PhotoVis: Visualization of Digital Photograph Metadata

Michael Hsueh

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2011-42
http://www.eecs.berkeley.edu/Pubs/TechRpts/2011/EECS-2011-42.html

May 9, 2011

PhotoVis: Visualization of Digital Photograph Metadata

Michael Hsueh
University of California, Berkeley
Computer Science Division
mhsueh@berkeley.edu

ABSTRACT

Metadata in digital photographs is an important reference tool for photographers aiming to improve their skills and learn new techniques. As digital pictures grow in prevalence and availability, it becomes increasingly important to find ways to summarize, explore, and make sense of the wealth of metadata available. This paper presents PhotoVis, a tool that approaches the problem of aggregate metadata analysis using visualization techniques. It incorporates data-centric views, filtering tools, and interactive exploration to provide users maximum flexibility in examining metadata. Tests of the application found it to perform well with large image sets and confirmed the value of visualizations in metadata analysis.

Author Keywords

digital photographs, Exif metadata, visualization, photography tools

INTRODUCTION

Digital image metadata records information about the equipment and settings used to capture individual photographs. The range of information encoded by metadata tags has grown with digital camera technology, with standardized as well as proprietary tag formats in use in virtually all modern devices. The most common standard, known as Exchangeable image file format (Exif), encapsulates tags for camera settings such as aperture, focal length, and ISO. Aside from camera settings, information about the date, time, and even geographic contexts of captured photos is commonly found amongst image tags. It should be noted that there are many widely used metadata formats. This work focuses primarily on the technical information included in Exif records.

Exif data serves mainly as a reference tool for the picture-taker, but its growing prevalence is turning it also into an instructive resource for photographers aiming to improve their skills and learn new techniques. For example, it is common practice to study and review metadata on a per-photo basis in order to identify the exact methods used to capture individual images. Side by side comparisons of metadata between photographs also aid understanding of the consequences that adjusting various camera settings have on results. This is especially important for specialized types of photography that demand precise manual control and for which automatic camera settings are usually not well suited. Examples of pictures requiring such techniques include night sky and unusual, artistic exposures. Unfortunately, the process of careful per-photo analysis is generally slow and laborious.

Photos tagged with Exif information, like many other types of fine-granularity data, exhibit interesting trends and patterns when grouped together in large numbers and analyzed. This is an important consideration as the digital camera revolution drives an explosion in the number of Exif-embedded photos that are taken and made widely available on public repositories. The problem is that while Exif studies of individual photos is straightforward, it would take an intractable amount of time and concentration to sort through even the several hundred images a photographer is likely to take in a single day.

The benefits of performing aggregate analysis are numerous. For example, an individual may want to see what all of his or her favorite photos have in common. It may also be useful for photographers to simply observe and understand how their personal shooting behaviors have evolved over time. Trends in such analyses can reveal deficiencies, preferences, and even best practices that exist between sets of photographs. Therefore, it is useful not only to focus on isolated collections belonging to individuals, but also to compare and combine them with other sets of photos during examination.

This paper describes PhotoVis, a tool that applies visualization techniques to help identify trends in Exif data across large sets of images. The visualizations used are interactive and facilitate exploration of image sets at multiple levels of granularity. They are intended to bridge the gap between the processes used for individual and aggregate studies of image metadata. The application was tested with a set of two hundred high-resolution photographs and found to perform well. Amateur photographers who evaluated the application found value in its flexibility and ability to present metadata in novel and helpful ways.

RELATED WORK

The majority of popular image viewers and editors feature some form of metadata processing. Commercial photo orga-

nization products such as Adobe Photoshop Lightroom[1] and Picasa[2] provide easy access to image metadata in human-readable form. Other independent viewing programs such as ExifTool[3] or Opanda IExif[4] are additionally able to extract and decode proprietary, manufacturer-specific metadata tags. Popular online photo sharing sites such as Flickr[5] offer options to display and download the metadata stored with uploaded images. It is also common for Flickr participants to annotate photographs with user-entered text tags.

Faceted search systems filter and browse large sets of photographs through queries on image attributes. These systems have commonly been "content-based," in that their query parameters target visual properties that are extracted from pictures. VisualFlamenco [8] is one of many examples of a system that incorporates standard image metadata as well as visual characteristics in its search queries. For a survey of content-based faceted search, see [12]. Researchers have also presented methods for searching and filtering based on high level metadata tags that encode abstract concepts such as themes, names, and hierarchies [13, 3]. Common to all faceted search systems is the goal of making it possible to specify precise criteria for images to be retrieved from typically large data sources.

