of this structure have been used in the speech community to model phonemes.

An Ergodic HMM is an HMM where a transition with a non-zero probability is defined for every possible pair of states. In other words, every state is reachable



		Destination State			
		S_1	S_2	S_3	S_4
Origin State	S_1	0.25	0.25	0.25	0.25
	S_2	0.25	0.25	0.25	0.25
	S_3	0.25	0.25	0.25	0.25
	S_4	0.25	0.25	0.25	0.25

Figure 3.7: An example of an Ergodic HMM with 4 states and its corresponding state transition matrix

from every other state (including itself). In Figure 3.7, we see that for 4 states, we have $\binom{4}{2} + 4$ transitions. In this structure, the generative process can be at any of the states at any point in time, but with possibly different probabilities.

In some works, the structure of the HMM is found using a randomized search strategy based on the Markov chain Monte Carlo (MCMC) algorithm (Xie et al., 2002). In others, the structure is hand-crafted based on domain knowledge (Freitag and Mccallum, 1999). In our work, we have adopted the latter approach to make the most out of our domain knowledge.

Some works have expanded on the basic HMM structure and proposed more sophisticated models like the hierarchical HMM (Xie et al., 2002) and the coupling of several HMMs (Brand, 1997).

3.6 HMM for Event Photo Stream Segmentation

To model an event photo stream with an HMM, consider the following semantics for the HMM. Let the HMM states represent PTSes such that transitions between states correspond to transitions between PTSes. In other words:

1. $Q = |S_i|$ — the number of states in the model
   corresponds to the number of PTSes in the event photo stream

2. $A = \{a_{ij}\}$ — the state transition probability distribution
   corresponds to the PTS transition probability

3. $B = \{b_j(v_t)\}$ — the observation symbol probability distribution in state j
   corresponds to the multivariate Gaussian distribution for PTS_j.

4. $\pi = \{\pi_i\}$ — the initial state distribution
   corresponds to the initial PTS distribution

With these semantics, we can use an HMM for event photo stream segmentation as follows:

1. Using the Baum-Welch algorithm, we train the HMM using the sequence of feature vectors from the event photo stream as the sequence of observations. The Baum-Welch algorithm will find the HMM parameters that will best generate the event photo stream[4].

2. Using the Viterbi algorithm, we can find the best (most probable) PTS (state) sequence to generate the event photo stream, given the HMM parameters found using the Baum-Welch algorithm.

3. With the best PTS sequence, we obtain the best alignment between photo and PTS. For example, if the best PTS sequence obtained from an event photo stream of 10 photos is $\langle PTS_1, PTS_1, PTS_1, PTS_2, PTS_2,$ $PTS_3, PTS_3, PTS_4, PTS_4, PTS_5\rangle$, then we know the alignment between the photos and the PTSes: The first three photos belong to PTS_1, the following two photos belong to PTS_2, and so on.

4. With the best alignment, we finally obtain the location of the segment boundaries, *i.e.* the location where adjacent photos belong to different PTSes.

In this simple application of an HMM for event photo stream segmentation, the HMM states generate feature vectors of a single type, *i.e.* for any two feature vectors v_l and v_k generated by the HMM, $\|v_l\| = \|v_k\|$ and the corresponding elements in v_l and v_k are of the same feature type, *e.e.* if the first element in v_l is an aperture diameter value, then the first element in v_k is also an aperture diameter value. Having all the HMM states generate feature vectors of the same type is typical of a standard HMMs.

In our case, since an event photo stream is comprised of alternating feature vector types (see Figure 3.2), we have to coalesce each pair of photo feature vector and photo gap feature vector into a single feature vector (see Figure 3.8). This simplification causes several issues which we will discuss in the next section.

[4] The solution found is at a local maxima.

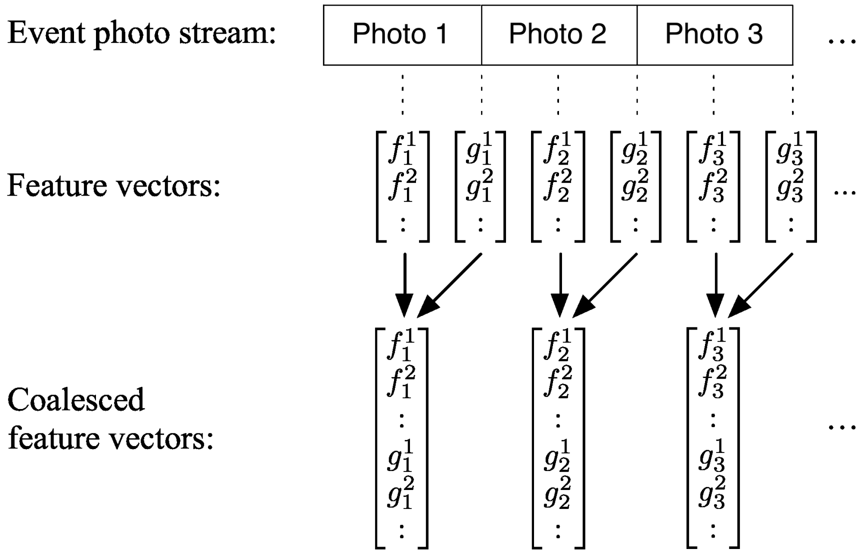


Figure 3.8: To simplify the feature vectors for the HMM, we coalesce each pair of photo feature vector and photo gap feature vector into a single feature vector.

3.7 Preliminary Models

We present several preliminary models in this section, neither of which is used in our final method pipeline (to be described in Section 3.12). These models are described here for completion and because they contribute to our understanding of using an HMM for event photo stream segmentation, as we analyze the shortcomings of each preliminary model. Together, these preliminary models illustrate the evolution of our approach from a simple Left-Right HMM up to its final form as an HMM with alternating observation types.

3.7.1 Left-Right HMM

We can use the Left-Right HMM (see Figure 3.6) for event photo stream segmentation by following the semantics described in the previous section, *i.e.* we model each PTS with a separate HMM state.

Because each PTS is modelled by a state, the HMM has as many states as there are PTSes in the event photo stream. Each state can transition to itself, producing a sequence of photos of the same PTS, or to the next state, marking a transition to the next PTS.

One disadvantage of the Left-Right HMM is that we cannot know *a priori* the number of states in the HMM, Q, because it relies on the number of PTSes in the event photo stream. This is similar to the problem of determining the number of clusters, k, in using the k-means clustering algorithm and we can adopt similar strategies to find Q, *e.g.* by finding Q that balances the complexity of the model (number of HMM parameters) and the goodness of fit (log likelihood of the observations given the HMM).

3.7.2 Ergodic HMM

While strategies exist to determine the number of states for the Left-Right HMM, the resulting complexity from having too many parameters — as a result of having the number of states equal to the number of PTSes — will aggravate data sparsity issues in training the parameters.

To resolve this issue, we can use an Ergodic HMM instead, taking advantage of our observation that some PTSes produce photos with similar features.

With an Ergodic HMM (see Figure 3.7), each state now corresponds to a canonical type of PTS, representing a group of PTSes that exhibit similar features. While we still have to find the number of states in this Ergodic HMM, the search space is smaller than finding the number of states in the Left-Right HMM, especially when the event photo stream has many PTSes.

3.7.3 Boundary HMM

With the Left-Right or Ergodic HMMs, we observe that when the model transitions from one PTS to another, the associated time gap boundary, *i.e.* the time gap at the boundary between PTSes, should not be an observation of either PTSes. The time gap boundary is merely an artefact of transitioning between PTSes. Consider the example in Figure 3.9. Time gaps tg_1, tg_2, and tg_3 occur in PTS_1. Time gaps tg_5

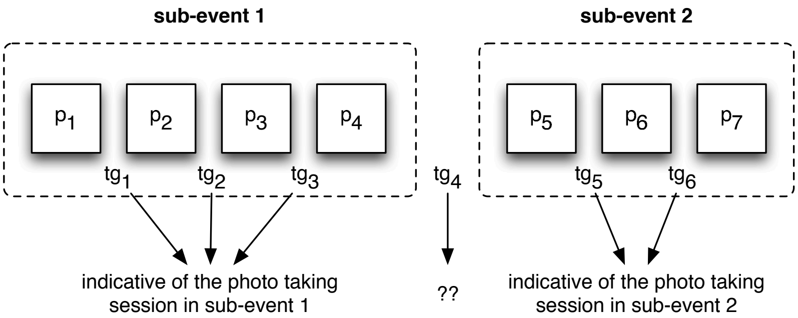

sub-event 1
sub-event 2
p1
p2
p3
p4
p5
p6
p7
tg1
tg2
tg3
tg4
tg5
tg6
indicative of the photo taking session in sub-event 1
??
indicative of the photo taking session in sub-event 2

Figure 3.9: While tg_1, tg_2, and tg_3 are indicative of the PTS in sub-event 1 and tg_5 and tg_6 are indicative of the PTS in sub-event 2, the time gap boundary tg_4 is indicative of neither PTS.

and tg_6 occur in PTS_2. The time gap boundary tg_4 however, is in neither PTS. So while tg_1, tg_2, and tg_3 are indicative of the state corresponding to PTS_1 and tg_5 and tg_6 are indicative of the state corresponding to PTS_2, the time gap boundary tg_4 corresponds to neither PTS and is indicative of neither state. As such, when time gaps are used to model PTSes in the Ergodic HMM, the states will incorrectly use time gap boundaries as samples to train the multivariate Gaussian distributions.

From this observation, the Left-Right and Ergodic HMMs can not correctly handle time gap information and a new HMM topology is needed to properly model time gaps at sub-event boundaries (PTS transitions). We introduce a new HMM structure, which we term a boundary HMM, that model PTS transitions as separate HMM states. We refer to these states as boundary states and the previously defined states as PTS states.

Note that as Figure 3.9 illustrates, this model only makes sense for photo gap features such as time gap and not for the coalesced feature vectors described in Figure 3.8 because we would be incorrectly aligning (attributing) photo features to a boundary state.

To model PTS transitions, one boundary state needs to be positioned between every pair of PTS states so that the HMM is forced to transition into the boundary state before transitioning into the other PTS state. So for 3 PTS states, we need 6 boundary states ($PTS_1 \rightarrow PTS_2$, $PTS_1 \rightarrow PTS_3$, $PTS_2 \rightarrow PTS_1$,

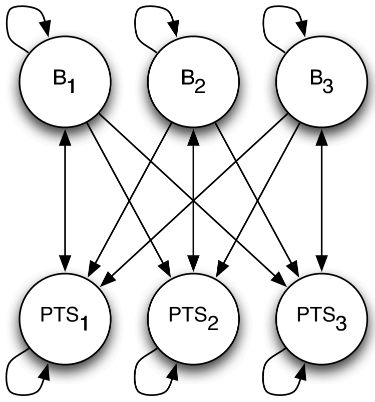


Figure 3.10: Boundary hidden Markov model for an event photo stream

$PTS_2 \rightarrow PTS_3, PTS_3 \rightarrow PTS_1, PTS_3 \rightarrow PTS_2$). The boundary state for each pair of PTS states needs to be distinct because the values that can constitute a time gap boundary depend on the distribution of the time gaps in the sub-event before as well as sub-event after the boundary. This is akin to the sliding window adaptive threshold methods reviewed in Chapter 2. However, to simplify the model and to reduce the effects of data sparseness, we only used one boundary state for each PTS state, not each pair. The structure of our boundary HMM is shown in Figure 3.10. A PTS state can only transition into one boundary state, but a boundary state can transition into any PTS state. So in effect, any PTS can transition to any PTS. The boundary states also have self loops so that the HMM can produce sub-events with a single photo.

3.7.4 Interweaved HMM

All three preliminary HMM models we have discussed so far have their own shortcomings. The Left-Right HMM has parameters that scale linearly with the number of photo taking sessions which is also unknown to us. Both the Ergodic and Boundary HMMs alleviates this problem but to avoid feature alignment issues (see Figure 3.11), coalesced feature vectors cannot be used and the Ergodic HMM should only be used with photo features and the Boundary HMM with photo gap features.

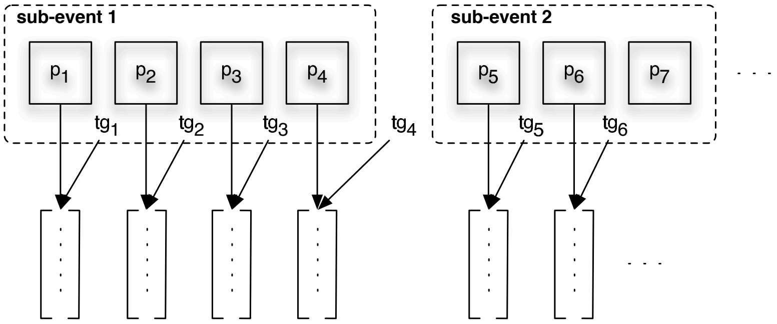


Figure 3.11: Forced alignment coalesces all feature types into a single vector for each photo, causing problems for the Ergodic HMM. The Boundary HMM suffers from a similar issue.

The primary issue with this forced alignment is in estimating the Gaussian parameters of the HMM states. When the time gap feature is used for the Ergodic HMM, time gap values that correspond to sub-event boundaries (*e.g.* tg_4 in Figure 3.11) will be aligned to the sub-events before the time gaps and erroneously used to estimate the Gaussian parameters for those sub-events. The problem also exists when we use context or visual features with the Boundary HMM. Feature values corresponding to photos near to sub-event boundaries will be aligned to boundary states instead of PTS states in the boundary HMM.

On the other hand, avoiding the forced alignment and not coalescing feature types means that we cannot make use of all the available features, which is just as unacceptable: when using the Boundary HMM, we cannot use context and visual features. When using the Ergodic HMM, we cannot use time gaps. We thus need a way to benefit from both models, but yet have each model use only the features that its designed for. In the literature (Brand, 1997), there are several ways to combine HMMs, depending on the type of coupling between the combined HMMs (see Figure 3.12):

- *Linked HMMs:* There is coupling between the HMMs for every pair of synchronous states. This is equivalent to a Cartesian product HMMs with a bias probability on each joint state.

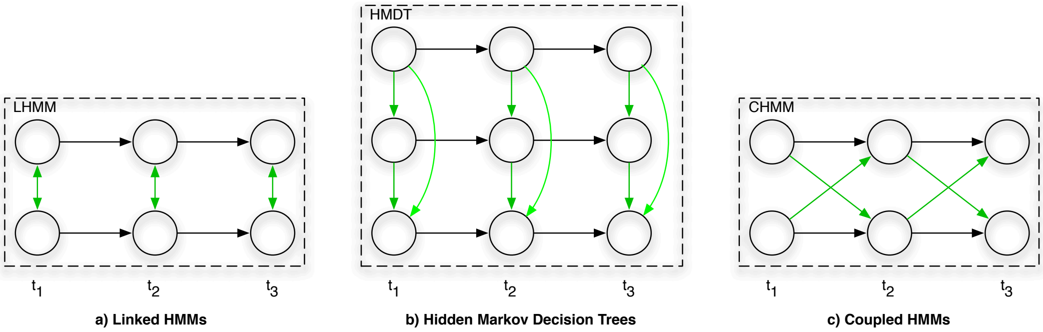


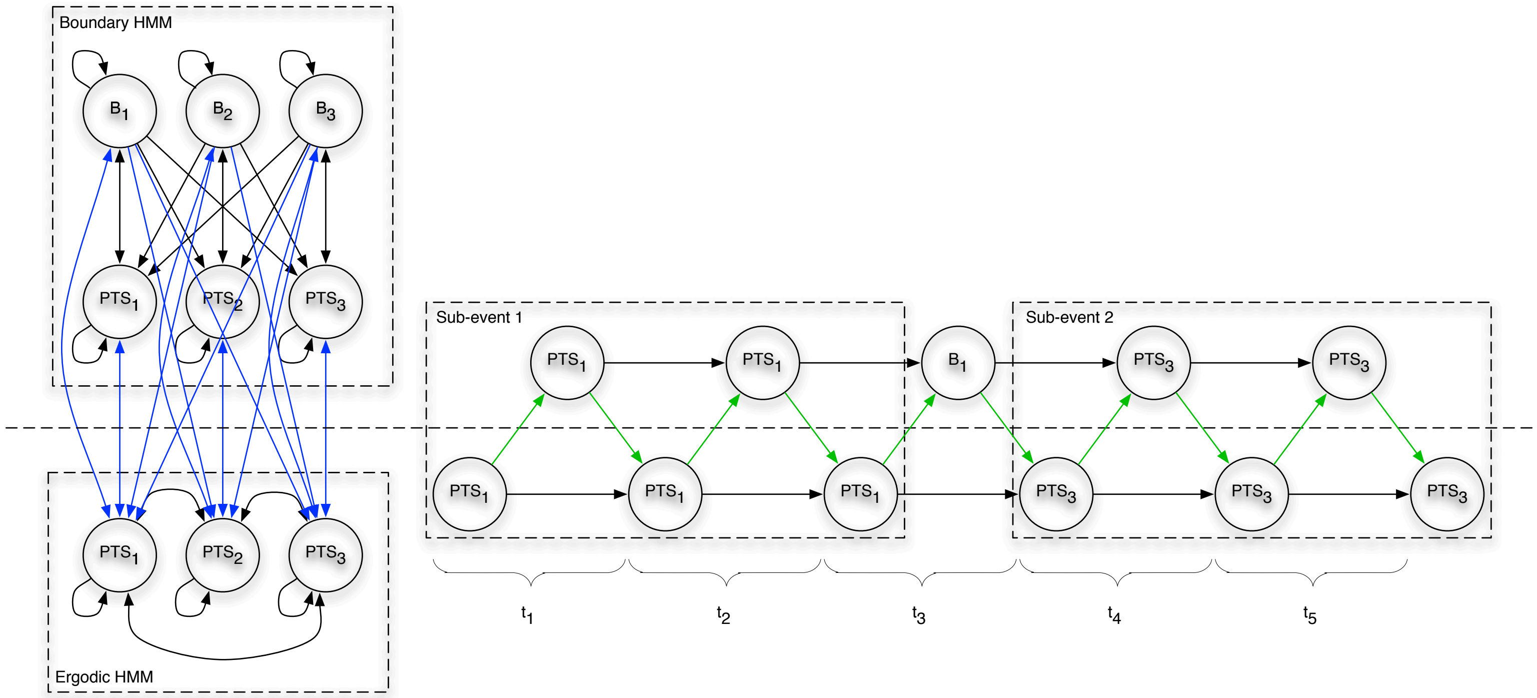
Figure 3.12: Varieties of couplings for the different ways of combining HMMs

- *Hidden Markov decision trees:* There is a cascade of synchronous conditional probabilities down an ordered hierarchy of HMMs. This is ideal when there are constraints from a "master" HMM that need to be imposed down the hierarchy.
- *Coupled HMMs:* The coupling between HMMs occur across time slices. This is appropriate for processes that influence each other asymmetrically and possibly causally.

None of the above methods are however, suitable for our case; the dependencies between the boundary and Ergodic HMMs occur both within time slices and across time slices. Consider the partial state trellis in Figure 3.13b. In the first time slice, $t = 1$, the PTS_1 state of the boundary HMM (D_{PTS_1}) is dependent on the PTS_1 state of the Ergodic HMM (E_{PTS_1}), *i.e.* if the probability that the Ergodic HMM is in PTS_1 at $t = 1$ is high, then the probability that the boundary HMM is in PTS_1 should be higher than the probability that the boundary HMM is in any other PTS state (i.e., PTS_2, PTS_3). Similarly, in the next time slice, $t = 2$, the PTS_1 state of the Ergodic HMM is dependent on the PTS_1 state of the boundary HMM. Dependencies for the remaining time slices follow similar reasoning.

Our combined HMM, which we term an Interweaved HMM, is shown in Figure 3.13a. In this figure, we can see that the dependencies are encoded as follows:

1. $E_{PTS} \rightarrow D_{PTS}$



a) Interweaved Boundary-and-Ergodic HMMs

b) Partial state trellis example with the interweaved HMMs

Figure 3.13: The figure in (a) depicts interweaved boundary and Ergodic HMMs. The double-headed arrow is a shorthand for transitions coming from and going to the two states. An example of using these interweaved HMMs can be seen by following the partial state trellis shown in (b). The dashed line separates states from the boundary HMM and ones from the Ergodic HMM.

$$\underset{\substack{\downarrow \\ \text{Posterior} \\ \text{probability of the} \\ \text{state sequences} \\ \text{in both HMMs}}}{P(S|O)} = \underbrace{P_{s_1^D}}_{\text{Prior}} \underbrace{p_{s_1^D}(o_1^D)}_{\text{1st output}} \underbrace{P_{s_1^E}}_{\text{Prior}} \underbrace{p_{s_1^E}(o_1^E)}_{\text{1st output}} \prod_{t=2}^{T} X$$

$$X = (\underbrace{p_{s_t^D}(o_t^D)}_{\text{outputs}} \underbrace{p_{s_t^E}(o_t^E)}_{\text{outputs}} \underbrace{P_{s_t^D|s_{t-1}^D}}_{\text{transition}} \underbrace{P_{s_t^D|s_t^E}}_{\text{coupling}} \underbrace{P_{s_t^E|s_{t-1}^E}}_{\text{transition}} \underbrace{P_{s_t^E|s_{t-1}^D}}_{\text{coupling}})/P(O)$$

Figure 3.14: Posterior probability of the state sequence of the Interweaved HMM

2. $E_{PTS} \rightarrow D_B$

3. $D_{PTS} \rightarrow E_{PTS}$

4. $D_B \rightarrow E_{PTS}$

The posterior probability of the state sequence in both HMMs can thus be computed according to the equation in Figure 3.14.

To efficiently solve the Interweaved HMMs (computing $P(O|\lambda)$, most probable state sequence, and model learning), we implemented an algorithm similar to the N-heads dynamic programming algorithm proposed for Coupled HMMs. This algorithm is a deterministic $O(T(CN)^2)$ approximation for MAP state estimation that samples the highest probability paths via expectation maximization (Brand, 1997).

With this framework, we can combine any number of HMMs together. In our case, we chose to combine one boundary HMM and one Ergodic HMM. Alternatively, we can also combine one boundary HMM and several Ergodic HMMs, one for each context / visual feature.

In our experiments however, this HMM structure produced poor results and most of the time, the log likelihood during parameter learning did not converge and strayed to negative infinity instead. We suspect that the complexity of the

HMM structure due to the number of parameters that had to be learned makes the parameter estimation problem intractable given the data sparsity problem of our task.

3.8 HMM with Alternating Observation Types

Here we describe the model we use in our final method pipeline. While the Interweaved HMM solves the feature vector alignment problem by having one HMM for each feature vector type, *i.e.* the Ergodic HMM for photo features and the Boundary HMM for photo gap features, the model described in this section solves the problem by modelling alternating observation types in a single HMM. This solves the alignment issues of the Ergodic and Boundary HMMs; and since we only have a single HMM, the parameter estimation is also much simpler and more tractable than for the Interweaved HMM.

To find which gaps between consecutive photos correspond to segment boundaries, this approach takes the view that an event photo stream is the result of a stochastic process that generates feature vectors, consisting of a set of foreground and background models. The foreground models generate the feature vectors that we want to find, *i.e.* the photo gap feature vectors corresponding to segment boundaries. The remaining models are background models that generate the surrounding feature vectors, *i.e.* photo feature vectors or photo gap feature vectors that do not correspond to segment boundaries.

To generate the event photo stream, the process emits alternating photo feature and photo gap feature vectors from the background models. At some point, the process switches to a foreground model at a segment boundary before switching back to a background model. This process continues until the end of the event photo stream (see Figure 3.15).

