High-Entropy Layouts for Content-based Browsing and Retrieval *

Ruixuan Wang Stephen J. McKenna Junwei Han
School of Computing, University of Dundee, UK
{ruixuanwang,stephen,jeffhan}@computing.dundee.ac.uk

April 10, 2009

Abstract

Multimedia browsing and retrieval systems can use dimensionality reduction methods to map from high-dimensional content-based feature distributions to low-dimensional layout spaces for visualization. However, this often results in displays in which many items are occluded whilst large regions are empty or only sparsely populated with items. Furthermore, such methods do not take into account the shape of the region of layout space to be populated. This paper proposes a layout method that addresses these limitations. Layout distributions with low Renyi quadratic entropy are penalized since these result in displays in which some regions are over-populated (i.e. many images are occluded), sparsely populated or empty. Experiments using two image datasets and a comparison with two related methods show the effectiveness of the proposed method.

1 INTRODUCTION

Image or multimedia browsing and retrieval systems need to enable users to visualize collections of multimedia items (or their thumbnails or icons) by laying them out appropriately for display. This paper proposes a method for arranging such items, especially images, in a lower-dimensional (e.g., 2D in [7] or 3D in [8, 10]) layout space. Image data are used to illustrate and evaluate the method although in principle it could be applied to other media such as video.

Browsing systems often categorize items (e.g., images) into different classes and simply lay them out on a 2D display as lists of thumbnails for each class of items [5]. Similarly, retrieval systems often lay out items as lists of thumbnails ordered by similarity to a query (e.g. VisualSEEk [13]). Such 1D lists do not portray the mutual relationships between items well. Alternatively, 2D map-based visualizations [4, 8, 10, 12] lay out items such that similar items appear close to one another on the 2D display while very different items will be further apart. These 2D layout techniques differ in how they extract high-dimensional feature vectors from items, measure pairwise item similarity, and

*This is a pre-publication version of the conference paper which is to appear in CIVR'09, July 8-10, Santorini, Greece.

perform dimensionality reduction to map the distribution of items from the high-dimensional space to a 2D space [11]. For example, Rubner *et al.* [12] used Earth Mover's Distance to measure pair-wise dissimilarity based on color and texture features, and multi-dimensional scaling (MDS) [3] to map to a 2D space. The dimensionality reduction methods that are used in these visualization techniques were formulated with the goal of approximating the distribution of items in the chosen high-dimensional feature space. While this provides useful visualization of this distribution, there may be more appropriate ways of laying out items as images for browsing. Firstly, such methods often result in displays in which many images occlude other images whilst large available areas of the layout space are empty or only sparsely populated with images. Secondly, such methods do not take into account the shape of the region in the layout space that is available to be populated.

The goal of the research described in this paper is to address these shortcomings. A method is proposed that generates layouts which conform to a pre-specified shape, approximate the feature space distribution, and result in rendered displays that are populated more uniformly with images. The shape of the layout region to be populated by the method can be specified to be annular (Figure 3(a) and (c)), rectangular (Figure 4(c)), or elliptical, for example. The relative sizes and the aspect ratios of the images are taken into account by the method. The images are taken to form a distribution in the low-dimensional layout space. Layout distributions with low entropy are penalized since these result in displays in which some regions are over-populated (i.e. many images are occluded) and other regions are sparsely populated or empty. High entropy layouts, on the other hand, arrange images more uniformly over the layout region.

The contributions of this paper are (i) a novel formulation for content-based visualization of multimedia collections based on combining existing manifold learning methods with entropy, (ii) more specifically, the use of Renyi quadratic entropy in this context based on pairwise measures between Gaussians, (iii) the use of a penalty term to obtain image layouts in a layout region with pre-specified shape, (iv) empirical evaluation using gradient-based optimization on two datasets of images, and (v) empirical comparison with two previously proposed methods for image overlap reduction.

The next Section formulates the problem as an optimization problem. Section 3 describes how this optimization can be performed using gradient-based methods. Section 4 briefly discusses the methods in the literature whose aims are closest to those of this paper. Section 5 presents experiments including comparisons with these previous methods.

2 FORMULATION

Given a set of images $\{I_i\}, i = 1, \ldots, N$, with extracted high-dimensional feature vectors $\{\mathbf{x}_i\}$ and image (or thumbnail) sizes $\{\mathbf{s}_i\}$, the problem of interest is to arrange the images on a (virtual) layout region $\mathbf{R}$ with predetermined shape by trading off two requirements: (1) the distance between images in the layout space should depend on the similarity of their content, and (2) images should spread out so as to make good use of the layout region. Note that the layout region is a bounded region of the layout space. It is not limited to be rectangular and can be annular (Figure 3(a)) for example.