Photographs are often spatially arranged to visually represent information contained in metadata. Jang et al. [6] took photos from the cameras of different tourists traveling in the same tour group and clustered them in a Cartesian plane based on their spatial and temporal similarities. Research by Cooper et al. [2] shows a similar clustering concept focusing on tags indicating events such as holiday gatherings. The Photohelix project [4] was a system designed for interacting with groups of photographs at different scales using interactive tabletops.

Much of the studies on spatial arrangements of photographs have focused on geotags, which have natural and obvious spatial mappings to physical locations. These photos are becoming more widespread as the number of imaging devices equipped with GPS sensors grows. Panoramio[6] is a popular photo sharing website that overlays geotagged images on real-time interactive maps to encourage user exploration. Exploration based on geographic data has also gone the opposite direction. Naaman et al. [9, 10] used geotags to infer higher level information that is context-dependent, such as weather conditions or daylight status at the time and place a photo was taken.

The area of geotagged-photo visualizations has seen a need for techniques that reduce the information overload arising from floods of user-submitted pictures. To this end, researchers have developed methods to generate summaries over groups of similarly geotagged photos by analyzing the groups over visual, metadata, and user-entered properties [1, 5]. These summary or "representative" tags are overlaid on the exploratory maps in place of the actual, underlying set of geotagged photos. Panoramio uses representative photos on its maps but allows users to zoom in on specific areas of interest to retrieve additional relevant images.

Visualizations based on other metadata tags have also been created. Kang and Schneiderman [7] describe an application targeting general users that allows personal photo browsing through dynamic queries. The program provides insight into the distribution of photographs over metadata tags by displaying histogram-like previews of results over the dynamic query interface. It can also spatially plot individual photos based on metadata and user-defined axes. As a reflection of its intended users, the application focuses on human-generated tags, such as the number of people or event depicted in a photograph, rather than the technical details contained in Exif tags. ExposurePlot[7] is a metadata visualization program that targets photographers. It takes a set of images as input, computes statistics over several Exif tags related to exposure settings, and generates a number of bar charts that summarize the results. The application is helpful in displaying broad patterns spanning the input images, but is limited in terms of interactivity and exploratory capabilities. The generated graphics are static and obscure the important relationships between the bars in the charts and the individual photos that they summarize.

THE PHOTOVIS APPLICATION

PhotoVis is built on the .NET 4.0 framework as a multiple document interface (MDI) application. The design of the program is centrally focused on flexible and interactive visualizations. The application interface consists of a workspace area for displaying charts and a menu bar to filter, manipulate, and retrieve more information about the images being displayed (see Fig. 1).

Each discrete photograph in the application is represented by a structure called a PhotoItem. PhotoItems are instantiated when the user loads images on the file system into the program. At load time, file information, metadata, and a low resolution thumbnail are extracted for each image and stored. All created PhotoItems are added to a global collection in memory and listed by filename in a file list located in the upper left corner of the application. Once data is loaded, users interact with the system via the controls described in the following sections.

Multiple Views

PhotoVis uses *views* to display aggregate Exif data. Each view consists of a chart in a window, implemented as a child form to the MDI parent window. When a view is created, the globally maintained list of PhotoItems is propagated to the window and then bound as the data source for the view's chart. Existing views are notified of new PhotoItems when additional images are subsequently loaded by the user. The chart graphics shown in views are implemented using Microsoft Chart Controls.

[1] http://www.adobe.com/products/photoshoplightroom/
[2] http://picasa.google.com/
[3] http://www.sno.phy.queensu.ca/ phil/exiftool/
[4] http://www.opanda.com/en/iexif/
[5] http://www.flickr.com/
[6] http://www.panoramio.com/
[7] http://www.cpr.demon.nl/prog_plotf.html

Figure 1. The PhotoVis application. The file list, filter, thumbnail preview, and metadata preview menus are located on the left. The workspace is on the right and shown here to be occupied by three views.

Two types of views are available. The first type is the plot view, which lays out PhotoItems as individual points in a scatter plot chart. The markers drawn are given a moderately transparent fill to alleviate clutter in areas with a high density of points. Users can choose any quantitative metadata tag to assign to either of the axes. Each axis is scaled in units based on its assigned tag and supports zooming and scrolling using interface controls. When changes to the axes are made by the user, the view is immediately updated to reflect the new settings.

