MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

Visualization, Estimation and User-Modeling for Interactive Browsing of Image Libraries

Baback Moghaddam *
Qi Tian †
Thomas S. Huang †

TR2002-53 July 2002

Abstract

We present a user-centric system for visualization and layout for content-based image retrieval and browsing. Image features (visual and/or semantic) are analyzed to display and group retrievals as thumbnails in a 2-D spatial layout which conveys mutual similarities. Moreover, a novel subspace feature weighting technique is proposed and used to modify 2-D layouts in a variety of context-dependent ways. An efficient computational technique for subspace weighting and re-estimation leads to a simple user-modeling framework whereby the system can learn to display query results based on layout examples (or relevance feedback) provided by the user. The resulting retrieval, browsing and visualization engine can adapt to the user's (time-varying) notions of content, context and preferences in style of interactive navigation. Monte Carlo simulations with synthetic "user-layouts" as well as pilot user studies have demonstrated the ability of this framework to accurately model or "mimic" users by automatically generating layouts according to their preferences.

International Conference on Image & Video Retrieval (CIVR'02)
London, UK, July 2002

* MERL - Research Laboratory
† University of Illinois at Urbana-Champaign

Published in: *International Conference on Image & Video Retrieval* (CIVR'02), July 2002.

Visualization, Estimation and User-Modeling for Interactive Browsing of Image Libraries

Baback Moghaddam[1], Qi Tian[2] and Thomas S. Huang[2]

[1] Mitsubishi Electric Research Laboratory, Cambridge, MA 02139, USA
{baback}@merl.com
[2] Beckman Institute, University of Illinois, Urbana-Champaign, IL 61801, USA
{qitian,huang}@ifp.uiuc.edu

Abstract. We present a user-centric system for visualization and layout for content-based image retrieval and browsing. Image features (visual and/or semantic) are analyzed to display and group retrievals as thumbnails in a 2-D spatial layout which conveys mutual similarities. Moreover, a novel subspace feature weighting technique is proposed and used to modify 2-D layouts in a variety of context-dependent ways. An efficient computational technique for subspace weighting and re-estimation leads to a simple user-modeling framework whereby the system can learn to display query results based on layout examples (or relevance feedback) provided by the user. The resulting retrieval, browsing and visualization engine can adapt to the user's (time-varying) notions of content, context and preferences in style of interactive navigation. Monte Carlo simulations with synthetic "user-layouts" as well as pilot user studies have demonstrated the ability of this framework to accurately model or "mimic" users by automatically generating layouts according to their preferences.

1 Introduction

With the advances in technology to capture, generate, transmit and store large amounts of digital imagery and video, research in content-based image retrieval (CBIR) has gained increasing attention. In CBIR, images are indexed by their visual contents such as color, texture, etc. Many research efforts have addressed how to extract these low level features [1, 2, 3], evaluate distance metrics [4, 5] for similarity measures and look for efficient searching schemes [6, 7].

In this paper, we present designs for optimal (uncluttered) visualization and layout of images (or iconic data in general) in a 2-D display space. We further provide a mathematical framework for user-modeling, which adapts and mimics the user's (possibly changing) preferences and style for interaction, visualization and navigation.

Monte Carlo simulation results with machine-generated layouts as well as pilot user-preference studies with actual user-guided layouts have indicated the power of this approach to model (or "mimic") the user. This framework is currently being incorporated into a broader system for computer-human guided navigation, browsing, archiving and interactive story-telling with large photo libraries.

2.1 Traditional Interfaces

The purpose of automatic content-based visualization is augmenting the user's understanding of large information spaces that cannot be perceived by traditional sequential display (*e.g.* by rank order of visual similarities). The standard and commercially prevalent image management and browsing tools currently available primarily use tiled sequential displays – *i.e.*. essentially a simple 1-D similarity-based visualization.

However, the user quite often can benefit by having a global view of a working subset of retrieved images in a way that reflects the relations between *all pairs* of images – *i.e.*, N^2 measurements as opposed to only N. Moreover, even a narrow view of one's immediate surroundings defines "context" and can offer an indication on how to explore the dataset. The wider this "visible" horizon, the more efficient the new query will be formed. In [8], Rubner proposed a 2-D display technique based on multi-dimensional scaling (MDS) [9]. A global 2D view of the images is achieved that reflects the mutual similarities among the retrieved images. MDS is a nonlinear transformation that minimizes the stress between high dimensional feature space and low dimensional display space. However, MDS is rotation invariant, non-repeatable (non-unique), and often slow to implement. These drawbacks make MDS unattractive for real time browsing or visualization of high-dimensional data such as images.