In this process, feature vectors in each photo taking session is generated by a pair of background models: one background model for photo features and another

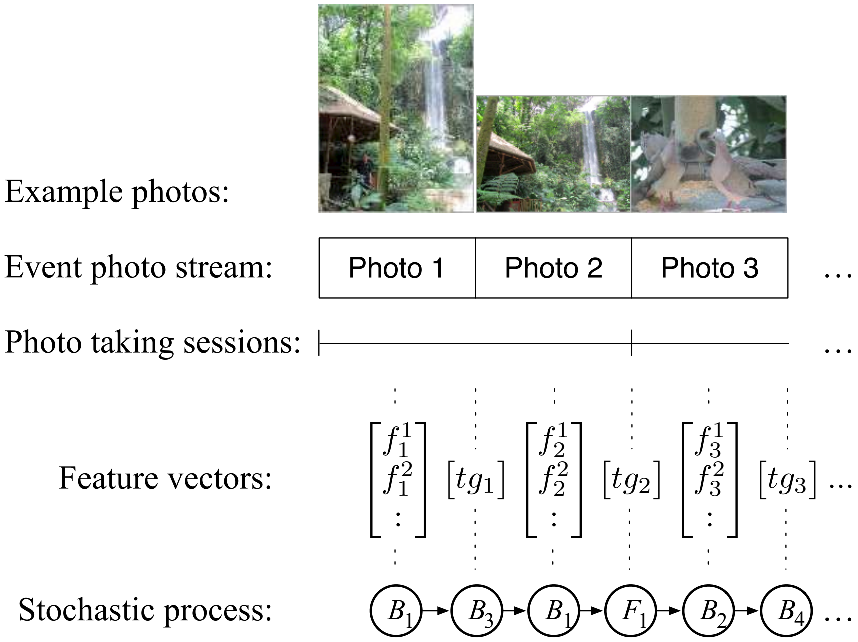


Figure 3.15: Our model views an event photo stream as the result of a stochastic process consisting of a set of foreground and background models. In the above, the first photo taking session consists of two photos. The time gap, tg_2, corresponding to the segment boundary between photo 2 and photo 3, is generated by the foreground model, F_1, of the stochastic process. The remaining models shown are the background models, B_i.

for photo gap features. For example in Figure 3.15, feature vectors in the photo taking session consisting of photos 1–2 are generated by the pair B_1 and B_3. This pair could generate feature vectors for other photo taking sessions in the stream. Suppose feature vectors in the photo taking session consisting of photos 6–10 are also generated by B_1 and B_3. The feature vectors in the two photo taking sessions, *i.e.* photos 1–2 and photos 6–10, would then follow the generated feature distributions of B_1 and B_3. For example, the feature distributions can be indicative of photos that are taken a few seconds apart, under good lighting conditions, at a medium distance from the participants, with a similar background view, etc. Similarly, other photo taking sessions are generated by other pairs of background models with their own feature distributions.

With our concept of foreground and background models, the simplest HMM structure consists of three states: two states for the pair of background models and one state for the foreground model. This 3-state HMM is shown in Figure 3.16a. For two or more pairs of background models, we can use the 3-state HMM as a basic building block to form larger HMMs. Figure 3.16b shows an HMM with two pairs of background models: (B_1, B_3) and (B_2, B_4).

Since the event photo stream consists of alternating feature types, our HMM has two types of states to generate each of the feature types. In Figures 3.16a and b, only states B_1 and B_2 generate photo features. The remaining four states generate photo gap features. Of these four states, F_1 and F_2 are the foreground models that generate photo gap features corresponding to segment boundaries. All states model their emissions with a single Gaussian distribution per dimension to simplify parameter estimation. With the state transitions in this structure, the HMM will alternatingly transition from a photo feature state to a photo gap feature state, thus generating alternating photo and photo gap feature vectors.

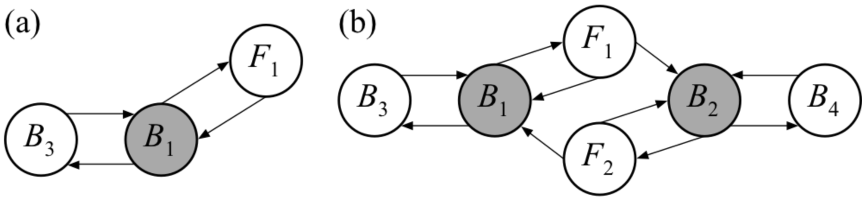


Figure 3.16: Grey HMM states generate photo features, while white HMM states generate photo gap features. States F_1 and F_2 represent foreground models that generate feature vectors corresponding to segment boundaries. States B_i represent background models that generate the surrounding feature vectors. The HMM in (a) has one pair of background models while the HMM in (b) has two pairs.

3.9 Feature and HMM Structure Analysis

Using our final model, the HMM with alternating observation types, we experimented with a wide variety of features to model PTSes, drawn from three features types: temporal, context, and visual.

Temporal. Like many inter-event photo organization methods discussed in Chapter 2, we employ time gaps in our study. We believe that time gap is a very important feature because unlike other features, time gap is not a feature *of* a photo, but *in between* photos, making it an excellent reflection of how PTSes change from one sub-event to another.

Context. The Exchangeable Image File Format (Exif) specifies the camera parameters stored in the image file of a digital photo (JEITA, 2002). Some of these parameters provide context information on how the photo was taken. For example, focal length is related to the camera's optical zoom and determines the magnification at which distant objects appear in the photo; depth of field affects the distances which objects would appear sharp in the photo and is increased with increasing aperture diameter, which measures the size of the opening through which light enters the camera. We believe that these parameters are features indicative of PTSes. In our study, we employ three context features: focal length, aperture diameter and the LogLight metric, a measure of the ambient light in an image (Sinha and Jain, 2008), derived from the exposure time, aperture area, ISO speed rating,

Type	Name		# Dims
Temporal	TG	Time Gap	1
Context	FL	Focal Length	1
	AD	Aperture Diameter	1
	LL	LogLight	1
Visual	CH	Color Histogram	8
	GA	Gradient Direction Autocorrelogram	16
	SD	SIFTdiff	1

Table 3.1: Feature Types

as well as focal length.

Visual. We evaluate three visual features in our approach. The first two are color histogram and gradient direction autocorrelogram. According to a recent study in similarity-based photo organization on a 2D virtual canvas (Strong and Gong, 2009), the combination of these two features performs best at low dimensions, an important criteria given our data sparseness problem. For the third visual feature, we propose a measure of visual difference between consecutive photos based on SIFT (Lowe, 2004) we call *SIFTdiff*. A single value is computed for each pair of consecutive photos by averaging the Euclidean distances between the best matching keypoint pairs from the two photos.

To evaluate the segmentation results, we used the error rate metric, Pr_{error}, proposed by Georgescul *et al.* (Georgescul et al., 2006). This metric improves on *WindowDiff*, previously used by Naaman *et al.* (Naaman et al., 2004) to evaluate their automatic albuming method. A lower Pr_{error} indicates better agreement with the manually segmented ground truth; a score of 0 indicates perfect agreement. Pr_{error} is an average of the miss and false alarm rates. As such, a method that proposes no segment boundaries or proposes segment boundaries everywhere will have an error rate of about 0.5.

We collected 28 event photo streams of various event types (see Table 3.2), *e.g.*

Set	#Photos	Time span	Event	Source
1	301	4h 54m	wedding	Flickr
2	321	7h 37m	wedding	Flickr
3	260	6h 53m	wedding	Flickr
4	209	5h 50m	wedding	Flickr
5	94	24h 47m	celebration	C2
6	132	12h 9m		
7	209	10h 22m	travel	
8	135	8h 59m		
9	160	6h 55m		
10	188	9h 58m		
11	173	16h 27m	travel	
12	236	15h 8m		
13	125	13h 46m		
14	177	13h 25m	travel	
15	224	11h 47m		
16	105	5h 47m		
17	149	1h 29m	beach	
18	150	2h 14m	river	
19	363	8h 42m	concert	C3
20	195	9h 18m	travel	C4
21	117	13h 37m		
22	157	14h 36m		
23	162	2h 16m	travel	C5
24	214	4h 5m	zoo	
25	162	3h 15m	wedding	C6
26	131	8h 34m		
27	207	10h 23m	travel	C7
28	132	16h 34m	travel	
Mean	185.3	9h 38m	–	–
Median	167.5	9h 8m	–	–

Table 3.2: We collected 28 photo sets with a variety of event types. Note that the calculated medians and means shows that the duration of the photo sets is fairly long and the number of photos per set is fairly large.

wedding, travel, cruise, concert, etc. Four event photo streams are from publicly available Flickr photo sets[5]. The remaining 24 were obtained from seven volunteers. In total, our evaluation data set consists of 5188 photos, with an average and median of 185 and 168 photos respectively.

For the four streams from Flickr, the photo owners were not available to annotate the sets. As such, the first author manually segmented the photos to provide ground truth. For the remaining 24, we asked the contributors — as photo owners — to provide the ground truth. This practice is in line with many photo stream segmentation works we reviewed in Chapter 2, which also require ground truth for their evaluation.

To find the best feature combination, we enumerated all possible feature combinations from Table 3.1. For our HMM, we need to have at least one photo feature and one photo gap feature. Thus, with five photo features, we have $\sum_{i=1}^{5} \binom{5}{i} = 31$ combinations. With two photo gap features, we have 2 combinations. Together, they make for 62 different feature combinations for our HMM. We also enumerated over a range of possible number of HMM states. Since the number of states in our HMM is in multiples of 3, we searched in the space of $\{3, 6, 9, 12, 15\}$ states.

Our experiment is conducted as follows: for each set (28), we used our HMM with alternating observation types and iterated over all possible feature combinations (62) and for each feature combination, we iterated over the range of number of HMM states (5).

The feature combination ranking based on averaging the resulting Pr_{error} over the range of number of HMM states over all photo sets is shown in Table 3.3. Here we can see that of the two photo gap features, *SIFTdiff* and time gap (TG), only the latter appears in the top five feature combinations. We also note that the LogLight feature, which is a measure of scene brightness, appears in all the top five positions. Aperture diameter and color histogram also appears prominently. On the other hand, the gradient direction autocorrelogram, that has the most number of

[5] Flickr photo set ID: 847825, 1068265, 72157601961445922, and 72157603826353321.

dimensions amongst our list of features, *i.e.* 16 dimensions, occupies the bottom half of the list. Most notably, focal length is absent from the list. On further investigation, we found that the focal length values do not vary by much (low standard deviation) in our dataset, making it a poor feature for our approach.

The number of HMM states ranking based on average the resulting Pr_{error} over all possible feature combinations and all photo sets is shown in Table 3.4. The table shows that the best performing HMM is the one with 6 states. From this result, we looked at the feature combinations again and looked for the best feature combination for the HMM with 6 states. The resulting rank is shown in Table 3.5. From this table, we conclude that our HMM should have 6 states and it should use a feature combination that on hindsight, consists of simple features that work best under our task constraint of data sparsity[6]:

1. *Aperture Diameter* – a photo feature measuring the size of the opening through which light enters the camera
2. *LogLight* (Sinha and Jain, 2008) – a photo feature measuring the ambient light in an image
3. *Color Histogram* – a photo feature measuring the color distribution in an image, and
4. *Time gap* – a photo gap feature measuring the time difference between capture times of consecutive photos.

3.10 Smoothing HMM Parameters

In Section 3.6, we described how we are using the Baum-Welch and Viterbi algorithms as a means to solve the alignment and parameter estimation problem we outlined in Section 3.4. In this approach, we train the HMM, using the Baum-Welch

[6] In Section 5.3, we explain how our photo browser, CHAPTRS ver. 2, handles photos with missing features in its implementation of our event photo stream segmentation algorithm.

Rank	Feature Combination	Average Pr_{error}
1	LL, TG	0.317
2	AD, LL, TG	0.318
3	AD, LL, CH, TG	0.321
4	AD, LL, CH, GA, TG	0.322
5	LL, CH, GA, TG	0.323

Table 3.3: Ranking of feature combinations by averaging Pr_{error} over all number of states ($\{3, 6, 9, 12, 15\}$). See Table 3.1 for the description of each feature abbreviation.

Rank	Number of HMM States	Average Pr_{error}
1	6	0.534
2	3	0.544
3	9	0.554
4	12	0.572
5	15	0.583

Table 3.4: Ranking of number of HMM states by averaging Pr_{error} over all feature combinations. See Table 3.1 for the description of each feature abbreviation.

Rank	Feature Combination	Pr_{error}
1	AD, LL, CH, TG	0.255
2	AD, CH, TG	0.265
3	AD, LL, CH, GA, TG	0.265
4	LL, CH, GA, TG	0.268
5	AD, LL, GA, TG	0.271

Table 3.5: Ranking of feature combinations for HMM with 6 states. See Table 3.1 for the description of each feature abbreviation.

algorithm, with the feature vectors from the given event photo stream because we want to find the best parameters to generate these very feature vectors.

As there is only one such event photo stream, the data sparsity problem ensues. As a generative model, an HMM typically needs to be trained with large amounts of data. A possible alternative would be to train the PTS states individually. This however, requires training data for each PTS state, which we also do not have because that would require collecting a large number of event photo streams, finding their ground truth segmentation, and labelling each PTS.

To resolve this situation, we turn to smoothing as a way to alleviate data sparsity by interpolating (smoothing) the parameters learnt from the given event photo stream with parameters learnt from a large number of event photo streams. Smoothing essentially allows us to account for probabilities of missing observations, observations that did not occur in the given event photo stream.

In automatic speech recognition (Lee, 1989), smoothing has been used to alleviate data sparsity issues associated with lack of data for speaker-dependent speech recognition. The HMM parameters are smoothed with those learnt from a speaker-independent dataset. More recently (Freitag and Mccallum, 1999), a smoothing method called shrinkage is also used to alleviate data sparsity issues with HMM parameter learning.

The smoothing method we adopt is the deleted interpolation method (Jelinek and Mercer, 1980) used in automatic speech recognition. Deleted interpolation works similar to k-fold cross-validation where the dataset is divided into k equal-sized portions and one portion is used to learn the smoothing coefficients (coefficients for interpolation) while the remaining $k-1$ portions are used as the smoothing HMM parameters to be interpolated with the HMM parameters learnt from the given event photo stream. This is repeated k times and the final smoothing coefficients is simply an average of the k sets of coefficients.

3.11 Filtering Spurious Solutions

So far, we have addressed the problem of data sparsity for the HMM. Another common pitfall for HMM-based methods has to do with the parameter initialization. For learning HMM parameters, the Baum-Welch algorithm is an efficient EM algorithm that finds a local maximum in the solution space. It does not guarantee that this solution is the best solution. The common practice is to iterate a handful of times and each time, initialize the parameters differently. The idea is to explore more of the solution space and possibly find different local maximums. Since the structure and thus the complexity of the HMM is the same, the only difference between the different iterations is the initial parameters for the HMM. So, the better estimate for the best solution is simply the solution that corresponds to the best of all the found local maximums.

Unfortunately, some of these solutions may be spurious solutions, *i.e.* they do not provide good HMM parameters even though the log likelihood is the local maxima (or even the best of several local maximums). This can be caused by a variety of factors but mainly due to the HMM parameters overfitting the training data.

To filter out spurious solutions, we check the solutions for indications of overfitting by looking at the state distribution, *i.e.* the number of times each state was visited in the state sequence of the solution. We assumed that an acceptable solution is one where:

1. For each feature type (photo feature and photo gap feature), the number of visits to the background states are balanced, *i.e.* in the HMM depicted in Figure 3.16b, $\frac{|B_1|}{|B_2|} \approx 1$ and $\frac{|B_3|}{|B_4|} \simeq 1$.

2. The pairs of background states are positively correlated, *i.e.* $\frac{|B_1|}{|B_3|} = k\frac{B_2}{B_4}$, where k is a positive real number.

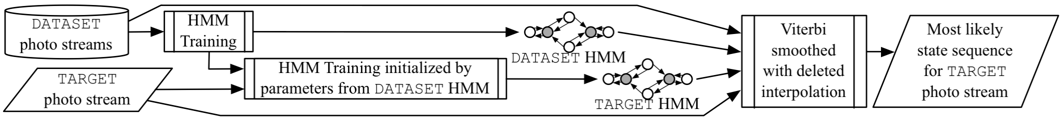

DATASET photo streams
TARGET photo stream
HMM Training
HMM Training initialized by parameters from DATASET HMM
DATASET HMM
TARGET HMM
Viterbi smoothed with deleted interpolation
Most likely state sequence for TARGET photo stream

Figure 3.17: We use a separate set of event photo streams (DATASET) to alleviate data sparsity in the event photo stream we want to segment (TARGET). All photo streams are unlabelled and unsegmented. The four inputs are needed to perform the Viterbi algorithm with deleted interpolation (Lee, 1989; Jelinek and Mercer, 1980).

3.12 Final Pipeline

Having gone through various aspects of our approach: features, structure, training, smoothing, and filtering of spurious solutions, in this section we outline the entire process.

Let us refer to the given event photo stream we want to segment as the TARGET photo stream. This photo stream is unlabelled and unsegmented. Let us then refer to the training data of unlabelled, unsegmented event photo streams as the DATASET photo streams. We note that while the term "training data" typically implies that the data is labelled, that is not the case here. We refer to this data as training data because it is used to train the parameters of the HMM.

First (see Figure 3.17), an HMM is trained using the DATASET photo streams. We call this the DATASET HMM. Parameters from this HMM is then used to initialize the parameters of a second HMM, the TARGET HMM, which is trained with the TARGET photo stream. In its training, the TARGET HMM parameters converge when they maximize the TARGET HMM's probability of generating the TARGET photo stream feature vectors. To determine the TARGET HMM's state sequence in generating the given feature vectors with maximum probability, we use the Viterbi algorithm (Rabiner, 1989) with deleted interpolation, a smoothing technique that finds the smoothing parameters between two distributions depending on how well-trained each distribution is. We use deleted interpolation, as is typical in speech recognition (Lee, 1989), to alleviate data sparsity by smoothing the parameters of the TARGET HMM with parameters from the DATASET HMM, which was trained with much more data. Deleted interpolation is a slow process and the execution time of our method is primarily spent on this step. In Section 5.3, we outline several practical optimizations implemented by our photo browser, CHAPTRS ver. 2, to alleviate this issue. Using the evaluation data set described in the next section, the average execution time was reduced from 134.9 seconds to 1.9 seconds.

Finally, with the state sequence we can determine which photo gap feature vectors were generated by the foreground models, and hence correspond to segment boundaries. A more detailed description is outlined in Figure 3.18.

3.13 Evaluation and Analysis

In our evaluation, we assess the usefulness of our approach extrinsically, by measuring its performance for event photo stream segmentation. We hope to answer two primary questions:

1. *Does modeling PTSes help event photo stream segmentation?* This validation is the primary goal of our evaluation. Favorable results would indicate that PTSes do correlate with moments in the event and that these PTSes can be modeled from the consistencies within sub-events.

2. *How do existing methods (including automatic albuming methods) perform for event photo stream segmentation?* In the introduction of this thesis in Chapter 1, we argue that the task of event photo stream segmentation and the task of automatic albuming are different, with the former being more challenging due to issues of data sparsity, indistinct time gaps, and visual similarities. We explore the validity of our argument by applying existing automatic albuming methods and comparing their performance for our task.

As baselines (see Table 3.6), we have implemented the cluster tree event photo stream segmentation algorithm (Graham et al., 2002) and five automatic albuming algorithms from Chapter 2: fixed threshold (Platt, 2000), best-first model merging (Platt et al., 2003), adaptive threshold (Platt et al., 2003), K-means (Loui and

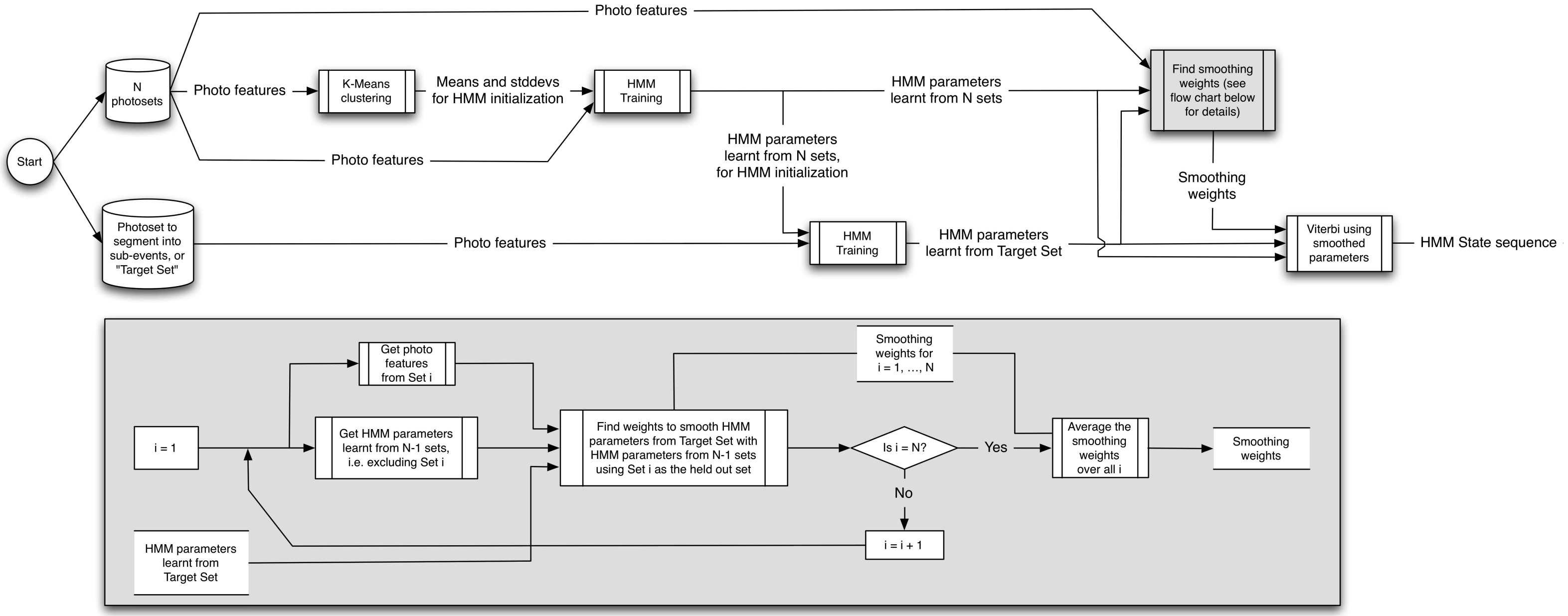


Photo features
N photosets
Photo features
K-Means clustering
Means and stddevs for HMM initialization
HMM Training
HMM parameters learnt from N sets
Find smoothing weights (see flow chart below for details)
Start
Photo features
HMM parameters learnt from N sets, for HMM initialization
Smoothing weights
Photoset to segment into sub-events, or "Target Set"
Photo features
HMM Training
HMM parameters learnt from Target Set
Viterbi using smoothed parameters
HMM State sequence
Get photo features from Set i
Smoothing weights for i = 1, ..., N
i = 1
Get HMM parameters learnt from N-1 sets, i.e. excluding Set i
Find weights to smooth HMM parameters from Target Set with HMM parameters from N-1 sets using Set i as the held out set
Is i = N?
Yes
Average the smoothing weights over all i
Smoothing weights
No
HMM parameters learnt from Target Set
i = i + 1

Figure 3.18: Complete pipeline of our automatic event photo stream segmentation method

Baseline Method	Feature Used
Fixed threshold (Platt, 2000)	Time gap
Adaptive threshold (Platt et al., 2003)	Time gap
Cluster Tree (Graham et al., 2002) (event photo stream segmentation)	Time gap
K-means (Loui and Savakis, 2003)	Time gap
Event ending probability (Zhao et al., 2006)	Time gap
Best-first model merging (Platt et al., 2003)	Color histogram

Table 3.6: Baseline Methods

Savakis, 2003), and event ending probability (Zhao et al., 2006). These baselines provide us with a variety of methods for comparison: heuristic, probabilistic, hierarchical, visual-based, and the state-of-the-art event photo stream segmentation algorithm.

For the best-first model merging baseline, we used the number of sub-events in the ground truth as the threshold for its termination condition, a necessary parameter for this method. While this gives this baseline an unfair advantage, as we shall see later, the baseline still does not perform very well.