The first requirement can be met by manifold learning techniques. By assuming that the images are distributed on a low-dimensional nonlinear manifold embedded in the high-dimensional space, manifold learning techniques can be applied to discover the structure of the data manifold and unfold the manifold into a vector space. Once the original high-dimensional data points can be faithfully embedded into the lower-dimensional vector space, the relative proximity of images in the database will be approximately preserved in the lower-dimensional (e.g., 2D or 3D) space. This is referred to as content structure preservation. Based on the manifold learning result, a large collection of high-dimensional images can be visualized in the 2D or 3D space. According to criteria for manifold structure and structure preservation when unfolding the manifold, many different manifold learning techniques have been proposed, e.g., Isomap [14], Laplacian eigenmaps [2], diffusion maps [9], and maximum variance unfolding [16]. In principle, any manifold learning technique can be used.

Here Isomap is used as an example for empirical comparison. Isomap first constructs a sparse graph based on $\{\mathbf{x}_i\}$ in which there is a one-to-one correspondence between images and vertices in the graph. Edges are constructed between similar images by the K-nearest neighbor (KNN) method. Each edge is assigned a weight w_{ij} which is the dissimilarity between the two neighboring images. An approximation, D_{ij}, to the geodesic distance between any two images is then obtained as the shortest path between their corresponding vertices in the graph. Without loss of generality, $\{D_{ij}\}$ are normalized such that the maximum D_{ij} is limited by the layout region size. Isomap can determine image positions $\{\mathbf{y}_i\}$ in the lower-dimensional (e.g., 2D or 3D) vector space by minimizing E_s:

$$E_s = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{N}(d_{ij} - D_{ij})^2 \,, \tag{1}$$

where d_{ij} is the Euclidean distance between $\mathbf{y}_i$ and $\mathbf{y}_j$. Note that when two images I_i and I_j are similar in content, the distance D_{ij} between them will be small and accordingly the two images in the lower-dimensional space will probably appear close to each other.

For the second requirement, we propose that layouts with high entropy are preferable. Given an image position $\mathbf{y}_i$ in the lower-dimensional layout space, a Gaussian distribution $G(\mathbf{y}_i, \mathbf{\Sigma}_i)$ is used to approximate the spatial distribution of this image in the space, where $\mathbf{\Sigma}_i$ is determined by image size and shape, the number of images, and the size of the layout region. Then, the Gaussian distributions of all the images can be combined to obtain a Gaussian mixture with equal weight for each Gaussian component, i.e.,

$$p(\mathbf{y}) \quad = \quad \frac{1}{N}\sum_{i=1}^{N} G(\mathbf{y} - \mathbf{y}_i, \mathbf{\Sigma}_i)\,. \tag{2}$$

Instead of Shannon entropy, the use of Renyi's quadratic entropy measure is proposed here. This is because the quadratic Renyi entropy, H, of a Gaussian mixture can be efficiently estimated as a sum of pair-wise measures between

Gaussian components [15], i.e.,

$$\begin{aligned} H &= -\log \int_{\mathbf{y}} p(\mathbf{y})^2 d\mathbf{y} \\ &= -\log\{\frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} G(\mathbf{y}_i - \mathbf{y}_j, \mathbf{\Sigma}_i + \mathbf{\Sigma}_j)\} . \end{aligned} \tag{3}$$

Maximizing H (or minimizing $-H$) would have the effect of arranging images so that their distribution is close to uniform. This means that large areas of empty space in the layout region are avoided and that the extent to which images overlap each other is kept small. This has to be traded-off against content structure preservation, so the goal is to minimize E_λ,

$$E_\lambda = (1 - \lambda)E_s - \lambda H , \tag{4}$$

subject to the constraint that each image should stay within the region $\mathbf{R}$, where $\lambda \in [0, 1]$ is a trade-off parameter. The value of λ should be determined in an application-dependent way. When λ is close to 0, preservation of manifold structure is emphasized. When λ is close to 1, spreading the images to maximize entropy is emphasized.

3 OPTIMIZATION

The constrained optimization problem in Equation (4) could be solved using various well-known optimization methods. Here we use a penalty function method to penalize image positions outside $\mathbf{R}$. Intuitively, the larger the Euclidean distance from the image position $\mathbf{y}_i$ to the layout region $\mathbf{R}$, the worse the image is positioned, and therefore the higher the penalty. Denote by E_b the mean penalty cost of all image positions, i.e.,

$$E_b = \frac{1}{N} \sum_{i=1}^{N} f(\mathbf{y}_i) , \tag{5}$$

where $f(\mathbf{y}_i)$ is a monotonically increasing non-negative function of the Euclidean distance from $\mathbf{y}_i$ to the layout region $\mathbf{R}$ (i.e., $\min_{\mathbf{y}\in\mathbf{R}} \|\mathbf{y} - \mathbf{y}_i\|$). Then, the problem can be finally transformed to that of minimising E, where

$$E = E_\lambda + \gamma E_b , \tag{6}$$

and γ is a constant to balance E_λ and E_b.

