PhotoTOC: Automatic Clustering for Browsing Personal Photographs

John C. Platt **Mary Czerwinski** **Brent A. Field**
Microsoft Research
1 Microsoft Way
Redmond, WA 98052
{jplatt,marycz,brentfi}@microsoft.com

Abstract

This paper presents Photo Table Of Contents (PhotoTOC), a system that helps users find digital photographs in their own collection of photographs. PhotoTOC is a browsing user interface that uses an overview+detail design. The detail view is a temporally ordered list of all of the user's photographs. The overview of the user's collection is automatically generated by an image clustering algorithm, which clusters on the creation time and the color of the photographs. PhotoTOC was tested on users' own photographs against three other browsers. Searching for images with PhotoTOC was subjectively rated easier than all of the other browsers. This result shows that automatic organization of personal photographs facilitates efficient and satisfying search.

1 Introduction

Taking photographs with a digital camera is so convenient and low cost that it is easy for a user to generate more than 1000 photographs per year. This flood of photographs presents a user interface challenge: how can a user find photographs in his or her collection?

Previous work in image browsing, search, and management has largely concentrated on solving the problem of a user interacting with a large, impersonal, possibly annotated image database. Unfortunately, many of the lessons learned from that problem may not carry over to searching through one's own personal photographs. Unlike interacting with impersonal databases, users have very good memories about photographs within their personal collection. Also, users are often reluctant to spend effort annotating their own images: the images will often be stored in a small, shallow hierarchy of folders on a computer. Users can therefore spend a large amount of effort with browsing tools searching through disorganized collections for their photographs. These issues have not been studied by previous work, because user studies have not been performed on users' own photographs and folder structures.

We propose that users should interact with their personal photographs through the use of an image browser which automatically organizes the user's images. To this end, we present Photo Table Of Contents (PhotoTOC), a browser for personal digital photographs that uses a clustering algorithm to automatically generate a table of contents of a user's personal photograph collection. The clustering algorithm segments the stream of photographs into events by analyzing both the creation time of the photographs and their color histograms. PhotoTOC then automatically chooses one representative image per cluster to place into a table of contents. This table of contents is presented in an overview+detail [10] user interface.

Section 2 describes a new user interface: PhotoTOC. We present the algorithmic details of PhotoTOC in section 3. We describe a user study in section 4, which compares PhotoTOC to other browsers on searches through users' own photographs and folder structure. In that study, PhotoTOC did not sacrifice performance compared to other browsers. In addition, PhotoTOC was rated by users as the most efficient browser. Thus, PhotoTOC is the first automatically organized media browser that has scored reliably higher in subjective satisfaction than browsing with a user's own folder structure.

1.1 Related Work

There have been several image browsers proposed in the literature. In Similarity Pyramids [2] and the work of [12], photographs are organized and clustered according to their color. In [5], photographs are clustered according to their creation time. In other work [3, 6], the browsing interfaces work by strongly encouraging users to annotate their images. In PicHunter [4], instead of a fixed organization, a dynamic organization is created by having a user iteratively select one photograph out of four that is the most similar to the desired photograph. PhotoMesa [1] uses a zoomable user interface to browse personal photographs. In [7], time and color clustering are combined in order to separate events in a photograph browser. Note that none of the previous image browsers were studied with users' own photographs, nor were they tested against browsing users' own folder structures.

Automatic table of content generation has previously been proposed for media types other than photographs. Hypertext is an example of a media type that is amenable to automatic table of content generation [8]. Video has been temporally segmented for scene detection to build a video table of contents [13].

2 PhotoTOC User Interface

We have created a new image browser, called "PhotoTOC," for Photo Table of Contents. PhotoTOC is shown in figure 1. PhotoTOC is an overview+detail interface. The selection of the overview photographs is performed by a clustering algorithm, described in the Section 3. The detail pane contains an array of all of the user's photographs, sorted by creation date and shown as thumbnails. Clicking on a detail thumbnail shows the full-sized image. Clicking on an overview thumbnail from the left pane scrolls the detail pane to show the corresponding cluster within the entire list, with the first thumbnail of the cluster at the top of the pane. The thumbnail that was selected in the overview pane is highlighted in red in the detail pane, to orient the user and give feedback for the overview selection action. The user is free to use the overview pane to "power scroll" the detail pane, or simply scroll the detail pane and ignore the overview pane. We correctly anticipated that allowing the user the freedom of either using overview selection or detail scrolling would increase user satisfaction.