As the evaluation metric, we used the error rate metric, Pr_{error}, just as we did in Section 3.9 for our feature and HMM structure analysis. We also used the same dataset here. Results are shown in Figure 3.19.

1. Does modeling PTSes help event photo stream segmentation? — Our method had the lowest error rate overall. Our method (with smoothing and filtering) is statistically significantly better than all the baseline methods ($p < 0.05$). All versions of our method have the lowest miss rate among all methods we studied, but the highest rate of false alarms. Looking at the figure however, our method gives the most balance between misses and false alarms. Furthermore, we believe that for end users, having a low miss rate is more valuable than having a low false alarm rate. To correct a false alarm is a one-step process of removing the incorrect

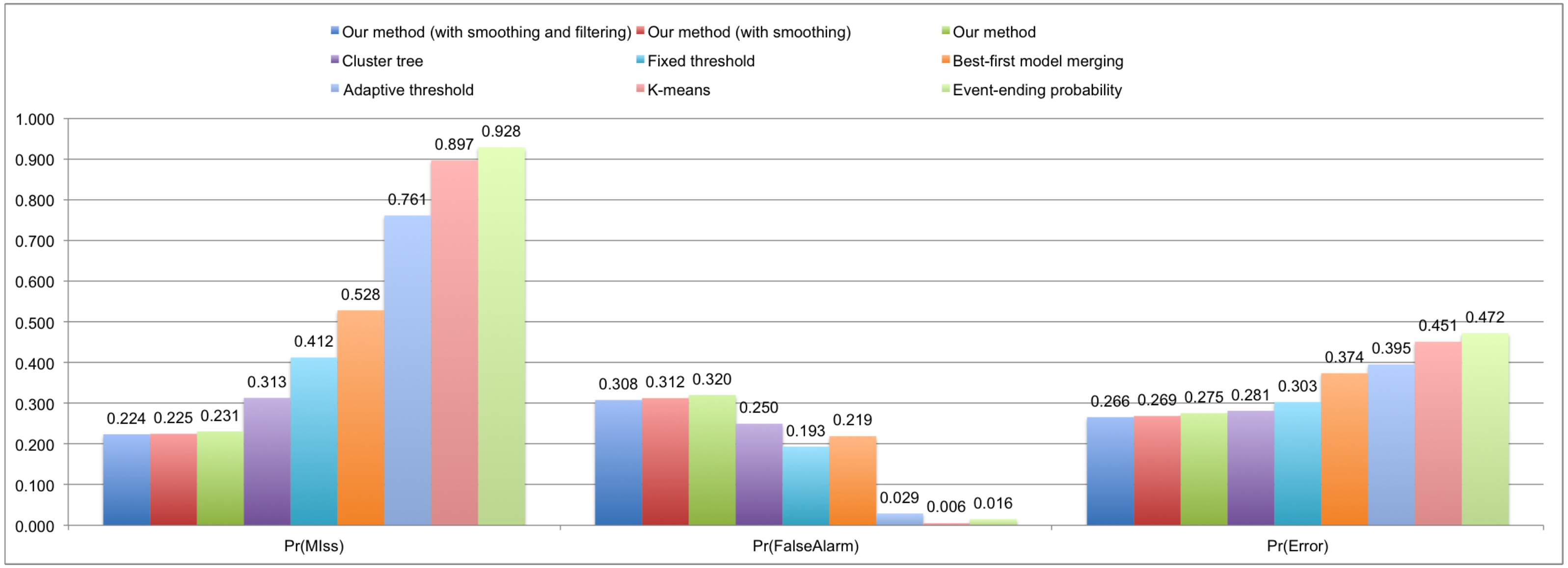


Figure 3.19: Comparison between our method and the baselines, averaged over all event photo streams, in terms of miss rate, false alarm rate, and error rate, against ground truth segmentations (smaller numbers / shorter bars are better)

segment boundary. But to correct a miss, the user must first realize that there is a miss, then figure out the position of the segment boundary.

Why does our method produce more false alarms? We believe it is produced during the Viterbi algorithm when the HMM — with its trained parameters — incorrectly finds that transitioning to a foreground model (*e.g.* transitioning from B_1 to F_1 in Figure 3.16b) has a higher probability than transitioning to a background model (*e.g.* B_3). One possible reason for the lower probability is the lack of training data for the feature vectors corresponding to the false alarms. A more likely reason is however, the lower accuracy associated with training the HMM without labelled data. Nonetheless, the error rate was computed by penalizing misses and false alarms equally. In this regard, our method outperformed all the baselines.

Table 3.7 shows a more detailed description of how our method's performance (with smoothing and filtering) compares with the best baseline for each method, as measured by Pr_{error}. Our method performed better than the best baseline for 22 of the 28 photo sets in our dataset. We also show the number of photos and the number of sub-events (as provided by the ground truth) for each set, to show that there is no pattern related to photo set size or number of sub-events in the six sets in which our method performed the least. Instead, we found that the low performance of our method in these six sets are primarily caused by a mismatch between the photo owner's subjective segmentation preference based on the semantics of the event and the segmentation that can be derived from the available features of the event photo stream.

We observe cases where our method produces boundaries at locations where the time gap is large and/or there are color differences in the adjacent photos. Nevertheless, these boundaries are incorrect according to the ground truth. For example, the lowest performing set, Set 16[7], actually only had 8 errors in total: 4 false alarm errors 4 miss errors. Of the 4 false alarm errors, 2 have large time gaps (166

[7] We have obtained permission from the photo owners to include some of their photos in this thesis.

seconds and 193 seconds in the 3rd and 4th error in the figure) and color differences that caused our method to produce the incorrect sub-event boundaries (see Figure 3.20). For the remaining 2 false alarm errors, the color differences, but not the time gap values, caused the incorrect sub-event boundaries. For the 4 miss errors, while 3 are legitimate errors, the 4th miss error is purely a matter of subjective preference, as the time gap is only 19 seconds apart and the adjacent photos look very similar to each other, as can be seen in Figure 3.21.

2. How do existing methods (including automatic albuming methods) perform for event photo stream segmentation? — The best baseline is the state-of-the-art cluster tree event photo stream segmentation algorithm. The best-first model merging method which utilizes visual information alone did not perform well and ranked fourth place. This was caused by a relatively high miss rate, suggesting that visual similarities amongst the photos hinder the method from finding any segment boundaries. The adaptive threshold method which is a simple and well-known automatic albuming method, performed worse than the simplest baseline — the fixed threshold method — when used to segment event photo streams. Methods that rely on heuristics such as the K-means and the event ending probability methods performed the worst, finding very few segment boundaries, resulting in very high miss rates and correspondingly high error rates.

3.14 Conclusion

To help make large event photo streams more manageable, we proposed a method for event photo stream segmentation, *i.e.* the process of finding contiguous groups of photos from an event photo stream, each of which corresponds to a photo-worthy moment in the event (Gozali et al., 2012a). Our model leverages our observation that photo streams exhibit alternating photo and photo gap feature types. We use it to formulate the problem and the structure of our proposed HMM. We motivated our final model, the HMM with alternating observation types, by describing the

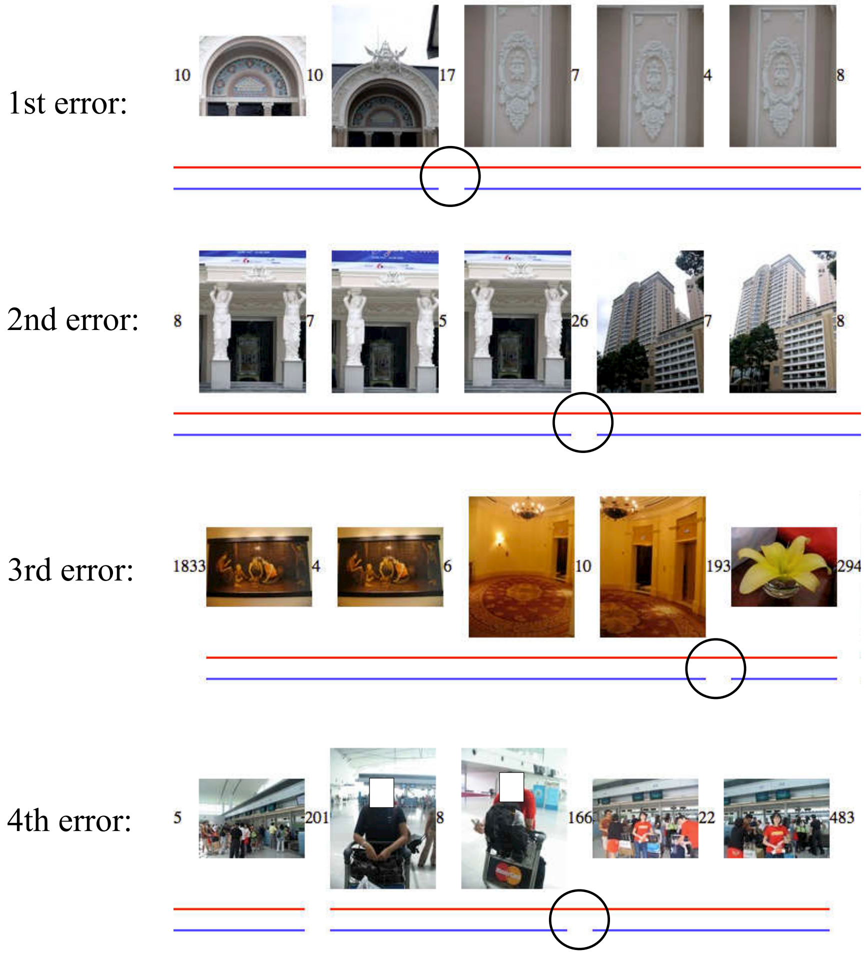


Figure 3.20: The 4 false alarm errors in Set 16 and its surrounding photos. The number shown between photos correspond to time gap values (seconds). The colored lines indicate sub-event membership, *i.e.* photos on the same line belong to the same sub-event. The first red line shows the ground truth while the second blue line is produced by our method. False alarm errors are circled in black.

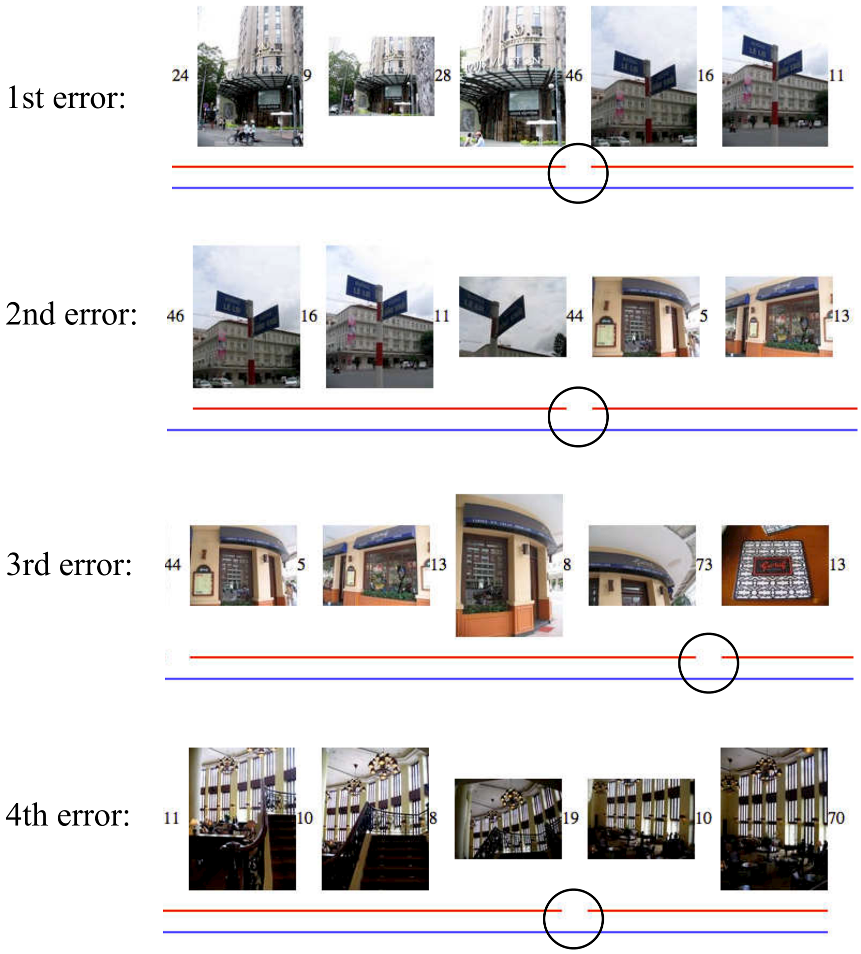


Figure 3.21: The 4 miss errors in Set 16 and its surrounding photos. The number shown between photos correspond to time gap values (seconds). The colored lines indicate sub-event membership, *i.e.* photos on the same line belong to the same sub-event. The first red line shows the ground truth while the second blue line is produced by our method. Miss errors are circled in black.

Set	Number of photos	Number of ground truth sub-events	ΔPr_{error}
1	301	73	0.142
2	321	77	0.014
3	260	42	0.002
4	209	68	0.003
5	94	13	-0.045
6	132	23	0.025
7	209	47	0.030
8	135	42	0.007
9	160	24	0.008
10	188	37	0.026
11	173	46	-0.035
12	236	62	0.078
13	125	26	0.066
14	177	41	0.052
15	224	50	0.013
16	105	20	-0.097
17	149	41	0.017
18	150	45	0.007
19	363	18	-0.025
20	195	40	-0.040
21	117	46	0.077
22	157	48	-0.063
23	162	20	0.061
24	214	56	0.016
25	162	40	0.007
26	131	40	0.022
27	207	44	0.012
28	132	52	0.067

Table 3.7: Comparison between our method (with smoothing and filtering) with the best baseline for each photo set. For each set, the ΔPr_{error} is shown. A positive number indicates that our method performed better.

drawbacks of several preliminary models. We performed a thorough feature and structure analysis to determine the best feature combination and number of HMM states to use for our model. We then described how the HMM can be trained without labelled data and how we addressed the issue of data sparsity and parameter initialization with deleted interpolation smoothing. We also outlined how spurious solutions can be filtered out by looking at the HMM state distributions.

In the evaluation, we showed that many existing photo stream segmentation methods are unsuitable for our task. While our method produces more false alarms, a deeper analysis reveals that this is primarily caused by the subjectivity of the ground truth segmentations provided by the photo owners. Overall, our method performed better than all baselines, including the state-of-the-art cluster tree algorithm, with statistical significance.

Chapter 4

Photo Organization Study and Photo Layout Study

The second and third component of this thesis address the user behavior and layout presentation for a chapter-based photo organization. For this, we conducted a user study to explore the following three questions:

1. How do people organize their photos in each event?
2. How does chapter-based photo organization affect photo-related tasks such as storytelling, searching, and interpretation tasks?
3. What photo layout aspects are important for chapter-based photo organization?

In the following sections, we describe the photo layouts used for the study, the participant demographics, photo sets used in the study, the task descriptions, and safeguards for validity, before going into the results and discussion.

4.1 Photo Layouts Used for Study

For this study, we developed the first iteration (ver. 1) of our chapter-based photo browser, CHAPTRS, with four layouts for displaying photos from a single event (see Figures 4.1, 4.2, 4.3, and 4.4). The first is our baseline, a plain grid layout commonly used by commercial photo browsers and offers no chapter-based photo organization. The other three layouts present chapter-based photo organizations but each emphasizes on a different key layout aspect. As such, they facilitated our study to explore which key aspects are important for chapter-based photo organization.

1. **Plain grid layout** is our baseline layout and it consists of a single grid of row-by-row chronologically-ordered photos. No chapter information is presented in this layout.

2. **Bi-level layout** consists of a split view where the bottom view displays a film strip of chronologically-ordered chapter thumbnails for selection and the top view displays photos of the selected chapter in a grid layout, in chronological order row-by-row.

3. **Grid-stacking layout** consists of chronologically-ordered vertically-stacked grids, each corresponding to a chapter. Photos in each grid are ordered chronologically row-by-row.

4. **Space-filling layout** consists of a single grid of row-by-row chronologically-ordered event photos with an outline surrounding each span of photos that are part of the same chapter.

CHAPTRS ver. 1 also affords users with a drag-and-drop interface to edit the chapter groupings in the bi-level layout. By default, our event photo stream segmentation algorithm automatically groups event photos into chapters so users only need to adjust the chapter groupings instead of starting from scratch. To combine adjacent chapters, users simply drag one chapter thumbnail onto another from the film strip. When users have a chapter selected in the film strip, its photos are shown in the top view. To move photos into a new chapter, users can select a span of photos at the beginning or end of the chapter and then drag the photos onto the film

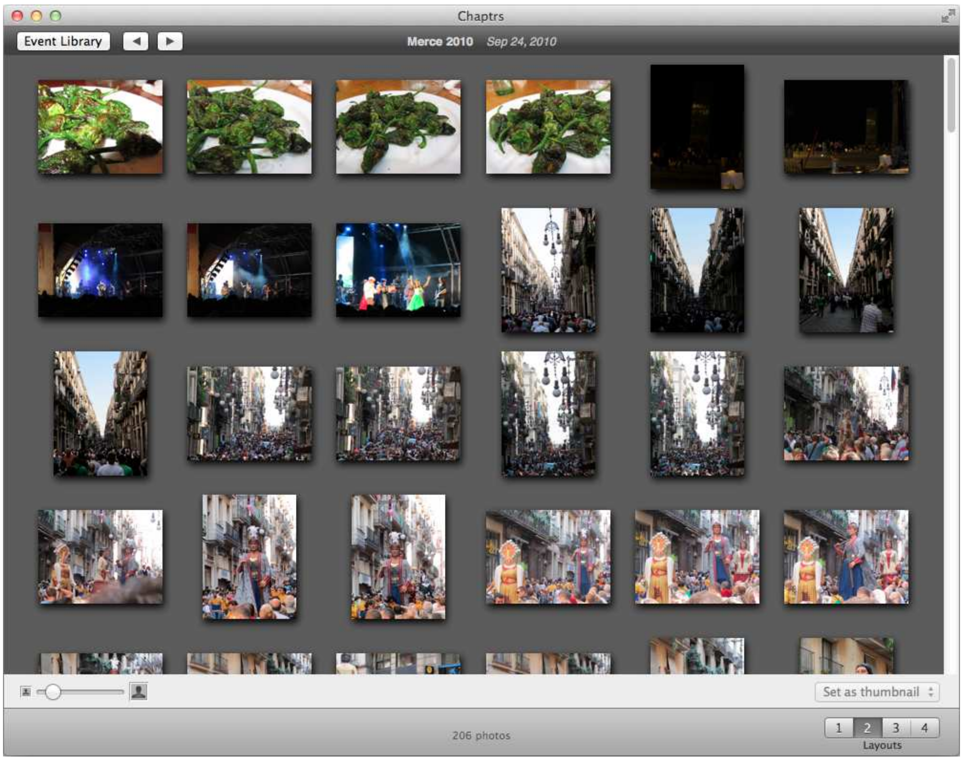


Figure 4.1: Plain grid layout

strip. Other kinds of selections are not valid to ensure that the chronological order of the photos in the stream is not violated.

The four layouts take inspiration from our review of existing photo layouts for personal digital photo libraries in Chapter 2. We adapt them to organize chapters, instead of other group types (*e.g.* events, similar photos). The bi-level layout takes inspiration from photo storytelling interfaces which present the chronological order unweaved in a single horizontal dimension, *i.e.* in contrast to a plain grid layout where the chronological order of the photos are weaved row-by-row. The space-filling layout takes inspiration from the bubblemap layout in PhotoMesa (Bederson, 2001) and maximizes screen space usage. The grid-stacking layout is similar to how Picasa[1] displays photos from all events at once with a separate grid for each event. Screen space is still wasted but not as much as in the bi-level layout. We now discuss each of the chapter-based layouts in more detail.

4.1.1 Bi-Level Layout

The bi-level layout consists of a split view where the bottom view provides an overview of all photos by displaying a scrollable film strip of chapter thumbnails. The top view displays photos from the selected chapter in a grid layout.

Chapter thumbnails are displayed in chronological order. Each thumbnail is labelled with the timestamp of the first photo in the corresponding chapter and, optionally labelled with a user-defined title. The film strip provides users with an overview of all photos. It acts as an index into the event photos, allowing users to glean over moments in the event through the chapter thumbnails without having to sift through individual photos. The chapter groupings allow users to collapse the timeline in a meaningful way and present chapter thumbnails in a linear structure that effectively conveys their chronological order.

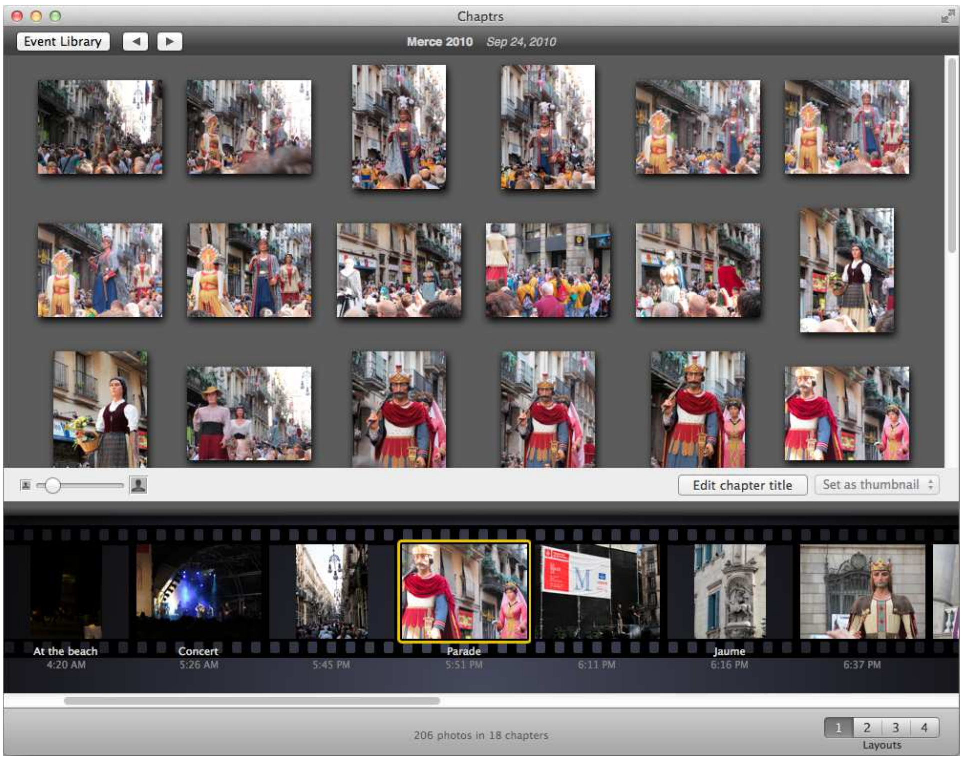


Figure 4.2: Bi-level layout

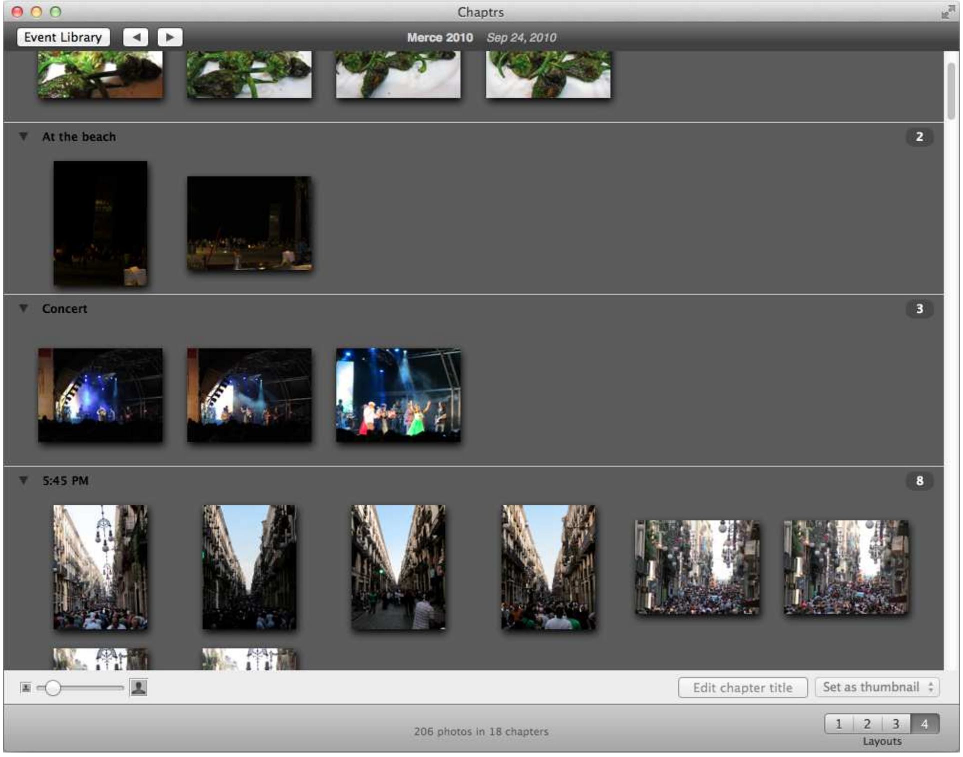


Figure 4.3: Grid-stacking layout

4.1.2 Grid-Stacking Layout

The grid-stacking layout displays all photos from the event with photos of each chapter in its own grid. Photos in each grid are ordered chronologically row-by-row. All grids have the same number of columns and are displayed in chronological order separated by a horizontal line and chapter title.

