Evaluating a Visualisation of Image Similarity as a Tool for Image Browsing

Kerry Rodden, Wojciech Basalaj
University of Cambridge Computer Laboratory
Pembroke Street
Cambridge CB2 3QG, UK
{kr205, wb204}@cl.cam.ac.uk

David Sinclair, Kenneth Wood
AT&T Laboratories
Trumpington Street
Cambridge CB2 1QA, UK
{das, krw}@uk.research.att.com

Abstract

A similarity metric based on the low-level content of images can be used to create a visualisation in which visually similar images are displayed close to each other. We are carrying out a series of experiments to evaluate the usefulness of this type of visualisation as an image browsing aid. The initial experiment, described in this paper, considered whether people would find a given photograph more quickly in a visualisation than in a randomly arranged grid of images. The results show that the subjects were faster with the visualisation, although in post-experiment interviews many of them said that they preferred the clarity and regularity of the grid. We describe an algorithm with which the best aspects of the two layout types can be combined.

1. Introduction

Improvements in storage and networking technology have made it possible for large collections of images to be made available in digital form. In order to facilitate access to these collections, it is necessary to carry out some indexing of their content. With textual document collections, conventional information retrieval algorithms and systems can be used to index them according to the words they contain, enabling searches based on these. With digital images, the task is more difficult, as they do not contain individually meaningful units like words. They may of course be given textual annotations manually, allowing them to be indexed conventionally, but this is a very time-consuming process, and the resulting annotations may be highly subjective.

Ideally, it would be possible to extract high-level semantic content from images automatically, and use this to index them objectively. This is of course a very difficult problem, and current systems for automatic indexing and retrieval of images can make use of only low-level visual features, such as colour and texture. There is some evidence that visual properties are important to people when searching for images. For example, a study by Jose et al. [6] found that designers searching for photographs to illustrate a brochure for tourists reported that they often had a reasonably well-defined “mental image” of a photograph that might satisfy their requirement.

Users are expected, however, to express their requirement in a visual form, and this can be much more difficult than constructing a textual query, and less likely to yield meaningful results. For example, a user looking for textual documents about jet aeroplanes would be sure to include words like “jet” and “aeroplane” as query terms. However, searching for photographs of jet aeroplanes, in current content-based image retrieval systems, would require the user to either attempt to characterise a prototypical image of a jet with a sketch or colour percentages, or to select a suitable example image by browsing the collection. The system would then return to the user the set of images that it measures to be most visually similar to the given image. However, because of the low-level nature of the features used to judge similarity, the system could be just as likely to return (for example) pictures of birds in flight, or mountains, as pictures of jet aeroplanes.

These difficulties in query construction mean that it becomes very important for systems to provide good support for browsing of images. Information visualisation is often proposed as a means of helping the user to gain an overview of a set of data objects, thus assisting browsing. Mutually similar objects can be arranged close to each other, enabling the user to see how they are related with regard to the similarity metric used.

Rubner [11] has proposed that a low-level content-based image similarity metric can be used to create visualisations of sets of images. Thumbnail versions of the images can be placed directly at the appropriate points in the visualisations, so that the variation in their visual content is easy to see. We have implemented software to generate such visualisations. For example, Figure 4 (in the colour plate) shows our visualisation of 100 images of Kenya. The combination of the metric and the layout algorithm has uncovered a natural structure in the set, where images of people are visible at the top left, images of wildlife on the right, and buildings are at the bottom.

Of course, because of the low-level nature of the metric, these visualisations are still subject to many of the same limitations as a purely query-based system. However, users at least are not forced to attempt to sketch their "mental image", but can instead simply allow their gaze to move into the areas of the visualisation where the images most resemble it. For example, when searching for aeroplanes, a user's eye would probably be drawn to pictures containing a lot of blue sky.

No experimental evaluation of this type of visualisation had been carried out, so we set out to investigate its utility for different image browsing tasks. As an initial experiment, we decided to consider a purely visual search task, since one would expect that this would be the type of task most favoured by the combination of a visualisation and a content-based similarity metric. If the visualisation proved itself in principle, we could then move on to tasks involving a more general requirement, putting the usefulness of content-based similarity metrics on trial as well as that of the visualisation.