Gradient-based methods can be used to find a local minimum of E. From Equation (6),

$$\frac{\partial E}{\partial \mathbf{y}_j} = (1 - \lambda)\frac{\partial E_s}{\partial \mathbf{y}_j} - \lambda \frac{\partial H}{\partial \mathbf{y}_j} + \gamma \frac{\partial E_b}{\partial \mathbf{y}_j} . \tag{7}$$

The gradient of E_s with respect to $\mathbf{y}_j$ has been derived by Kruskal [3]:

$$\frac{\partial E_s}{\partial \mathbf{y}_j} = -2 \sum_{i \neq j} \frac{(d_{ij} - D_{ij})}{d_{ij}} \cdot (\mathbf{y}_i - \mathbf{y}_j) , \tag{8}$$

From Equation (3), we can derive the gradient of H with respect to $\mathbf{y}_j$:

$$\begin{aligned}\frac{\partial H}{\partial \mathbf{y}_j} &= -\frac{1}{\alpha}\sum_i \{G(\mathbf{y}_i - \mathbf{y}_j, \mathbf{\Sigma}_i + \mathbf{\Sigma}_j)(\mathbf{\Sigma}_i + \mathbf{\Sigma}_j)^{-1} \\ &\qquad \cdot(\mathbf{y}_i - \mathbf{y}_j)\}\,, \end{aligned} \tag{9}$$

where $\alpha = \sum_i \sum_j G(\mathbf{y}_i - \mathbf{y}_j, \mathbf{\Sigma}_i + \mathbf{\Sigma}_j)$.

For the gradient of E_b with respect to $\mathbf{y}_j$, a discrete approximation is adopted because, in general, it is difficult to parametrically represent the function $f(\mathbf{y}_i)$ due to the freeform shape of the layout region. In the approximation, the k-th component of the gradient of E_b with respect to $\mathbf{y}_j$ is computed by

$$\frac{\partial E_b}{\partial y_{jk}} \approx \frac{1}{N}\sum_{i=1}^{N} \frac{f(\mathbf{y}_j + \delta \mathbf{u}_k) - f(\mathbf{y}_j)}{\delta}\,, \tag{10}$$

where δ is the discrete unit scale and $\mathbf{u}_k$ is the basis vector for the k-th dimension of the layout space.

For optimization, good initial positions $\{\mathbf{y}_i\}$ can be obtained by minimizing E_s using the Isomap method.

4 RELATED WORK

Methods have been proposed previously for reducing image overlap when visualizing small collections of images. Moghaddam *et al.* [8] and Nguyen *et al.* [10] used gradient descent methods to move overlapped images towards unoccupied layout space without constraining image positions to be within a layout region. Basalaj [1] and Liu *et al.* [6] used an analogue of MDS in a discrete domain to display each image within a single cell of a grid. While these approaches can help to reduce image overlap, they mainly deal with small numbers of images (about $20 \sim 200$).

5 EMPIRICAL EVALUATION

The method is not limited by the dimensionality of the layout space, nor by the shapes of the images. However, in order to experimentally evaluate it, in the following, we consider a 2D display and each image I_i is assumed to be aligned with the axes of the layout space. In this case, the covariance matrix $\mathbf{\Sigma_i}$ is diagonal, i.e.,

$$\mathbf{\Sigma_i} = \begin{pmatrix} \sigma_{i1}^2 & 0 \\ 0 & \sigma_{i2}^2 \end{pmatrix}$$

Note that σ_{i1} and σ_{i2} should be neither very small nor very large because in both cases, H (Equation (3)) will converge to a constant function and therefore the image layout cannot be effectively spread. Here we propose a method to automatically determine suitable values of σ_{i1} and σ_{i2} based on image sizes, layout region size, and the number of images, i.e.,

$$\sigma_{i1} = \tfrac{w_i}{2}\sqrt{\tfrac{|\mathbf{R}|}{N\,\overline{w}\,\overline{h}}}\ , \quad \sigma_{i2} = \tfrac{h_i}{2}\sqrt{\tfrac{|\mathbf{R}|}{N\,\overline{w}\,\overline{h}}}\ ,$$

where $|\mathbf{R}|$ is the area of the layout region, h_i and w_i are the height and width of the i^{th} image, and $\overline{w}$ and $\overline{h}$ are the average width and height of the images. $\sqrt{\frac{|\mathbf{R}|}{N\,\overline{w}\,\overline{h}}}$ is a global scale for each image such that the combination $p(\mathbf{y})$ (Equation (2)) of all images' spatial distributions can effectively cover the layout region. For a fixed layout region $\mathbf{R}$, the greater the number of images and the larger the mean image size, the smaller the global scale. For a fixed set of images, the larger the layout region, the larger the global scale. Every pair $(\sigma_{i1}, \sigma_{i2})$ is linearly related to the corresponding image size and the global scale.

