BROWSING AN ELECTRONIC MAIL-ORDER CATALOGUE WITH PICSOM CONTENT-BASED IMAGE RETRIEVAL SYSTEM

Ville Viitaniemi and Jorma Laaksonen

Helsinki University of Technology
Laboratory of Computer and Information Science
P.O. Box 5400, FIN-02015 HUT, Finland

This paper describes an example case where the PicSOM general-purpose content-based image retrieval system is applied onto a task of browsing electronic mail order catalogues. The images are indexed by their visual features calculated after segmenting the images into object and background. Colour and shape features are extracted from the foreground objects. The images are browsed by querying the image database through the normal web-based user interface of the PicSOM system. The special needs and possibilities of the narrow application domain are observed and some improvements are outlined.

1 INTRODUCTION

Content-based image retrieval (CBIR) systems index image databases with descriptors derived from the visual content of the images. Most practical CBIR systems are concerned with approximate queries where the goal is to find images visually similar to a specified target image.

In a general case assessing the image similarity presents a difficult problem. The problem is related to the general image understanding problem which has defied decades of research in field of artificial intelligence. The depth of image understanding involved in the characterisation of image content can be quantified into semantic levels [3]. Most existing CBIR systems operate on the first, most superficial level. To the user of CBIR systems the subsequent, sementically more involved levels are more meaningful. One of the principal challenges of CBIR is bridging the semantic gap, i.e. making the systems and the users to use similar kind of criteria to judge the similarity of images. However, if the images are constrained to come from a specialised domain the situation is not that difficult anymore. In such a narrow image domain the semantically low level image features more directly correspond to the high level interpretations of the images.

A popular model for CBIR is the vector space model, either used as such or as a part of a larger system. In the vector space model the images are represented as points in a multi-dimensional feature space. In other words, contents of an image is described with a fixed-length list of numbers. Similarity between images is equated with the distance of the corresponding points in the feature space.

One way to categorise the image queries that might be posed on a CBIR system is to do it according to the specificity of the response the user hopes for. In his survey of CBIR [5], Johansson presents the following as examples of possible types of tasks:

1. *Target search*. In target search the user already knows that there exists a certain image in the image database. The user's recollection of the image may be more or less exact. Moreover, depending on the system's flexibility, the user is able to transfer varying proportion of her information about the image to the system so that it can be used as a specification in a database search.

2. *Category search*. In category search the user is interested in any/all the images that possess a given property. The property might be for example the presence of airplane in the image or the (subjective) similarity to a given template image.

3. *General browsing*. In general browsing the goal is to show the user the parts of the image database she might be interested in. The goal itself is more or less vague. It might be desirable for an image database browsing system to:

   - provide the user with an overview of the image database
   - provide a detailed view of individual images
   - create a reasonable (from the user's point of view) ordering of images, thus enabling gradual navigation in the image space
   - provide means to alter the ordering criterion of images

In this article we describe the testing of the suitability of a general purpose CBIR system for browsing a narrow domain database. More specifically, the PicSOM CBIR system was used to browse a database which has been collected from the electronic mail order catalogues of two garment stores.

The vector space model facilitates the modularity of a CBIR system. When using the model the system can be divided into a backbone module for general purpose vector space manipulation on one hand and to an application area specific image analysis module on the other hand. The backbone module takes care of such tasks as storing and querying the databases of feature vectors and providing an user interface for the queries. The task of the image analysis module is to produce a feature vector as output when an image is given as input. Such a division into modules can be found in the PicSOM CBIR system.

2 THE PICSOM SYSTEM

As a part of the current research project, an experimental content-based image retrieval system called PicSOM has been implemented [11–13]. PicSOM uses a World Wide Web browser as the user interface and the Tree Structured Self-Organising Map (TS-SOM) [7, 8] as the image similarity scoring method. TS-SOMs are used instead of plain Self-Organising Maps (SOM) [6] for the sake of computational effectiveness. In addition, the use of TS-SOMs incorporates a hierarchical view to the database through the TS-SOM levels. A separate TS-SOM is created for each image feature type used.

