



ELSEVIER

Available online at www.sciencedirect.com

SCIENCE @ DIRECT®

Neural Networks xx (xxxx) 1–13

Neural
Networkswww.elsevier.com/locate/neunet

Class distributions on SOM surfaces for feature extraction and object retrieval

Jorma T. Laaksonen*, J. Markus Koskela, Erkki Oja

Laboratory of Computer and Information Science, Neural Networks and Research Centre, Helsinki University of Technology,
P.O. Box 5400, FI-02015 HUT, Finland

Received 22 December 2003; accepted 12 July 2004

Abstract

A Self-Organizing Map (SOM) is typically trained in unsupervised mode, using a large batch of training data. If the data contain semantically related object groupings or classes, subsets of vectors belonging to such user-defined classes can be mapped on the SOM by finding the best matching unit for each vector in the set. The distribution of the data vectors over the map forms a two-dimensional discrete probability density. Even from the same data, qualitatively different distributions can be obtained by using different feature extraction techniques.

We used such feature distributions for comparing different classes and different feature representations of the data in the context of our content-based image retrieval system PicSOM. The information-theoretic measures of entropy and mutual information are suggested to evaluate the compactness of a distribution and the independence of two distributions. Also, the effect of low-pass filtering the SOM surfaces prior to the calculation of the entropy is studied.

© 2004 Published by Elsevier Ltd.

Keywords: Self-organizing map; Feature extraction; Probability distribution; Entropy; Mutual information; Content-based image retrieval

1. Introduction

The Self-Organizing Map (SOM) (Kohonen, 2001) is a powerful tool for exploring huge amounts of high-dimensional data. Many studies have been made on the clustering, visualization, and data mining capabilities of the SOM (Kohonen, Oja, Simula, Visa, & Kanfas, 1996; Kraaijveld, Mao, & Jain, 1995; Lampinen & Oja, 1992; Ultsch & Siemon, 1990; Vesanto & Alhoniemi, 2000). In a typical data mining, visualization, or information retrieval application, a Self-Organizing Map is trained in a fully unsupervised mode, using a large batch of training data. Yet, it is often known that the data contain some semantically related object groupings or classes, and there are available subsets of vectors belonging to such user-defined classes. Such a set of vectors can be mapped on

a trained Self-Organizing Map (SOM) by finding the best matching unit for each vector in the set. These ‘hits’ over the map units form a discrete probability distribution over the two-dimensional SOM surface which characterizes the object class: even from the same data, qualitatively different distributions can be obtained by using different feature extraction techniques, leading to different numerical representations of the data items.

In this paper we do not discuss the SOM training at all, but assume that a properly trained SOM exists. Instead, we address here the following two problems of great practical importance:

- Assume that a user-defined set of feature vectors, similar to the ones used in training the map exists, and it is known that this set represents semantically related items. Typically, the set might be a part of the training set, or a new batch of samples that were not used for training. How do we map these items on the SOM so that a Bayes optimal classification can be made for any new object?

* Corresponding author. Tel.: +358 9 451 3269; fax: +358 9 451 3277.

E-mail addresses: jorma.laaksonen@hut.fi (J.T. Laaksonen), markus.koskela@hut.fi (J. Markus Koskela), erkki.oja@hut.fi (E. Oja).

169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224

List of symbols

\mathbf{x}	feature vector	H_{\max}	theoretical maximum of entropy
$p(\mathbf{x})$	probability density function	\bar{H}	normalized entropy
N	total number of feature vectors	$G(\cdot)$	kernel function
\mathcal{V}_i	Voronoi region of i th SOM unit	\mathbf{w}_i	weight vector of i th SOM unit
P_i	probability of i th SOM unit being the BMU	h	convolution mask
$\#\{\cdot\}$	cardinality of a set	l	size of the convolution mask
\mathcal{C}	image class	$I(P, Q)$	mutual information of distributions P and Q
$N(\mathcal{C})$	cardinality of class \mathcal{C}	R	estimated joint probability of two distributions
$\rho(\mathcal{C})$	a priori probability of class \mathcal{C}	\bar{I}	normalized mutual information
$P_i(\mathcal{C})$	probability of i th SOM unit being the BMU for class \mathcal{C}	L	number of parallel feature spaces
K	numbers of symbols in an information source, number of SOM units	q_j	qualification value or score for image j
$H(P)$	entropy of distribution P	\mathcal{D}_t	set of images retrieved on t th round
		\mathcal{R}_t	set of images marked relevant on t th round
		\mathcal{H}_t	query history up to the t th round

- In the above situation, assume we have several different feature extraction schemes, leading to several alternative representations of the data. Then several separate SOMs will result for the training data, each based on one of the representations. When the user-defined class is mapped on each of these SOMs, which one of them will give the best discrimination for the class?