The method was evaluated using two image databases: 1000 images of textile designs from a commercial archive and 1000 art images from a public museum collection. Two kinds of features were used to represent images. Color histograms with 512 bins were extracted by regularly quantizing hue into 32 values and saturation into 16 values in the HSV color space. Texture features were extracted by performing multi-scale Gabor filtering and then computing the means and variances of the normalised magnitude responses at each scale and orientation, giving 108 texture features. For both kinds of features, Euclidean distance was used to determine the nearest neighbors for constructing the manifold structure (see Section 2). In the tests, each image I_i was resized such that the maximum of its height and width was $0.08\sqrt{|\mathbf{R}|}$. All the tests were performed using a Matlab R2007a implementation running on an Intel Core 2 Quad 2.4 GHz PC with 3.5GB RAM.

5.1 Overlap versus Structure Preservation

The method was compared to the methods of Moghaddam *et al.* [8] and Nguyen *et al.* [10]. Since the stated aim of these methods was overlap reduction, the proposed method was compared to these previous methods using a measure of overlap similar to that used by those authors. Specifically, overall image overlap e_o, was measured as the sum of all pair-wise image overlaps $\sqrt{z_{ij}}$ in the layout space, i.e.,

$$e_o = \sum_{i=1}^{N}\sum_{j=1}^{N}\sqrt{z_{ij}}\,, \tag{11}$$

where the area z_{ij} of the overlap region can be directly computed from the image positions, $\mathbf{y}_i$ and $\mathbf{y}_j$, and the image sizes, (w_i, h_i) and (w_j, h_j). Note that before computing image overlap e_o, all $\mathbf{y}_i$'s and $\mathbf{y}_j$'s have to be normalized such that all images are positioned in the default layout region.

In order to compare these methods with the proposed method, $\gamma = 0$ in Equation (6) because E_b was not used in the previous methods. In this case, for each $\boldsymbol{\Sigma_i}$, $\sigma_{i1} = w_i/2$ and $\sigma_{i2} = h_i/2$ without global scaling.

In the evaluation, structure preservation error, e_s, was measured as

$$e_s = \min_{\beta}\{\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{N}(\beta \cdot d_{ij} - D_{ij})^2\}\,, \tag{12}$$

where the value of the normalization factor β at this minimum can be analytically computed as:

$$\beta^* = \frac{\sum_{i=1}^{N}\sum_{j=1}^{N} d_{ij} \cdot D_{ij}}{\sum_{i=1}^{N}\sum_{j=1}^{N} d_{ij}^2}\,. \tag{13}$$

β is necessary to compute e_s because, intuitively, the structure of the image distribution should be the same if all d_{ij} are scaled by the same amount.

For fair comparison and consistent notation, the same E_s was used for structure preservation in Nguyen's method such that their cost function (Equation (9) in [10]) is written $E = (1-\lambda)E_s + \lambda \cdot E_V$. The cost function (Equation (1) in [8]) of Moghaddam's method was reformulated as $E = (1-\lambda) \cdot S \cdot G + \lambda \cdot F$. Please refer to [8] and [10] for explanation of E_V, G, S and F. In both these previous methods, each image was approximated by a circular image with radius $s_{max}/2$.

In this test, 100 images were uniformly sampled from the set of textile images, and each image was represented by the Gabor features. The trade-off parameter λ was gradually varied from 0 to 1. For every λ value, the structure error e_s and overlap e_o were measured based on the convergent result of each method. In implementation, stochastic gradient descent was adopted since it helped escape local minima and reduced the number of iterations compared to a non-stochastic gradient descent. For each of the three methods, it took between 20 and 100 iterations to obtain convergent results, with each iteration taking about 30 milliseconds. From the relationships between structure preservation and overlap (Figure 1), it can be seen that for any given structure error, the proposed method can always obtain equivalent or lower image overlap than the other two methods. In addition, the minimum image overlap (i.e., ~ 2) obtained by the proposed method is much less than those (i.e., ~ 4.5 and ~ 4 respectively) obtained by the other two methods. This can be perceptually verified from the corresponding visualizations of the image collection (Figure 2). Compared to the initial visualization obtained by Isomap (Figure 2(a)), there was much less image overlap in the visualizations obtained by Moghaddam's method and Nguyen's method (Figure 2(b)(c)). However, the smallest image overlap appears in the visualization obtained by the proposed method (Figure 2(d)). In Figure 2(d), almost every image is clearly visualized with very small overlap by other images. The proposed method performs better probably for two reasons. Firstly, both width and height of each image are embedded in the cost function, by which the pair-wise image overlap can be more effectively approximated. In the two previous methods, each image was approximated by a circular image. Secondly, in the proposed cost function (Equation (4)), the cost term $-H$ that penalizes overlap is smoother and has a larger region of effect on the pair-wise distance, which indicates that a good minimum of the cost function can be more easily found. In comparison, the cost terms for image overlap in both previous methods are piecewise smooth functions with no effect on pair-wise distance when there is no image overlap between two images, making it more difficult to find a good minimum.