The image retrieval in the PicSOM system is based on approximate queries by image group examples. The query process of PicSOM is iterative. In order to refine the retrieval results relevance feedback is performed on the image sets the queries return. The user identifies each example image as relevant or non-relevant to the current retrieval task and the system uses the information to select new database images the user is most likely to be interested in. Furthermore, the weighting of different feature TS-SOMs is adjusted according to the relevance of the retrieval results they produce.

Technically, the PicSOM system implements the vector space model of CBIR. Images are first mapped to vectors (i.e. points) in the feature space. There is a separate feature space for each image feature. Each of the spaces is indexed through a TS-SOM. The system responds to queries by identifying database images whose mappings on the TS-SOMs are close to the mappings of the relevant example images.

3 THE IMAGE DATABASE

As an example of a data browsing application we use the PicSOM system to browse a collection of 421 images which was captured from the WWW-based mail order catalogues of two garment stores [4] [14]. The images were automatically retrieved from the web pages in February 2000 with an image retrieval web robot [10].

This sort of an application could be favourable for CBIR as the goodness or suitability of the garments for given needs is largely judged based on their visual appearance. It is reasonable to expect that the low-level visual features that are commonly used in CBIR may offer a feasible criterion for organising the collection of images.

The images display a variety of garments: trousers, jeans, skirts, shoes and others. Most of the images serve the purpose of displaying an item that is on sale. However, the collection also includes images that are used for other purposes, such as decorating and structuring the catalogues. Most of the catalogue images contain a single garment. Some others contain several, mostly two, pieces of clothing which often partly occlude one another. One special group of such images are images of human models wearing an outfit consisting of several garments. In most images the background is more or less constant. However, some images have a patterned background. Some of the images embody text. The database images measure couple of hundred pixels (typically around 200 pixels) in each spatial dimension. Most of the images are colour images whose colours are represented in the RGB space with eight-bit precision. In practice almost all the images use less than ten colours.

4 IMAGE ANALYSIS

The image analysis module of a vector space CBIR system is the part of the system that could be very specific to the application at hand. The garment images form a narrow domain, facilitating the use of specialised image analysis techniques. In this case, however, we were more interested of testing the general idea, not tuning the performance of the system to the maximum. Therefore only simple image analysis techniques were used which would be useful also for general domain images. The narrowness of image domain was exploited in that the images could be first reliably segmented using straightforward methods (Section 4.1). After segmentation three feature vectors were calculated for each image (Section 4.2). It was also tested how the system could be used for browsing automatically formed outfits from separate garment images. Section 4.3 describes how the combinations were formed.

4.1 Segmentation

The task of the image browsing system is to show the user images depicting products, here particularly garments. Any other content of the images can be regarded as unessential from the point of view of browsing. Therefore the system should be able to determine which properties of the images do not reflect the properties of the garments in order to filter them away from distracting the browsing process. Fortunately, the images can be considered to form a narrow domain and consequently the needed automatic image processing is simplified. Furthermore, many of the images in the considered database have the garment as the single central object in the foreground and a plain background. Then it is easy to separate the objects from their backgrounds.

There are two main reasons for segmenting the images to distinct object/background regions. Some feature types, in particular the colour features, are affected by the large background region. In this application we consider the background to be irrelevant to the judgement of image content. By eliminating the image backgrounds we avoid their properties steering the browsing of images. Image segmentation also facilitates the use of object shape features.

Besides feature calculation the segmentation results are needed for visualisation purposes. In particular, the system can be instructed to combine two garments together to form an outfit (e.g. a blouse and a skirt). In this case the segmentation of the individual images is essential in order to align and scale the garments properly. The visualisation sets the strictest demands for the segmentation. The human observer easily spots even minor mistakes in the segmentation as she has a clear *a priori* insight how the garments may look like. The feature calculation is more robust toward inaccuracies in segmentation. Here the object shape features require most from the segmentation, the other features generally tolerate deficient or even nonexistent segmentation in a greater degree.