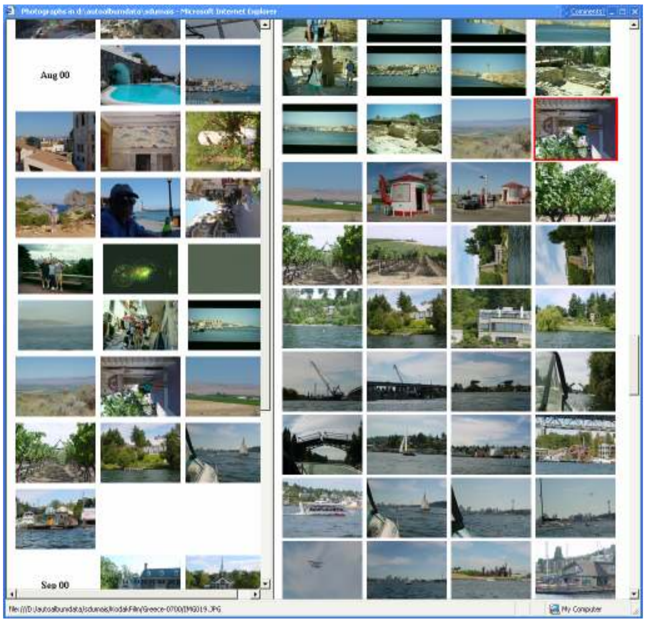

Figure 1: A screen shot of the PhotoTOC user interface

A demo of the PhotoTOC user interface is available on the web at http://research.microsoft.com/∼jplatt/ autoAlbum/ex2.html

3 Algorithms

PhotoTOC attempts to identify events in a user's collection. Identifying events from pure image information is very difficult. However, almost all digital cameras time stamp each photograph when the image is created.

Unfortunately, time stamps are sometimes incorrect due to an improperly set camera clock. Also, some users have scanned photographs, which do not contain photo creation time. The file creation date, file modification date, or filename can still be used to order the photographs, although extracting events is more difficult when the true creation time is missing. For cases where the creation time is missing or corrupt, PhotoTOC uses the order of the photographs plus the color information in the photographs to identify events [11]. Because digital photographs can be ordered in time even when the exact creation time is unavailable, adding color information is sufficient to identify events.

Thus, there are two different clustering algorithms that can be applied to a user's collection of photographs. One is time-based clustering, where the creation time is used to cluster the photographs. The other is content-based clustering, where the creation time is used only to order the photographs, and color information is then used to cluster. The time-based clustering is preferred when the data is reliable: the content-based clustering is used as a backup algorithm.

3.1 Clustering Algorithms

The goal of time-based clustering is to detect noticeable gaps in the creation time. A cluster is then defined as those photographs falling between two noticeable gaps. These gaps are assumed to correspond to a change in event. The time gap detection is adaptive: it compares a gap to a local average of temporally nearby gaps. A gap is then considered a change of event when it is much longer than the local gap average. Time gaps have a very wide dynamic range, thus the gap detection algorithm operates on logarithmically transformed gap times.

Time-based clustering first sorts the photographs by creation time. Then, if g_i is the time difference between picture i and picture $i + 1$ in the sorted list, g_N is considered a gap between events if it is much longer than a local log gap average:

$$\log(g_N) \geq K + \frac{1}{2d+1} \sum_{i=-d}^{d} \log(g_{N+i}), \qquad (1)$$

where K is a suitable threshold (chosen empirically to be $K = \log(17)$), and d is a window size (chosen to be $d = 10$). If $N + i$ refers to a photograph beyond the ends of the collection, the term is ignored, and the denominator $2d + 1$ is decremented for every ignored term. In effect, equation (1) compares a gap to the a local geometric average of gap times, and declares an new event when the difference is large enough. The algorithm that adaptively determines the gap between events is new to this paper: previous versions of time-based clustering [11] used a fixed threshold.

PhotoTOC combines time-based and content-based clustering because the creation times are not always reliable. A signal of unreliable creation time is that time-based clustering yields large clusters. Therefore, PhotoTOC uses this signal to combine the time-based and content-based kinds of clustering.