The second type of view is the bar view, which displays the distribution of values over a selected tag in the form of a bar chart. Unlike plot views, only the horizontal axis can be changed to show different metadata tags; the vertical axis always represents a total count of PhotoItems.

PhotoItems that have a null value for a charted tag are automatically excluded from being displayed in views showing that tag. Therefore, newly loaded PhotoItems must be filtered for null values upon arriving at the individual views. Setting an axis to a different tag also results in re-filtering of the data source, as a different set of PhotoItems may become valid upon the change. The views are responsible for filtering null tags and maintaining their own data sources.

Brushing and Linking

PhotoVis implements the idea of "brushing and linking" as described by Stolte, Tang and Hanrahan [11]. Specifically, all the views and menus in the application are linked. When the user moves the cursor over a particular point, the corresponding PhotoItem is identified as the application-wide hovered item by a method call to the parent form. This information is propagated by the parent to each open view, which accordingly highlights the point corresponding to the hovered item (see Fig. 2).

Plotted PhotoItems can be selected either individually or by dragging selection rectangles, which are rendered as annotations to the chart controls. Selections are linked across views in the same way as hovered items, except that selections may span several items. In addition, selections are reflected in bar views while hovered items are not.

When a group of points is selected, the areas they correspond to are highlighted in the bar charts. The resulting views are effectively stacked bar charts with two separate areas per bar. The lower stacked series represent the currently selected items and the higher areas show non-selected items. Clicking on an individual bar results in all the PhotoItems contained in the bar being selected. Likewise, selecting items in the file list causes the corresponding items to become selected in the views.

The stacked bar chart controls require that all charted data series be aligned. This means the collections of selected and non-selected PhotoItems maintained by each bar view must be padded to contain the exact same buckets in sorted order. Padded buckets are given an empty count value of 0.

Filtering

The menu on the left side of the main application window allows users to control which PhotoItems are included in the views based on their metadata. Filters are available for nominal Exif fields and applied at the global level. The filter menu consists of a list of metadata tags and an enumera-

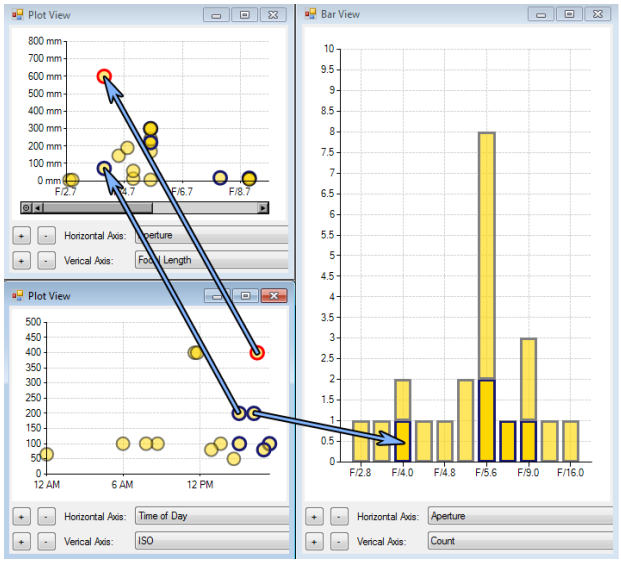

Figure 2. Views showing brushing and linking. The hovered item is outlined in red and the selected items are highlighted in blue. The regions of the bars corresponding to the selected items are outlined in blue.

tion of all possible values for each tag (see Fig. 3). To enable filtering on a given tag, the user activates the checkbox next to its name. Once enabled, the individual boxes next to the tag's possible values control whether or not PhotoItems with those values are included in the views. For example, if the "Camera Make" filter is enabled and the user subsequently deactivates the "Canon" checkbox, all images with "Canon" as its manufacturer tag value will be excluded from the views. PhotoItems with null values for any of the tags being filtered are automatically excluded.

Because the application is interactive, filtering tasks should exhibit low response times. It is inefficient to re-filter the entire collection of PhotoItems every time a filter setting is modified. Changing a single filter setting should result in the items being re-checked only on the tag corresponding to the filter. For further optimization, it is helpful to know whether a filter was changed to become more restrictive or less restrictive. This information is sufficient to determine whether only already-passing items or already-excluded items need to be re-filtered.