The used segmentation method is a simple variation of the k-means segmentation [16] in the RGB colour space. The standard k-means iteration is supplemented by an additional relaxation step which makes use of a simple Markov random field (MRF) model. The first-order MRF model penalises nearest neighbour pixels that belong to different clusters. In effect, spatially contiguous regions of pixels belonging to same clusters are encouraged. After classifying the image pixels into k classes, one of the clusters is marked as background and all the others as object. The selection process relies on the fact that in the considered images the objects are generally in the central region and, on the other hand, the background region is usually quite large. After determining the background and object regions the segmentation image is post-processed by filtering out all but the largest contiguous object region. Finally, all holes within the boundary of the object are filled.

The typical catalogue images that display one garment on a homogeneous background can be segmented reliably in most cases by applying the outlined procedure. Some small image details may be missed, though. Some images have a thin rectangular frame around the objects. It is taken care of by a custom filter that is sensitive to thin horisontal and vertical lines. A more serious problem arises from the shadows that the objects sometimes cast on their backgrounds. Currently the shadows are often segmented as part of the object. Fortunately the resulting erroneous region is usually only a few pixels wide and reflects the actual object shape quite well. Consequently such errors are considered tolerable. There are a couple of images in the database that conform to the central object/plain backround model, but still the segmentation results are considerably bad. The problems occur as a region of exactly the same colour penetrates from the backround into the object. In order to reach a good segmentation in these cases a model of allowable shapes could be used to constrain the object outline. All in all, these cases are rare and in general the segmentation results for the anticipated type of images can be considered more than adequate.

Other sorts of images can present problems to the system. In the catalogues there are images that are used for other purposes than presenting products, e.g. as title images. This sort of images can safely be ignored. More problematic are images which contain several objects or have an uneven background, for example the products might be embedded in a natural scene. In almost all such images mannequins pose wearing an outfit typically consisting of several pieces of clothing. As such images were not too predominant in our test database, it was not considered worth the trouble devising a scheme for the extraction of the garments from such images. That should be doable in a pretty straightforward manner for the stereotypical human model images that appear in the garment catalogues. However, it would present quite a laborious image processing problem by itself.

The implementation of the segmentation method is relatively slow. Even when the images are as small as 200×200 pixels, the segmentation takes approximately four seconds per image when run on a Silicon Graphics POWER Challenge 10000 server. Currently this poses no problem, but the issue has to be reconsidered if larger databases and/or more sizable images are to be segmented.

4.2 Features

In feature extraction only the foreground object revealed by the segmentation phase was taken into account. Three feature vectors were calculated for each image. Two of them describe the colour distribution and one characterises the shape of the foreground object. A separate TS-SOM was trained for each of the three features.

4.2.1 Colour

The colour features were calculated only for the foreground object in the images. Two different colour features were used. The first was the average colour in the RGB space. Thus the feature vectors have three components. The other colour feature was formed from the first three central moments of the colour distribution in the HSV space. The colour moment feature vector comprises of the average values C_i, standard deviations σ_i and the third roots of the skewness s_i, calculated separately for each colour channel i. In this case i assumes the values in the set $\{H, S, V\}$ and therefore a nine-component feature vector results. The components are calculated as follows:

$$C_i = \frac{1}{WH} \sum_{x=1}^{W} \sum_{y=1}^{H} c_i(x, y), \tag{1}$$

$$\sigma_i = \sqrt{\frac{1}{WH} \sum_{x=1}^{W} \sum_{y=1}^{H} (c_i(x, y) - C_i)^2}, \text{ and} \tag{2}$$

$$s_i = \sqrt[3]{\frac{1}{WH} \sum_{x=1}^{W} \sum_{y=1}^{H} (c_i(x, y) - C_i)^3}. \tag{3}$$

Here $c_i(x, y)$ is the value of colour channel i in image pixel (x, y) and W and H are the width and height of the image, respectively. The low-order colour moments have been found [18] to be an efficient and robust way to characterise colour distributions.

4.2.2 Shape

The browsing system uses normalised Fourier descriptors [1, 9] of object boundaries as shape features. In the images for which the system is intended there is no occlusion which would diminish the usability of Fourier descriptors. Unnormalised Fourier descriptors are derived from the following expansion of the object contour:

$$z(s) = \sum_{n=-\infty}^{\infty} z_n e^{\frac{2\pi j n s}{L}} \tag{4}$$

Here the complex function $z(s)$ is a parametrised description of the contour of an object whose coordinates are taken to form the real and imaginary parts of complex numbers. The real and imaginary components of the summation coefficients z_n form the set of Fourier descriptors.