Compared to the bi-level layout, the grid-stacking layout makes better use of screen space. While the grids may not be fully occupied with photos, the grids are stacked one after another. The chronological order of the chapters are also presented in a linear structure by stacking the grids in one dimension.

4.1.3 Space-Filling Layout

The space-filling layout displays all photos from the event in a single grid. Photos are ordered chronologically row-by-row. In addition, an outline is drawn around photos of the same chapter. To keep photos contiguous within each chapter outline, some grid elements may be left empty (see Figures 4.4 and 4.5). This layout is similar to the bubblemap layout in PhotoMesa but maintains a row-by-row chronological order. As such, the space-filling layout is not as densely packed and may still waste some screen space.

Of the three chapter-based layouts, the space-filling layout is the one that wastes the least amount of screen space and displays the most number of thumbnails at once while still presenting the chapter groupings. These space savings are however, at the expense of the chronological order of the chapters. Unlike the grid-stacking layout, the chronological order of the chapters is weaved into two dimensions row-by-row, instead of linearly top-down.

[1] `http://picasa.google.com`

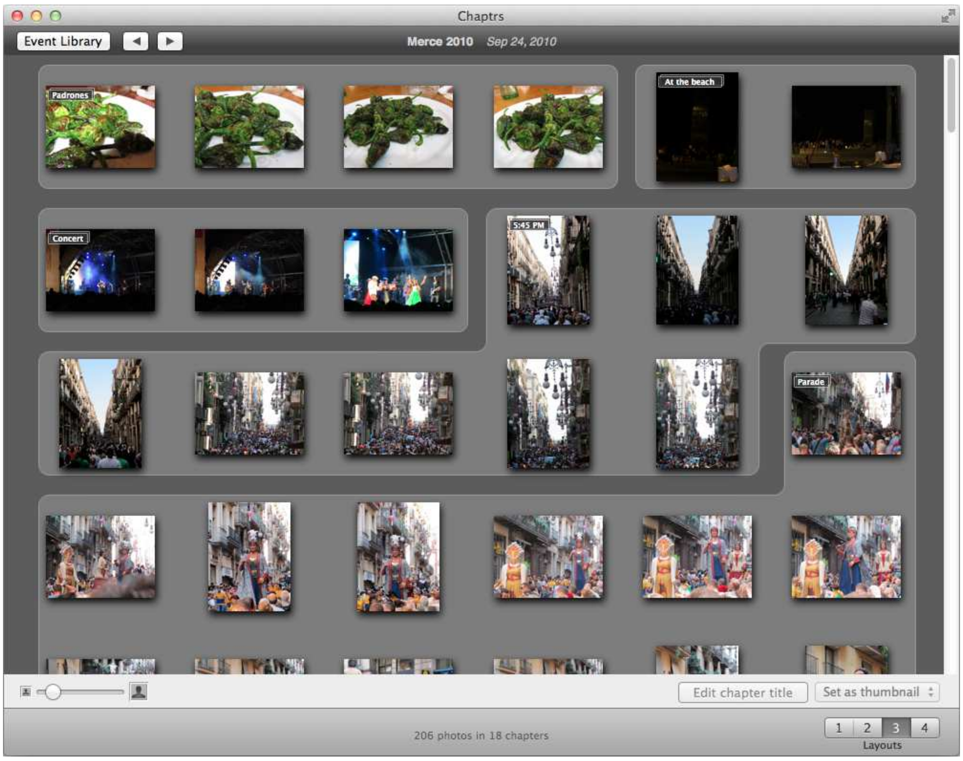


Figure 4.4: Space-filling layout: Event photos are displayed in a grid layout, in chronological order row-by-row, with an outline surrounding photos of the same chapter.

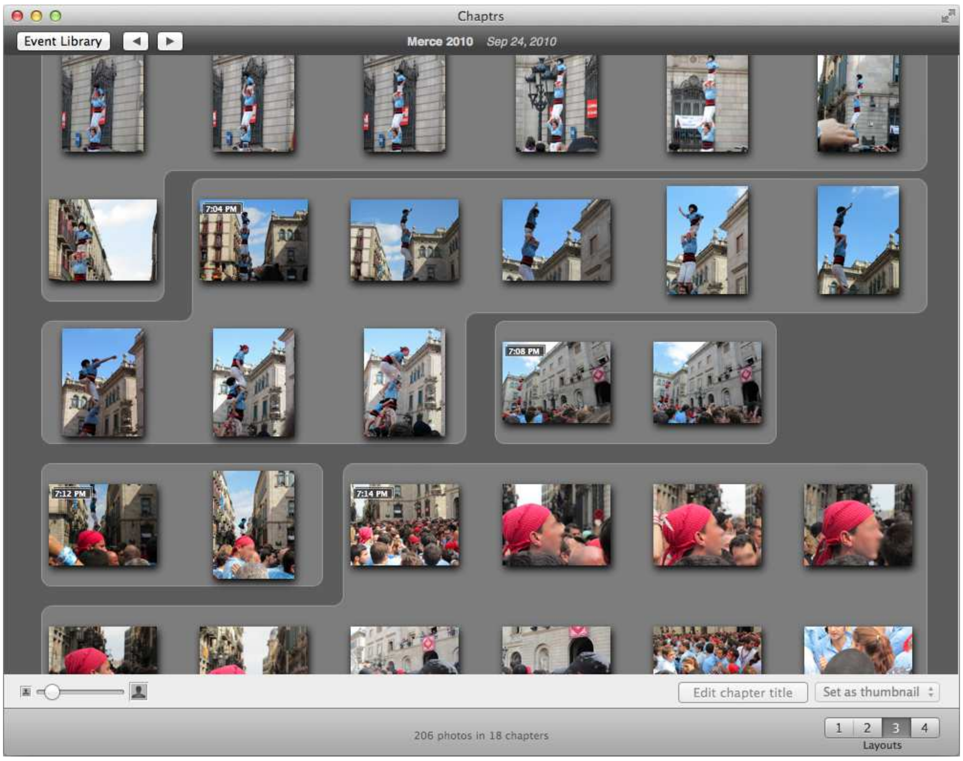


Figure 4.5: Space-filling layout: Some grid elements may be left empty in order to keep photos contiguous within each chapter outline.

4.2 Participant Demographics

For the study, we recruited all 23 college students that responded to our call for user study participation. In our email, we stated that familiarity with one or more desktop photo browser applications was required. We also explained that they would be required to perform three tasks with a new photo browser and answer some questions after each task. We suspect that the use of personal photos may explain the low number of responses. Nonetheless, the 23 participants that responded come with a variety of photography backgrounds: one participant, *P4*, is a professional photographer who often participates in photography trips at public events or at leisure. Another participant, *P12*, maintains an active food blog and always has a digital camera at hand. Some are enthusiastic amateur photographers who carry their digital cameras for social events (*P1, P3, P6, P7, P9, P11, P12, P15, P17, P18, P20*). Others only carry their digital cameras during special occasions or big events like holiday trips (*P2, P5, P8, P10, P13, P14, P16, P19, P21, P22, P23*). Most participants use Windows Explorer or Windows Live Photo Gallery (*P1, P4, P6, P7, P8, P9, P10, P11, P12, P13, P18, P20, P21, P22*) as their primary photo browser. Some use Picasa (*P2, P9, P14, P15, P17, P19, P23*), two participants use iPhoto (*P5, P16*), and one participant (*P3*) uses Aperture.

Following our Institutional Review Board exemption guidelines, photos were immediately discarded at the end of each study session and all collected data was anonymized.

4.3 Photo Sets

Participants were asked to bring four sets of personal photos, each from a different event. While most events are associated with holiday trips, others span a variety of event types: a public cosplay event, a college orientation camp, talks at a conference, a stage performance, visit to the museum, etc. The total number of photos in the study is 8096 photos from 92 photo sets. We asked the participants to bring at least one set with more than 100 photos and at least one with 40-60 photos. This allowed us to ask the participants to reflect on sets with many photos or few photos.

To place these sizes in context, CHAPTRS ver. 1 displays 40-60 photos in less than two screens using the Plain grid or Space-filling layouts. So participants would only need to scroll the user interface by a little to view all the photos. We did not want the participants to bring photo sets with too few photos because the storytelling task in our study inherently assumes that the photo set represents an event worth telling and thus non-fleeting[2].

Before we imported the participant's photo sets into CHAPTRS ver. 1, we asked the participant to choose four different favorite photos from the set with the most photos, using the default file explorer application for Microsoft Windows. These photos were later used in the searching task.

After the photo sets were imported, we asked the participant to *"group the photos into chapters according to their preference and liking"*. Additionally, we randomly selected two photo sets from the participant for s/he to group into chapters *without help* from our event photo stream segmentation algorithm, *i.e.* the participant started with *no initial chapter groupings*. For his/her photo sets, we asked the participant to group the photos to his/her satisfaction; the participant's final organization for the photo sets is used for the study tasks. This protocol allowed us to analyze the effects of initializing the chapter groupings on how the participants group their photos into chapters.

4.4 Study Tasks

Participants were asked to complete three tasks. Participants were also asked to fill a questionnaire after each task, and another overall questionnaire after all three tasks. All questionnaires use a standard 5-point Likert scale from 1 (strongly

[2] We note that the number of photos in the photo stream has no implications on the performance of our event photo stream segmentation algorithm. Our photo browser, CHAPTRS, which implements the algorithm can be used to automatically organize photo sets of various sizes

disagree) to 5 (strongly agree). Finally, each study session ended with a semi-structured interview[3]. The audio from the interview session was recorded for note-taking purposes.

In our study, we focused on common photo-related tasks for users — tasks that fit the STU (Situations, Tasks, and Users) context (Olsen, 2007). In particular, the first two tasks have been used in the related works we reviewed in Chapter 2. We describe each task in more detail next, followed by more details on how we eliminated confounding variables.

Task 1: Storytelling from familiar event photos

In this task, participants were asked to tell the story of each event from their personal photo sets. We asked participants to imagine sharing about the event and its photos, as they normally would, to their friends. We used a within-subject design where each participant carries out the task four times, each with a different layout. To avoid learning effect on the story told, each layout was used with a different photo set.

Task 2: Finding a given photo from familiar event photos

In this task, participants were asked to find the favorite photos they chose at the beginning of the study. We used a within-subject design where each participant carries out the task four times, each for a different favorite photo and with a different layout. At each iteration, the target favorite photo was clearly displayed on an adjacent external monitor. The four favorite photos were chosen from the same photo set to make the iterations comparable. There is no learning effect between iterations on the photo set because the participant — who also owns the photo set — has been through the photos at least twice from the storytelling task and from grouping the photos into chapters at the beginning of the study.

[3] Questionnaires and interview questions available in (Gozali et al., 2012b)

Task 3: Interpreting unfamiliar event photos

In this task, participants were shown and asked to interpret unfamiliar event photos, not belonging to the participants. We asked the participants: *"Tell me about the event. What do you think was happening?"*. For this task, we prepared four sets of event photos that were not used in any other part of the study. The photo sets were titled, grouped into chapters, but chapters were left untitled. We used a within-subject design where each participant carries out the task four times, each with a different layout. To avoid any learning effects, each layout was used with a different photo set. This task is the most synthetic of the three tasks in our user study. While participants are unlikely to find themselves having to interpret event photos without any context other than the photos themselves and the event title, our goal was simply to create a scenario where the participants have very little knowledge of the event, similar to how they would find themselves when faced with an old set of event photos but not remembering any details of the event (Frohlich et al., 2002).

4.5 Internal Validity

We chose a within-subject design, *i.e.* repeated measurements per participant, to have better internal validity, as is common for user studies with few participants. The personal nature of the photos and the length of the study per participant made recruiting hundreds of participants impractical.

As mentioned in Section 4.4, we have tried to eliminate any learning effects. In addition, we eliminated learning effects on the four layouts by demonstrating CHAPTRS ver. 1, its four layouts, and all their features at the beginning of the study, prior to any of the tasks. We prepared five sets of photos, grouped into chapters, exclusively for this purpose. The participants were also asked to spend five minutes to familiarize themselves with the four layouts and ask any questions.

To eliminate ordering effects from the four layouts, we balanced the user study for each task, *i.e* the order in which participants used the four layouts was systematically varied for each task; each participant used a different order from the other participants for each task[4]. Participants were also asked to revisit all four layouts with all photo sets when they answer each questionnaire.

4.6 How Do People Organize Their Photos in Each Event?

At the beginning of the user study, we asked participants to *"group the photos into chapters according to their preference and liking"*. This allowed us to first observe and later inquire on the criteria they used to decide the chapter groupings. We have gathered three insights into this process:

First, **users value chapter consistency more than the chronological order of the photos.** While past findings have shown that people want their photos displayed in chronological order (Rodden and Wood, 2003), all but one (*P11*) of the participants in our study requested that they be allowed to combine non-adjacent chapters in the timeline, effectively displaying the photos out of their chronological order.

Almost all participants had at least one photo set where in the midst of photos capturing one moment in the event, *e.g.* a performance on stage, there were a handful of photos that did not belong, *e.g.* photos of the audience. Another example is where in the midst of scenic photos of a nearby landscape, there were photos of friends and/or family. In these cases, participants wanted to keep all but the handful of photos in one chapter. This observation is similar to how people keep printed photos in albums in chronological order, but with small adjustments done for aesthetic reasons (Rodden, 1999).

By allowing the participants to create meaningful chapters as the organizational unit for their photos, what becomes important to them is the consistency of the photos within each chapter. In explaining why they wanted certain photos taken out of

[4] There are 24 distinct permutations in ordering the four layouts.

a chapter, participants said that the photos *"do not belong there"* (*P8*). This importance supercedes displaying the photos in chronological order. Some participants mentioned that they *"don't really care"* (*P5*) if the photos are not in chronological order, that *"sometimes [it] is not that important"* (*P18*).

Secondly, **criteria for chapters include moment, object, location, photography type,** and **intention.** These criteria pertain to the kind of consistency discussed in the first point. From our study, we observed that the participants commonly adopted one of the following five criteria for their chapters:

1. **Moment** — This criteria is the most common and refers to chapters that correspond to moments in the event. Several participants refer to photo sets whose chapters followed this criteria as being *"according to time"* (*P11*).

2. **Object** — Participants wished to group photos of the same object or object type in the same chapter. For example, in a photo set of a trip to a defunct railroad, the participant *P7* wanted all photos depicting the track in its own chapter, regardless of when the photos were taken.

3. **Location** — Participants also commonly organized their photos with a chapter for each location, for example, in holiday photos where photos were captured from a variety of different locations (*e.g.* tourist spots).

4. **Photography type** — For example, participants wished to group photos of their friends in the same chapter. Another example is to have a chapter for all the scenic photos.

5. **Intention** — On several occasions, participants wished to have a different chapter for photos of different groups of individuals, *e.g.* one chapter for photos with friends and another chapter for photos with colleagues. Another example is where one participant, *P3*, has several *"silly shots"* taken at very different times during the event but would like to have them all in the same chapter.

Lastly, **choice of criteria and granularity for segmentation are very subjective**. We found that deciding a criteria for the chapters is a very subjective process. For example, in a photo set of performances on stage, the participant, *P17*, *separated visually similar photos* into several chapters to have one chapter for each performance. For another photo set however, the same participant wanted to *combine visually similar photos* of different speakers into the same chapter to create a summary of the event in a single chapter. Several participants noted that they would group photos of the same location, even if taken at different times, *e.g.* night and day, into the same chapter. However, they will separate portrait photos of their friends/family into a different chapter, separate from the chapter with scenic photos of the same location.

Participants also had different notions of granularity for their chapters. One participant, *P18*, wanted to create a chapter with many photos to depict *"photos of the path [he] took from the entrance to the mountain"*. Photos taken near the path would be grouped into separate chapters. Another participant, *P2*, mentioned that he would like to group his photos *"by visual similarity"* unless *"[the photo set] is for a big event because there will be too many chapters"*. Some participants (*P7, P19, P22*) disliked having a chapter with just one or two photos and would combine the chapter with an adjacent one simply because s/he *"want[s] to combine it with something else"* (*P22*).

While deciding the chapter grouping is a subjective process, participants agree that *"grouping [their] photos by chapter makes sense"* (μ=4.3, δ=0.6). In response to the subjectivity, more participants found it *"easy to decide the correct chapter groupings"* (μ=3.7, δ=1.0). These participants said that they will know what to do when they see the photos.

To assess how automatically grouping photos into chapters affected their final organization by participants, at the beginning of the study we randomly selected two photo sets from each participant for s/he to group without the help of our event photo stream segmentation algorithm. The other two photo sets of each participant

Initialized?	**Num**	**Average**	**Average**	**Average**
	Photo Sets	Pr_{miss}	Pr_{fa}	Pr_{error}
No	30	0.193	0.508	0.350
Yes	47	0.118	0.290	0.204
Improvement		38.7%	43.0%	41.8%

Table 4.1: Comparison between the chapter groupings by our algorithm with the ground truth by the participants as measured by miss rate, Pr_{miss}, false alarm rate, Pr_{fa}, and error rate, Pr_{error}. A smaller number indicates better agreement. One group of photo sets were initialized by our algorithm and further organized by the participants. The other was done by the participants without help.

were initialized with a chapter organization given by our algorithm. This allows us to compare the chapter groupings from our algorithm with those by the participants (as ground truth) for two kinds of photo sets: 1) photo sets that were organized by the participants without help[5], and 2) photo sets that were initialized by our algorithm and further organized by the participants.

Some photo sets were from older generation cameras that did not embed photo metadata[6] in the image files. Since the metadata is necessary for our event photo stream segmentation algorithm, we could not run our algorithm on these photo sets. For this initialization analysis, we have a total of 7073 photos in 77 sets.

To perform the comparisons, we used the error rate metric, Pr_{error}, that was used in our evaluation in the previous chapter. Recall that a lower Pr_{error} indicates better agreement with the ground truth by the participants; a score of 0 indicates perfect agreement. Pr_{error} is an average of the miss and false alarm rates. As such, a method that proposes no chapter boundaries or proposes chapter boundaries everywhere will have an error rate of about 0.5.

In Chapter 3[7], we noted that our event photo stream segmentation algorithm

[5] We ran our algorithm on these photo sets but the results were neither used nor shown to the participants.

[6] Exchangeable Image File Format (Exif) data

[7] Also in (Gozali et al., 2012a).

has a tendency to propose more fine-grained segmentations. We can see this in Table 4.1 where the false alarm rate, Pr_{fa}, is markedly smaller — a 43% improvement from a high rate of 0.508 — for the initialized photo sets. With initialization, participants were provided with the opportunity to explicitly agree or disagree with our fine-grained results. The effect is that participants found meaningful chapter boundaries among the many proposed. Without initialization, participants had to find meaningful chapter boundaries for themselves, resulting in higher false alarm rates for our algorithm in comparison.

While the error rate values we report in Table 4.1 were computed by penalizing misses and false alarms equally, we found through our user study that in practice, having a high miss rate is more detrimental to the user experience than having a high false alarm rate. Many participants in our study mentioned that it was easier to decide if two chapters should be combined than to decide how to split up a chapter. For example, one participant, *P11*, mentioned that *"its better to make it small small so then if the user want[s] to merge then they [can] do it themselves. Its not that difficult."* To correct a false alarm is a one-step process of combining the two chapters. But to correct a miss, the user must first realize that there is a miss, then figure out the best position to split the chapter.

4.7 How Does Chapter-based Photo Organization Affect The Study Tasks?

In this section, we present quantitative and qualitative results from each task of the study. We also present the level of statistical significance of the quantitative results, *i.e.* the p-value from a two-tailed paired Student's t-test in comparison with the plain grid layout. While our findings have different levels of significance, we note that most are significant at $p < 0.005$. We present the participants' mean response values from the questionnaire in Table 4.2 for easy reference. Values that are statistically significantly in comparison with the plain grid layout are shown with their p-values in subscript. We elaborate on the tabulated results in the following subsections, but defer comparisons between the three chapter-based layouts to Section 4.8.

Task 1: Story-telling from familiar event photos

Participants agree that *"having chapters helps present the event's story for sets with many photos"* (μ=4.3, $p < 0.001$). We obtained similar results for sets with few photos (μ=3.9, $p < 0.05$), but less statistically significant. When asked for each layout specifically however, participants agree that each of the chapter-based layouts helps present the event's story for sets with many or few photos, all with $p < 0.005$.

We also asked the participants whether having chapters helps them remember what to say about the event. One participant, *P17*, said that the chapters *"help give focus"* in remembering. Participants agree that *"having chapters helps [them] remember the event's story"* for sets with many or few photos (μ=4.7, μ=4.1; both $p < 0.005$). When asked for each layout specifically, participants agree that each of the chapter-based layouts helps them remember the event's story for sets with many photos ($p < 0.001$). We obtained similar results for sets with few photos, but only the grid-stacking and space-filling layouts are statistically significant at $p < 0.001$; the bi-level layout is less statistically significant at $p < 0.05$.

Chapters can guide users with their storytelling. In the plain grid layout where no chapter information is presented, one participant, *P23*, said that s/he was *"scrolling, scrolling, scrolling"* and did *"not know where to stop and say something more"*. In contrast, participants use the chapter information presented in the other chapter-based layouts to pace their story. Participants would refer to a particular chapter and start a part of their story with, *e.g.* *"this chapter is about..."* (*P18, P21, P22*). Participants also gesture around chapter outlines with their forefingers or cursors in the space-filling layout to highlight the photos relevant to their stories at the

Questionnaire Statement	**Bi-Level**	**Grid-Stacking**	**Space-Filling**	**Plain Grid**
The layout helps present the event's story for sets with many photos	$4.2_{0.005}$	$4.2_{0.005}$	$3.7_{0.005}$	2.4
The layout helps present the event's story for sets with few photos	$4.1_{0.005}$	$4.3_{0.005}$	$4.1_{0.005}$	3.2
The layout helps them remember the event's story for sets with many photos	$4.0_{0.001}$	$4.3_{0.001}$	$3.9_{0.001}$	2.6
The layout helps them remember the event's story for sets with few photos	$4.0_{0.05}$	$4.4_{0.001}$	$4.1_{0.001}$	3.2
The layout helps to find a photo in a set with many photos	$3.6_{0.01}$	$4.4_{0.001}$	$3.7_{0.001}$	2.7
The layout helps to find a photo in a set with few photos	3.6	$4.4_{0.001}$	$4.0_{0.001}$	3.1
The layout helps to interpret photos of an event with many photos	$3.9_{0.005}$	$4.6_{0.005}$	$4.0_{0.005}$	2.9
The layout helps to interpret photos of an event with few photos	$3.7_{0.05}$	$4.4_{0.001}$	$3.9_{0.001}$	3.1

Table 4.2: Mean response values from the participants to various questionnaire statements for each layout. The values follow a standard 5-point Likert scale from 1 (strongly disagree) to 5 (strongly agree). Values that are statistically significant in comparison with the plain grid layout are shown with their p-values in subscript.

time. One participant, *P12*, however, adopted a purely photo-driven storytelling method (Balabanović et al., 2000) where s/he would double-click to maximize the photo and subsequently use the navigation keys on the keyboard to go to the next or previous photos.