5.2 Layout Region Shape

The proposed method can be applied to visualize collections of images on layout regions of various shapes. Here, an annular layout region (Figure 3(a)) and a rectangular layout region with a triangular hole (Figure 3(b)) were used to visualize 200 images of textile designs. Each image was represented by its color histogram. The function $f(\mathbf{y}_i)$ in Equation (5) was simply the square of the Euclidean distance from $\mathbf{y}_i$ to the layout region $\mathbf{R}$. The algorithm was initialized

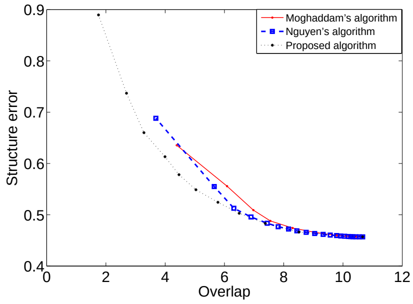

Figure 1: The relationships between structure error and image overlap obtained by Moghaddam's method, Nguyen's method, and the proposed method.

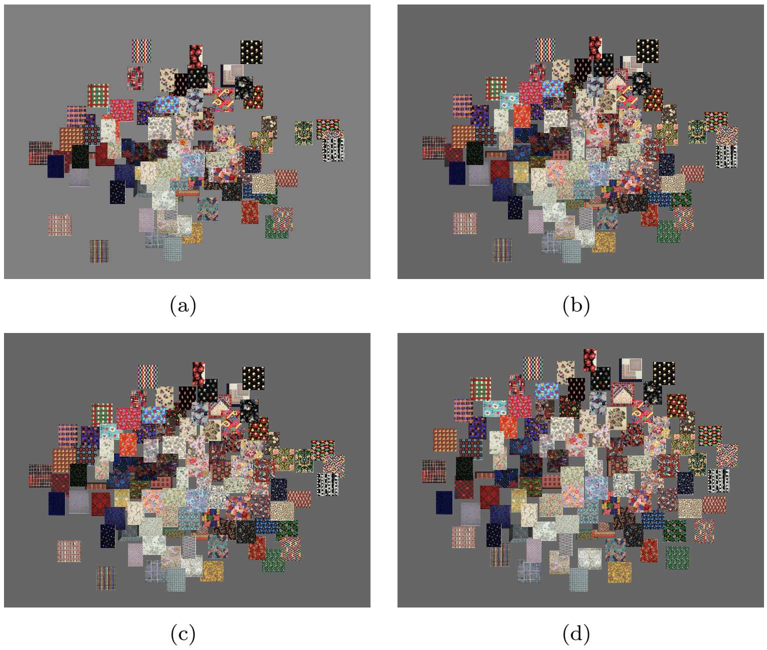

Figure 2: Visualizations of 100 textile images. (a) Visualization based on the image positions obtained by Isomap. (b–d) Visualizations with least image overlap (by setting $\lambda = 1$) obtained by Moghaddam's method, Nguyen's method, and the proposed method.

using Isomap and then run with $\lambda = 0.9$ followed by further iterations with $\lambda = 1.0$ in order to spread out images in each layout region. γ was experimentally set to 10 which was sufficient to constrain all image positions to lie in the layout region. The method took approx 90 milliseconds per iteration of the stochastic gradient-based optimization. Figures 3(c) and (d) show that all images are spread out in the layout region, while Figures 3(e) and (f) qualitatively confirm that images similar in color are still positioned close to one another.

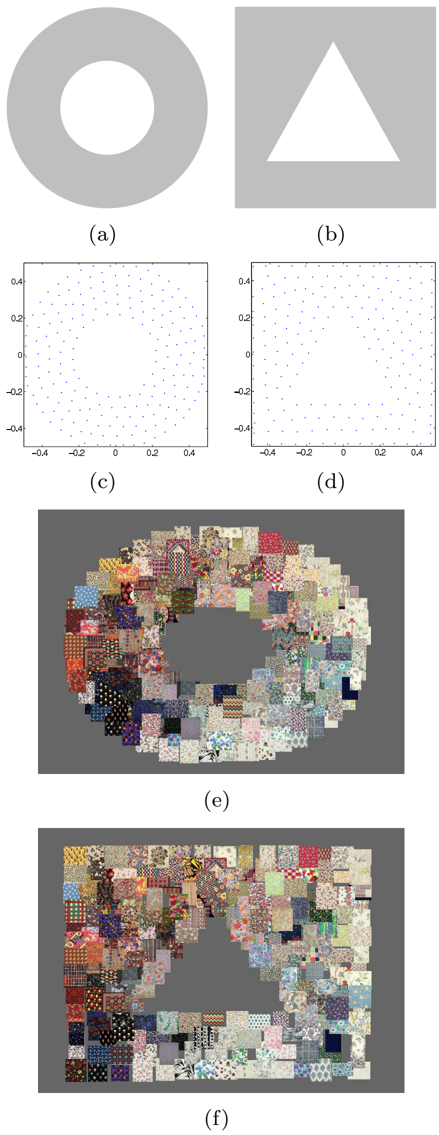

Figure 3: Visualization of 200 textile images on two different layout regions. (a) Annular layout region. (b) Rectangular layout region with a triangular hole. (c,d) Positions of image centers in layouts obtained. (e,f) Corresponding visualizations of the images based on the layouts.