Figure 1: The surface of the 16×16-sized SOM formed with the Fourier descriptor feature.

The number of coefficients and thus the length of the feature vectors is limited by storing only the low-order coefficients (i.e. z_n with small absolute values of n).

The Fourier descriptors are affected both by the exact way the coefficients are calculated and by geometrical transformations to the image. We choose to normalise the Fourier descriptors against image translation, scaling and rotation. Different ways of parametrising a given curve are also normalised. The normalising procedure used here anchors the image transformation parameters by imposing the following constraints on the normalised Fourier coefficients $\tilde{z}$:

$$\tilde{z}_0 = 0 \tag{5}$$
$$\tilde{z}_{\pm 1} \in \mathbb{R} \tag{6}$$
$$\tilde{z}_1 + \tilde{z}_{-1} = 1 \tag{7}$$

The normalisations still allow two different sets of Fourier descriptors for the same object, depending on whether the parametrisation traverses the contour of the object clockwise or counterclockwise. These possibilities could be incorporated into the equations. Here another approach has been chosen instead. The feature extraction algorithms have been designed so that they always traverse object contours counterclockwise.

Fourier descriptors up to order 10 have been used, resulting in a feature vector with the dimensionality of 42. Five of the components are redundant because of the normalisations. The higher-order components are quadratically emphasised:

$$z'_n = n^2 \tilde{z}_n \tag{8}$$

This takes into account that higher-order descriptors translate into quadratically more energy, i.e. variance. Figure 1 shows the bottom level of a TS-SOM trained with the Fourier feature. For example in the upper left corner we see a cluster of shirts and in the upper right corner a cluster of pants.

4.3 Creation of virtual outfits

It was also examined how the system could be applied to browsing of garment combinations. The idea is that the user would be able to explore different costumes made from the garments in the catalogue. To this end two classes of images were picked from the database: garments for the upper part of the body (shirts, blouses, jackets) and those for the lower part (jeans, trousers and skirts). The classes had 78 and 177 images, respectively. The Cartesian product of the classes was then formed (i.e. each image in the first class was paired with all the images in the second class) resulting in 13 706 composite images. Corresponding composite feature spaces were created by concatenating the feature vectors pairwise for all the combinations of upper/lower part garments. The number of feature spaces thus remains the same as with the original images. Indexing TS-SOMs were trained for these feature spaces and the PicSOM system was used almost in the same manner as with sigle images. The only exception was that the combined images were not stored in a database but generated on demand using the upper/lower part images, segmentation results and alignment information which was extracted automatically when the database was created.

The method of combining upper and lower part images was quite simple. The approach was to extract horisontal alignment lines in both the bottom and top parts of the images. This was done by detecting of edges in the horisontal and vertical projections of segmented images with a box filter. Two images were then combined by translating and scaling them so that the top alignment line of the lower image overlapped the bottom alignment line of the upper image.

Based on visual observations the bottom alignment lines were shortened with a constant factor of 0.87, which improved the scale matching somewhat. Also the alignment is only approximate. In a real-world situation the resolution and scale of the image could be known beforehand, which would help in eliminating these problems. Figure 2 shows some examples of combined garment images. The results of combining images are roughly acceptable in most cases.

Figure 2: Automatically formed images of garment combinations.

5 BROWSING THE IMAGES

Figures 3 through 5 show a sample garment browsing session. The web query interface can be seen in Figure 3. The system shows a set of images from the image database and asks the user to mark the relevant ones. When this has been done, the user can press the "Continue query" button in the browser window. Then the system proceeds to the next query round and picks a new set of images to be shown. The relevant images are accumulated in a set and shown alongside the new candidate images on every query round. If the user desires, she can still remove any images from the set of relevant images by unclicking checkmarks beside them. This can be used to first steer the search to roughly the right direction and later refine the requirements and exclude some images that are not exactly what is desired. The rectangles in the upper part of the window represent the surfaces of the TS-SOMs of the respective features.