The first experiment was designed in order to establish whether users would find a given image more quickly in a visualisation than in a baseline, random arrangement. This simulates a situation where the user is trying to find a particular image that he or she has seen before, rather than searching with a general requirement in mind. Experiments by Jacobs et al. [5] have shown that a person's sketch from memory of a given image can often be good enough for a content-based retrieval system to match it to the required image. However, it has not been shown that people are actually willing to make the effort to create such sketches in real situations. People have an excellent memory for recognition of photographs [14, p.192], and so one might expect that it would be easier for them to recognise a desired image than to recall and sketch it, particularly if they are not confident about their artistic ability. Of course, in this experiment we are asking the subjects to retrieve the images from their short-term memory, rather than the long-term memory that would be used in a real situation. However, images may be remembered in a similar way in both short- and long-term memory [14, p.198]. Colours, for example, are always remembered in a more simplified form than that in which they were originally seen.

Before explaining the experiment, we first describe the similarity metric and layout algorithm we used to create our visualisations. It would have been possible to choose any metric and any algorithm, but those used represent our own research in each of these areas. Future studies could compare different metrics and different layout algorithms in this context.

2. The image similarity metric

Researchers in the information retrieval and image processing communities have proposed a large number of image similarity metrics. These are used in image retrieval systems to find the images in a collection that are most visually similar to a given query image.

Such metrics can be qualitatively partitioned into those that use a *global* summary of image content and those that use a *region by region* comparison. Metrics based on global image properties generally use colour histograms, as well as some representation of the texture present in the image (e.g. Virage[1]). The relative orientation of regions is not reflected in this kind of metric. Systems actively using segmented image regions (e.g. QBIC[2] and NETRA[3]) permit richer query generation. The relative importance of region colour, shape, location and texture in the metric are set at query time. The metric employed in this paper is designed to reflect both global image properties and the broad spatial layout of regions in the image.

An image is segmented into regions with broadly homogeneous colour properties [13]. Regions have descriptors for colour, colour variance, area, shape, location and texture. Regions are then classified as either large (with an area greater than 0.1% of the total image area) or small. The large regions are further classified as either textured or smooth, and the small regions as regularly or irregularly shaped (based on their isoperimetric area [12]).

An image summary is then created as follows. The image is partitioned into nine equal areas, in the obvious way. Sets of four global colour histograms (for large smooth regions, large textured regions, small regular regions, and small irregular regions) are made for each of these nine areas, and the largest (dominant) region in each of the nine areas is recorded. The histograms are normalised by image area.

The chi-squared statistic [7] is used to determine the distance between like colour histograms across image pairs. The distance between dominant regions in corresponding ninths of each image is given by the Mahalanobis distance in RGB colour space. The metric is then a weighted sum of the above distances.

The metric gives a balance between global image colour properties and local region based properties. Experiments should help to fine tune the relative weighting of its different components.

3. The layout algorithm

Given a similarity metric for a collection of objects, such as images, how does one arrive at a layout that is a faithful representation? Multidimensional scaling (MDS) achieves this objective by treating inter-object dissimilarities as distances in some high dimensional space, and approximating them in a low dimensional output configuration. Two-dimensional layouts are most

[1] http://www.virage.com/static/products/image.html
[2] http://www.almaden.ibm.com/cs/showtell/qbic/
[3] http://maya.ece.ucsb.edu/Netra/

useful here, as thumbnail versions of the images can be placed at the corresponding points.

The MDS algorithm [1] that was used in this experiment emulates a fully connected spring system with one anchor point for every object in the collection. The relaxed length of a spring connecting two points is taken to be the dissimilarity between the corresponding pair of objects. The actual length of the spring is the distance between its two anchor points. The algorithm attempts to arrive at a minimum energy state (an optimal configuration) by reducing the disparity between actual and desired lengths. This process is iterative and terminates when the amount of improvement becomes negligible.

During a single iteration of the algorithm, all points in the configuration are considered one at a time, and their locations improved. This is achieved by applying a single Newton-Raphson iteration to the gradient of the energy function at the point currently considered. Successive applications of this procedure will converge to a configuration that is a root of the first derivative of the energy function. The topology characteristics of the energy function guarantee that this is a minimum.