First, PhotoTOC extracts the creation time from the digital image. If this time is unavailable or considered corrupt (i.e., before Jan 1, 1999), then the file creation time is used. The images are sorted and then the time-based clustering algorithm is applied to the images. If any of the time-based clusters are too large (i.e., more than 23 images), then content-based clustering is applied to each large cluster, which produces a number of smaller clusters. The number of content-based clusters is 1/12 the number of photographs in the large cluster(rounded to the nearest integer). This choice of cluster number yields a "zoom factor" of the detail view of approximately 12. All of the resulting clusters are then displayed in the overview and detail panes. Details on the content-based clustering can be found in [11].

3.2 Representative Photographs

We use a new algorithm to choose one photograph from a cluster that is the most representative of that cluster. The photograph is chosen by measuring the Kullback-Leibler (KL) divergence between the histogram of every photograph in the cluster and the averaged histogram over all photographs in the cluster. More specifically, let P_{ij} be the normalized histogram count in bin i for picture j. Let A_i be the average histogram count in bin i over all images in the cluster. Then, the picture j is chosen to be representative when it maximizes

$$\sum_i P_{ij} \log(P_{ij}/A_i). \qquad (2)$$

If images of an event have a number of uniquely colored regions, then the image with the highest number of those regions will tend to get selected by the KL divergence metric. Poor quality images very rarely get selected by the KL divergence criteria: all of the selected overview photographs in the user study were of good quality. More sophisticated algorithms, based on object recognition, would be a subject for future work.

4 Experiment

The experiment was designed to test PhotoTOC versus two standard browsers (Folders and LightBox), and versus a previous version of the user interface (AutoAlbum [11]). Folders is a traditional folder browser with thumbnails for each image. LightBox is a thumbnail browser that shows all of the pictures in a flat, scrollable list, ordered by creation time (see Figure 2) . AutoAlbum is similar to PhotoTOC, except that the detail view only shows photos in the cluster selected by the user in the overview pane. In this experiment, AutoAlbum and PhotoTOC shared the same clustering and representative photograph algorithms. All browsers (except Folders) provided calendar hints (month and year).

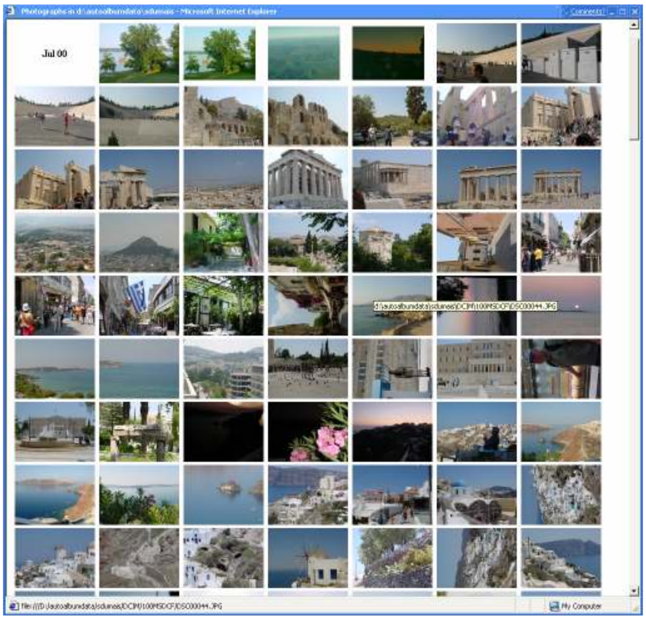

Figure 2: A screen shot of the LightBox user interface

The Folders browser is a standard hierarchical folder browser. Each folder was represented with an icon of a folder, four small thumbnails, plus a folder label; rather than with a representative thumbnail without a label, as in the AutoAlbum browsers. The contents of a folder are shown as thumbnails. A user can double-click on a folder to show its contents, and click on an "up" button to go up in the folder hierarchy. A folder tree view was also available.

The Folders browser was included to test whether automatic organization was the same or better than the user's organization using existing tools. The LightBox browser was included to test whether organizing by clustering (versus merely sorting) improved user performance and satisfaction. The AutoAlbum browser was included to test whether design iteration improved the browser.