Figures 4a-g display the candidate image sets the system shows the user during a sample query. This time we assume the user wants to find "blue jeans with some coloured decorations in them." In each image set the user marks all the relevant images with a check mark. These evaluations together with all the evaluations accumulated on earlier query rounds constitute the relevance feedback to the next query round. There are not many relevant images in the small testing database and on the final round (Figure 4g) the system does not find any more relevant images as they all have already been shown earlier. This is confirmed also by manually examining all the database images. Figure 5 shows the final set of accumulated result images.

6 OBSERVATIONS AND CONCLUSIONS

It seems that the adaptive components of the PicSOM system are too slow for browsing the small database of 421 images. Convergence does not fully take place before the database is exhausted from the desired images. Furthermore, the PicSOM system returns some randomly chosen images along with the best-matching ones. This kind of exploration of the image space prevents the search to get stuck in a small local neighbourhood in case of large image databases. However, as the amount of random exploration is decided regardless of the database size, searches in very small image databases begin to resemble random searches. The browsing of the database of garment combinations is better suited for the standard machinery of the PicSOM system as the database is large enough with its 13 706 images. The visual quality of the automatically-generated image combinations, however, is not fully satisfactory. This is no surprise since the images were aligned using very simple methods. As the visual appearance is among the most important factors when evaluating garments, some more effort needs to be invested in the image combining method if a decent garment combination browsing application is desired.

The experiences with the system suggest that the standard relevance feedback user interface of image searches might not be the best one for browsing image collections. The standard relevance feedback cycle is too mechanical and rather restricting, therefore the user should be provided with better means to explore the image database more freely. Also the way of presenting query results could be more informative. Currently the returned images are organised in a rectangular grid according to their similarity scores with the query images. The information in the image set could be more accessible to the user if the images were organised spatially according to some of their visual properties.

The PicSOM system includes the possibility to directly browse the surfaces of the TS-SOM levels. The user interface is somewhat clumsy but a tailored user interface employing the same principle could be a good solution. A problem in directly browsing the TS-SOMs is that they are static. The browsing does not take into account any relevance information. Another major