On average, the grid-stacking layout is most preferred, followed by the bi-level, space-filling and plain grid layouts. The difference in preference between each of the chapter-based layouts with the plain grid layout is statistically significant ($p < 0.001$).

Task 2: Find a given photo from familiar event photos

From the measured completion times, we determined the layout that allowed participants to complete the task the fastest. On average, the space-filling layout was the fastest (7.0s), followed by the plain grid (7.8s), grid-stacking (11.2s), and bi-level (14.2s) layouts. The difference between the grid-stacking and bi-level layouts ($p < 0.005$); and the plain grid and bi-level layouts ($p < 0.05$) are statistically significant. We note that this ranking aligns closely with how well the layouts make use of screen space, making our results consistent with past findings that propose displaying many thumbnails at once to help users with their visual search tasks (Rodden and Wood, 2003).

While the plain grid layout ranks second for the fastest completion time, participants actually preferred the plain grid layout the least for this task. On average, the most preferred layout for this task is the grid-stacking layout, followed by the space-filling, bi-level, and plain grid layouts. The difference in preference between each of the chapter-based layouts with the plain grid layout is statistically significant ($p < 0.001$).

Note that participants were not informed on how fast they performed with each layout. This was done so that their layout preference for this task was not affected by the completion time rankings. The contrast between the layout preference and the completion time rankings suggests that for the task of finding a photo within a familiar set, where the fastest and slowest times only differ by several seconds, completion time does not play a major role for their preference.

One participant, *P23*, noted that for tasks like this, *"they like to find the chapter first"*. Participants agree that *"having chapters helps [them] find a photo in a set with many photos"* (μ=4.4, $p < 0.001$). We obtained similar results for sets with few photos (μ=4.0, $p < 0.05$), but with less statistical significance. Participants also agree that each of the chapter-based layouts helps them find a photo in a set with many photos ($p < 0.001$, except the bi-level layout with $p < 0.01$). For sets with few photos, only results for the grid-stacking and space-filling layouts are with statistical significance ($p < 0.001$).

While the participants' layout preference contradicts with the completion time rankings, the behavior to find chapters first before finding the photo is similar to past findings. The same study we quoted above (Rodden and Wood, 2003) found that when users want to search for a particular photo, they will first attempt to remember the event at which it was taken. In our case, we observed that participants use the chapter groupings to skip chapters that they know will not contain the photo, and look deeper into chapters that might. This process is easiest to perform with the grid-stacking layout, which is the most preferred layout for this task.

Task 3: Interpreting unfamiliar event photos

Participants agree that *"having chapters helps [them] interpret photos of an event with many photos"* (μ=4.6, $p < 0.001$) as well as those with few photos (μ=4.0, $p < 0.001$). When asked for each layout specifically, participants agree that each of the chapter-based layouts helps them interpret photos of an event with many photos ($p < 0.005$). For sets with few photos, only the grid-stacking and space-filling layout are statistically significant at $p < 0.001$; the bi-level layout is less statistically significant at $p < 0.05$.

We observed that generally, the participants fall into two groups, each with a different approach to the task. Participants in the first group rely on gathering a visual overview of all the photos to interpret the event. They would scroll up and down fairly quickly to gather a general idea of the event. For this group, a layout that displays many thumbnails at once is most preferred and not having chapter information presented in the layout is not a loss. One participant, *P22*, disliked the bi-level layout for this reason: *"I can't grasp what's happening because it [displays] one chapter at a time"*. Participants would give a very general interpretation of the event and only comment for every other chapter. Participants who chose the space-filling layout as their most preferred overall layout fall into this group (*P3, P6, P15, P16, P22*).

Participants in the second group rely on chapter information to guide them through the event photos. Some would still gather a visual overview from all the event photos, but they would describe each chapter in chronological order: *"Here they went to... and then to..."* (*P5*). With the plain grid layout where no chapter information is presented, these participants are at a loss and *"can't tell if the photos are apart or together"* (*P23*). In contrast, the layouts with chapter groupings *"look [very|more] organized"* (*P4, P11, P12, P14, P18, P20*). The twelve participants who chose the grid-stacking layout as their most preferred overall layout fall into this group (*P2, P7, P8, P11, P13, P14, P17, P18, P19, P20, P21, P23*).

In our categorization of participants, we found that more participants fell into the second group. As such, the most preferred layout for this task is the grid-stacking layout, followed by the bi-level, space-filling, plain grid layouts. The difference in preference between each of the chapter-based layouts with the plain grid layout is statistically significant ($p < 0.001$).

4.8 What Layout Aspects are Important for Chapter-based Photo Organization?

Among the chapter-based layouts, the grid-stacking layout was the only layout that outperformed some others with statistical significance; and it does so for each task.

For helping to present the event's story for sets with many photos, participants agree more with the grid-stacking layout than with the space-filling layout ($p < 0.01$). For helping to find a photo in a set with many or few photos, participants agree more with the grid-stacking layout than with the bi-level layout ($p < 0.01$). For helping to interpret photos of an event with many or few photos, participants agree more with the grid-stacking layout than with all the other layouts ($p < 0.01$).

Regarding the methods used by the chapter-based layouts to present the chapters, participants like the grid-stacking layout (μ=4.6) statistically significantly more ($p < 0.005$) than the bi-level (μ=3.9) and space-filling layouts (μ=3.6). They liked how the layout shows the *"chapter groupings each in a separate grid"* (*P1*). In all tasks and in overall ranking, most participants indicated the grid-stacking layout as their top preference. All this suggests that participants value the strength of the grid-spacing layout — a clear top-down presentation of the chronological order of the chapters — more than the strengths of the bi-level and space-filling layouts.

The bi-level layout features an overview of all the event photos afforded by the film strip of chapter thumbnails. Participants like the film strip (μ=4.4, δ=0.7) as it shows *"the flow of the event"* (*P1*). Participants also found it is easy to navigate the user interface (μ=4.2, δ=0.8). On the other hand, for the statement *"I do NOT like the wasted screen space"*, participants only somewhat disagree (μ=2.7[8], δ=1.0). This contrast suggests that while participants like and appreciate having the overview, they prefer not to waste much screen space imposed by the restricting view hierarchy, even if its easy to navigate.

The space-filling layout maximizes screen space usage; minimal screen space is wasted while still presenting the chapter groupings. A number of participants do value maximizing screen space usage more than the chronological order of the chapters; five participants chose this layout as their most preferred layout overall. Most participants (12 of 23) however, prefer the grid-stacking layout. These

[8] 2 — Disagree, 3 — Neither agree nor disagree

participants found the space-filling layout to be *"confusing"* (*P23*).

4.9 Conclusion

In this chapter[9], we have explored chapter-based photo organization and report results — qualitative and quantitative with statistical significance — that advocates its use for personal digital photo libraries.

We developed a photo browser, CHAPTRS ver. 1, and integrated the event photo stream segmentation algorithm from our previous work to explore how people organize their photos in each event. Our algorithm helps users by automatically grouping event photos into smaller groups of chapters. We implemented a baseline plain grid layout and three chapter-based photo organization layouts in CHAPTRS ver. 1 to explore how chapter-based photo organization affects storytelling, searching and interpretation tasks; and what photo layout aspects are important for such tasks.

Our participants found chapter-based photo organization to be helpful in all three tasks. Our study also revealed how the participants employed chapters in these tasks. The grid-stacking layout was preferred the most in all three tasks and the baseline plain grid layout was preferred the least. Among the results, the following are our primary findings from the study:

1. Users value chapter consistency more than the chronological order of the photos in grouping photos into chapters

2. Choice of chapter criteria and granularity for chapter groupings are very subjective

3. Having low misses is more important than having low false alarms for automatic event photo stream segmentation

[9] Also in (Gozali et al., 2012c).

4. Users value chronological order of the chapters more than maximizing screen space usage in photo layouts

While we discovered that the preference for criteria and granularity of our participants were very subjective, our study also shows that our algorithm helps participants discover chapter groupings.

With our findings on the key layout aspects, we will use the grid-stacking layout and the film strip overview in the next design iteration of CHAPTRS, *i.e.* ver. 2, described in the next chapter.

Chapter 5

CHAPTRS PHOTO BROWSER

Figure 5.1 shows the main user interface of the final version of our photo browser, CHAPTRS ver. 2. In this chapter, we describe a typical usage scenario for CHAPTRS ver. 2 and highlight how it works harmoniously with existing photo digital libraries on the user's computer, in line with our goal to complement event-based photo organization. We also describe in detail how CHAPTRS ver. 2 embodies our segmentation method and our findings from the user study on chapter-based photo organization and photo layouts.

5.1 Usage Scenario

While there are different use cases for CHAPTRS ver. 2, *e.g.* as a quick way to search / access / visualize your photo libraries, here we outline a basic use case where the user starts CHAPTRS ver. 2 to browse photos (see Figure 5.2). Parts of the use case will be explained in detail in the sections that follow.

1. User starts CHAPTRS ver. 2 (see Figure 5.3).
2. CHAPTRS ver. 2 automatically scans for existing iPhoto or Aperture photo libraries and populates the **Event Sidebar** with events from these libraries (see Figure 5.4).

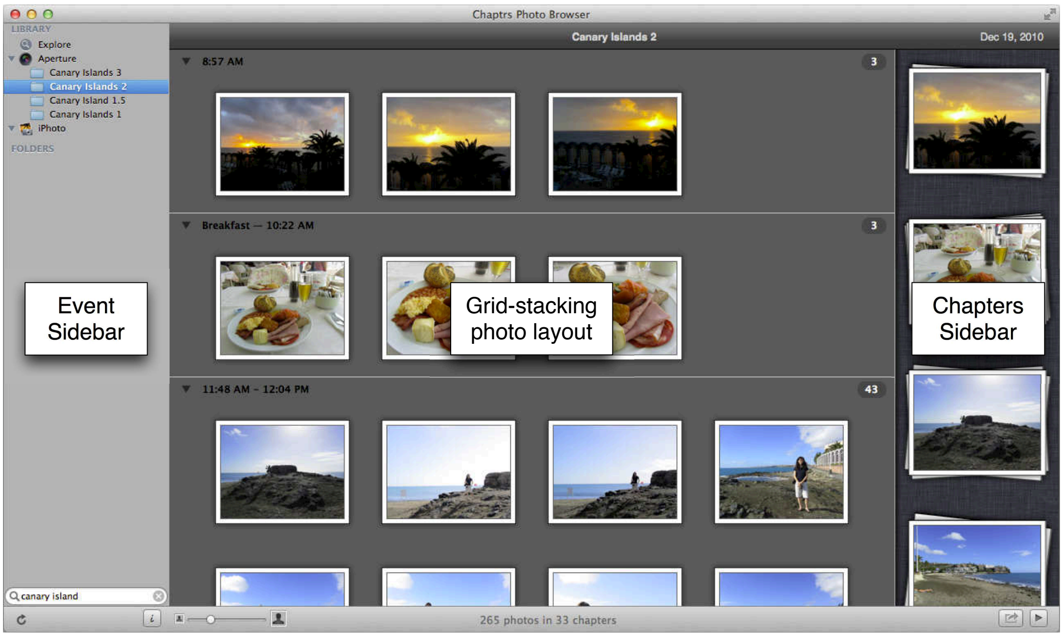


Figure 5.1: The main user interface for CHAPTRS ver. 2

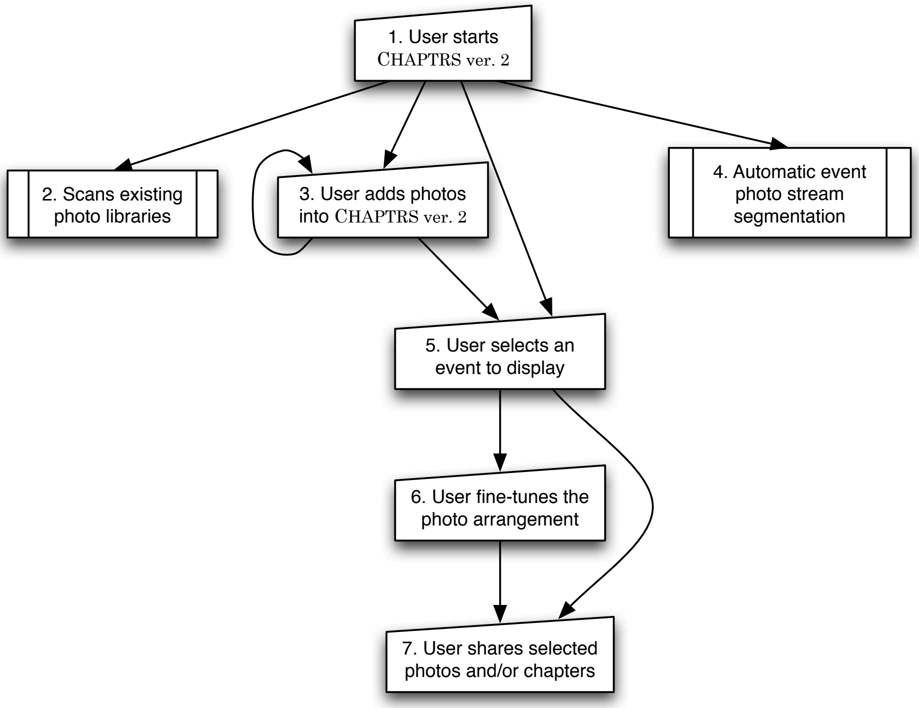

1. User starts CHAPTRS ver. 2
2. Scans existing photo libraries
3. User adds photos into CHAPTRS ver. 2
4. Automatic event photo stream segmentation
5. User selects an event to display
6. User fine-tunes the photo arrangement
7. User shares selected photos and/or chapters

Figure 5.2: Example use-case diagram for CHAPTRS ver. 2

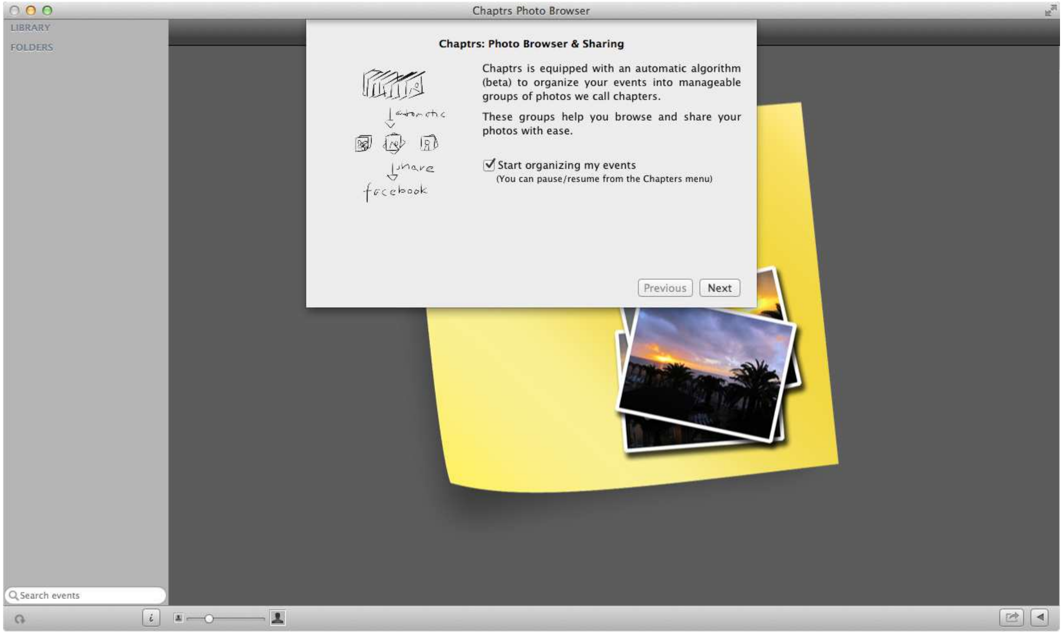
3. The user may drag-and-drop a selection of photo files into the **Event Sidebar** to add them as an event in CHAPTRS ver. 2. Users may also drag-and-drop folders, in which case each folder is added as an event (see Figure 5.5).

4. By default, CHAPTRS ver. 2 automatically finds chapters in each event, inconspicuous to the user, in the background (see Figure 5.11 and Section 5.3).

5. User selects an event from the **Event Sidebar** and is presented with photos from the event, grouped by chapter, in a grid-stacking layout. The **Chapters Sidebar** on the right displays chapter thumbnails (see Figure 5.6 and Section 5.5).

6. User performs drag-and-drop operations to arrange and fine-tune the photo arrangement (see Figure 5.7 and Section 5.4).

7. User shares selected photos and/or chapters to his/her social networks, or performs a drag-and-drop operation to a folder to copy the photos into the folder, *e.g.* the desktop (see Figure 5.8).



Chaptrs: Photo Browser & Sharing

Chaptrs is equipped with an automatic algorithm (beta) to organize your events into manageable groups of photos we call chapters.

These groups help you browse and share your photos with ease.

☑ Start organizing my events
(You can pause/resume from the Chapters menu)

Previous | Next

Figure 5.3: User starts CHAPTRS ver. 2.

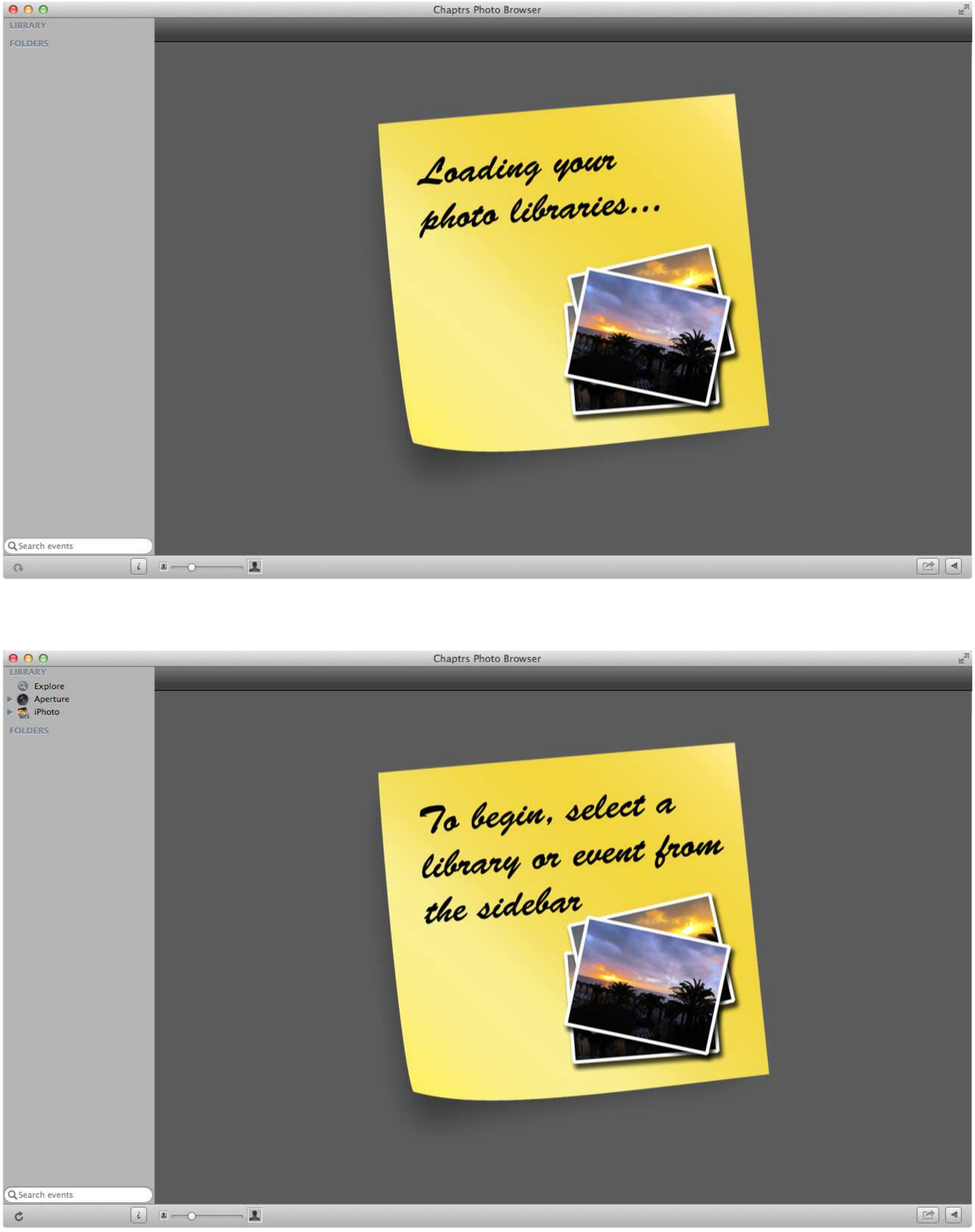
5.2 Complementing Event-based Photo Organization

As a photo browser with a chapter-based photo organization, CHAPTRS ver. 2 was designed to be complementary to existing event-based photo organization. CHAPTRS ver. 2 works amicably with any existing workflows using other applications:

1. CHAPTRS ver. 2 understands photo libraries of iPhoto and Aperture — two popular photo management applications on Mac OS X — and displays their events and albums in its **Event Sidebar**, making use of existing event boundaries from these libraries.

2. CHAPTRS ver. 2 supports multiple iPhoto libraries and multiple Aperture libraries and also allows users to add photo files or folders of photos directly into the **Event Sidebar**.

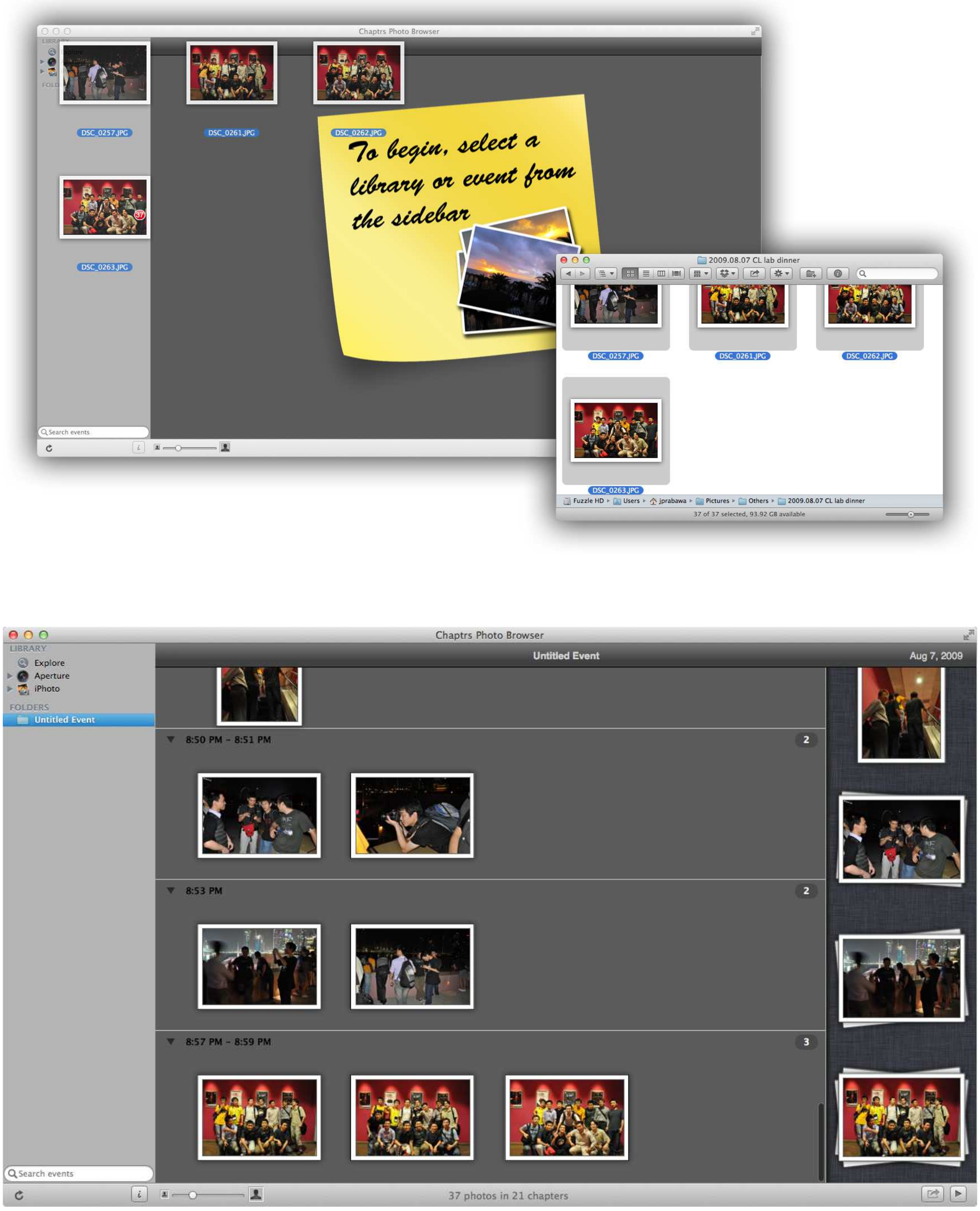
3. Even when photos in an event are arranged into chapters by CHAPTRS ver. 2, the original photo files and its corresponding iPhoto / Aperture photo library, if any, are not modified in any way.