2.2 Layout & Visualization

We propose an alternative 2-D display scheme based on Principle Component Analysis (PCA) [10]. Moreover, a novel window display optimization technique is proposed which provides a more perceptually intuitive, visually uncluttered and informative visualization of the retrieved images.

Traditional image retrieval systems display the returned images as a list, sorted by decreasing similarity to the query. The traditional display has one major drawback. The images are ranked by similarity to the query, and relevant images (as for example used in a relevance feedback scenario) can appear at separate and distant locations in the list. We propose an alternative technique to MDS in [8] that displays mutual similarities on a 2-D screen based on visual features extracted from images. The retrieved images are displayed not only in ranked order of similarity from the query but also according to their mutual similarities, so that similar images are grouped together rather than being scattered along the entire returned 1-D list.

2. 3 PCA Splats

To create such a 2-D layout, Principle Component Analysis (PCA) [10] is first performed on the retrieved images to project the images from the high dimensional feature space to the 2-D screen. Image thumbnails are placed on the screen so that the screen distances reflect as closely as possible the similarities between the images. If the computed similarities from the high dimensional feature space agree with our perception, and if the resulting feature dimension reduction preserves these similarities reasonably well, then the resulting spatial display should be informative and useful.

For our image representation, we have used three visual features: color moments [1], wavelet-based texture [2], and water-filling edge-based structure feature [3]. We should note that the choice of visual representation is not important to the methodology of this paper. In our experiments, the 37 visual features (9 color moments, 10 wavelet moments and 18 water-filling features) are pre-extracted from the image database and stored off-line. The 37-dimensional feature vector for an image, when taken in context with other images, can be projected on to the 2-D $\{x, y\}$ screen based on the 1st two principal components normalized by the respective eigenvalues. Such a layout is denoted as a PCA Splat. We note that PCA has several advantages over nonlinear methods like MDS. It is a fast, efficient and unique linear transformation that achieves the maximum distance preservation from the original high dimensional feature space to 2-D space among all possible linear transformations [10]. The fact that it fails to model nonlinear mappings (which MDS succeeds at) is in our opinion a minor compromise given the advantages of real-time, repeatable and mathematically tractable linear projections.

Let us consider a scenario of a typical image-retrieval engine at work in which an actual user is providing relevance feedback for the purposes of query refinement. Figure 1 shows an example of the retrieved images by the system (which resembles most traditional browsers in its 1D tile-based layout). The database is a collection of 534 images. The 1st image (building) is the query. The other 9 relevant images are ranked in 2nd, 3rd, 4th, 5th, 9th, 10th, 17th, 19th and 20th places, respectively.

Figure 2 shows an example of a PCA Splat for the top 20 retrieved images shown in Figure 1. In addition to visualization by layout, in this particular example, the sizes (alternatively contrast) of the images are determined by their visual similarity to the query. The higher the rank, the larger the size (or higher the contrast). There is also a number next to each image in Figure 2 indicating its corresponding rank in Figure 1.

Clearly the relevant images are now better clustered in this new layout as opposed to being dispersed along the tiled 1-D display in Figure 1. Additionally, PCA Splats convey N^2 mutual distance measures relating all pair-wise similarities between images while the ranked 1-D display in Figure 1 provides only *N*.

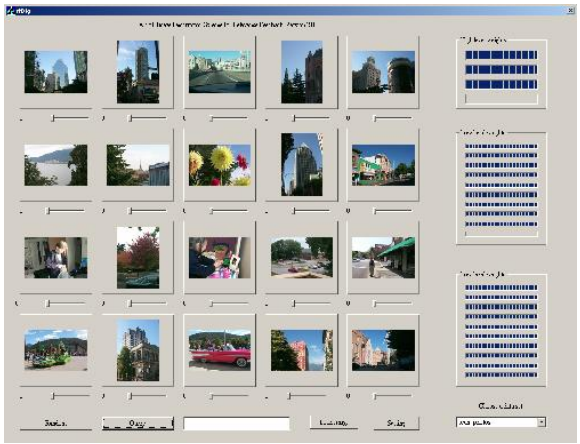

Figure 1. Top 20 retrieved images (ranked in scan-line order; the query is first in the list)