Numerical minimisation techniques can only be expected to give a local minimum, and this algorithm is not an exception. However, it does have a tendency to avoid particularly bad minima, by allowing energy-increasing changes to the configuration, especially early on in the minimisation process. In any case, it is always useful to run the algorithm a few times and only take the best layout.

4. The experiment

Our hypothesis was that, in general, subjects would find a given target image in a set more quickly if that set is arranged according to visual similarity, than if it is arranged randomly. The method we adopted was to display a full-size target image to the subject, remove it from the screen, and then measure how long he or she takes to find it in a set of thumbnail images. This process was repeated through a series of trials.

The image set was laid out in one of two different ways. In the *random* condition, the set was arranged in a random order, in a grid. A grid was used instead of a purely random layout, since current applications usually display images in this way. In the *MDS* condition, the set was arranged in a similarity-based visualisation (as in Figure 5, in the colour plate).

4.1. Subjects

16 subjects were recruited from among the students and staff of the University of Cambridge. All had either normal or corrected-to-normal vision, with no colour blindness (self-reported).

4.2. Apparatus

48 sets of 80 images were randomly selected from a 20,000-image stock photograph collection, available from Corel on CD-ROM. Target images were presented to the subjects at 768x512 pixels, and thumbnails were displayed at 96x64 pixels. Only landscape-oriented images were used, because in pilot studies it became obvious that it was possible to “filter” the image set according to the orientation of the target. For each of the 48 sets of images, a visualisation and a random arrangement were created, and each was saved in a large, maximum quality JPEG file. Subjects were tested one at a time, all on the same PC, with a 17-inch monitor set at 1280x1024 resolution. A Java program was used to display images and record timings.

A trade-off between thumbnail size and number of images per set was necessary, since the more images that are displayed on screen at once, the smaller they have to be to avoid obscuring each other. This is especially true of the MDS layouts, which have an irregular structure that results in thumbnails overlapping. The attributes were fixed after experimenting with several pilot subjects. It would have been possible to avoid overlap by implementing a mechanism to bring an image to the front when the user moves the mouse over it. However, rather than complicate the experiment, we simply decided to only use target images that had more than 70% of their area visible in the layout, so that the subjects were never asked to find an image that was badly obscured. This meant that approximately 10% of the images in each MDS layout could not be used as targets.

4.3. Design

The dependent variable was the time taken (in milliseconds) between the set of images appearing and the subject clicking on the target image. Any trials where the subject failed to click within the time limit, or clicked on the wrong image, were regarded as missing data. The independent variable was the way in which the image set was arranged, and the experiment was within-subjects (repeated measures), so that all of the subjects received both conditions of this variable. The design was balanced, so half of the subjects received the MDS condition first, and half received the random condition first. Each subject saw the same layouts, but the order of these within each condition was randomised, so that there would be no effect of ordering. The 48 trials in each condition were presented in four consecutive blocks of 12, with breaks between blocks. The subjects also received one “practice” block at the start of each condition.

Since some images always seem to “stand out” from a set more than others, we expected that an image’s distinctiveness would play an important part in how quickly it was located, regardless of the layout type.

Because this is difficult to measure, we did not attempt to control it as a variable, and instead decided that it should be randomised. Distinctiveness would then be a source of random variation, and should not affect the overall outcome. Thus, the target image for each layout was selected at random for each subject (from those at least 70% visible in the layout, as described in the previous section).

We also expected that the screen position of a target image in a layout would have some effect on the time taken to find it. Firstly, subjects may adopt a search strategy that favours a particular area of the screen, such as a "scanning" (top left to bottom right) strategy. Secondly, images that are placed at a larger distance from the user's initial mouse position will take longer to reach. This was also left as a source of random variation, dealt with by the randomised selection of targets. To give some consistency across all of the trials, subjects had to move the mouse to the centre of the screen before the set of images appeared, and were asked not to move it again until they had found the target image with their eyes.

4.4. Procedure

Subjects were asked to read a set of instructions, and were then given a practice block of whichever condition they were receiving first, followed by the four blocks of that condition. They then received the practice block and four real blocks of the other condition, as described above. In a single trial, a subject was shown a target image for 10 seconds. It was then removed from the screen, giving the subject 20 seconds to find it (as quickly as possible) in a set of 80 thumbnail images, and click on it using the mouse. The timeouts were set after pilot studies. As soon as an image was selected, or when the time ran out, the correct image in the set was highlighted. This gave the subjects some feedback about their performance, and was intended to help them remain interested in the task.