To answer these questions, we study how object class histograms on SOMs can be given interpretations in terms of probability densities and information-theoretic measures such as entropy and mutual information (Cover & Thomas, 1991). The latter measures arise when the distributions of the same objects after two different feature extraction stages are compared. Shortly, the entropy of a certain feature vector's distribution is a measure of how uniformly the used feature distributes the class over the map. A good feature is obviously such that the class is heavily concentrated on only a few nearby map elements, giving a low value of entropy. Then the posterior probabilities of the classes are clearly distinct over most of the map surface. The mutual information of two features' distributions is a measure on how independent those features are. Obviously independent features are better in describing objects than heavily correlated features. These concepts are defined in the following, and fairly extensive experimental evaluations are given.

To use the class histograms for density presentation, we use low-pass filtering of the SOM surfaces. This is related to a mixture density or reduced kernel representation of the density in the original feature space. This technique further facilitates analyzing the compactness and internal structure of an object class after mapping on the SOM surface, and makes possible to use Bayes optimal decision rules in classifying new data items.

The discussions in the paper are meant to be general in their nature, suitable for any SOM application in which

the data have inherent semantic classes. However, the results will be illustrated throughout the paper by a specific case study, which is *content-based image retrieval* (CBIR) (Castelli & Bergman, 2002; Del Bimbo, 1999; Rui, Huang, & Chang, 1999; Smeulders, Worring, Santini, Gupta, & Jain, 2000), namely our PicSOM CBIR system. CBIR is a very good case study for illustrations and examples of the techniques outlined above, due to the statistically significant sample sizes, a variety of possible feature extractions, and the availability of user-defined classes.

We do not review PicSOM in this paper, because the emphasis is not on image retrieval as such; for details, see Laaksonen, Koskela, Laakso, and Oja (2001), and Laaksonen, Koskela, and Oja (2002). PicSOM uses SOMs in implementing *relevance feedback* (Zhou & Huang, 2003) and *query by example* (QBE) paradigms (Lew, 2001) in interactive and iterative information retrieval from unannotated databases. In all the illustrations and examples, we use MPEG-7 (Manjunath, Salembier, & Sikora 2002; MPEG, 2003) and keyword features extracted from a Corel Gallery database of 59 995 images with miscellaneous content. The MPEG-7 features are low level, describing the coarse color, texture, and shape of an image, while the keyword feature is computed from related textual data (Koskela & Laaksonen, 2003). The dimensionalities of MPEG-7 feature vectors are from several tens to hundreds and the dimensionality of the statistical keyword feature has been 150 in our experiments. In PicSOM, a separate SOM for each feature type is trained by using the Tree Structured SOM (TS-SOM) algorithm (Koikkalainen, 1994; Koikkalainen & Oja, 1990) which makes training large SOMs much faster than by using a conventional SOM. The sizes of the TS-SOM layers are 4×4 , 16×16 , 64×64 , and 256×256 map units. Each layer is essentially a normal SOM trained with the same data, and thus it is possible to make quantitative comparisons of our results with different SOM sizes.

Sections 2–4 review the class histograms on the SOM surface and the normalized entropy measure. SOM surface convolutions are introduced in Section 5 and related to kernel-based density estimation. Mutual information is used in Section 6 to measure the independence of separate feature representations. Section 7 discusses optimal Bayes classification based on the convoluted distributions on SOM surfaces, and addresses the question how to choose the most representative sample from a class. Finally, conclusions are given in Section 8.