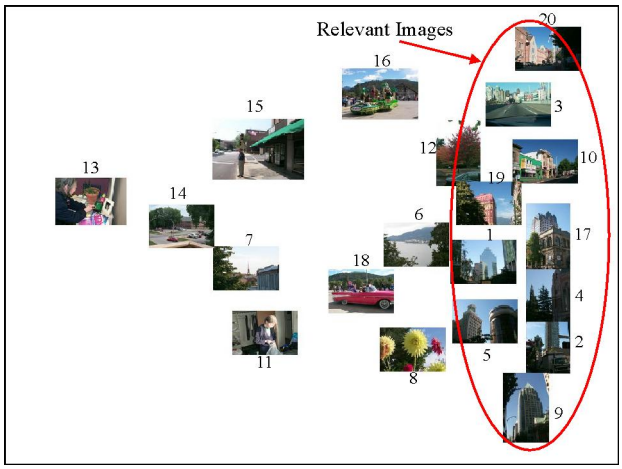

Figure 2. PCA Splat of top 20 retrieved images in Figure 1

One drawback of PCA Splats (or any low-dimensional mapping) is that inevitably some images are partially or totally overlapped (due to proximity or co-location in 2D space) which makes it difficult to view all the images at the same time. This image overlap will worsen when the number of retrieved images becomes larger. To address this problem, we have devised a unique layout optimization technique [13]. The goal is to minimize the total image overlap by finding an optimal set of image size and image positions with a little deviation as possible. A nonlinear optimization method was implemented by an iterative gradient descent method.

We note that Figure 2 is, in fact, just such an optimized PCA Splat. The overlap is clearly minimized while the relevant images are still close to each other to allow a global view. With such a display, the user can see the relations between the images, better understand how the query performed, and subsequently formulate future queries more naturally. Additionally, attributes such as contrast and brightness can be used to convey rank. We note that this additional visual aid is essentially a 3rd dimension of information display. A full discussion of the resulting enhanced layouts is deferred to our future work.

3. User-Modeling

Image content and "meaning" is ultimately based on semantics. The user's notion of content is a high-level concept, which is quite often removed by many layers of abstraction from simple low-level visual features. Even near-exhaustive semantic (keyword) annotations can never fully capture context-dependent notions of content. The same image can "mean" a number of different things depending on the particular circumstance.

By user-modeling or "context awareness" we mean that our system must be constantly aware of and adapting to the changing concepts and preferences of the user. A typical example of this human-computer synergy is having the system learn from a user-generated layout in order to visualize new examples based on identified relevant/irrelevant features. In other words, design smart browsers that "mimic" the user, and over-time, adapt to their style or preference for browsing and query display. Given information from the layout, *e.g.*, positions and mutual

distances between images, a novel feature weight estimation scheme, noted as $\boldsymbol{\alpha}$-estimation is proposed, where $\boldsymbol{\alpha}$ is a weighting vector for different features e.g., color, texture and structure (and possibly semantic keywords as well).

3.1 Subspace Estimation of Feature Weights

The weighting parameter vector is denoted as $\boldsymbol{\alpha} = \{\alpha_c, \alpha_t, \alpha_s\}^T$, where α_c is the weight for color, α_t is the weight for texture, and α_s is the weight for structure. The number of images in the preferred clustering is *N*, and $\mathbf{X}_c$ is a $L_c \times N$ matrix where the *i*th column is the color feature vector of the *i*th image, $i = 1, \cdots, N$, $\mathbf{X}_t$ is the $L_t \times N$ matrix, the *i*th column is the texture feature vector of the *i*th image, $i = 1, \cdots, N$, and $\mathbf{X_s}$ is the $L_s \times N$ matrix, the *i*th column is the structure feature vector of the *i*th image, $i = 1, \cdots, N$. The lengths of color, texture and structure features are L_c, L_t, and L_s respectively. The distance, for example Euclidean,-based between the *i*th image and the *j*th image, for $i, j = 1, \cdots, N$, in the preferred clustering (distance in 2-D space) is d_{ij}. These weights α_c, α_t, α_s are constrained such that they always sum to 1.