4.1 Task Definition

When a user searches or browses for a digital photograph, they have an end goal in mind. For example, it could be showing the photograph to a friend or placing it on a web page. Under many of these scenarios, the user is searching for a particular photograph that has some significance. The user has a mental image of the desired photograph and searches his or her collection until a photograph that matches the mental image is found.

We could ask a user to repeatedly think of a photograph in his or her collection, and then find it. However, this could introduce uncontrolled variability into task difficulty. For example, the user may select the next photograph based on how easy it is to find that photograph in that particular browser.

In order to achieve better experimental control on task difficulty, we select a randomly chosen photograph from the user's own image set. This photograph is presented to the participant as a target for search. Showing this photograph emulates the mental image that the user has when searching for a desired photograph. In debriefing sessions with our study participants, we were told that this task fairly represents the actual task of trying to find a picture some period of time after storing it in a digital collection of images.

4.2 Experimental Details

There were 8 participants (1 female) with an average age of 36.9 years. Two participants were from Microsoft Research, the rest were not Microsoft employees. Each provided a personal set of digital pictures (their libraries ranged from 345 pictures to 1298 pictures, average size was 850 pictures). All users were at least intermediate Windows users as assessed by a background screening questionnaire. Participants had a range of experience with photography: skill levels ranged from casual photographers who simply took photographs of their vacations all the way up to a professional wedding photographer.

The picture browsers were executed on a high-end PC running a beta version of Windows XP. A NEC MultiSync FE1250 21" monitor was used. The user's display was a CRT set to 1280 by 1024 resolution: the browser occupied a 1024 by 1024 window, while the search target occupied a 256 by 256 image in the upper left hand corner of the screen. The maximum dimension of the image thumbnails for all browsers except Folders was 128. In the Folders browser, folder contents were shown as image thumbnails plus a filename for each image. The thumbnail plus filename filled a region of 144 by 96 pixels, which was comparable to the screen area per thumbnail of the other browsers. All thumbnails are computed only once and then cached, so that system response differences across the browsers were minimized.

The four browsers were presented to each participant in random order. For each browser, there were two practice trials and ten measured trials, for a total of 320 data points over all subjects. The experimenter ensured that the participant did indeed find the correct photo. When the participant locates the target image in the browser, he or she is instructed to press a "Finished" button in the experimental program window, which is displayed to the upper left side of the browser's window. This allows us to collect timing information across the various image browsers for comparison purposes. After each browser was used, a satisfaction questionnaire about that browser was presented to the participant.

4.3 Results

Two outliers were identified from the 320 data points. These were the only points more than 5 standard deviations from the mean (in log space), and both appeared to be unrealistically fast responses. These two values were replaced with the mean response rate (in log space) for a given browser and trial.

The completion times were transformed logarithmically, as is standard in the statistical analysis of response times. A Browser×Trial RM ANOVA was performed on the log task completion time data. The type of browser contributed significantly to the overall variance ($F(3, 21) = 3.191$, $p < 0.047$), resulting in a reliable main effect. The Light Box condition had a mean completion time of 28.4s, PhotoTOC had 37.3s, AutoAlbum had 45.1s, and the Folders browser had 58.7s, as shown in Figure 3.

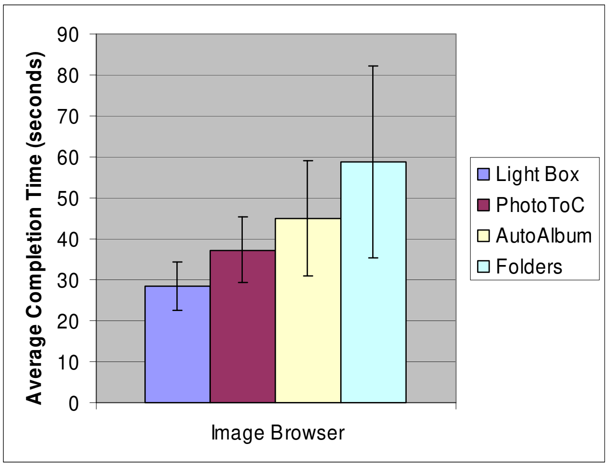

Figure 3: Mean task completion time for all four browsers, with error bars representing ±1 standard error of the mean

Pair-wise *post hoc* comparisons across the browser conditions showed that there was no significant difference in task completion time between any two browsers.