Specifically, if a filter is made more restrictive, it is only necessary to re-check those items that were already passing all filters at the time the change occurred. It is not possible for strengthening the criteria of a filter to cause any already-excluded photos to become included. Likewise, relaxing a filter can only result in already-excluded items being brought into the passing set. Excluding additional tag values or enabling new filters constitute strengthening of filter criteria. Doing the opposite of either is considered relaxation of criteria.

In addition to limiting the set of PhotoItems to be processed during filtering, it is also necessary to avoid applying filters that have already been evaluated. Take for example a case in which the filter on tag X was just disabled. Because this is a relaxation, previously excluded items must be filtered to determine whether any of them can be brought into the passing set. But it is inefficient to always check each tag of a given item in order to determine its passing status. Perhaps the item was previously verified to pass the tag X filter (while it was still enabled), and thus the item was excluded due to failing a filter on a different tag. To avoid the redundancy of checking all tags (that have filters enabled) again, it is necessary for each PhotoItem to remember its previous passing status for each filter. In this example, if a PhotoItem confirms that it had indeed previously passed the tag X filter, then no further processing needs to occur. This is because simply disabling the filter on X could not have possibly resulted in the item passing as a whole. The application thus knows right away to leave the item excluded and to forgo checking any other tags.

PhotoItems contain a structure called FilterFlags that encapsulate this bookkeeping information. The structure maintains a boolean for each filter indicating whether it is currently satisfied by the PhotoItem. It also maintains a count of the total number of filters the PhotoItem currently fails. Therefore, a PhotoItem that is currently in the passing set will have all of its FilterFlags boolean members set to true and a failing filter count of 0.

An outline of the filtering process is provided (details are excluded):

```
struct FilterFlags
  filter1 = true;
  filter2 = true;
  ...
  failedCount = 0;

class PhotoItem
  FilterFlags filterFlags;
  ...

Filter(bool filterStrengthened,
  filter changedFilter)
  if(filterStrengthened)
    /* Need to check all currently
    passing items for new exclusions */
    foreach(PhotoItem pi
      in passingItems)
      if(pi.filterFlags.changedFilter
        == true)
        bool passed = CheckForPassing(
          pi, changedFilter);
        if(!passed)
          pi.filterFlags.changedFilter
            = false;
          pi.failedCount++;
          excludedItems.Add(pi);
  else
    /* Need to check all current
```

```
exclusions for new passing items */
foreach(PhotoItem pi
  in excludedItems)
  if(pi.filterFlags.changedFilter
    == false)
    bool passed = CheckForPassing(
      pi, changedFilter);
    if(passed)
      pi.filterFlags.changedFilter
        = true;
      pi.filterFlags.failedCount--;
      if(pi.filterFlags.failedCount
        == 0)
        passingItems.Add(pi);
```

A filter can be added by simply implementing a CheckForPassing() method for it. The method takes a PhotoItem as input and returns a boolean specifying whether the item passes the filter.

PhotoVis supports a fixed number of filterable tags, but the enumerations of the possible values for those tags are determined dynamically as images are loaded into the program. The enumerations are populated by examining the tags across all PhotoItems and taking the maximal set of unique values found.

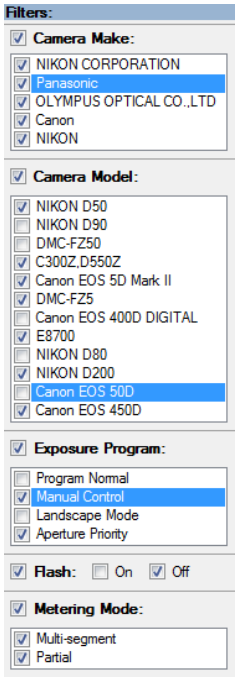

Figure 3. A populated filter menu.

Item-specific Information

An important part of interactive data exploration is having the ability to "drill down" in granularity to investigate points of interest that stand out. To facilitate this, the bottom left portion of the main window populates with detailed information about a PhotoItem when it is selected or hovered over (see Fig. 4). A preview window shows the PhotoItem's thumbnail, and directly beneath is a human-readable summary of relevant metadata. The user can double click on the thumbnail or equivalent item in the file list to launch the original image in the system's default viewing application. Connecting the views to the actual images on the file system gives users quick access to detailed metadata or close inspection of peculiar photos.