We then define an energy term to minimize with an *Lp* norm (with $p = 2$). This cost function is defined in Equation (1). It is a nonnegative quantity that indicates how well mutual distances are preserved in going from the original high dimensional feature space to 2-D space. Note that this cost function is similar to MDS stress, but unlike MDS, the minimization is seeking the optimal feature weights $\boldsymbol{\alpha}$. Moreover, the low-dimensional projections in this case are already known. The optimal weighting parameter recovered is then used to weight original feature-vectors prior to a PCA Splat, resulting in the desired layout.

$$J = \sum_{i=1}^{N}\sum_{j=1}^{N}\{{d_{ij}}^p - \sum_{k=1}^{L_c}{\alpha_c}^p \mid \mathbf{X}_{c(i)}^{(k)} - \mathbf{X}_{c(j)}^{(k)} \mid^p - \sum_{k=1}^{L_t}{\alpha_t}^p \mid \mathbf{X}_{t(i)}^{(k)} - \mathbf{X}_{t(j)}^{(k)} \mid^p - \sum_{k=1}^{L_s}{\alpha_s}^p \mid \mathbf{X}_{s(i)}^{(k)} - \mathbf{X}_{s(j)}^{(k)} \mid^p\}^2 \quad (1)$$

The parameter $\boldsymbol{\alpha}$ is then estimated by a constrained non-negative least-squares optimization procedure. Defining the following *Lp*-based deviations for each of the 3 subspaces:

$$V_{(ij)}^c = \sum_{k=1}^{L_c} \mid \mathbf{X}_{c(i)}^{(k)} - \mathbf{X}_{t(j)}^{(k)} \mid^p \quad , \quad V_{(ij)}^t = \sum_{k=1}^{L_t} \mid \mathbf{X}_{t(i)}^{(k)} - \mathbf{X}_{t(j)}^{(k)} \mid^p \quad , \quad V_{(ij)}^s = \sum_{k=1}^{L_s} \mid \mathbf{X}_{s(i)}^{(k)} - \mathbf{X}_{s(j)}^{(k)} \mid^p$$

Equation (1) can be re-written as the following cost function:

$$J = \sum_{i=1}^{N}\sum_{j=1}^{N}({d_{ij}}^p - {\alpha_c}^p V_{(ij)}^c - {\alpha_t}^p V_{(ij)}^t - {\alpha_s}^p V_{(ij)}^s)^2 \quad (2)$$

which we differentiate and set to zero to obtain a linear system in the *p*-th power of the desired subspace coefficients, α^p. This system is easily solved using *constrained* linear least-squares, since the coefficients must be *non-negative*: $\alpha^p > 0$. Subsequently, the subspace parameters we set out to estimate are simply the *p*-th root of the solution. In our experiments we used $p = 2$.

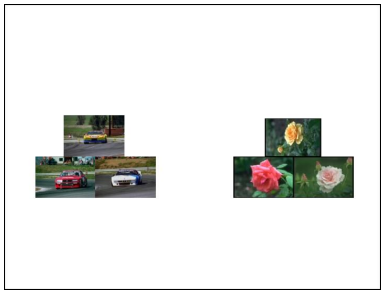

Figure 3. An example of a user-guided layout

Figure 3 shows a simple user layout where 3 car images are clustered together despite their different colors. The same is performed with 3 flower images (despite their texture/structure). These two clusters maintain a sizeable separation thus suggesting two separate concept classes implicit by the user's placement. Specifically, in this layout the user is clearly concerned with the distinction between *car* and *flower* regardless of color or other possible visual attributes.

Applying the $\boldsymbol{\alpha}$-estimation algorithm to Figure 3, the feature weights learned from this 2-D layout are $\alpha_c = 0.3729$, $\alpha_t = 0.5269$ and $\alpha_s = 0.1002$. This shows that the most important feature in this case is texture and not color, which is in accord with the concepts of car *vs*. flower as graphically indicated by the user in Figure 3.

(a) (b)

Figure 4. PCA Splat on a larger set using (a) estimated weights (b) arbitrary weights

Now that we have the learned feature weights (or modeled the user) what can we do with them? Figure 4 shows an example of a typical application: automatic layout of a larger (more complete data set) in the style indicated by the user. Fig. 4(a) shows the PCA splat using the

learned feature weight for 18 cars and 19 flowers. It is obvious that the PCA splat using the estimated weights captures the essence of the configuration layout in Figure 3. Figure 4(b) shows a PCA splat of the same images but with a randomly generated $\boldsymbol{\alpha}$, denoting an arbitrary but coherent 2-D layout, which in this case, favors color ($\alpha_c = 0.7629$). This comparison reveals that proper feature weighting is an important factor in generating the desired layouts.