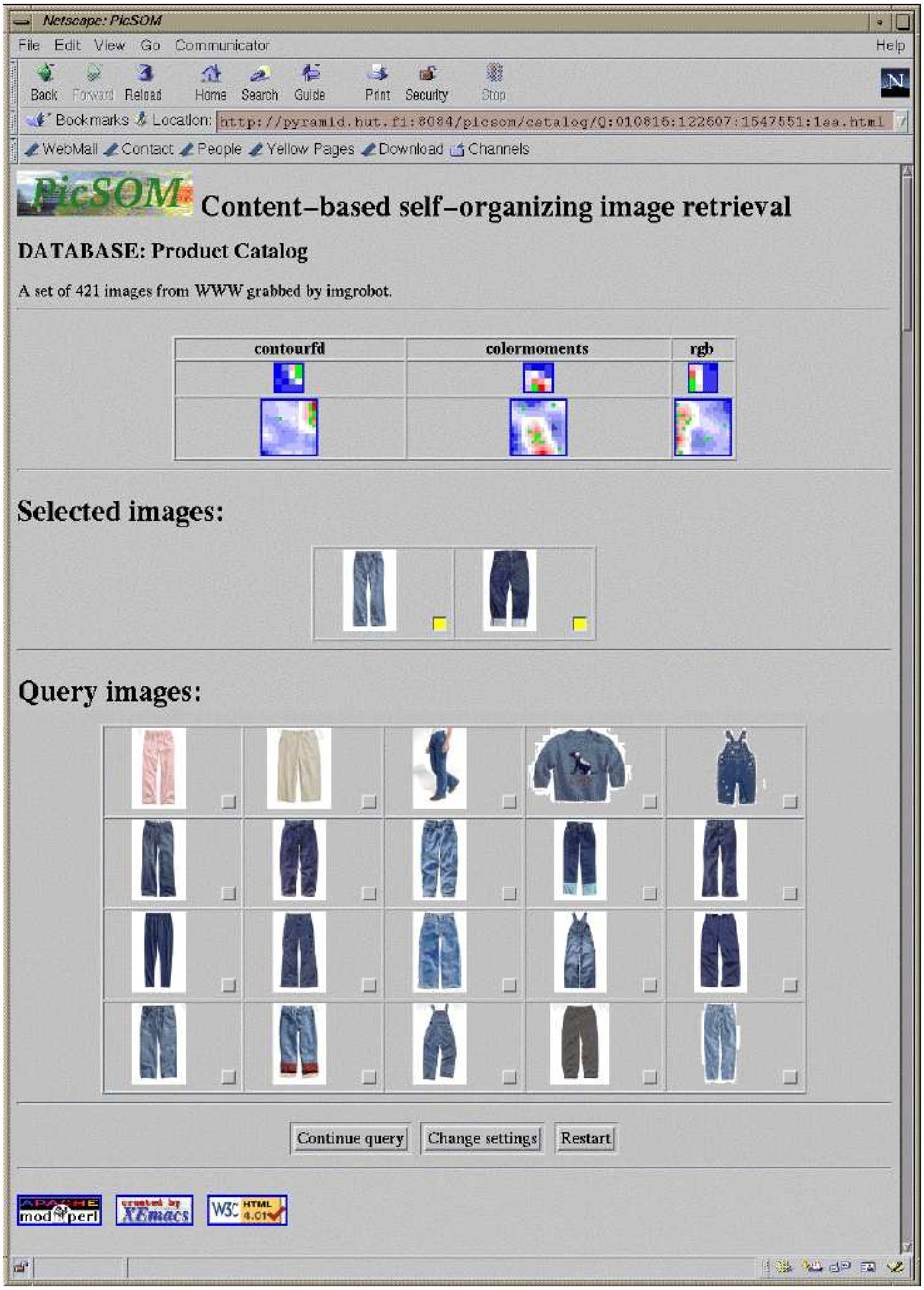
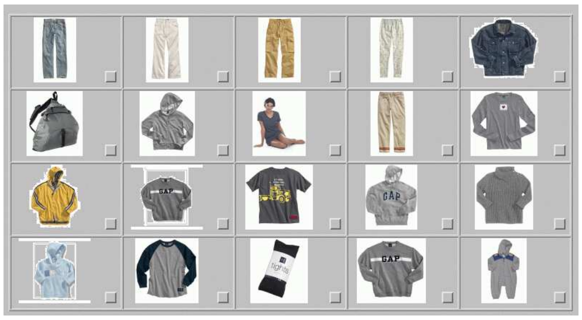


Figure 3: Example search for decorated blue jeans, based on relevance feedback from the user. The figure portrays the user interface on the second round of the query.

(a) (b)

(c) (d)

(e) (f)



(g)

Figure 4: Example search for decorated blue jeans. Results from seven sequential query rounds are shown in subfigures a) through g). Images selected relevant on each round are marked with checkmarks. In the initial state of subfigure a) the system had no *a priori* knowledge about how to select the images. The relevance feedback begins to function only from subfigure b) on. The clarity of the images has been improved by emphasising the selected check boxes with manually drawn check marks.

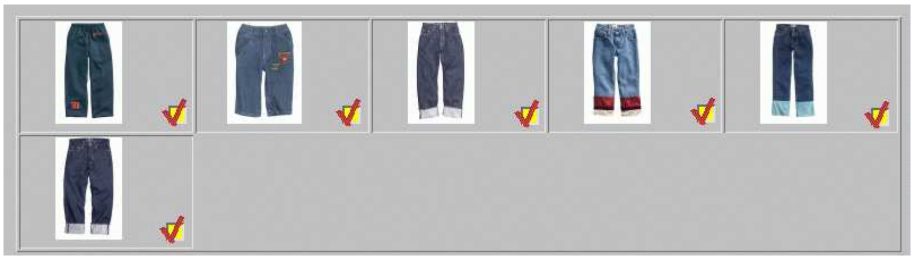


Figure 5: The final set of images found in the example search.

shortcoming is that the browsing considers only one feature TS-SOM at time. A custom browsing interface should overcome both of these problems. One approach would be to maintain an additional combined topographic mapping which would be constructed according the feature weighting in use at the moment. The user would then be provided with means to navigate in the target space of this mapping. The space could be three-dimensional even if it would have to be projected back to two dimensions for display. The human visual system is namely accustomed to dealing with 2D projections of 3D world. Research results show [17] that notably larger amount of information can be retrieved from a mapping of 3D space compared with the direct use of 2D space. To make the data display still more effective focus-and-context teqhniques [2] could be applied. Probably it would be computationally too expensive to dynamically maintain a mapping of the whole image database. Instead, it might be possible to use a local mapping of the image space near the example images.