5.3 Visualizations for Browsing

The performance of the method for larger collections of images was tested using the two sets of 1000 images. Gabor features were used to represent the textile images and color histograms were used to represent the art images. γ was set to 10 as in the previous test. The method took about 1.5 seconds per iteration. Figure 4 illustrates the image positions and the corresponding image visualizations in the 2D display for the art image set. In the initial image distribution obtained by Isomap (Figure 4(a)), most images are clustered around the center of the rectangular layout region and a few images are irregularly distributed near the boundaries. By trading off the structure preservation and entropy, the images are more uniformly distributed without strong clusters (Figure 4(b)). If entropy is emphasized (i.e., $\lambda = 1$), the images are most uniformly separated from their neighbors (Figure 4(c)). The corresponding image visualizations (Figures 4(d)(f)(h)) produced by rendering the images to a display show that images similar in color are still positioned close to one another when the requirement of structure preservation is relaxed. Obviously, total image overlap is always large when visualizing 1000 images on a small 2D display. In order to better show the effect of the method, the three distributions are zoomed in around one image near the layout center and then the corresponding visualizations are shown in Figures 4(e)(g)(i). Note that the image positions are scaled by the zoom operation but the images themselves are not. Figures 4(e)(g)(i) clearly show that the image overlap can be effectively reduced by the proposed method. In image browsing, the proposed method can provide an effective way for users to zoom in to a large collection of images to view subsets of the images with less occlusion.

Figure 5 illustrates the image positions and corresponding image visualizations for the textile images, and the zoom-in visualizations around an image near the layout center. Similar observations can be made as above for the art images. Here the images are distributed according to texture features instead of color features. In Figures 5(d)(f)(h), the roughness of the image texture changes smoothly from top to bottom in the display. The smooth change of texture can help users to browse a large collection of images and find images of interest with specific texture information.

6 CONCLUSION

A new problem formulation for arranging items for display was developed by combining manifold learning with Renyi entropy, subject to the constraint that each item should stay within a specified layout region. The inclusion of the constraint into the cost function can ensure that all images are positioned in the layout region. Renyi entropy can effectively incorporate image size and aspect ratio information and be applied to spread out the images in the layout region. Experiments suggested that the proposed method performs better than related methods, and it provides an effective way to arrange a large collection of images for content-based browsing and retrieval applications.

7 Acknowledgements

The authors are grateful to Annette Ward, Ian Ricketts and Mike Stapleton for helpful discussions, and to Anna Buruma, Peter Taylor and James Stevenson for providing the image collections. This research was supported by the UK Technology Strategy Board grant "FABRIC: Fashion and Apparel Browsing for Inspirational Content" in collaboration with Liberty Fabric, System Simulation, Calico Jack Ltd., and the Victoria and Albert Museum. The Technology Strategy Board is a business-led executive non-departmental public body, established by the government. Its mission is to promote and support research into, and development and exploitation of, technology and innovation for the benefit of UK business, in order to increase economic growth and improve the quality of life. It is sponsored by the Department for Innovation, Universities and Skills (DIUS). Please visit www.innovateuk.org for further information.

References

[1] W. Basalaj. *Proximity visualisation of abstract data.* PhD thesis, University of Cambridge, 2000.

[2] M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *NIPS*, volume 14, pages 585–591, 2002.

[3] M. Cox and M. Cox. *Multidimensional Scaling.* Chapman and Hall, 2001.

[4] J. Fan, Y. Gao, H. Luo, D. A. Keim, and Z. Li. A novel approach to enable semantic and visual image summarization for exploratory image search. In *ACM Conf. on Multimedia Information Retrieval*, pages 358–365, Vancouver, 2008.

[5] H. Kang, B. Bederson, and B. Suh. Rcapture, annotate, browse, find, share: Novel interfaces for personal photo management. *International Journal of Human-Computer Interaction*, 23(3):315–337, 2007.

[6] H. Liu, X. Xie, X. Tang, Z. Li, and W. Ma. Effective browsing of web image search results. In *ACM SIGMM Workshop on Multimedia Information Retrieval*, pages 84–90, 2004.

[7] T. Meiers, T. Sikora, and I. Keller. 3D browsing environment for mpeg-7 image databases. In *Proc. SPIE Storage and Retrieval for Media Databases*, pages 324–335, 2002.