4. Performance Analysis

Given the lack of sufficiently numerous (and willing) human subjects to test our system with, we undertook a Monte Carlo approach to evaluating our user-modeling estimation algorithm. Thus, we generated 1000 synthetic "user-layouts" (with random values of $\boldsymbol{\alpha}$'s representing the "ground-truth") to sample the space of all possible *consistent* user-layouts that could be conceivably generated by a human subject (an *inconsistent* layout would correspond to randomly "splattered" thumbnails in the 2-D display). In each case, the $\boldsymbol{\alpha}$-estimation method was used to estimate (recover) the original ("ground-truth") values. We should note that the parameter recovery is non-trivial due to the information lost whilst projecting from the high-dimensional feature space down to the 2-D display space. As a control, 1000 randomly generated feature weights were used to see how well they could match the synthetic user layouts (*i.e.*, by chance alone). Note that these controls were also *consistent* layouts.

Our primary test database consists of 142 images from the COREL database. It has 7 categories of car, bird, tiger, mountain, flower, church and airplane. Each class has about 20 images. Feature extraction based on color, texture and structure has been done off-line and pre-stored. Although we will be reporting on this test data set -- due to its common use and familiarity to the CBIR community -- we should emphasize that we have also successfully tested our methodology on larger and much more heterogeneous image libraries. For example: real personal photo collections of 500+ images (including family, friends, vacations, etc.).

The following is the Monte Carlo procedure was used for testing the significance and validity of user-modeling with $\boldsymbol{\alpha}$-estimation:

> Simulation: Randomly select *M* images from the database. Generate arbitrary (random) feature weights $\boldsymbol{\alpha}$ in order to simulate a "user-layout". Do a PCA Splat using this "ground truth" $\boldsymbol{\alpha}$. From the resulting 2-D layout, estimate $\boldsymbol{\alpha}$. Select a new distinct (non-overlapping) set of *M* images from the database. Do PCA Splats on the second set using the original $\boldsymbol{\alpha}$, the estimated $\boldsymbol{\alpha}$ and a third random $\boldsymbol{\alpha}$ (as control). Calculate the resulting stress and layout deviation (2-D position error) for the original, estimated and random (control) values of $\boldsymbol{\alpha}$. Repeat 1000 times

The scatter-plot of 1000 Monte Carlo trials of $\boldsymbol{\alpha}$-estimation from synthetic "user-generated" layouts is shown in Figure 5. Clearly there is a direct linear relationship between the original weights and the estimated weights. Note that when the original weight is very small (<0.1) or very large (>0.9), the estimated weight is zero or one correspondingly. This means that when one particular feature weight is very large (or very small), the corresponding feature will become the most dominant (or least dominant) feature in the PCA, therefore the estimated weight for this feature will be either 1 or 0.

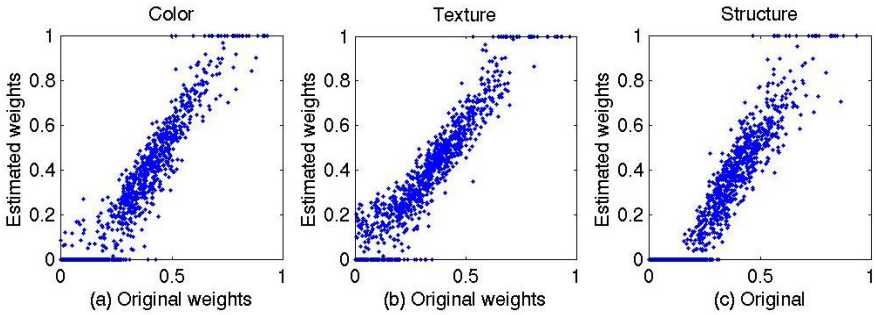

Figure 5. Scatter-plot of α-estimation: Estimated weights vs. original weights

In terms of actual measures of stress and deviation we found that the α-estimation scheme yielded the smaller deviation 78.4% of the time and smaller stress 72.9%. The main reason these values are less than 100% is due to the nature of the Monte Carlo testing and the fact that in low-dimensional (2-D) spaces, random weights can become close to the original weights and hence yield similar "user" layouts (in this case apparently ~ 25% of the time).

Another control other than random weights is to compare the deviation of an α-estimation layout generator to a simple scheme which assigns each new image to the 2-D location of its (unweighted) 37-dimensional nearest-neighbor from the set previously laid out by the "user". This control essentially operates on the principle that new images should be displayed on screen at the same location as their nearest neighbors in the original 37-dimensional feature space and thus ignores the subspace defined by the "user" in a 2-D layout.