2. Class distributions

Assume that we have trained a SOM in an unsupervised fashion, using a large set of high-dimensional vectors. Let us choose a subset of vectors, which may be included in the original training set or may be a new sample of similar data. The subset contains objects that are semantically related, as defined by a human user. Such a subset is standardly *mapped* on the trained SOM by finding the best matching unit for each vector and counting the number of hits for each map unit. Normalized to unit sum, the hit frequencies give a discrete histogram which is a sample estimate of a probability distribution of the class on the SOM surface.

The shape of the distribution on the SOM surface depends on several factors:

- The distribution of the *original data* in the very-high-dimensional pattern space is generally given and cannot be controlled.
- The *feature extraction* technique in use affects the metrics and thus the distribution of all the generated feature vectors.
- The *overall shape* of the training set, after it has been mapped from the original data space to the feature vector space, determines the, overall organization of the SOM.
- The *class distribution* of the studied object subset or class, relative to the overall shape of the feature vector distribution, specifies the layout of the class on the formed SOM.

Fig. 1 visualizes how the pattern space is projected to feature space, the vectors of which are then used in training

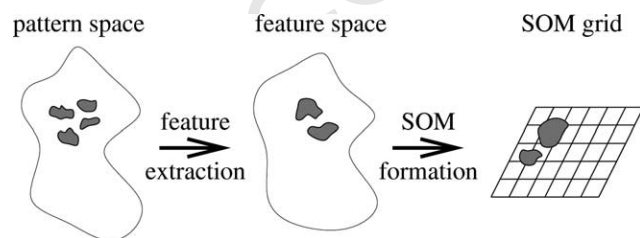


Fig. 1. Stages in dimensionality reduction from the very-high-dimensional pattern space through the high-dimensional feature space to the two-dimensional SOM grid.

the SOM. The areas occupied by objects of a particular class are shown with gray shades.

In the very-high-dimensional pattern space the distribution of any nontrivial object class is most certainly sparse. As a consequence, in most cases it is meaningless to talk about the uni- or multimodality of class distributions in the pattern space. On the other hand, if the feature extraction stage is working properly, semantically similar patterns will be mapped nearer to each other in the feature space than semantically dissimilar ones. In the most advantageous situation, the pattern classes match clusters in the feature space, i.e. there exists a one-to-one correspondence between feature vector clusters and pattern classes.

In feature extraction, some pattern space directions are retained better than others. This is known as a particular type of *feature invariance*. Depending on the application, different types of invariances are needed. The relative distances between the feature vectors of a class compared to the overall distribution of the feature space data determine how well the class is concentrated in nearby SOM units. If the class is truly multimodal with wide relative variance, one cannot in general avoid its splitting in noncontiguous map regions.

In the open literature, there exists a number of different measures for assessing the *denseness* or *locality* of feature vectors on a SOM. Some of the proposed techniques are merely qualitative in their nature and mostly applicable to visualization purposes. Such methods include, e.g. the smoothed data histogram (SHD) (Pampalk, Rauber, & Merkl, 2002). In SHD, each data point is mapped not only to its nearest SOM unit but to *s* nearest units with reciprocally decreasing fractions. With a properly selected value for the smoothing parameter *s*, the SHD method is able to visually identify the cluster structure of the data. Also the *U*-matrix technique (Ultsch & Siemon, 1990; Kraaijveld et al., 1995) serves well for visualizing the data distribution and its clusters.

Proposed quantitative locality measures include, e.g. *map usage*, *average pair distance*, *fragmentation*, and *purity* used in Pullwitt (2002). The map usage and purity as such are limited in their usability as their values are invariant to random permutations of the SOM units. As a consequence, they fail to take into account the topological structure of the class. On the other hand, the fragmentation measure, which counts the number of isolated areas on the map, does not pay attention to the size of the fragments, i.e. the numbers of SOM units and mapped data points in each fragment. The average pair distance is thus the only potential locality measure of the ones mentioned here.