[8] B. Moghaddam, Q. Tian, N. Lesh, C. Shen, and T. Huang. Visualization and user-modeling for browsing personal photo libraries. *International Journal of Computer Vision*, 56(1):109–130, 2004.

[9] B. Nadler, S. Lafon, R. Coifman, and I. Kevrekidis. Diffusion maps, spectral clustering and eigenfunctions of Fokker-Planck operators. In *NIPS*, volume 18, 2005.

[10] G. Nguyen and M. Worring. Interactive access to large image collections using similarity-based visualization. *Journal of Visual Languages and Computing*, 19(2):203–224, 2006.

[11] K. Rodden. *Evaluating similarity-based visualisations as interfaces for image browsing*. PhD thesis, University of Cambridge, 2002.

[12] Y. Rubner, C. Tomasi, and L. Guibas. A metric for distributions with applications to image databases. In *ICCV*, pages 59–66, Bombay, India, 1998.

[13] J. Smith and S. Chang. VisualSEEk: a fully automated content-based image query system. In *ACM Multimedia*, pages 87–98, Boston, 1997.

[14] J. Tenenbaum, V. de Silva, and J. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[15] K. Torkkola. Feature extraction by non-parametric mutual information maximization. *Journal of Machine Learning Research*, 3:1415–1438, 2003.

[16] K. Q. Weinberger, B. D. Packer, and L. K. Saul. Nonlinear dimensionality reduction by semidefinite programming and kernel matrix factorization. In *Proc. of Int. Workshop on AI and Statistics*, 2005.

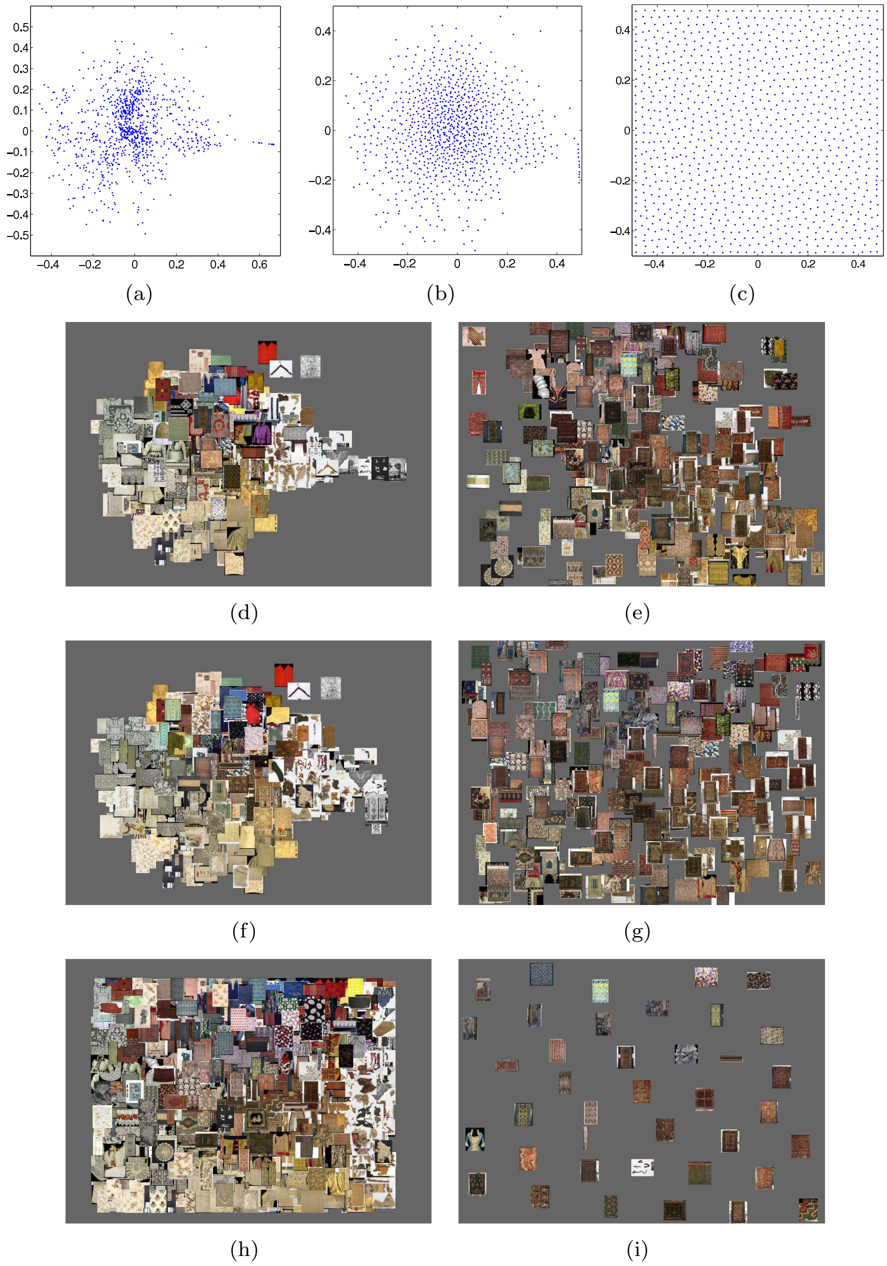

Figure 4: Visualization of 1000 art images by the proposed method using color features. Image positions obtained using (a) Isomap, (b) a trade-off ($\lambda = 0.8$) between structure preservation and image overlap, and (c) an emphasis on maximizing entropy. (d)(f)(h) The corresponding visualizations of the image collection. (e)(g)(i) The corresponding visualizations after scaling the image positions around an image at the layout center.