There was a significant linear correlation between the log of the number of photographs in a participant's database and the log of the task completion time. This accounted for 10.9% of the variance ($R^2 = 0.109$, $F(1, 318) = 38.85$, $p < 0.001$). The slope of the correlation is 0.883 ± 0.142, which is not significantly different from a linear time relationship between task completion time and size of database. No significant differences in the linear correlation were found between browsers. This confirms the linear time relationship in image browsing reported by [3], in this case on somewhat larger data sets. Unlike [3], each participant browsed their full personal database of images. Therefore, in the present experiments, participant speed is confounded with image database size. The linear time result should thus be considered preliminary.

The satisfaction scores, taken after each condition was completed, are shown in Table 1. These scores showed that PhotoTOC was viewed most favorably on average, followed by Folders, Light Box, and AutoAlbum. A Browser×Question RM ANOVA indicated that variance explained by browser was non-significant. However, there was a significant effect of questionnaire item ($F(6, 42) = 4.77$, $p < .01$), as well as a significant interaction between browser and questionnaire item, ($F(18, 126) = 2.1$, $p < .01$). Planned comparisons for each individual question using the Bonferroni correction for multiple tests revealed several significantly higher ratings for the PhotoTOC browser compared to the other browsers. All statistically significant differences are shown in boldface in Table 1.

Overall, the individual questionnaire data indicates that users think that using PhotoTOC to browse photographs is subjectively easier than a folder browser, a detail-only view, and AutoAlbum. Subjects felt comfortable with their own organization of photos, but we suspect that, over time, they would begin to trust the automatic clustering algorithm.

The PhotoTOC versus AutoAlbum results show that good interface design is important for high satisfaction: the two browsers share the same underlying technology, yet have very different satisfaction results.

	Folders	Light-Box	Auto-Album	PhotoTOC
I like this image browser.	2.63	2.88	2.50	3.25
This browser is efficient.	2.63	2.88	2.38	**3.38**
This browser is easy to use.	3.25	3.63	3.50	3.88
This browser feels familiar.	3.88	3.63	3.00	3.00
It is easy to find the photo I am looking for.	2.75	2.75	2.50	**3.75**
A month from now, I would still be able to find these photos.	3.63	3.25	3.25	**4.13**
I was satisfied with how the pictures were organized.	**3.50**	2.75	2.63	2.88
Average	3.18	3.11	2.82	3.46

Table 1: Mean satisfaction scores across participants, using a 5 point Likert Scale with 1 being strongly disagree and 5 being strongly agree (boldface marks significant differences).

4.4 Discussion

Although the completion time data and the overall questionnaire data provide some initial evidence of the superiority of PhotoTOC to some or all of the other browsers studied, only certain individual questionnaire items revealed statistically significant differences between browsers. The lack of a reliable performance advantage could be due to the limited number of users in the user study, which limits the statistical power of our comparisons.

Many interesting image search behaviors were identified during the user study. For example, it was often observed that subjects were quite good at determining the approximate time that a picture was taken. However, sometimes their hypotheses would be wrong (probably based on some cue in the target image itself), and these would lead users down garden path searches that they strongly believed were correct. For example, one participant misrecognized one target as taking place on a different lake, which caused them to look around for the target in the wrong month. When their theories failed, subjects would resort to serial search, effectively scrolling through their entire database either forward or backward. In addition, it was observed that participants would often return to a given category of items multiple times when they held a strong but mistaken belief about the date or event of an image. This multiple return behavior was most noticeable in the Folders browser, where the user descended up and down the folder hierarchy repeatedly. Participants sometimes found this quite frustrating, which confirms research that has shown that searching through hierarchies is problematic, even for fairly shallow hierarchies [9]. Analogous to [3], it would be interesting to combine content-based image retrieval with image browsing to help users find their photographs even when they are confused about how the photograph fits into its context.

A strong difference in organizational behavior was noted between the professional and high-end consumer photographers and the more casual photographers. Serious photographers, due to their long history of taking pictures and their large databases of images, have built a categorical hierarchy that is well-honed and memorized by the user. Casual photographers had fewer, less well-defined categories. Serious photographers used their folder system very effectively, with minimal incorrect hypotheses, and would most likely reject a software tool that didn't support their rich folder structure. Casual photographers are grateful for any sensible organizational guidance the system provides. Therefore, PhotoTOC can be redesigned to allow a switch between a table of contents view and a folder view.