3. BMU probabilities

In theory, one can calculate the a priori probability of each SOM unit for being the *best-matching unit* (BMU) for any

vector \mathbf{x} of the feature space. This is possible if the *probability density function* (pdf) $p(\mathbf{x})$ is known. Denote the SOM unit by i and its surrounding *Voronoi region* by \mathcal{V}_i . This is the set of vectors in the original feature space that are closer to the weight vector of unit i than to any other weight vector. One may now calculate the unit's a priori probability P_i as

$$P_i = P(\mathbf{x} \in \mathcal{V}_i) = \int_{\mathcal{V}_i} p(\mathbf{x}) d\mathbf{x}. \quad (1)$$

In mapping a data set $\mathbf{x}_j, j=0,1,\dots,N-1$ on the SOM surface, we are actually replacing the continuous pdf with a discrete probability *histogram*, by counting the number of times that any given map unit is the BMU for the vectors in the data set. Without danger of confusion, the probability can still be denoted as P_i

$$P_i = P(\mathbf{x} \in \mathcal{V}_i) = \frac{\#\{j|\mathbf{x}_j \in \mathcal{V}_i\}}{N}, \quad (2)$$

where $\{\cdot\}$ stands for the cardinality of a set. One needs to note that the original probability density of the continuous feature space cannot be directly approximated with the discrete P_i , because the sizes of the histogram bins, i.e. the Voronoi regions, are not equal. It will suffice, however, that the one-directional mapping from the continuous distribution to the discrete one can be performed.

In what follows, we will concentrate on the distributions of specific *subsets* of data. We may assume that the members of such a subset fulfill a specific *ground truth* criterion by which each object can be classified as either a member or nonmember of the class. The probability histogram of class \mathcal{C} on the SOM surface will thus be

$$P_i(\mathcal{C}) = P(\mathbf{x} \in \mathcal{V}_i | \mathbf{x} \in \mathcal{C}) = \frac{P(\mathbf{x} \in \mathcal{V}_i, \mathbf{x} \in \mathcal{C})}{P(\mathbf{x} \in \mathcal{C})}, \quad (3)$$

which will be estimated in the SOM mapping as

$$P_i(\mathcal{C}) = \frac{\#\{j|\mathbf{x}_j \in \mathcal{V}_i, \mathbf{x}_j \in \mathcal{C}\}}{N(\mathcal{C})} \quad (4)$$

with $N(\mathcal{C})$ the cardinality of the subset of vectors in class \mathcal{C} .

4. BMU entropy

We will now turn to study the uniformity of the distributions of feature vectors' BMUs on the SOM surface. A simple and commonly used measure for the randomness of a symbol distribution is its *entropy* (Cover & Thomas, 1991). In our case, the BMU indices for the vectors of the training set play the role of symbols. The entropy H of a distribution $P=(P_0, P_1, \dots, P_{K-1})$ is calculated as

$$H(P) = H = - \sum_{i=0}^{K-1} P_i \log P_i, \quad (5)$$

where K is the number symbols in the alphabet of the stochastic information source, in our case thus the number

of map units. P_i is the probability of map unit i being the BMU of an input vector, as defined before.

If one assumes that every map unit is equally probable as an input vector's BMU, i.e. the distribution is uniform, then one can easily calculate a theoretical maximum for the entropy of the BMU distribution

$$H_{\max} = \max_{\{P_i\}} \left\{ - \sum_{i=0}^{K-1} P_i \log P_i \right\} = -K \cdot \frac{1}{K} \log \frac{1}{K} = \log K. \quad (6)$$

For example, for a map of size $K=16 \times 16=256$ units, $H_{\max}=8$ when logarithm base two is assumed.

The entropy of BMU histograms can be made to some extent independent of the size of the SOM by dividing the entropy H by its theoretical maximum. The *normalized entropy* \bar{H} of the distribution is thus

$$\bar{H} = \frac{H}{H_{\max}}. \quad (7)$$

One should, however, note that the above value for H_{\max} in (6) is really an upper limit also in the sense that $P_i \approx 1/K$ only when $N \gg K$ and can hold exactly when N is divisible by K . Consequently, \bar{H} will be biased toward smaller values especially in cases when the distribution of a small subset of objects is studied.

In general it can be assumed that the usual unsupervised training of a SOM distributes the training vectors fairly evenly over the map surface (Kohonen, 2001). Therefore, the normalized entropy \bar{H} of the whole training set should be near unity. On the other hand, all object subsets of semantic similarity should be concentrated in few SOM units if the feature extraction and SOM training phases have been favorable to that specific subset. If that really is the case, the normalized entropy will be clearly smaller than one. The normalized entropy $\bar{H}(\mathcal{C})$ of class \mathcal{C} can simply be calculated by replacing P_i s in (5) with $P_i(\mathcal{C})$ s of (3).