Subjects were told that they were being timed, and that this was being done in order to compare the two layout methods to each other. They were not told how the layouts were created, to avoid giving them the impression that the experimenter had any investment in one type doing better than the other. In particular, this meant they were not told that visually similar images were arranged together in the MDS layouts, but it was assumed that they would notice this during the practice block.

In order to provide qualitative data, the subjects were also asked to fill in a post-experiment questionnaire. This asked them, in an open-ended way, to describe what sort of searching strategies they had used for each of the layouts, as well as what they thought were the advantages and disadvantages of each layout method, and to explain which one they preferred. They were also asked if some kinds of target image were easier or more difficult to find than others. The whole procedure took about an hour per subject.

5. Results

An analysis of variance (ANOVA) was performed, with time as the dependent variable, and layout type and subject as independent variables. The times were not normally distributed, and so a log transform of the time was used in its place, as is standard. The results showed that subjects were significantly faster with the MDS layout than with the random layout ($p<0.0001$), with a mean time of 4311 milliseconds for MDS, versus 5107 milliseconds for the random layout. In the post-experiment questionnaires, we were surprised to find that 7 of the 16 subjects said that they had not noticed that visually similar images were arranged together in the MDS layout. Yet these subjects were just as fast as those who did notice, so we concluded that the visualisation must have guided them in the direction of the target, without them being consciously aware of it.

12.3% of the trials had to be regarded as missing data, because the subject either failed to select an image within the time limit (9.3%), or selected the wrong image (3.0%). This constituted 14.5% of the random trials and 10.2% of the MDS trials. The subjects said that when they had failed to find an image, it was often because it looked different to the way they had remembered it. In particular, reducing an image to the size of a thumbnail may result in the appearance of remembered objects changing dramatically. Those subjects who were consciously making use of the grouping of visually similar images in the MDS layouts said that sometimes the target thumbnail did not appear in the area of the screen that they expected. Consequently, they would scan the area repeatedly before realising that the target was just outside it. This may be because the metric or layout algorithm has not succeeded in arranging the images according to the way a human would arrange them, or because the subject's memory of the image is different to its actual appearance.

5.1. The effect of distinctiveness and position

All of the subjects definitely felt that some target images were easier or more difficult to find than others. In particular, they said that the easiest images to find were those containing large areas of bright, saturated, or contrasting colour, especially close-ups. The most difficult targets were those containing dull colours and showing distant objects, for example landscapes or city scenes. The subjects noted that this had affected their search strategies, reporting that with both layout types they had first quickly scanned the image set to see if the target "jumped out", and only if that failed did they search in a more methodical way.

The experiment design attempted to deal with the distinctiveness of target images by giving each subject a different random set of target images, so that it would appear only as random variation. We felt it would be interesting to attempt to retrospectively introduce some measure of distinctiveness into the statistical analysis. An image's distinctiveness cannot be quantified in an absolute way, as it is always relative to the set of images in which it is presented: the most distinctive images are likely to be those which are most visually dissimilar to all of the others. The distinctiveness of an image was thus estimated quantitatively as the average similarity (according to the metric) of the image to each of the other images in its set. For the image sets used in the experiment, the measurement seems to be in broad agreement with the subjects' opinions. For example, close-ups of brightly coloured flowers or exotic fruits are usually measured as highly distinctive. Despite the crudeness of this measurement, once it is introduced into the analysis as an independent variable it is highly significant ($p<0.0001$), showing that distinctiveness does indeed have an effect on search time. This did not affect the significance of layout type.

We also attempted a similar retrospective analysis of the effect of a target's screen position, introducing the distance in pixels of the target from the centre of the screen as an independent variable. This showed, again at a high level of significance ($p<0.0001$), that images appearing closer to the centre of the screen were located more quickly. Again, the significance of layout type was not affected.

5.2. The problem of image overlap

It is generally expected that objects will be represented in information visualisations by points. However, when the objects have on-screen representations that are any bigger than a single pixel, as is the case with image data, there is a risk of overlap between them, resulting in some objects being partially or wholly obscured. Before we conducted the experiment, we did not expect overlap to be much of a problem. Images that overlap are likely to be very visually similar, and we were not asking subjects to find any badly obscured images.