5 Conclusions

Users are being overwhelmed by an incoming flood of their own digital photographs. They are starting to demand automatic organization tools: specifically, systems that automatically group photographs into albums or clusters. This paper presents PhotoTOC as an example of an automatic organization tool. PhotoTOC is a system that automatically clusters photographs: it allows a user to browse his or her collection in an overview+detail view.

We compare PhotoTOC to other image browsers by performing a user study on users' own photographs and their own folder organization. This study allows us to objectively compare traditional folder browsing user interfaces to specialized image browsing user interfaces that utilize various forms of automatic organization.

The user study showed that automatic organization of images combined with a suitably designed UI is subjectively more satisfying to browse than standard browsing interfaces that can leverage a user's own organization. Automatic organization does not sacrifice browsing performance. This increased user preference was found for collections up to and beyond 1000 photographs. For consumers, higher user preference may be more important than improved search performance. Thus, automatic organization is a practical management technique for personal photographic collections.

The observed differences between AutoAlbum and PhotoTOC in the main study show that simply automatically organizing personal photographs is not enough. AutoAlbum and PhotoTOC were based on the same underlying automatic algorithm, but produced very different satisfaction results due to differing user interface designs. Therefore, it is not adequate to simply design automatic organization algorithms in a vacuum. Any automatic organization algorithm development must be coupled to iterative interface design and user studies in order to be truly useful.

Acknowledgments

The authors would like to acknowledge the suggestions and support of our co-workers, including Susan Dumais, Hagai Attias, David Heckerman, Dan Robbins and Ken Hinckley.

References

[1] B. B. Bederson. Quantum treemaps and bubblemaps for a zoomable image browser. In *Proc. User Interface Systems and Technology*, pages 71–80, 2001.

[2] J. Chen, A. Bouman, and J. C. Dalton. Similarity pyramids for browsing and organization of large image databases. In *Proc. SPIE/IS&T Conf. on Human Vision and Electronic Imaging III*, volume 3299, pages 563–575, 1998.

[3] T. T. A. Combs and B. B. Bederson. Does zooming improve image browsing? In *Proc. Digital Libraries*, pages 130–137, 1999.

[4] I. J. Cox, M. L. Miller, S. M. Omohundro, and P. N. Yianilos. PicHunter: Bayesian relevance feedback for image retrieval. In *Proc. ICPR*, pages 361–369, 1996.

[5] A. Graham, H. Garcia-Molina, A. Paepcke, and T. Winograd. Time as essence for photo browsing through personal digital libraries. In *Proc. Joint Conf. Digital Libraries*, 2002.

[6] H. Kang and B. Shneiderman. Visualization methods for personal photo collections: Browsing and searching in the PhotoFinder. In *Proc. IEEE Intl. Conf. on Multimedia and Expo*, 2000.

[7] A. Loui and M. Wood. A software system for automatic albuming of consumer pictures. In *Proc. ACM Multimedia*, pages 159–162, 1999.

[8] D. A. Nation, C. Plaisant, G. Marchionini, and A. Komlodi. Visualizing websites using a hierarchical table of contents browser:WebTOC. In *Proc. 3rd Conf. on Human Factors and the Web*, Denver, CO, 1997.

[9] K. L. Norman. *The Psychology of Menu Selection: Designing Cognitive Control and the Human/Computer Interface*. Ablex Publishing, 1991.

[10] C. Plaisant, D. A. Carr, and B. Shneiderman. Image browsers: Taxonomy and design guidelines. *IEEE Software*, 12(2):21–32, 1995.

[11] J. C. Platt. AutoAlbum: Clustering digital photographs using probabilistic model merging. In *Proc. IEEE Workshop on Content-Based Access of Image and Video Libraries*, pages 96–100, 2000.

[12] K. Rodden, W. Basalaj, D. Sinclair, and K. Wood. Does organisation by similarity assist image browsing? In *Proc. ACM CHI 2001*, pages 190–197, 2001.

[13] Y. Rui, T. S. Huang, and S. Mehrotra. Constructing table-of-content for videos. *ACM Multimedia Systems Journal*, 7(5):359–368, 1999.