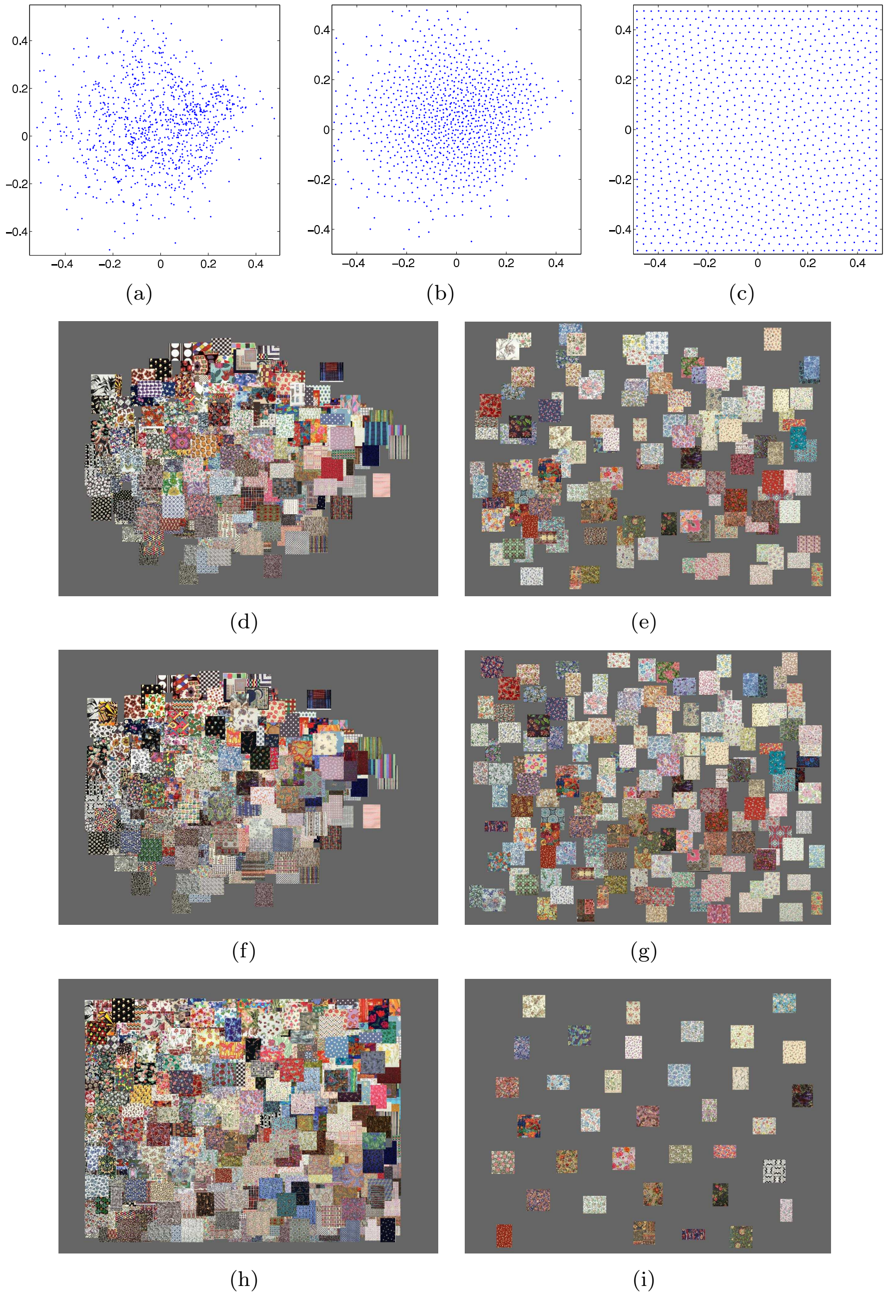

Figure 5: Visualization of 1000 textile images by the proposed method using texture features. Image positions obtained using (a) Isomap, (b) a trade-off ($\lambda = 0.8$) between structure preservation and image overlap, and (c) an emphasis on maximizing entropy. (d)(f)(h) The corresponding visualizations of the image collection. (e)(g)(i) The corresponding visualizations after scaling the image positions around an image at the layout center.