However, in the post-experiment questionnaire all of the subjects said that the overlap in the MDS layouts had given them problems, making it difficult to see the edges of images and sometimes resulting in a remembered detail being obscured. Five of the subjects expressed a preference for the random grid, citing lack of overlap and ease of systematic scanning (along rows or down columns) as their reasons. We realised that it should be possible to adapt the "pure" MDS layout so that the images are arranged in a more regular way, thus combining the advantages of the two techniques.

6. Grid algorithm

A visualisation of a MDS configuration will not suffer from overlap if the separation between object centres is greater than the object diameter. This observation led to development of an algorithm that guarantees a minimum amount of separation by forcing object centres to lie on a grid. A MDS configuration and desired grid size form input to the algorithm. An optimal allocation of points in the configuration to grid cells is the goal of the algorithm. This task is particularly hard for dense grids, where all or most of the cells will become occupied. As an example of the desired effect, Figure 6 (in the colour plate) shows the 80-image layout of Figure 5, transformed to fit into a 12x12 grid. This leaves 64 cells empty, and results in much of the original structure of the layout being preserved.

Of course, a similar effect could be achieved by adapting a MDS algorithm to work in a discrete rather than the continuous domain. However, there are a number of advantages of having it as a separate post-processing step. An exact grid size can be enforced, since continuous co-ordinates have been established in advance by MDS, and are therefore fixed. Moreover, different grid sizes can be tried for a single configuration, and their quality directly compared. This also applies to dense grids, which would no doubt be very difficult to deal with inside a MDS algorithm (a situation resembling a sliding tiles puzzle). The grid algorithm can also be used in conjunction with configurations from other sources.

In the initial stage of the algorithm, continuous co-ordinate values are mapped into a discrete range. Since only square grids are used, the grid length determines how many unique locations exist in each dimension. Naturally, it is possible for two points to be mapped to the same grid cell. Many strategies could be implemented to resolve this. Since we wanted to avoid introducing another numerical optimisation process, we adopted a greedy approach. This will only give an approximate solution. However, the computational cost is low, and the amount of introduced error can be controlled by adjusting the grid length.

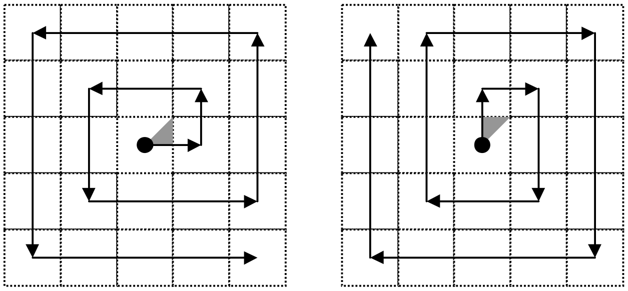

Figure 1: An example of spiral search. The grey triangle represents the set of all possible positions of point *p* that would result in this particular orientation of the spiral. The black circle denotes the optimal cell *o* for this point.

Each point **p** in the configuration is considered in turn. If the optimal grid cell **o** is unoccupied, **p** is allocated to it. Otherwise, the closest empty cell **e** is found by performing a spiral search starting from cell **o**. The second location visited by the spiral is the second closest cell to point **p**, taking Euclidean distance into account. This determines the orientation of the spiral in an unbiased way. Figure 1 is an example of a spiral search when **o** is below and to the left of **p** (the remaining three cases are analogous). If there are at least as many grid cells as points this procedure is guaranteed to find **e.** Once this happens three strategies can be adopted, as illustrated in Figure 2:

a. **p** is allocated to **e** (*basic* strategy)
b. the point allocated to **o** is relocated to **e**, and **p** is allocated to **o** (*swap* strategy)
c. points allocated to cells on the line from **o** to **e** are relocated outwards by one cell, and **p** is allocated to **o** (*bump* strategy)

The order in which points are considered is important with each strategy. It should be independent of permutations of the input configuration. The Minimum Spanning Tree (MST) of the original data that was subject to MDS defines such a unique order [1]. Shortest MST edges link most closely related data objects. Considering the corresponding points first in the *basic* strategy will ensure that they occupy neighbouring cells. The same effect would be achieved by considering them last in the *swap* and *bump* strategies.