Loading your photo libraries...

To begin, select a library or event from the sidebar

Figure 5.4: CHAPTRS ver. 2 automatically scans for existing iPhoto or Aperture photo libraries and populates the **Event Sidebar** with events from these libraries.



To begin, select a library or event from the sidebar

Figure 5.5: The user may drag-and-drop a selection of photo files into the **Event Sidebar** to add them as an event in CHAPTRS ver. 2. Users may also drag-and-drop folders, in which case each folder is added as an event.

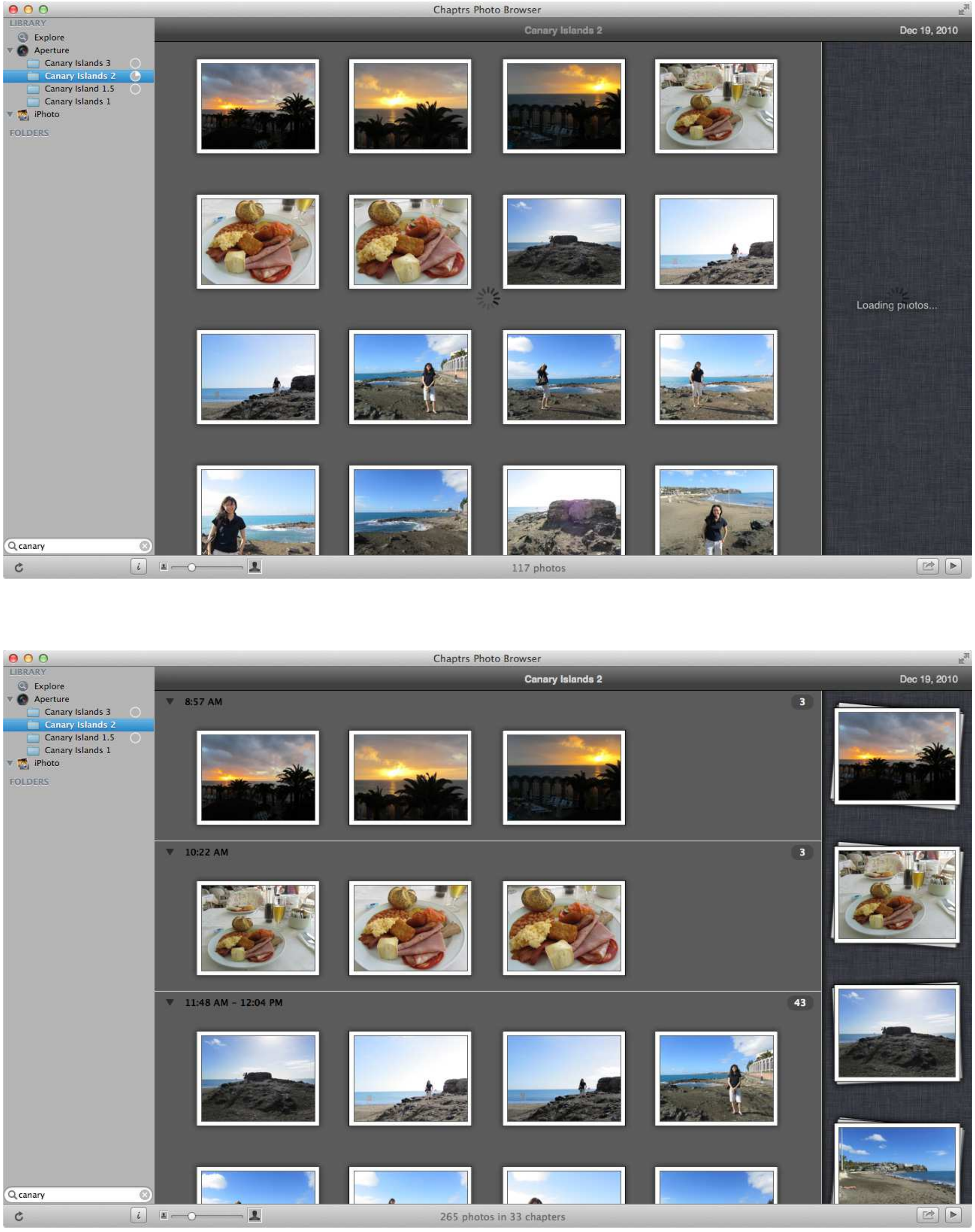


Figure 5.6: User selects an event from the **Event Sidebar** and is presented with photos from the event, grouped by chapter, in a grid-stacking layout. The **Chapters Sidebar** on the right displays chapter thumbnails.

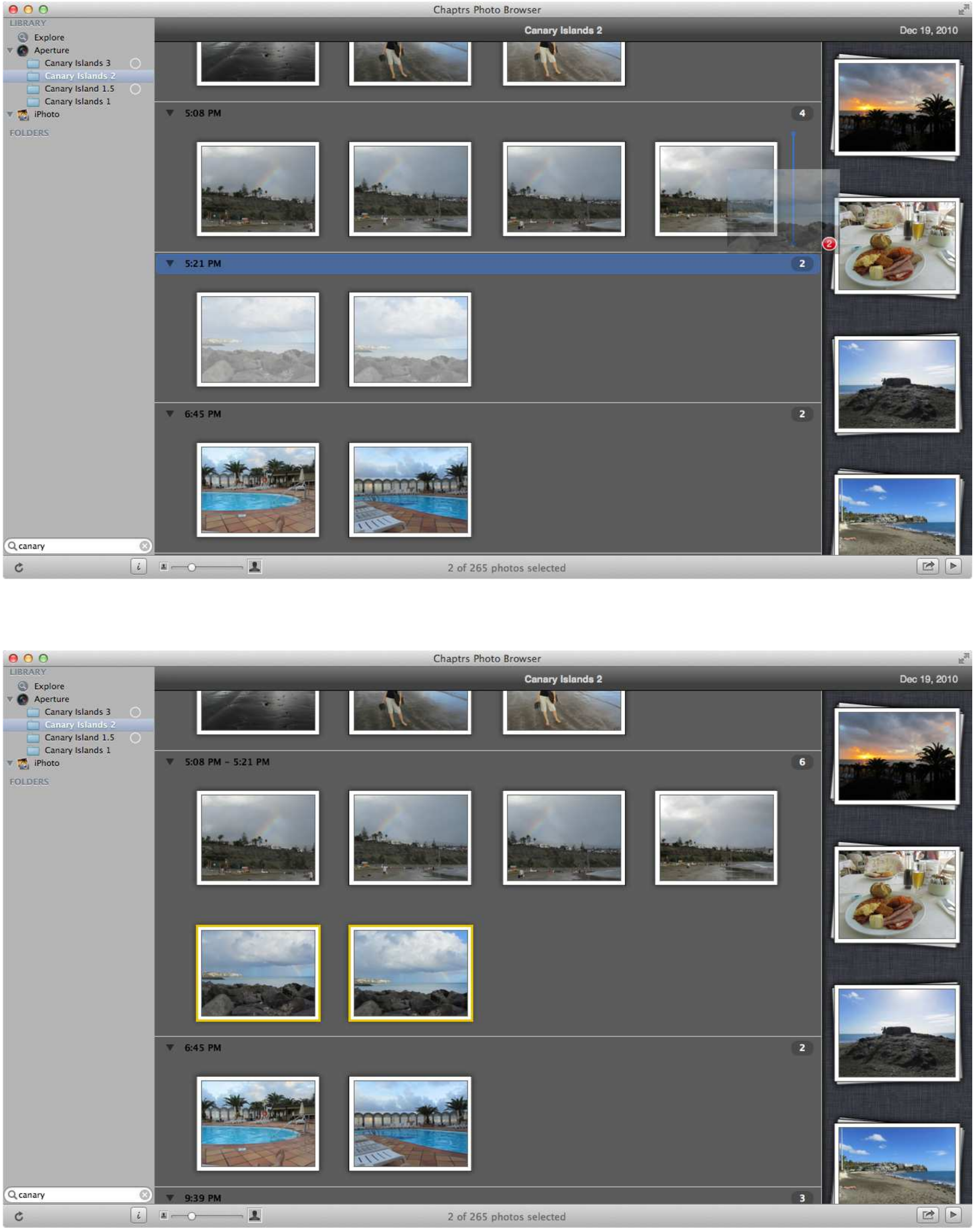


Figure 5.7: User performs drag-and-drop operations to arrange and fine-tune the photo arrangement.

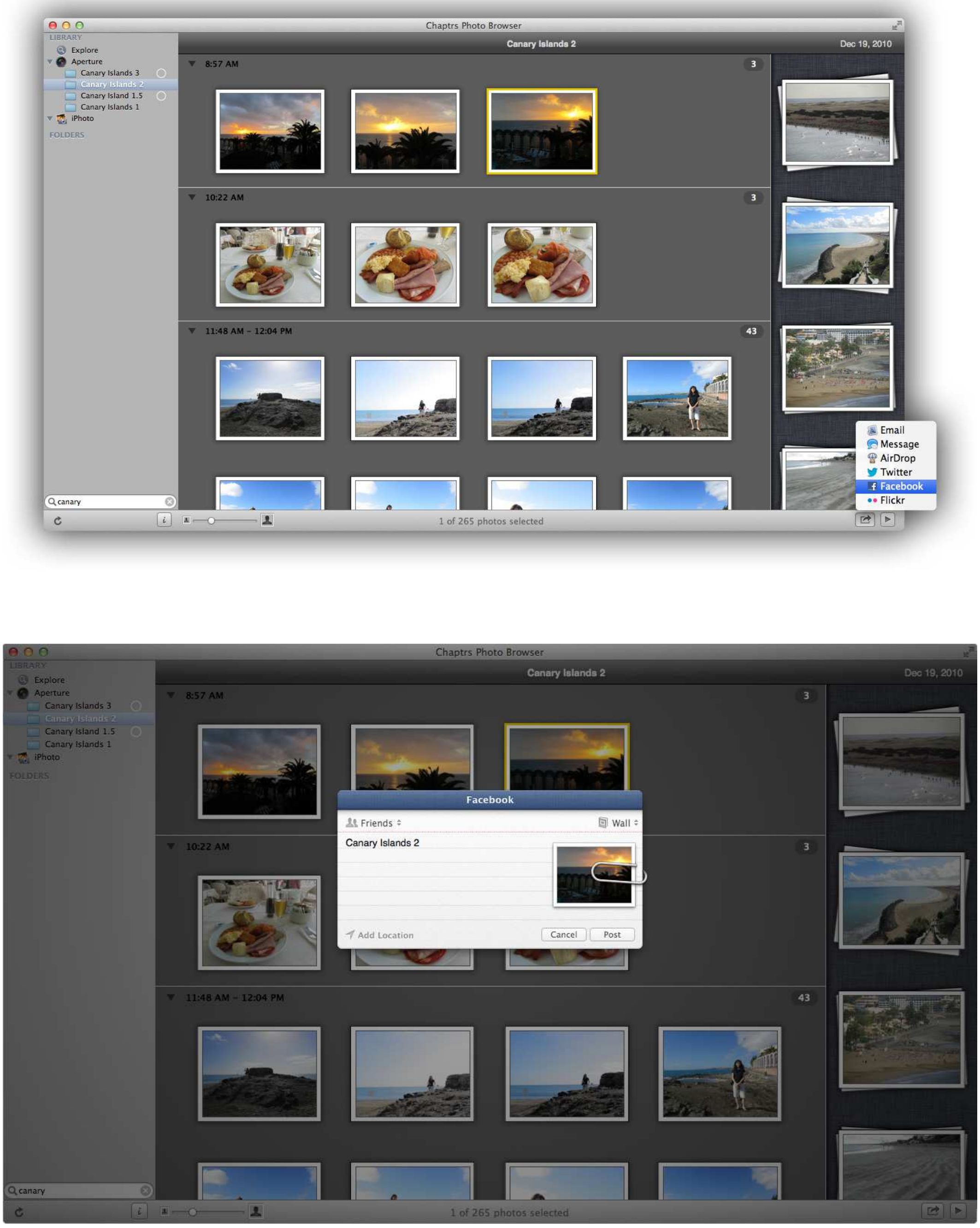
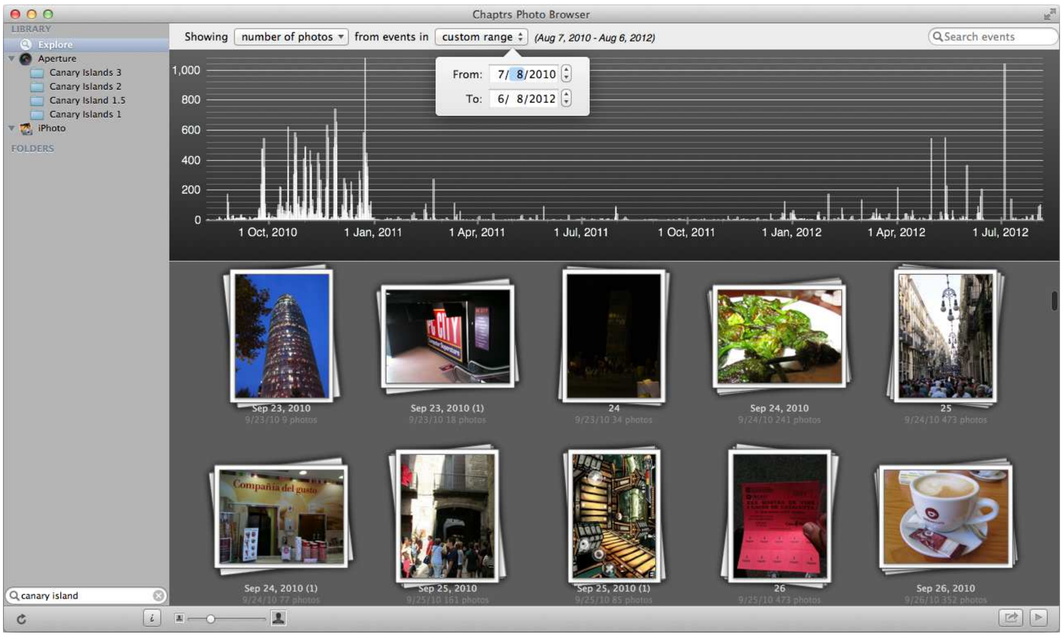


Figure 5.8: User shares selected photos and/or chapters to his/her social networks, or performs a drag-and-drop operation to a folder to copy the photos into the folder, *e.g.* the desktop.



Showing number of photos from events in custom range (Aug 7, 2010 - Aug 6, 2012)

From: 7/ 8/2010

To: 6/ 8/2012

Figure 5.9: The Explore user interface in CHAPTRS ver. 2 allows user to navigate events from all their photo libraries using a graphical overview.

In addition, CHAPTRS ver. 2 also provides event-level navigation through searching and browsing with a customizable sort (alphabetically on event title or by event time). Searching and browsing is further enhanced by coupling event thumbnails with a graphical visualization of all events (see Figure 5.9). When the user enters a query to search, the graph and the event thumbnails update to reflect the results of the search. When there is no search query, all the events are shown. The graph can be zoomed-in and out. When the graph's zoom level is changed, thumbnails corresponding to events that are outside of the zoom region are not displayed. When an event thumbnail is selected, its corresponding bar in the graph is similarly selected and highlighted with the selection color.

5.3 Event Photo Stream Segmentation

We implemented our automatic event photo stream segmentation algorithm in Objective-C and C++. In particular, we implemented our own hidden Markov model library with alternating observation types using the robust and fast Eigen linear algebra C++ library[1].

While there are no technical challenges in implementing our algorithm for CHAPTRS ver. 2, there were some practical considerations.

Some events read from iPhoto or Aperture contain non-photo image files, *e.g.* desktop wallpapers, screenshots, clipart images. This causes a problem for our algorithm which requires Exif data not present in such images. Our solution is to remove such images from the input into the algorithm. After we obtain the chapter boundaries of the filtered event photo stream from the algorithm, we add the non-photo images back into the event photo stream and surround each group of non-photo images with a chapter boundary. In doing so, we are assuming that a non-photo image is never in the same chapter as its adjacent photos and that non-photo images is always in the same chapter as its adjacent non-photo images.

Some events, *e.g.* "Vacation in Barcelona" may actually be a composite event that spans several days or even weeks, *i.e.* it contains smaller groups of photos each of which can be considered as an event on its own, *e.g.* "First day in Barcelona". Our solution for such composite events is to first insert chapter boundaries in between adjacent photos with a time gap of more than 4 hours. This is similar to how the constant threshold segmentation algorithm works. As a result, the composite event now consists of several segments of photo streams. We then run our algorithm on each segment and combine the multiple segmentation results to obtain the final segmentation.

Another consideration has to do with execution time. In our proposed algorithm, multiple runs are made to find various local maxima solutions. Subsequently, the results can be filtered for spurious solutions and the best of the remaining runs is taken as the final segmentation. To minimize execution time, we optimized the algorithm as follows:

1. We do not recompute the HMM parameters used for smoothing. These pa-

[1] `http://eigen.tuxfamily.org`

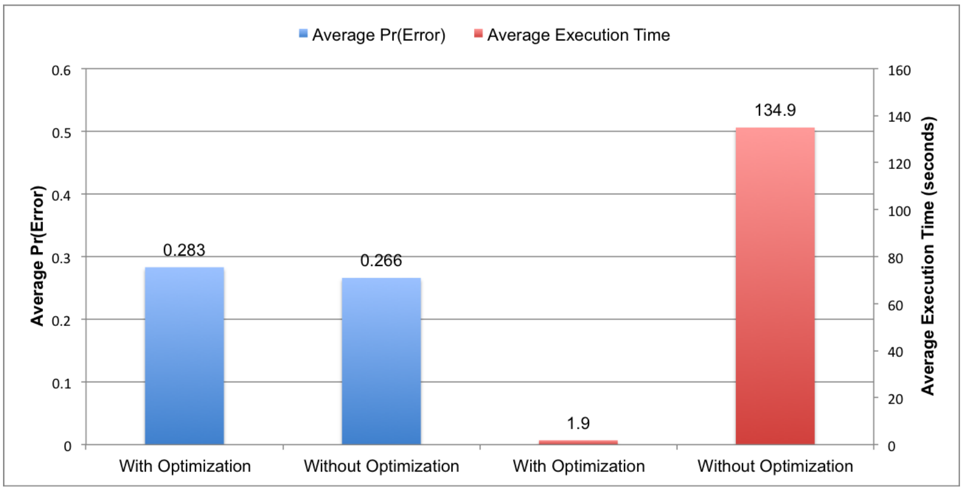


Figure 5.10: The optimizations allow CHAPTRS ver. 2 to have a significant reduction in execution time with only a minor reduction in performance.

rameters takes a considerable amount of time to learn due to the size of the dataset used for smoothing.

2. We use the same smoothing weights for any event photo stream we want to segment. Like the first optimization, learning the smoothing parameters takes a considerable amount of time due to the size of the dataset used for smoothing. Unlike the first optimization however, this may result in suboptimal segmentation results (see Figure 5.10). This reduction in performance brings down our results to be only comparable with the best state-of-the-art baseline discussed in Chapter 3, which scored a Pr_{error} of 0.281.

3. Because of the previous two points, the algorithm becomes deterministic and multiple runs yield the same solution. As such, we only need to run the algorithm once.

We justify the difference in performance through our optimization because the reduction in execution time is much more significant, as shown in Figure 5.10, *i.e.* close to a 99% reduction from 134.9 seconds to 1.9 seconds.

We note that the time values reported in Figure 5.10 excludes the time taken to compute the photo features. In CHAPTRS ver. 2, to extract features from a photo with a resolution of 3264 by 2448 pixels takes 0.1 seconds. An entire event with 200 of such photos however, takes only 13.1 seconds due to code-level multi-core and multi-threaded optimizations. This still amounts to a considerable amount of time depending on the number of photos in the event. In practice, CHAPTRS ver. 2 alleviates these concerns in several ways:

1. CHAPTRS ver. 2 runs the event photo stream segmentation as a background thread. The default setting is to have the segmentation run automatically in the background so that when the user wants to view an event, it would already be segmented into chapters (see Figure 5.11).

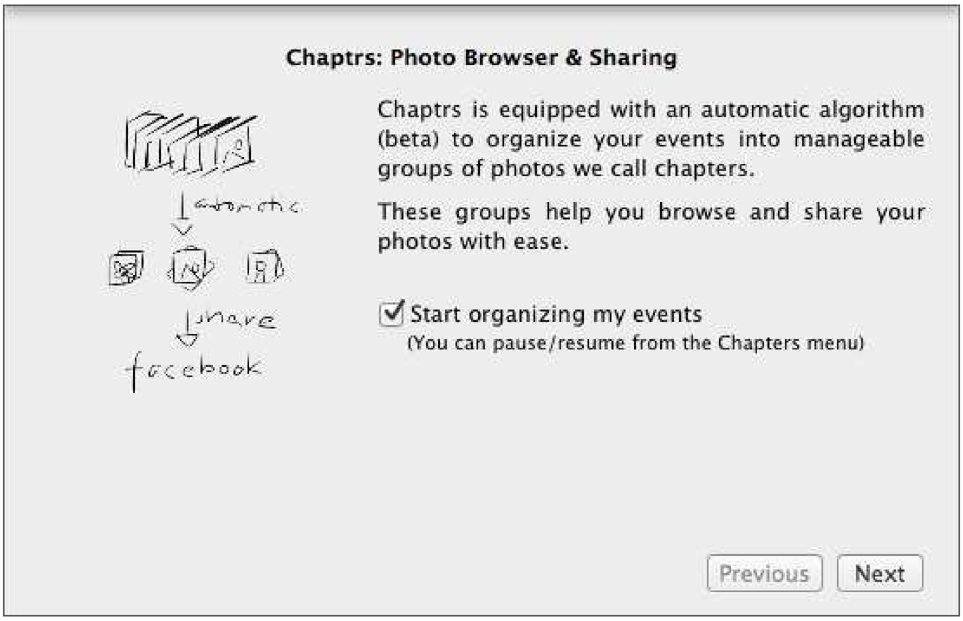
2. If the user views a new event where the features have not been read yet, CHAPTRS ver. 2 visualizes the feature extraction progress by incrementally loading the photos for display as it processes the features. This transparency aims to provide real-time feedback to the user. At the same time, users can be occupied with browsing through the unsegmented but already-loaded photos of the event.