Table 1 illustrates this using image data having some user-defined classes. The images can be represented as numerical vectors using a number of different features. The data used for training the SOMs were the 59 995 Corel database images. The normalized entropies of all images in the database, as well as in six semantic image classes are shown. The used image classes were faces (1115 images, a priori probability 1.85%), cars (864 images, 1.44%), planes (292 images, 0.49%), sunsets (663 images, 1.11%), horses (486 images, 0.81%), and traffic signs (123 images, 0.21%). All classes were manually gathered by a single subject; for more details, see Laaksonen et al. (2002). Four feature extraction methods defined in the MPEG-7 standard were experimented with, viz. the *Color Structure* (dimensionality 256), *Scalable Color* (256), *Edge Histogram* (80), and *Homogenous Texture* (62) descriptors (Manjunath et al., 2002). In addition, a *keyword* (150) feature obtained from the original keyword annotations provided by Corel for the images was used. The keyword feature was composed of

Table 1

Normalized entropies of different SOM sizes with the *Color Structure* (CS), *Scalable Color* (SC), *Edge Histogram* (EH), *Homogenous Texture* (HT), and *keyword* (KW) descriptors, respectively

SOM size	All	Faces	Cars	Planes	Sunsets	Horses	Tr. signs
<i>CS</i>							
4×4	0.990	0.882	0.961	0.709	0.611	0.880	0.960
16×16	0.995	0.877	0.928	0.762	0.710	0.859	0.762
64×64	0.994	0.776	0.783	0.650	0.665	0.701	0.564
256×256	0.947	0.627	0.607	0.510	0.572	0.552	0.430
<i>SC</i>							
4×4	0.979	0.877	0.966	0.863	0.749	0.898	0.879
16×16	0.995	0.885	0.927	0.799	0.775	0.799	0.788
64×64	0.995	0.788	0.777	0.653	0.697	0.680	0.570
256×256	0.943	0.629	0.606	0.507	0.575	0.544	0.433
<i>EH</i>							
4×4	0.975	0.888	0.925	0.670	0.559	0.922	0.677
16×16	0.996	0.843	0.843	0.763	0.694	0.910	0.616
64×64	0.995	0.759	0.752	0.648	0.686	0.725	0.519
256×256	0.941	0.624	0.605	0.507	0.572	0.558	0.424
<i>HT</i>							
4×4	0.989	0.914	0.949	0.880	0.711	0.960	0.779
16×16	0.997	0.864	0.916	0.816	0.801	0.921	0.727
64×64	0.995	0.783	0.780	0.663	0.720	0.721	0.543
256×256	0.948	0.627	0.607	0.510	0.580	0.557	0.425
<i>KW</i>							
4×4	0.865	0.401	0.507	0.564	0.601	0.627	0.327
16×16	0.922	0.499	0.384	0.288	0.621	0.495	0.360
64×64	0.905	0.513	0.456	0.326	0.543	0.477	0.282
256×256	0.844	0.499	0.456	0.375	0.499	0.422	0.319

4538 keywords and then reduced to 150 dimensions with latent semantic indexing (LSI) (see Koskela & Laaksonen, 2003 for more details). Four SOM grids of different sizes were used (4×4, 16×16, 64×64, and 256×256 map units). In estimating the data distributions over the SOMs, this results in approximately 3750, 234, 15, and 0.92 training set vectors on the average per SOM unit, respectively.

First of all, it can be observed from Table 1 that the normalized entropies of the whole database (column ‘all’), for all SOM sizes and for all features, are close to one and clearly higher than the ones computed with only one image class. On the other hand, normalized entropies of semantic image classes express distinct differences over the features, providing estimates about the discriminating abilities of those particular feature extraction methods with the used object classes.