RESULTS

PhotoVis was tested on a 3.0GHz workstation with 4GB RAM. Two hundred sample images tagged with the term "nature" were downloaded at full resolution from Flickr and loaded into the application. With six separate views open, filtering, zooming, and hovering exhibited near instantaneous response times. The selection feature experienced delays of up to a few seconds when used with datasets of several hundred photos. At the time of this writing, the delay is believed to be due to an optimization rather than fundamental design issue. At higher view counts such as nine and above, limited display area becomes a usage bottleneck before any serious performance obstacles do. Total memory usage for the application remained around 65MB throughout the test case.

The currently supported quantitative metadata tags are aperture, focal length, ISO, shutter speed, date, original image width, original image height, time of day, actual image width, and actual image height. Supported nominal tags include camera manufacturer, camera model, exposure program, flash on/off, and metering mode.

The application, along with a brief video demonstration, was distributed for evaluation to four amateur photographers of varying skill levels. Feedback was generally positive and the testers expressed enthusiasm about the concept of metadata visualization. One user who was enrolled in a photography course suggested that the application would be helpful in exploring homework assignments submitted by the students in his class. Photos that are turned in can be quickly summarized to reveal patterns or outliers in the strategies students employed to complete their assignments.

Take for example the common photography course module on night photography. Submitted night shots are likely to cluster towards those using high ISO settings or long exposure times. Visualizations over the photos can be used to identify, with examples, the strengths and weaknesses of either technique. Filters can be applied on top of this to reveal interesting uses of various metering modes.

A majority of the tested users expressed the desire for the application to support additional tags, in particular for use in filtering and bar views. The application currently handles only a subset of the most common and useful Exif tags. Manufacturer-specific "makernote" tags are often embedded in images and contain more sophisticated information such as subject distance and white balance settings. One user suggested supporting user-entered and other nominal tags as buckets in bar views. Such a feature would enable the application to address questions such as, "what proportion of all

Figure 4. The thumbnail and metadata preview panels.

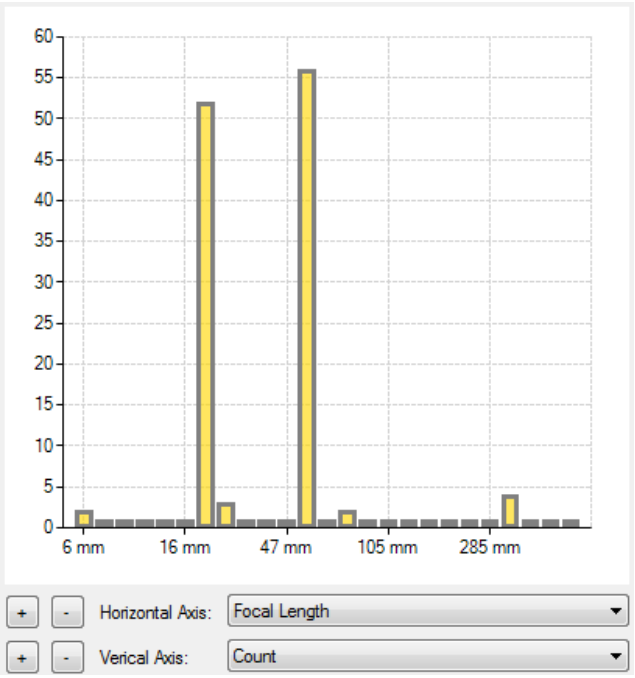

Figure 5. Bar view showing an apparent dichotomy in focal length use.

'nature' photographs are characterized by a high focal length and low f-stop?"

DISCUSSION

The main goal for PhotoVis is to enable exploration of digital photographs based on metadata. It achieves this by providing visualizations that facilitate aggregate analysis, exploration, and drilling down to finer levels of granularity. As a result, users are given the ability to evaluate their own photographic work, gain insight into the work of others, and receive actionable guidance on how to improve. To illustrate these tasks, several example use cases are provided below.

Self Evaluation

Photographers can load in sets of their own images to find important trends in their shooting behavior. Consider for example an individual who owns and takes photos primarily with a walk-around, 24-70mm zoom lens. The user might notice upon loading his or her photos into the program that the vast majority of shots fall either on the wide angle or telephoto ends of the lens, but rarely in-between (see Fig. 5). The pattern perhaps indicates that the user prefers to only shoot at narrow or wide angles, and thus two appropriately selected prime lenses might better suit him or her than a single zoom lens. Conversely, the trend might reaffirm that the user has chosen a versatile lens that conveniently allows shooting at his or her two favorite focal lengths. Either of these conclusions can be helpful in future lens-buying decisions. Finally, a reflective photographer might simply conclude that he or she needs to practice shooting at a greater variety of focal lengths. What is ultimately most important is that the trend is revealed to the user, who can then consider its implications.