3. CHAPTRS ver. 2 follows a strict separation between its main user interface thread and its background worker threads to ensure a responsive user interface and a good user experience, *e.g.* the user can switch to browsing / searching other events with no apparent penalty while the current event photo stream is being segmented.

5.4 Chapter-based Photo Organization

In our user study in Chapter 4, we obtained the following insights on how people organize their photos in each event:

1. Users value chapter consistency more than the chronological order of the photos in grouping photos into chapters



Chaptrs: Photo Browser & Sharing

Chaptrs is equipped with an automatic algorithm (beta) to organize your events into manageable groups of photos we call chapters.

These groups help you browse and share your photos with ease.

☑ Start organizing my events
(You can pause/resume from the Chapters menu)

Figure 5.11: Dialogue window in CHAPTRS ver. 2 explaining the automatic event photo stream segmentation, which is enabled by default to run in the background and can be toggled with the provided checkbox

2. Criteria for chapters include moment, object, location, photography type, and intention.

3. Choice of chapter criteria and granularity for chapter groupings are very subjective

A direct implication of these findings on the design of CHAPTRS ver. 2 is that users must be afforded with the freedom to customize the arrangement of the photos with as much ease as possible. In a desktop environment where mouse / trackpad use is the norm, users are already familiar with the drag-and-drop operation. CHAPTRS ver. 2 lets users perform drag-and-drop operations on the photos and/or chapters to fine-tune their photo arrangements (see Figure 5.12).

In our findings for the second task of our user study: *Find a given photo from a familiar event photo* in Section 4.7, we observed that participants used chapter groupings to skip chapters that they know will not contain the photo, and look deeper into chapters that might. To support this type of search behavior, we imple-

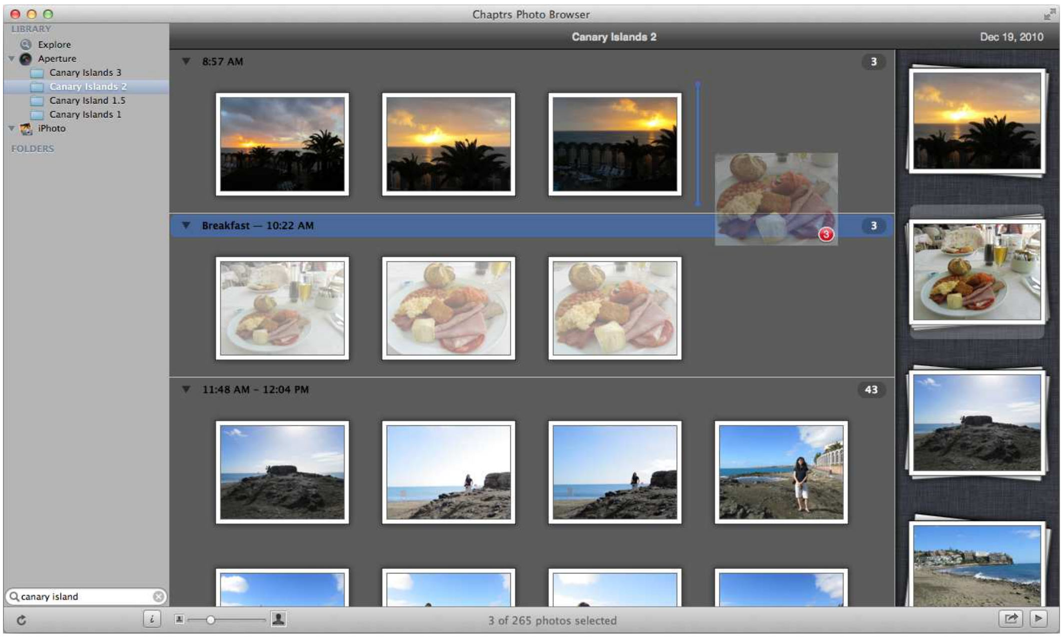


Figure 5.12: Photos can be rearranged in the grid-stacking layout. Similarly, chapters can be rearranged in the Chapter Sidebar. Dropping photos or chapters into a chapter in the Chapter Sidebar moves the photos or chapters into the chapter. Dropping photos into an empty space in the Chapter Sidebar creates a new chapter with the photos.

mented a feature that is activated when the user hovers the mouse pointer over a chapter thumbnail. As the mouse pointer hovers over the chapter thumbnail from left to right, the thumbnail is replaced with photos from the chapter, consecutively from the first photo to the last.

We note that the chapter thumbnails are presented with aesthetics suggesting that it represents a stack of photos. This design decision was done to drive the chapter analogy further, that a chapter is a group of photos. These aesthetics were also independently suggested by two of our user study participants.

To continue to support users seamlessly transitioning between photo-driven and story-driven storytelling methods (Balabanović et al., 2000), as was observed in the user study, we also implemented QuickLook support in CHAPTRS ver. 2. QuickLook is a Mac OS X system-wide mechanism where the user can press the spacebar key to preview a selected item. The preview appears in a separate window above the currently active application. CHAPTRS ver. 2 supports this preview mechanism which can also be triggered by simply double-clicking on a photo. While the preview window is visible, arrow keys will let users navigate to adjacent photos, effectively changing the photo currently being previewed. This essentially supports the photo-driven storytelling mechanism that we saw used by some participants in the user study. At any time, the user can press escape to close the preview window and resume a story-driven storytelling method or by summarizing the story, chapter by chapter.

In the user study, participants also mentioned that they often share their photos to Facebook or via email. Additionally, they also mentioned that only a subset of the photos would be shared, not all the photos from an event. To support this sharing behavior, CHAPTRS ver. 2 lets users share selected photos and / or chapters to their social networks, or perform a drag-and-drop operation to a folder or to other applications (*e.g.* into a Gmail compose window in a web browser).

5.5 Layout

In our user study, the grid-stacking layout was the most preferred layout for chapter-based photo organization. In addition, participants like the film strip in the bi-level layout as it afforded an overview of all the event photos using chapter thumbnails. However, they still prefer the bi-level layout less than the grid-stacking layout due to the wasted screen space.

In CHAPTRS ver. 2, we kept the grid-stacking layout as the primary means of presenting the event photos, grouped by chapter. In addition, we added a **Chapters Sidebar** on the right to display chapter thumbnails, similar to the film strip from the bi-level layout. Unlike the film strip however, the chapter thumbnails are laid out vertically, not horizontally. This change simplifies the navigation for the user as both the event photos and the chapter thumbnails now scroll vertically.

To further harmonize the navigation of the event photos and the chapter thumbnails, CHAPTRS ver. 2 synchronizes the selections of the photos and the chapter thumbnails as follows:

1. When all the photos of a chapter is selected, the corresponding chapter thumbnail will also be selected.

2. Similarly, when a chapter thumbnail is selected, all the photos of that chapter will be selected.

3. Double-clicking on a chapter thumbnail will cause its corresponding event photos to scroll into view.

This synchronization of selections further drives the association between the chapter thumbnails and the event photos, working on top of the aesthetics of the chapter thumbnails as a stack of photos.

5.6 Conclusion

In this chapter, we have described CHAPTRS ver. 2, a chapter-based photo browser that complements event-based photo organization. We outlined how we integrated our event photo stream segmentation method into the browser and described the practical considerations and the resulting optimizations. We also described how findings from our user study affected the design decisions in terms of layout and functionality.

CHAPTRS ver. 2 was built as a Mac OS X application and has been released on the Mac App Store for free. We describe our rationale for this decision in the next chapter as we describe our methodology for using the Mac App Store as a platform to reach a large user base in constructing a dataset to further research in personal digital photo libraries.

Chapter 6

Data Collection

Researchers in personal digital photo libraries (DLs) require access to such DLs to conduct their studies. For example, works on photo summarization (Sinha et al., 2012), photo stream alignment (Yang et al., 2012), automatic albuming (Platt, 2000), and event photo stream segmentation (Gozali et al., 2012a) require various features and ground truth annotations from DLs. Accessing them and acquiring the data however, tends to be a challenging process especially when sizable data is desired. Common methods to obtain photos, such as from volunteers or from study participants, do not scale well to thousands of photo sets due to the remuneration costs and the limited reach that study advertisement has in gathering interested participants.

To collect ground truth annotations on such collected photos, even more human effort is required. For example, in automatic albuming, the ground truth is the true grouping of photos into separate events. In some works, the authors themselves produced the ground truth (Platt, 2000) or external annotators were employed (Pigeau and Gelgon, 2003), which may be problematic due to unfamiliarity, bias, or ignorance of events that transpired in the photos. The semantics associated with personal photos render these tasks difficult to annotate by parties not privy with the context of the photos.

For such reasons, studies often require that the photo owners themselves produce the ground truth (Loui and Savakis, 2003). Data collection thus involves both 1) accessing DLs, as well as 2) acquiring the efforts of the photo owners themselves to produce the ground truth annotations. These two issues exacerbate the difficulty in scaling up the data collection process.

In this chapter, we propose using popular application distribution channels such as the Mac App Store (MAS) to alleviate issues with cost and reaching potential study participants. We use our own research needs as a case study to explore using the MAS as a platform to acquire the needed data. To the best of our knowledge, this is the first study to explore collecting anonymous data from personal digital photo libraries at a large scale, *i.e.* our data collection application was downloaded by over 2,500 users in 60 days.

The contributions of our study is two-fold. First, we report and discuss our experiences with the design of the data collection application, timeline, visibility, and cost in using the MAS in Section 6.1. Secondly, we present the large collected data to the research community in Section 6.2, providing an in-depth analysis of a few pertinent features.

6.1 Data Collection

The goal of our study is to explore the MAS for data collection in personal digital photo libraries. Primarily, we were motivated by its large user base: on Jan 7th, 2011, after only 24 hours of being available, the MAS had received over one million downloads[1]. We hypothesize that with its large user base in multiple countries, using the MAS will increase the visibility of our study and thus yield more collected data.

[1] `http://apple.com/pr/library/2011/01/07macappstore.html`

6.1.1 Design

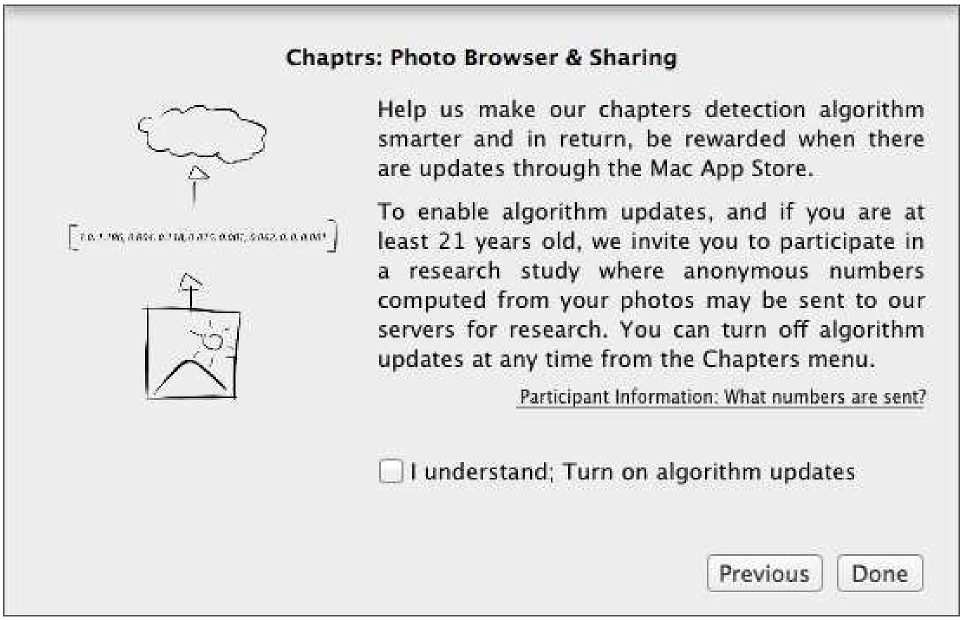
With any data collection method, a means for the collection needs to be designed and created. Even when the data to collect is small in scale, researchers still need to create a way to collect the data (*e.g.* from the volunteers) and a way for annotators to provide ground truth (*e.g.* for parameter tuning, supervised learning, or evaluation). When large-scale data collection is necessary, other scaling issues arise. For example, with crowd-sourcing platforms like Amazon Mechanical Turk (MTurk), recent works (Lee and Hu, 2012) have noted that verification questions or a qualification task is necessary to ascertain if the annotators are suited for the actual annotation task. Results also often have to be monitored and filtered for fake data from cheating crowd-sourcing users (Bloodgood and Callison-Burch, 2010).

For the MAS, its Review Guidelines[2] outline very specific *functionality requirements* for any application it distributes. One of the requirements states that applications *"that are not very useful"* may be rejected. As such, in the design of our application, we needed to relegate the data collection to a secondary function. While this seems counter-intuitive, we argue that generally, the data is collected to ultimately serve some practical purpose for the users; this purpose is a natural fit as the primary function of the application.

In our case, we published CHAPTRS ver. 2 in the MAS with the primary purpose of helping users organize their event photo streams. At the same time, we can use CHAPTRS ver. 2 for data collection, *i.e.* as a secondary function.

When CHAPTRS ver. 2 is launched for the first time, a window appears and explains how the automatic segmentation works and then appeals to the user to participate in the study to help improve the algorithm (Figure 6.1). Participation is voluntary and opt-in, but we entice users by stating that a future improved algorithm would be provided exclusively to participants. We also explained that the data is anonymized and the study has been approved by our Institutional Review

[2] `https://developer.apple.com/appstore/mac/resources/approval/guidelines.html`



Chaptrs: Photo Browser & Sharing

Help us make our chapters detection algorithm smarter and in return, be rewarded when there are updates through the Mac App Store.

To enable algorithm updates, and if you are at least 21 years old, we invite you to participate in a research study where anonymous numbers computed from your photos may be sent to our servers for research. You can turn off algorithm updates at any time from the Chapters menu.

Participant Information: What numbers are sent?

☐ I understand; Turn on algorithm updates

Figure 6.1: Window inviting users to participate in a study to help improve our algorithm

Board, as described in detail in a provided link[3].

To perform the data collection, CHAPTRS ver. 2 simply checks the user settings for study participation. If toggled `true`, CHAPTRS ver. 2 sends photo features to our server as and when they are computed or when the features was found to not have been sent yet. CHAPTRS ver. 2 also records ground truth annotations, by maintaining a log of all user annotations, *i.e.* the grouping of photos within an event into separate chapters, and sends this information to the server if the user is a study participant.

6.1.2 Cost

Currently, there is no mechanism in the Mac Software Development Kit (SDK) to allow MAS developers to send money to their users, and thus we opted not to remunerate participants monetarily. This reduces the overall cost of the study as it no longer grows with the number of participants. At the same time, the participants are

[3] `http://chaptrs.com/research`

more likely to be users who are genuinely interested in helping to improve the algorithm so they can benefit from the future algorithm, unlike many crowd-sourcing users who cheat to get their monetary rewards (Bloodgood and Callison-Burch, 2010).

We made CHAPTRS ver. 2 a free application to maximize number of downloads. All the cost in the study is then attributed to the Mac Developer Program annual fee of 99 USD. Past works with MTurk (Lee and Hu, 2012) have reported paying about 0.02 USD per annotation *on top of the 60.50 USD Amazon fee* and some paid 0.10 USD per translation (Urdu into English) (Bloodgood and Callison-Burch, 2010). For data collections that involve no human judgement or annotation, *e.g.* collecting Short Message Service (SMS) messages, a recent work (Chen and Kan, 2012) has reported paying at most 0.01 USD per message.

In our case, we collected features from 20,778 photo sets, comprising of 473,772 photos, of which 60 sets have ground truth segmentations, comprising of 8,107 photos. This translates to 0.0002 USD per photo or if we attribute all the cost to the collected annotations, 0.012 USD per annotation[4].

When we consider the first 19 days of the study — the time taken by (Lee and Hu, 2012) to collect 2,500 annotations from MTurk — we collected 5,787 photo sets, comprising of 227,969 photos, of which 23 sets have ground truth segmentations, comprising of 4,559 photos. This translates to a similar cost of 0.02 USD per annotation, but without any other additional fees.

This illustrates another difference between our study and existing data collection methods. Because the cost of our study does not scale with the amount of data collected, the cost per collected data (*e.g.* photo or annotation) decreases with the duration of the study and with the number of concurrent studies.

[4] *i.e.* whether there are segment boundaries in the pairs of consecutive photos in a photo set

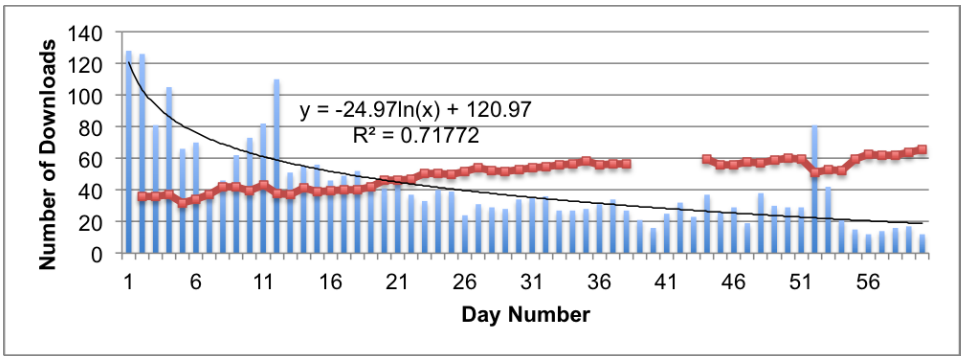


Figure 6.2: Daily number of downloads (columns) with trendline and average rankings (line) for CHAPTRS ver. 2 in the 60 days of study

6.1.3 Visibility

We define visibility as the exposure obtained by CHAPTRS ver. 2 to MAS users. This includes both MAS users who downloaded CHAPTRS ver. 2 and those who did not. While visibility is difficult to ascertain, we can produce a lower bound by determining the number of MAS users who downloaded CHAPTRS ver. 2. In the 60 days that we conducted the study, the daily number of downloads can be seen in Figure 6.2.

In the figure, the trendline that best matches the decrease in number of downloads over time is logarithmic: $y = -24.97ln(x) + 120.97$ with a coefficient of determination, $R^2 = 0.71$, where y and x correspond to the number of downloads and the day number respectively. We report this trendline in hope that the research community can find it helpful to estimate future downloads of their apps given their initial download counts.

We note that there are two anomalous spikes in the number of downloads on Day 12 and Day 52. Both spikes is attributed to the unusually high number of downloads in the Japan MAS on those days (47 and 38). These high number of downloads are caused by a snowballing effect from CHAPTRS ver. 2 taking the number 2 and 4 positions in the top photography category in the Japan MAS. The line graph in Figure 6.2 plots the average photography category ranking for CHAP-

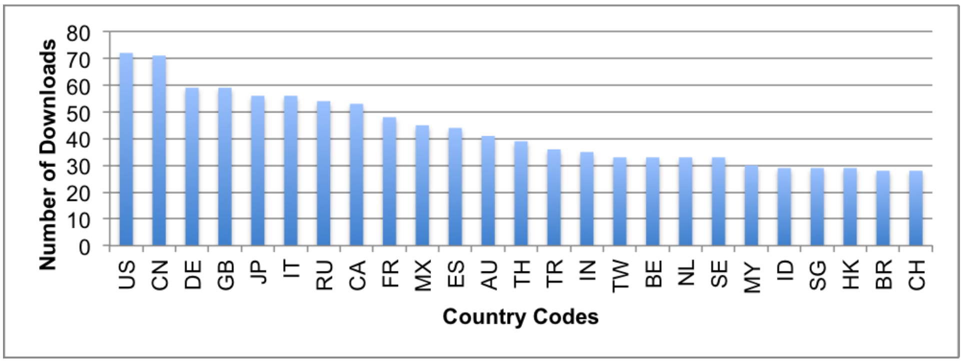


Figure 6.3: Top 25 countries with highest number of downloads

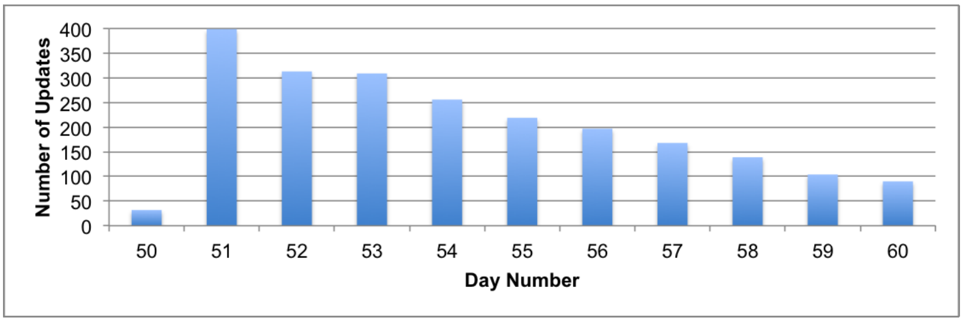


Figure 6.4: Number of updates from Day 50 to 60

TRS ver. 2 among various MAS stores. We can observe that the ranking decays linearly with time. Figure 6.3 shows a time series plot of the top 25 countries with the highest number of downloads. This ranking shows relative market sizes that would be useful for planning pilot studies.

As CHAPTRS ver. 2 is a free application, one tendency is for users to download and delete the application after only a brief experience. This is undesirable especially if the data collection is meant to contribute to a longitudinal study. To estimate the percentage of deletions, we submitted an update to the MAS. As the MAS only notifies updates to users with the application still installed, this gives us a good estimate. The update was released on Day 50 (see Figure 6.4). Comparing the number of downloads in the first 49 days (2,261) and the number of updates in the last 11 days (2,226), we can estimate that there is only at most a 1.5% deletion rate.

6.1.4 Timeline

It took 19 days to collect 23 photo sets with ground truth annotations, comprising of 4,559 photos. In the same amount of time, (Lee and Hu, 2012) collected 2,500 music mood annotations using MTurk. A work on SMS collection (Chen and Kan, 2012), which was considerably simpler as it involved no annotations from contributors, reported less success with 43 submissions (over 200 SMS per submission on average) over 40+ days.

We note that there is some temporal overhead with using MAS as a distribution channel. This is because applications need to undergo a review process before it becomes available for download. The review time fluctuates over time and usually takes 1-2 weeks[5]. Additional time is required for resubmission if the application is rejected.

6.2 Dataset

While there are publicly available datasets, *e.g.* COREL database, there are none that are event photos from personal photo libraries. We have previously noted that researchers have so far made use of their own collections to conduct studies. This poses a hurdle for new researchers. In practice, producing a public dataset of personal photos is challenging due to the private nature of the photos and their semantics.

We believe that a compromise is possible. The data we collected is a “blind” dataset of personal photos because the photos themselves are not in the dataset. Instead, only anonymized photo features and annotations are contained[6].

The dataset currently contains features that we use for our own work on event photo stream segmentation: time gap, focal length, aperture diameter, LogLight, and an 8-bin color histogram, but can be easily extended to collect others.

[5] Trend is reported at reviewtimes.shinydevelopment.com

[6] `http://wing.comp.nus.edu.sg/~jeprab/chaptrs_dataset/`

In the absence of the original photos, any micro or qualitative analysis that involves accessing semantic information would not be feasible. Instead, the focus of this dataset is the availability of data for quantitative analysis. Here we provide some quantitative analysis of the data set, details of which are packaged with the dataset.

Using K-means, we clustered the color distributions and searched for an optimal value for k, $k < 9$, which was found to be 6. Figure 6.5 shows the color distributions of the cluster centroids. We observe that there is a large percentage of black in all clusters due to the binning of dark colors to the nearest color, black. We also observe that Cluster 2 represents the blue/cyan photos while the red/yellow photos are represented by Cluster 3. These two clusters thus show the color distribution of the "blue/cyan" photos and "red/yellow" photos in the dataset. The other three clusters seem to represent different ratios of white to black while the ratios of the remaining 6 colors remain fairly constant.