The results of this Monte Carlo simulation are shown in Figure 6 where we see that the lay-out deviation using α-estimation (red: mean=0.9691 std=0.7776) was consistently lower -- by almost an order of magnitude -- than nearest neighbor (blue: mean=7.5921, std=2.6410).

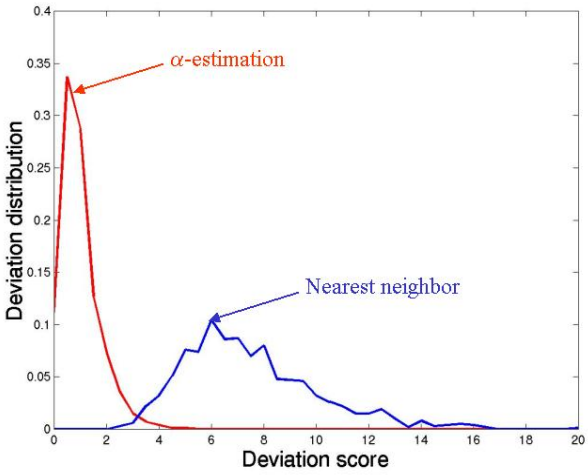

Figure 6. Comparison of the distribution of α-estimation vs. nearest neighbor deviation scores (note that in every one of the random trials the α-estimation deviation was smaller)

6. Discussion

There are several areas of future work. First, more extensive user-layout studies are needed to replace Monte Carlo simulations. It is critical to have real users with different notions of content do layouts and to see if our system can model them accurately. Moreover, we have already integrated hybrid visual/semantic feature weighting that requires further testing with human subjects. We have designed our system with general CBIR in mind but more specifically for personalized photo collections. The final visualization and retrieval interface can be displayed on a computer screen, large panel projection screens, or -- for example -- on embedded tabletop devices [12], designed specifically for purposes of story-telling or multi-person collaborative exploration of large image libraries.

Finally, we note that although the focus of this paper has been on visual content analysis, the same framework for visualization and user-modeling would apply to other data entities such as video clips, audio files, specialized documents (legal, medical, etc), or web pages. The main difference would be the choice of features used, their representation in high-dimensional spaces and the appropriate metrics.

References

1 M. Stricker and M. Orengo, “Similarity of Color Images”, *Proc. SPIE Storage and Retrieval for Image and Video Databases*, 1995

2 J. R. Smith and S. F. Chang, “Transform Features for Texture Classification and Discrimination in Large Image Database”, *Proc. IEEE Intl. Conf. on Image Proc.*, 1994

3 S. X. Zhou, Y. Rui and T. S. Huang, “Water-filling algorithm: A novel way for image feature extraction based on edge maps”, in *Proc. IEEE Intl. Conf. On Image Proc.*, Japan, 1999

4 S. Santini and R. Jain, “Similarity measures”, *IEEE PAMI*, vol. 21, no. 9, 1999

5 M. Popescu and P. Gader, “Image Content Retrieval From Image Databases Using Feature Integration by Choquet Integral”, in *SPIE Conference Storage and Retrieval for Image and Video Databases VII*, San Jose, CA, 1998

6 D. M. Squire, H. Müller, and W. Müller, “Improving Response Time by Search Pruning in a Content-Based Image Retrieval System, Using Inverted File Techniques”, *Proc. of IEEE workshop on CBAIVL*, June 1999

7 D. Swets, J. Weng, “Hierarchical Discriminant Analysis for Image Retrieval”, *IEEE PAMI*, vol. 21, no.5, 1999

8 Y. Rubner, “Perceptual metrics for image database navigation”, Ph.D. dissertation, Stanford University, 1999

9 W. S. Torgeson, *Theory and methods of scaling*, John Wiley & Sons, New York, NY, 1958

10 Jolliffe, I.T., *Principal Component Analysis*, Springer-Verlag, New-York, 1986

11 S. Santini, Ramesh Jain, "Integrated browsing and querying for image databases", *July-September Issue, IEEE Multimedia Magazine*, pp.26-39, 2000

12 B. Moghaddam *et al*, "Visualization and Layout for Personal Photo Libraries," International Workshop on Content-Based Multimedia Indexing (CBMI'01), September, 2001.

13 Q. Tian ., B. Moghaddam, T.S. Huang, "Display Optimization for Image Browsing," International Workshop on Multimedia Databases and Image Communication (MDIC'01), September, 2001.