Evaluation of Other Photographs

Valuable insights can be obtained by studying photo collections belonging to others. Consider an example in which the application is being used to analyze a set of pictures from nature. An explorative user may be curious to see what kind of photos can be produced using a slow shutter speed and low ISO. This combination might seem unusual to a less experienced photographer.

The user configures a plot view to display shutter speed over ISO, and selects the points in the upper left region of the plot (which corresponds to the settings of interest). An inspection of the low ISO, slow shutter speed images reveals a high frequency of photographs of waterfalls or streams exhibiting "soft water" effects. The user has gone from being curious about a certain combination of exposure settings to seeing examples of pictures that can be produced with those settings.

The user may choose to further explore characteristics of the soft water photos. He or she opens an additional plot view showing aperture versus focal length and immediately notices that most of the previously selected photos fall in the lower right quadrant of the plot. Images in this area are taken with high f-stops and low focal lengths. After some thought, the user makes sense of the trend since the scenic, wide-angle pictures depicting soft water commonly employ small apertures to get deep depth of field along with long exposure times.

Instructive Tool

Collective analysis can provide a starting point for users experimenting with techniques for specialized photography. Consider the examples of night sky or star trail pictures, which

require very specific techniques to capture properly. The problem with consulting individual photos for example settings is that they are strongly influenced by local conditions under which the pictures are shot. Light pollution, the lunar phase, and atmospheric effects all impact the camera settings required for a proper exposure. Fortunately, loading a broad collection of night sky photos into PhotoVis and then plotting them, for example by ISO versus shutter speed, can reveal a good average starting point. To further target specific conditions, the charted photos can be plotted and brushed based on the time of day they were taken.

Figure 6 shows a plot of ISO over shutter speed for one hundred photos tagged with the term "aurora borealis" on Flickr. Though exposure settings vary across the images, there is a noticeable clustering of items in the lower left quadrant of the plot. The points in this region are selected and correspond to photos taken with shutter speeds in the range of 2 to 30 seconds and ISO speeds in the range of 100 to 1600. A reasonable strategy for someone photographing the northern lights for the first time would be to start with settings corresponding roughly to the center of the selected region (for example, a shutter speed of 16 seconds and ISO of 800).

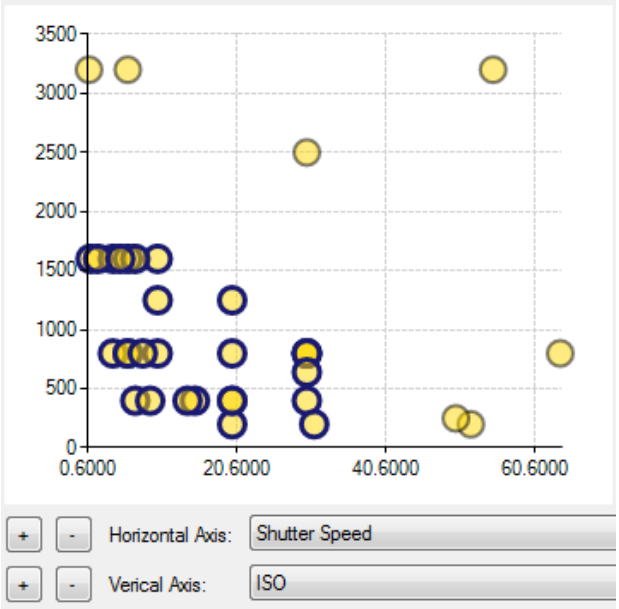

Figure 6. Plot of ISO over shutter speed for images tagged with "aurora borealis" on Flickr.

FUTURE WORK

A number of potentially helpful future enhancements to PhotoVis are described below.

Support More Tags

The utility of the application will increase if additional metadata tags are supported. Currently, only a subset of the most common Exif tags is collected by the application. Support for user-entered data, such as IPTC or XMP tags, would significantly enhance the flexibility of PhotoVis. Extracting proprietary, manufacturer specific tags would also be useful, as they occasionally encapsulate data that is valuable to photographers such as subject distance and white balance settings.