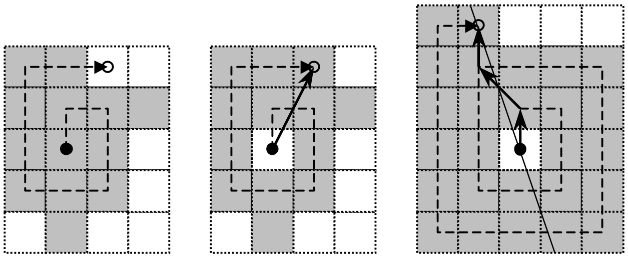

Figure 2: Examples of the basic, swap, and bump strategies (from left to right). In each, the solid black circle denotes the optimal cell, and the empty circle shows the closest empty cell.

To determine the best strategy to use in the algorithm, testing was performed on a MDS configuration of 400 images. For each grid length tested, all strategies, in both ascending and descending MST order (a total of six possibilities) were applied. In each case maximum and average error were noted. The amount of error in positioning each image was defined to be the distance between its optimal and actual grid cell. The distance between two grid cells was taken as either the horizontal or vertical grid separation, whichever was greater. The test ranged over all grid lengths from 20 (the smallest possible for this collection), up to the smallest length that gave no error for all possible strategies.

Figure 3 is a plot of maximum error for each grid length and strategy. The difference between applying a strategy in descending and ascending order is very slight. However, there are definite differences between strategies. The bump strategy gives best results for small grids, which will be densely populated with objects. The basic strategy appears to be marginally better for larger, more sparsely populated grids. The swap strategy consistently gives the highest maximum error. This is because a single point can be relocated more than once with this strategy, increasing the amount of error. This is also true for the bump strategy, and becomes apparent for larger grids, where it is equivalent to a swap strategy, as can be discerned from Figure 3.

The average error is very similar for each grid length and strategy. Bump appears marginally better for smaller grids. This is due to the tendency of this strategy to spread error over many points, rather than positioning some items optimally and delegating potentially large error to others. The descending bump strategy seems best overall, as it minimises both maximum and average error for dense grids, which are most useful as they give large separation between objects. This strategy will also preserve distances between most closely related objects.

The algorithm's space requirement is $O(\mathbf{m}^2)$, where

$\mathbf{n}$	number of points in the configuration
$\mathbf{m}$	grid length, $\mathbf{m}^2 \geq \mathbf{n}$

Then, assuming:

$O(\mathbf{n})$	time taken by initial discretisation
$O(\mathbf{m}^2)$	time for initialising the grid
$\mathbf{l_k} = \mathbf{k}\text{-}1$	number of operations to find an empty cell (length of spiral search for $\mathbf{k}^{th}$ point)
$\mathbf{r_k} = O(\sqrt{\mathbf{k}})$	number of operations to perform bump (radius of spiral for $\mathbf{k}^{th}$ point)

the worst-case running time, using the bump strategy, is
$O(\mathbf{n})+O(\mathbf{m}^2)+\sum(\mathbf{l_k}+\mathbf{r_k})=O(\mathbf{m}^2)+\sum(\mathbf{k}-1)=\underline{O(\mathbf{m}^2)+O(\mathbf{n}^2)}$

7. Discussion and further work

The experiment described here has shown that laying out a set of random images according to their visual similarity can help people to find a given target image more quickly than when those images are laid out in a random order, in a grid. This was intended as an initial investigation into the potential usefulness of a visualisation for image browsing tasks. The task in this experiment was purely visual, simulating a search for a specific photograph, which the user has seen before and wants to find again.

It is not clear how common this type of search is in practice. Enser's analysis of the requests received by a major photograph library [2], for example, does not mention any for specific images. However, the study was done in a context where experienced search staff locate images matching a verbal request submitted by clients who have very little knowledge of the collection. He did not study to what extent the trained searchers make use of their memory of particular images from previous searches. One might expect that specific requests will become more common as technology enables more people to search for images directly, and become more familiar with the collection.