As was mentioned earlier, the choice of features that were used was not optimal in terms of performance. As the application domain is narrow, it would be advantageous to use domain-specific features, such as eigenfeatures e.g. [15].

REFERENCES

[1] K. Arbter. Affine-invariant fourier descriptors. In J. C. Simon, editor, *From Pixels to Features*, pages 153–164. Elsevier Science Publishers B.V.(North-Holland), 1989.

[2] S. Card, J. Mackinlay, and B. Schneiderman, editors. *Readings in information visualization*. Morgan Kaufmann, 1999.

[3] J. P. Eakins. Automatic image retrieval — are we getting anywhere? In *Third International Conference on Electronic Libraries and Visual Information Research (ELVIRA3), April 30 - May 2*, pages 123–135, Milton Keynes, UK, 1996. De Montfort University.

[4] Gap Online. `http://www.gap.com/` [2001-12-07], February 2000.

[5] Börn Johansson. A survey on: Contents based search in image databases. Electronic report, Linköping University,Computer Vision Laboratory, December 2000.

[6] Teuvo Kohonen. *Self-Organizing Maps*, volume 30 of *Springer Series in Information Sciences*. Springer-Verlag, third edition, 2001.

[7] Pasi Koikkalainen. Progress with the tree-structured self-organizing map. In A. G. Cohn, editor, *11th European Conference on Artificial Intelligence*. European Committee for Artificial Intelligence (ECCAI), John Wiley & Sons, Ltd., August 1994.

[8] Pasi Koikkalainen and Erkki Oja. Self-organizing hierarchical feature maps. In *Proceedings of 1990 International Joint Conference on Neural Networks, June 17-21*, volume II, pages 279–284, San Diego, USA, 1990. IEEE, INNS.

[9] O. Kübler and G. Gerig. Bildverarbeitung und Computer Vision II. Lecture notes, ETH Zürich, Institut für Kommunikationstechnik, 1996.

[10] Sami Laakso. Implementation of content-based WWW image search engine. Master's thesis, Laboratory of Computer and Information Science, Helsinki University of Technology, 2000.

[11] J. T. Laaksonen, J. M. Koskela, S. P. Laakso, and E. Oja. PicSOM - Content-based image retrieval with self-organizing maps. *Pattern Recognition Letters*, 21(13-14):1199–1207, December 2000.

[12] Jorma Laaksonen, Markus Koskela, Sami Laakso, and Erkki Oja. Self-organizing maps as a relevance feedback technique in content-based image retrieval. *Pattern Analysis & Applications*, 4(2+3):140–152, June 2001.

[13] Jorma Laaksonen, Markus Koskela, and Erkki Oja. PicSOM–self-organizing image retrieval with MPEG-7 content descriptors. *IEEE Transactions on Neural Networks*, 13(4):841–853, July 2002.

[14] macys.com. `http://www.macys.com/` [2001-12-07], February 2000.

[15] Alex Pentland, Rosalind W. Picard, and Stan Sclaroff. Photobook: Content-based manipulation of image databases. Technical Report #255, M.I.T Media Laboratory, 1995.

[16] Robert J. Schalkoff. *Pattern Recognition: Statistical, Structural and Neural Approaches*. John Wiley & Sons, Ltd., 1992.

[17] M. Sheelagh, T. Carpendale, D. J. Cowperthwaite, and F. D. Fracchia. Extending distortion viewing from 2D to 3D. In S. Card, J.Mackinlay, and B. Schneiderman, editors, *Readings in information visualization*. Morgan Kaufmann, 1999.

[18] Markus Stricker and Markus Orengo. Similarity of color images. In *Storage and Retrieval for Image and Video Databases III (SPIE)*, volume 2420 of *Proceedings of SPIE*, pages 381–392, February 1995.