Dynamic Queries for Quantitative Tags

Filtering is currently only available on nominal metadata fields. This was done to keep the query logic in the PhotoVis prototype simple. Allowing quantitative dynamic queries with relational or range operators would give the user significantly more control over how to filter images.

More View Types

Extending existing and adding new views would increase the number of ways in which data can be analyzed. Bar views currently only support the same tags as plot views, namely quantitative ones. It would be helpful if bar views could additionally be used to display distributions over nominal tags. In general, additional types of views such as stacked area charts would improve the overall visualization capabilities of the application.

Plot Points as Thumbnails

PhotoItems are currently marked as circular points on plot views. It may be useful to instead use scaled down versions of the images themselves as markers. Though PhotoVis already provides a visual reference in the form of a thumbnail preview, displaying thumbnails on the charts may reduce a user's need to pay attention to two portions of the interface simultaneously. A similar idea can be used in bar views. That is, thumbnails can be rendered inside the bars and "stacked" up to construct their heights.

CONCLUSION

This paper presented PhotoVis, a tool for visually exploring digital photograph metadata. The application incorporates multiple views, linked menus, filtering capabilities, and fast response times to grant users maximum flexibility in examining metadata. Initial tests of the application reflected good overall performance and confirmed the benefits of using visualizations in metadata analysis.

REFERENCES

1. Ahern, S., Naaman, M., Nair, R., and Yang, J. H.-I. World explorer: visualizing aggregate data from unstructured text in geo-referenced collections. In *Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries*, JCDL '07, ACM (New York, NY, USA, 2007), 1–10.
2. Cooper, M., Foote, J., Girgensohn, A., and Wilcox, L. Temporal event clustering for digital photo collections. *ACM Trans. Multimedia Comput. Commun. Appl. 1* (August 2005), 269–288.
3. Garber, S. R., and Grunes, M. B. The art of search: a study of art directors. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '92, ACM (New York, NY, USA, 1992), 157–163.
4. Hilliges, O., Baur, D., and Butz, A. Photohelix: Browsing, sorting and sharing digital photo collections.

In *Horizontal Interactive Human-Computer Systems, 2007. TABLETOP '07. Second Annual IEEE International Workshop on* (Oct. 2007), 87 –94.

5. Jaffe, A., Naaman, M., Tassa, T., and Davis, M. Generating summaries and visualization for large collections of geo-referenced photographs. In *Proceedings of the 8th ACM international workshop on Multimedia information retrieval*, MIR '06, ACM (New York, NY, USA, 2006), 89–98.

6. Jang, C.-J., Lee, J.-Y., Lee, J.-W., and Cho, H.-G. Smart management system for digital photographs using temporal and spatial features with exif metadata. In *Digital Information Management, 2007. ICDIM '07. 2nd International Conference on*, vol. 1 (Oct. 2007), 110 –115.

7. Kang, H., and Shneiderman, B. Visualization methods for personal photo collections: browsing and searching in the photofinder. In *Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference on*, vol. 3 (2000), 1539 –1542 vol.3.

8. Muller, W., Zech, M., Henrich, A., and Blank, D. Visualflamenco: Dependable, interactive image browsing based on visual properties. In *Content-Based Multimedia Indexing, 2008. CBMI 2008. International Workshop on* (June 2008), 568 –575.

9. Naaman, M., Harada, S., Wang, Q., Garcia-Molina, H., and Paepcke, A. Context data in geo-referenced digital photo collections. In *Proceedings of the 12th annual ACM international conference on Multimedia*, MULTIMEDIA '04, ACM (New York, NY, USA, 2004), 196–203.

10. Naaman, M., Paepcke, A., and Garcia-Molina, H. From where to what: Metadata sharing for digital photographs with geographic coordinates. In *11th International Conference on Cooperative Information Systems (COOPIS 2003)* (November 2003).

11. Stolte, C., Tang, D., and Hanrahan, P. Polaris: a system for query, analysis, and visualization of multidimensional relational databases. *Visualization and Computer Graphics, IEEE Transactions on 8*, 1 (Jan/Mar 2002), 52 –65.

12. Veltkamp, R. C., and Tanase, M. Content-based image retrieval systems: A survey. Tech. rep., Department of Computing Science, Utrecht University, 2002.

13. Yee, K.-P., Swearingen, K., Li, K., and Hearst, M. Faceted metadata for image search and browsing. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '03, ACM (New York, NY, USA, 2003), 401–408.