In interviews carried out as part of an earlier study of how people organise their personal photographs [8], locating a specific image was the most commonly mentioned search task. People are very familiar with their own photographs, and do not normally want to use them to satisfy a general requirement; a specific photograph, associated with a particular memory, is usually enough. We are currently integrating a visualisation facility into our system for organising personal photographs, Shoebox, and we plan to study people's use of different searching and browsing tools in this context.

As discussed earlier, people who are searching for images with only a general requirement in mind may often have a "mental image" of a photograph that would meet their needs. One could argue that this experiment has also simulated looking for images based on a mental image, where the picture is just given to the subject rather than being conjured up in his or her own imagination. However, since a mental image is likely to be much less well-defined than a memory of a particular photograph, and since requirements often evolve and change during the search process [4], we are reluctant to make such a claim based on these results. We have since carried out a second experiment, where the subjects were given a general verbal description of images to find, and were expected to create their own mental image. The results showed that, in general, people found matching images more quickly with an MDS layout than a random layout. More details can be found in a separate paper [9].

The issue of scale has to be addressed: our studies have looked only at sets of 80 or 100 images, but of course there are thousands of images in real collections. In cases where no annotation or organisation of the collection exists, the search would first have to be narrowed down with simple visual queries, as Rubner [11] suggests. It is also possible to attempt a hierarchical visual organisation of the entire collection, as described by Chen et al. [2]. However, this sort of hierarchical grouping would have to use a metric that mapped very closely to users' own perceptions of visual similarity, otherwise they may become lost, having descended the wrong branches of the hierarchy.

Of course, for most image searching tasks it is crucial that some sort of annotation exists, so that the user knows exactly what a photograph actually depicts. Image libraries have already made a lot of investment in creating well-annotated collections. For example, to produce the layout of images of Kenya shown in Figure 4 (in the colour plate), it was first necessary for a human indexer to classify all of these photographs under the heading "Kenya". Without such a categorisation, there would be no way of knowing that these images were of Kenya, and it would certainly be impossible for any visual indexing system to identify them as such.

A designer working on a tourist guide to Kenya might want to look for photographs of the country, to use as illustrations. He or she could find the "Kenya" category in a pre-classified image collection, or perhaps enter a textual query in an annotated collection. In a conventional system, the images would either be displayed in some default order, or ranked in order of their similarity to a query. A visualisation like that of Figure 4 could be used as an alternative way of displaying the images, and might help the designer to get a feel for all of the different types of photographs of Kenya that are available in the collection, and choose those which are most visually suitable for the guide. We are currently examining the use of a visualisation for arranging the contents of a single category in a pre-organised collection, or the results of a text-based query.

As well as examining different image browsing tasks, we could also attempt to determine experimentally the best possible way in which to create the visualisations. Rogowitz et al. [10] created multiple MDS visualisations of the same set of images, using different similarity metrics, in order to qualitatively compare them. They compared content-based metrics with the similarity judgements of human subjects. It is also possible to use a text-based metric, with a collection whose images have been annotated. It will be very interesting to see which of these metrics produces the most useful visualisations. It is possible that metrics which work well in image retrieval systems may not be the most appropriate for use in a visualisation, and vice versa.

8. Conclusions

Information visualisations created using content-based metrics of image similarity offer some promise as tools for image browsing. The experiment described in this paper showed that people can locate a specific image more quickly in a visualisation than in a randomly arranged image set. Further experiments will provide more insight into the usefulness of this type of visualisation in support of a wider range of image browsing tasks.

Thumbnails are used to represent images in a visualisation, and since these take up more than a single pixel of screen space, arranging them according to the original configuration of points produced by the layout algorithm leads to some overlap. The subjects in our study disliked this, expressing a preference for a more regular arrangement that would eliminate overlap. We developed an algorithm to create grid-like versions of the visualisations, thus combining the best aspects of both approaches.