We also analyzed for bursts of photo taking activity (Kleinberg, 2002), *i.e.* a sequence of photos (> 1) taken in succession with a certain average time gap. In our analysis, we looked for 15 kinds of bursts, each with a different average time gap[7]. Figure 6.6 shows the number of bursts found and the average number of photos for each kind of burst. We observe that the most frequent burst has an average time gap of 9 seconds. Also, the burst with the lowest average time gap in our analysis has the highest average number of photos. This suggests that when people take photos in quick succession ($\sim$1 seconds), they do so with 4 photos on average.

Lastly, Figure 6.7 shows a histogram of LogLight values. We have also fitted a two-mixture Gaussian to the histogram ($\mu = \{-4.91, -1.47\}$, $\sigma = \{0.74, 2.35\}$, $\lambda = \{0.26, 0.74\}$), suggesting that the LogLight values correspond to two normal distributions, that plausibly represent day (left mixture) and night (right mixture)

[7] While photos taken > 1 min apart can hardly be considered a burst, we analyze such "bursts" for completeness

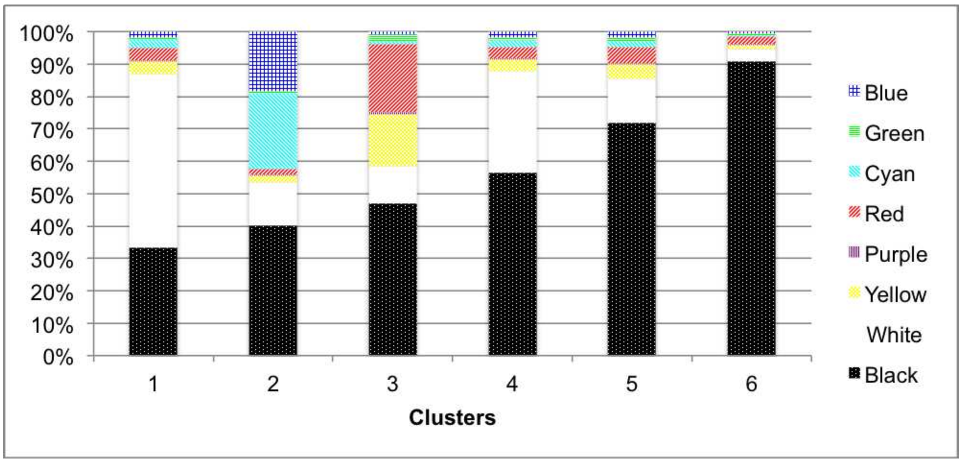


Figure 6.5: Color distributions of the six cluster centroids in the dataset

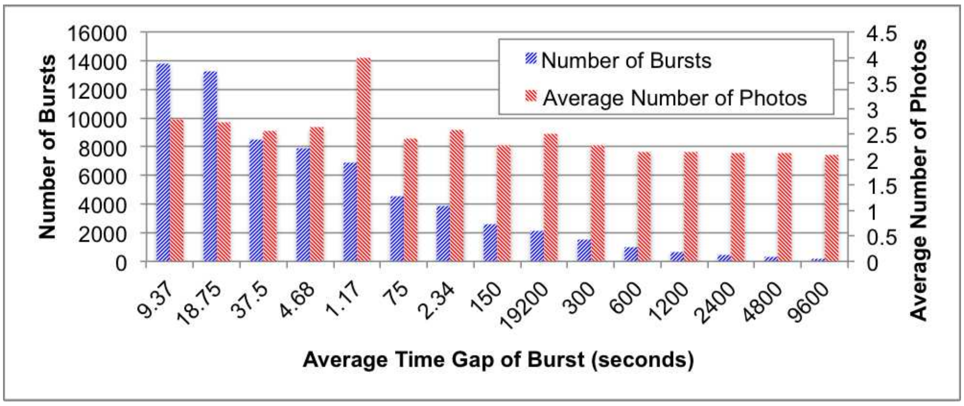


Figure 6.6: Dataset statistics of photo taking bursts

photos[8].

6.3 Conclusion

There is a lack of publicly available datasets for personal photos and we believe that the challenge lies is in the issue of privacy and in the difficulty in collecting any sizable amount of data. In this chapter[9], we have demonstrated how such a dataset can be constructed by collecting anonymous photo features and ground

[8] The LogLight value is small and large for high and low ambient lights respectively

[9] Also in (Gozali et al., 2013).

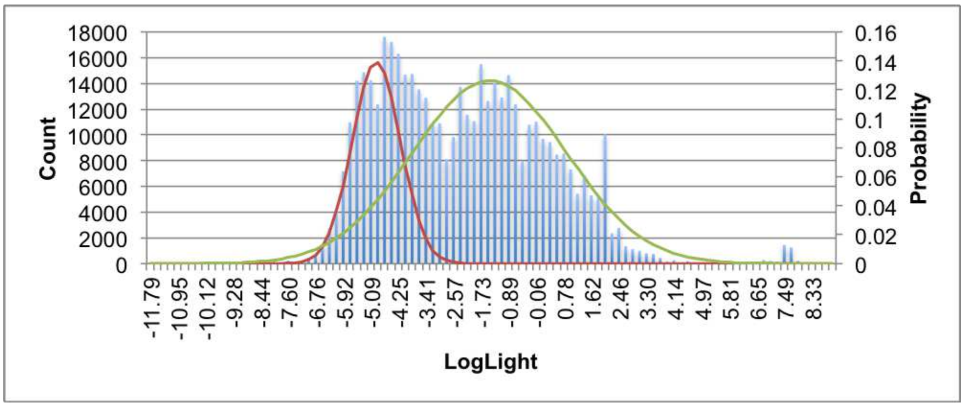


Figure 6.7: Histogram of LogLight values and the estimated Gaussian mixtures. The probabilities of the mixtures have been multiplied by their mixture ratios $(0.26, 0.74)$ to aid with the visualization.

truth annotations using an application distributed through the Mac App Store.

Aside from the review time overhead and conceptual overhead of designing the data collection application, we have demonstrated that the MAS with its large user base allows CHAPTRS ver. 2 to achieve high number of downloads, collects data at a faster rate and with lower cost than the data collection experiences from some recent works.

Ultimately, there is a self-filtering process because only genuinely interested users would volunteer to participate in the studies. This is in contrast with other data collection means, *e.g.* crowd-sourcing platforms where some users may only be interested in the monetary remunerations.

We note that in the works that we have reviewed in this chapter, the types of data and annotations collected are very different and thus we should not discount the possibility of confounding variables affecting our comparisons. Nonetheless, our experiences with CHAPTRS ver. 2 can stand on its own and shows that the MAS provides a fruitful and viable alternative for data collection especially in reaching out to personal digital photo libraries. In the same spirit, applications like CHAPTRS ver. 2 can be used to collect other anonymous features from the photos to expand on our dataset and its analysis.

Chapter 7

Conclusion

We began this thesis with the hypothesis that a "*chapter-based photo organization provides a better user experience than event-based photo organization in a photo browser for a personal digital photo library*". In the preceding chapters, we have made several key findings in support of this hypothesis.

We found that for event photo stream segmentation, visual or time features alone do not work well. In using features from an event photo stream, we made the key observation that the feature types alternate in the event photo stream. In our feature and structure analysis, we found that simple features and structures work best. While the reason for this is rooted in the data sparsity of the task, using simple features and structures also helped us to reduce the time taken for feature extraction in our photo browser, CHAPTRS ver. 2, an important goal to ensure less waiting time and good user experience.

In our user study, we found that users care more for how the chapters group their event photos than for the chronological order of the photos. We found a variety of different criteria that users may employ to group event photos into chapters: moments in the event, object, location, photography type, or by intention. The grid-stacking layout, the most preferred photo layout in the study, supports these findings. It displays each chapter as a grid of photos, with each chapter displayed separately from one another. Users were less concerned with the screen space usage of such a layout.

Additionally, the user study also revealed that for event photo stream segmentation, having a low miss rate, *i.e.* the method misses a low number of segment boundaries, is more important than having a low false alarm, *i.e.* the method produces a low number of false segment boundaries. If we factor this finding into the metric we used for our evaluation, our method would further outperform the baselines because of their tendency for high miss rates.

In constructing a dataset of anonymous photo features, we also found that using a popular application distribution channel, the Mac App Store, allows researchers such as ourselves to reach a large number of potential study participants and their personal digital photo libraries. Traditionally, even a small-scale data collection would have to be done with a lot of manual effort to publicise the study and attract volunteers. With this methodology, datasets can be created to further research in personal digital photo libraries.

7.1 Contributions

In supporting our hypothesis, this thesis makes the following contributions in the field of personal digital photo libraries:

1. **Event Photo Stream Segmentation** — We explored and proposed an unsupervised method for event photo stream segmentation. In doing so, we explored and analyzed a variety of photo features and model structures. We evaluated our method with a variety of baselines and showed how our approach outperforms all the baselines with statistical significance.

2. **Chapter-based Photo Organization User Study** — We conducted the first user behavior study on chapter-based photo organization. We drew insights from exploring fundamental issues of organization criteria and the affects on common photo-related tasks, such as storytelling, searching, and interpretation.

3. **Chapter-based Organization Photo Layout** — We conducted the first photo layout study on chapter-based photo organization. We explored several well-known photo layout aspects — view hierarchy, chronological order, and screen space usage — and their effects on common photo-related tasks.

4. **CHAPTRS Photo Browser** — We developed a fully-implemented publicly available chapter-based photo browser, CHAPTRS ver. 2. Our photo browser embodies all our work and findings from the unsupervised method, the photo organization study and photo layout study. Using CHAPTRS ver. 2, we constructed a dataset of anonymous photo features for the research community and report on our experience in assembling such a large anonymous dataset from personal digital photo libraries.

7.2 Limitations and Future Work

We recognise that this thesis has several limitations and also makes room for further work in the area of chapter-based photo organization. First, our method for event photo stream segmentation is only complementary to automatic albuming methods for event-based photo organization. Our method cannot be used for automatic albuming, *i.e.* to find events from a photo collection. This limitation is caused by the nature of our generative approach and the structure of the HMM used in our approach. While a unified solution may seem more elegant, we believe that our current framework where our method complements existing event-based photo organization methods is better because the framework allows less coupling between the two levels of organization — event and chapter — so that each level can be organized independently with different methods. In particular, chapters following different grouping criteria can be organized by different methods. The challenge for future work would then be to predict user organizational needs, automatically select the appropriate methods, and present them as suggestions to the user.

Second, our approach is unsupervised and as such, does not make use of information from available ground truth segmentations. At present, the amount of available ground truth segmentations is still limited, even including the ones in our dataset. Going forward, we hope more features and ground truth will be accessible for personal digital photo libraries. With such data — as is the case in the speech community and its usage of HMM-based solutions — supervised solutions trained using ground truth segmentations and labelled data will be feasible. The challenge for the research community would be to create supervised models that are semantically grounded with how photographers take photos, similar to Barry's cognitive model (2005) of how videographers think when creating a story; they observe the world, decide what to record, record a shot, and then reflect on its influence on the story.

Third, existing literature on personal photography reported that users did not find grouping photos by their visual appearance as useful at the photo collection level. In our study on chapter-based photo organization, we found the opposite to be true. As such, there is room for such automatic organization tools based on visual appearance to help users group event photos into chapters. The challenge here would be to balance the use of computationally-intensive features and the accuracy of the resulting visual organization.

Lastly, our photo layout study has identified photo layout aspects that are important for chapter-based photo organization. We hope these findings and that from the photo organization study will inform the design of future novel user interfaces for chapter-based photo browsers. The challenge would be to apply these user interfaces to both traditional and emerging use cases, *e.g.* accessing online digital photo libraries ("in the cloud") such as Apple's iCloud Photo Stream where a user's online photos are presented as a single continuous stream of photos from the past 30 days.

7.3 Towards An Automatic Personal Digital Photo Library

Our personal photos are our treasure troves. While we often find ourselves disinclined to invest our precious time to organize them, the memories our photos represent is truly priceless. And unlike the pixels which we can preserve for posterity in a variety of physical media, the semantics that are associated with the photos cannot be so easily preserved, not without effort and annotations on our part.

One ultimate goal for personal digital photo libraries is then to automate our tasks. Central to this automation is organization, an essential pre-processing step useful for other tasks such as annotation, summarization, and life logging. As our knowledge in automatic photo organization grows, the other tasks can subsequently benefit as well.

References

Paul André, Max L. Wilson, Alistair Russell, Daniel A. Smith, Alisdair Owens, and m.c. schraefel. 2007. Continuum: designing timelines for hierarchies, relationships and scale. In *Proc. of ACM Symposium on User Interface Software and Technology*, pages 101–110.

Marko Balabanović, Lonny L. Chu, and Gregory J. Wolff. 2000. Storytelling with digital photographs. In *Proc. of ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 564–571.

Barbara A. Barry. 2005. *Mindful Documentary*. Ph.D. thesis, Massachusetts Institute of Technology, June.

Leonard E Baum, Ted Petrie, George Soules, and Norman Weiss. 1970. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics*, 41(1):164–171.

Benjamin B. Bederson. 2001. Photomesa: a zoomable image browser using quantum treemaps and bubblemaps. In *Proc. of ACM Symposium on User Interface Software and Technology*, pages 71–80.

Michael Bloodgood and Chris Callison-Burch. 2010. Using Mechanical Turk to build machine translation evaluation sets. In *Proc. of NAACL-HLT 2010 Workshop on AMT*.

Matthew Brand. 1997. Coupled hidden Markov models for modeling interacting processes. Technical Report 405, MIT Media Lab, June.

Tao Chen and Min-Yen Kan. 2012. Creating a live, public short message service corpus: The NUS SMS Corpus. *Language Resources and Evaluation*, pages 1–37, Aug.

Chufeng Chen, Michael Oakes, and John Tait. 2006. Browsing personal images using episodic memory (time + location). In *Proc. of European Conference on Information Retrieval*, pages 362–372.

Ya-Xi Chen, Michael Reiter, and Andreas Butz. 2010. Photomagnets: supporting flexible browsing and searching in photo collections. In *Proc. of International Conference on Multimodal Interfaces and Workshop on Machine Learning for Multimodal Interaction*, pages 25:1–25:8.

Pei-Yu Chi and Henry Lieberman. 2010. Raconteur: from intent to stories. In *Proc. of International Conference on Intelligent User Interfaces*, pages 301–304.

Matthew Cooper, Jonathan Foote, Andreas Girgensohn, and Lynn Wilcox. 2003. Temporal event clustering for digital photo collections. In *Proc. of the 11th ACM International Conference on Multimedia*, pages 364–373.

Sally Jo Cunningham and Masood Masoodian. 2007. Metadata and organizational structures in personal photograph digital libraries. In *Proc. of International Conference on Asian Digital Libraries*.

Steven M. Drucker, Curtis Wong, Asta Roseway, Steven Glenner, and Steven De Mar. 2004. Mediabrowser: reclaiming the shoebox. In *Proc. of International Working Conference on Advanced Visual Interfaces*, pages 433–436.

Scott Fertig, Eric Freeman, and David Gelernter. 1996. Lifestreams: an alternative to the desktop metaphor. In *Proc. of ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 410–411.

Dayne Freitag and Andrew Kachites Mccallum. 1999. Information extraction with HMMs and shrinkage. In *Proc. of AAAI Workshop on Machine Learning for Information Extraction*, pages 31–36.

David Frohlich, Allan Kuchinsky, Celine Pering, Abbe Don, and Steven Ariss. 2002. Requirements for photoware. In *Proc. of ACM conference on Computer Supported Cooperative Work*, pages 166–175.

Yuli Gao, Clayton Brian Atkins, Phil Cheatle, Jun Xiao, Xuemei Zhang, Hui Chao, Peng Wu, Daniel Tretter, David Slatter, Andrew Carter, Roland Penny, and Chris Willis. 2009. Magicphotobook: designer inspired, user perfected photo albums. In *Proc. of the 17th ACM International Conference on Multimedia*, pages 979–980.

Ullas Gargi. 2003. Modeling and clustering of photo capture streams. In *Proc. of the International Workshop on Multimedia Information Retrieval*, pages 47–54.

Maria Georgescul, Alexander Clark, and Susan Armstrong. 2006. An analysis of quantitative aspects in the evaluation of thematic segmentation algorithms. In *Proc. of SIGdial Workshop on Discourse and Dialogue*, pages 144–151.

Andreas Girgensohn, Frank Shipman, Thea Turner, and Lynn Wilcox. 2010. Flexible access to photo libraries via time, place, tags, and visual features. In *Proc. of ACM/IEEE Joint Conference on Digital Libraries*, pages 187–196.

B. Gong and R. Jain. 2007. Segmenting photo streams in events based on optical metadata. In *Proc. of the 1st IEEE International Conference on Semantic Computing*.

Jesse Prabawa Gozali, Min-Yen Kan, and Hari Sundaram. 2012a. Hidden Markov model for event photo stream segmentation. In *Proc. of ICME 2012 Workshop on Human-Focused Communications in the 3D Continuum (HFC3D)*.

Jesse Prabawa Gozali, Min-Yen Kan, and Hari Sundaram. 2012b. How do people organize their photos in each event and how does it affect storytelling, searching and interpretation tasks? Technical Report TRC4/12, National University of Singapore Department of Computer Science, April.

Jesse Prabawa Gozali, Min-Yen Kan, and Hari Sundaram. 2012c. How do people organize their photos in each event and how does it affect storytelling, searching and interpretation tasks? In *Proc. of ACM/IEEE Joint Conference on Digital Libraries*, pages 315–324.

Jesse Prabawa Gozali, Min-Yen Kan, and Hari Sundaram. 2013. Constructing an anonymous dataset from the personal digital photo libraries of mac app store users. In *Proc. of ACM/IEEE Joint Conference on Digital Libraries*, pages 305–308.

Adrian Graham, Hector Garcia-Molina, Andreas Paepcke, and Terry Winograd. 2002. Time as essence for photo browsing through personal digital libraries. In *Proc. of ACM/IEEE Joint Conference on Digital Libraries*, pages 326–335.

David Huynh, Steven Drucker, Patrick Baudisch, and Curtis Wong. 2005. Time quilt: scaling up zoomable photo browsers for large, unstructured photo collections. In *Proc. of ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 1937–1940.

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

F. Jelinek and R. Mercer. 1980. Interpolated estimation of Markov source parameters from sparse data. In *Proc. of the Workshop on Pattern Recognition in Practice*.

David Kirk, Abigail Sellen, Carsten Rother, and Ken Wood. 2006. Understanding photowork. In *Proc. of ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 761–770.

Jon Kleinberg. 2002. Bursty and hierarchical structure in streams. In *Proc. of ACM Conference on Knowledge Discovery and Data Mining*, pages 91–101.

Allan Kuchinsky, Celine Pering, Michael L. Creech, Dennis Freeze, Bill Serra, and Jacek Gwizdka. 1999. Fotofile: a consumer multimedia organization and retrieval system. In *Proc. of ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 496–503.

Jin Ha Lee and Xiao Hu. 2012. Generating ground truth for music mood classification using Mechanical Turk. In *Proc. of ACM/IEEE Joint Conference on Digital Libraries*, pages 129–138.

K-F Lee. 1989. *Automatic Speech Recognition: The Development of the Sphinx System*. Kluwer Academic Publishers, AH Dordrecht.

Alexander C. Loui and Andreas E. Savakis. 2003. Automated event clustering and quality screening of consumer pictures for digital albuming. *IEEE Transactions on Multimedia*, 5(3):390–402, September.

David G. Lowe. 2004. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110.

Tao Mei, Bin Wang, Xian-Sheng Hua, He-Qin Zhou, and Shipeng Li. 2006. Probabilistic multimodality fusion for event based home photo clustering. In *Proc. of IEEE International Conference on Multimedia and Expo*, pages 1757–1760.

Timothy J. Mills, David Pye, David Sinclair, and Kenneth R. Wood. 2000. Shoebox: A digital photo management system. Technical Report 2000.10, AT&T Research.

P. Van Mulbregt, I. Carp, L. Gillick, S. Lowe, and J. Yamron. 1998. Text segmentation and topic tracking on broadcast news via a hidden Markov model approach. In *Proc. of the International Conference on Spoken Language Processing*, pages 2519–2522.

Mor Naaman, Yee Jiun Song, Andreas Paepcke, and Hector Garcia-Molina. 2004. Automatic organization for digital photographs with geographic coordinates. In *Proc. of ACM/IEEE Joint Conference on Digital Libraries*, pages 53–62.

Dan R. Olsen, Jr. 2007. Evaluating user interface systems research. In *Proc. of ACM symposium on User interface software and technology*, pages 251–258.

Antoine Pigeau and Marc Gelgon. 2003. Spatial-temporal organization of one's personal image collection with model-based ICL clustering. In *Proc. of the International Workshop on Content-Based Multimedia Indexing*.

Catherine Plaisant, Brett Milash, Anne Rose, Seth Widoff, and Ben Shneiderman. 1996. Lifelines: visualizing personal histories. In *Proc. of SIGCHI Conference on Human Factors in Computing Systems*, pages 221–227.

John C. Platt, Mary Czerwinski, and Brent A. Field. 2003. PhotoTOC: Automatic clustering for browsing personal photographs. In *Proc. of the 4th International Conference on Information, Communications & Signal PRocessing – 4th IEEE Pacific-Rim Conference on Multimedia*, pages 6–10.

John C. Platt. 2000. AutoAlbum: Clustering digital photographs using probabilistic model merging. In *IEEE Workshop on Content-based Access of Image and Video Libraries*, pages 96–100.

Lawrence R. Rabiner. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. of the IEEE*, 77(2):257–286.

Kerry Rodden and Kenneth R. Wood. 2003. How do people manage their digital photographs? In *Proc. of ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 409–416.

Kerry Rodden. 1999. How do people organize their photographs? In *Proc. of BCS IRSG 21st Annual Colloquium on Information Retrieval Research.*

Dong-Sung Ryu, Woo-Keun Chung, and Hwan-Gue Cho. 2010. Photoland: a new image layout system using spatio-temporal information in digital photos. In *Proc. of the ACM Symposium on Applied Computing*, pages 1884–1891.

Philipp Sandhaus and Susanne Boll. 2011. Semantic analysis and retrieval in personal and social photo collections. *Multimedia Tools Appl.*, 51:5–33.

Philipp Sandhaus, Sabine Thieme, and Susanne Boll. 2008. Processes of photo book production. *Multimedia Systems*, 14(6):351–357.

Pinaki Sinha and Ramesh Jain. 2008. Classification and annotation of digital photos using optical context data. In *Proc. of the International Conference on Image and Video Retrieval*, pages 309–317.

Pinaki Sinha, Sharad Mehrotra, and Ramesh Jain. 2012. Summarization of personal photologs using multidimensional content and context. In *Proc. of ACM International Conference on Multimedia Retrieval.*

Grant Strong and Minglun Gong. 2009. Organizing and browsing photos using different feature vectors and their evaluations. In *Proc. of the ACM International Conference on Image and Video Retrieval.*

Jun Xiao, Nic Lyons, C. Brian Atkins, Yuli Gao, Hui Chao, and Xuemei Zhang. 2010. iphotobook: creating photo books on mobile devices. In *Proc. of the 18th ACM International Conference on Multimedia*, pages 1551–1554.

L. Xie, S.-F. Chang, A. Divakaran, and H. Sun. 2002. Learning hierarchical hidden Markov models for video structure discovery. Technical report, Columbia University, December.

Jianchao Yang, Jiebo Luo, Jie Yu, and T.S. Huang. 2012. Photo stream alignment and summarization for collaborative photo collection and sharing. *IEEE Transactions on Multimedia*, 14(6):1642 –1651, Dec.

Ming Zhao, Yong Wei Teo, Siliang Liu, Tat-Seng Chua, and Ramesh Jain. 2006. Automatic person annotation of family photo album. In *Proc. of the International Conference on Image and Video Retrieval*, pages 163–172.