9. References

1. Basalaj, W. Incremental multidimensional scaling method for database visualization. In *Visual Data Exploration and Analysis VI* (Proc. SPIE, volume 3643), January 1999.
2. Chen, J.Y., Bouman, C.A., and Dalton, J.C. Similarity pyramids for browsing and organization of large image databases. In *Human Vision and Electronic Imaging III* (Proc. SPIE, volume 3299), January 1998, 563–575.
3. Enser, P.G.B. Query analysis in a visual information retrieval context. *Journal of Document and Text Management*, 1(1), 1993, 25–52.
4. Garber, S.R, and Grunes, M.B. The art of search: a study of art directors. In *Proceedings of CHI'92*, ACM, May 1992, 157–163.
5. Jacobs, C.E., Finkelstein, A., and Salesin, D.H. Fast multiresolution image querying. In *Proceedings of SIGGRAPH'95*, ACM, August 1995.
6. Jose, J.M., Furner, J., and Harper, D.J. Spatial querying for image retrieval: a user-oriented evaluation. In *Proceedings of SIGIR'98*, ACM, August 1998, 232–241.
7. Puzicha, J., Hofmann, T., and Buhmann, J. Non-parametric similarity measures for unsupervised texture segmentation and image retrieval. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, 1997, 267–272.
8. Rodden, K. How do people organise their photographs? *Proceedings of the BCS IRSG Colloquium*, April 1999.
9. Rodden, K., Basalaj, W., Sinclair, D., and Wood, K. Evaluating a visualisation of image similarity. Poster, in *Proceedings of SIGIR'99*, ACM, August 1999.
10. Rogowitz, B.E., Frese, T., Smith, J.R., Bouman, C.A., and Kalin, E. Perceptual image similarity experiments. In *Human Vision and Electronic Imaging III* (Proc. SPIE, volume 3299), January 1998, 576–590.
11. Rubner, Y., Tomasi, C., and Guibas, L.J. A metric for distributions with applications to image databases. In *Proceedings of the IEEE International Conference on Computer Vision*, January 1998.
12. Sinclair, D., and Blake, A. Isoperimetric normalisation of planar curves. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(8), 1994, 769–777.
13. Sinclair, D. Voronoi seeded colour image segmentation. Technical Report 1999.3, AT&T Laboratories Cambridge, 1999.
14. Stroebel, L., Todd, H., and Zakia, R. *Visual concepts for photographers*. Focal Press, London, 1980.

Acknowledgements

Thanks to Alan Blackwell and Jonathan Pfautz for their advice on experiment design, and to Gill Ward for her assistance with statistical analysis. Matthew Chalmers gave useful criticism of a draft of this paper.

Kerry Rodden's work was supported by a studentship from the UK EPSRC, and by AT&T Laboratories, Cambridge. Wojciech Basalaj's work is supported by Trinity College, Cambridge, and by the Overseas Research Students Awards Scheme.

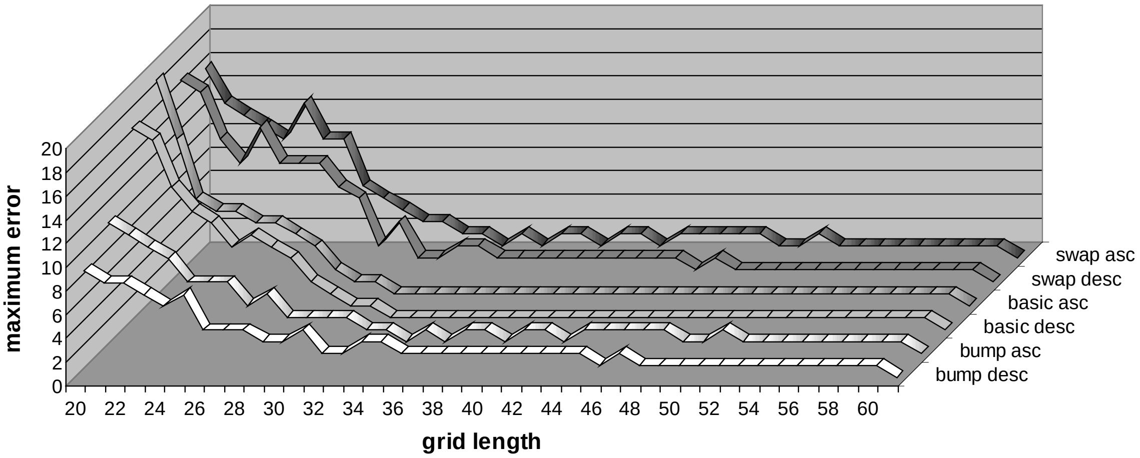

Figure 3: Maximum error for all strategies and directions of the grid algorithm.