In our previous experiments on image retrieval (e.g. Laaksonen et al., 2002; Koskela & Laaksonen, 2003), it has been determined that of the six studied classes, sunsets and traffic signs are the ‘easiest’ ones with the used low-level visual features, i.e. they yield by far the best retrieval results. The four remaining classes exhibit more similar retrieval behavior, with the class of cars showing the lowest retrieval precision compared to the a priori probability of the class. These findings agree with the normalized entropies of the classes. With the *keyword* feature, all the class-wise relevant entropies are smallest, indicating that the feature is

able to cluster all six semantic classes. Again, this agrees with previous experiments (Koskela & Laaksonen, 2003), in which the superior retrieval performance of the *keyword* feature was perceived.

An overall trend seems to be that the normalized entropy decreases as the size of the SOM increases, especially with the distributions of the semantic classes. An explanation for this behavior can be found from the fact that with the two largest SOMs the ratio of the number of images in the studied class and the total number of map units becomes overly small. In this setting, the actual maximum of entropy is considerably smaller than H_{\max} and the entropy value thus mostly reflects just the size of the image class. The SOM algorithm is by nature a trade-off between clustering and topological ordering. This trade-off depends on the size of the SOM; the clustering property is dominant with relatively small SOMs whereas the topology of the map becomes more significant as the size of the SOM is increased. For these reasons, with large SOMs the class-wise normalized entropy measure is less informative and the spatial configuration of the data on the SOM grid should be taken into account.

5. SOM surface convolutions

As discussed above, the calculation of entropies does not yet take into account the spatial topology of the SOM units in any way. This is a direct consequence of the fact that the SOM indices are regarded as discrete symbols without any

561 connection to the SOM grid structure. This is clearly a
 562 drawback, because it is the topological order of the units that
 563 separates SOM from other vector quantization methods. It
 564 should be possible to exploit the ordering property of the
 565 SOM also in the entropy calculations.

566 Some authors have considered probabilistic models for
 567 Self-Organizing Maps, for example, the Generative Topo-
 568 graphic Mapping (GTM) by Bishop, Svensén, and Williams
 569 (1998), coding models by Luttrell (1991), and various
 570 energy functions that can be given probabilistic interpret-
 571 ations (Heskes, 1999; Kostianen & Lampinen, 2001;
 572 Lampinen & Kostianen, 2002). Especially, in Kostianen
 573 and Lampinen (2001), the authors present a generative
 574 probability density model for which the usual SOM training
 575 gives the maximum likelihood estimate. In the model, the
 576 density has Gaussian form within each Voronoi cell.

577 Instead of using density models in the feature space, we
 578 are here looking at the discrete distributions over the map
 579 surface. If the BMU of each training vector is given a small
 580 positive ‘hit’ value equal to the inverse of the number of
 581 training vectors, and the hit values are summed in each
 582 SOM unit, then the distribution (2) results.

583 One may now force the neighboring SOM units to
 584 interact by *low-pass filtering* or *convolving* the hit
 585 distributions on the SOM surface. When the surface is
 586 convolved, the one-to-one relationship between input
 587 vectors’ SOM indices and hits on the SOM surface is
 588 broken. Instead, each hit results in a spread point response
 589 around the BMU.

590 This is actually related to a kernel-based estimation of a
 591 class density in the original feature space (Duda & Hart,
 592 1973). Assume that we are using a reduced kernel expansion
 593 in which the kernel centers are only located on the SOM
 594 weight vectors. The weight of a kernel is equal to the a priori
 595 probability of that unit as a BMU

$$596 \quad p(\mathbf{x}|\mathcal{C}) = \sum_{i=0}^{K-1} P_i(\mathcal{C})G(\|\mathbf{x} - \mathbf{w}_i\|), \quad (8)$$

597 where $G(\cdot)$ is the kernel function, e.g. a spherical Gaussian,
 598 and the SOM weight vectors \mathbf{w}_i are the corresponding
 599 centroids. Instead of fitting this mixture model onto the data,
 600 as, e.g. in the GTM method (Bishop et al., 1998), we take the
 601 weights $P_i(\mathcal{C})$ from the computed hit histogram for class \mathcal{C}
 602 and the weight vectors from the trained SOM. Depending on
 603 the variance of the kernel function, kernels will overlap and
 604 weight vectors close to each other in the feature space will
 605 partially share each other’s probability mass.

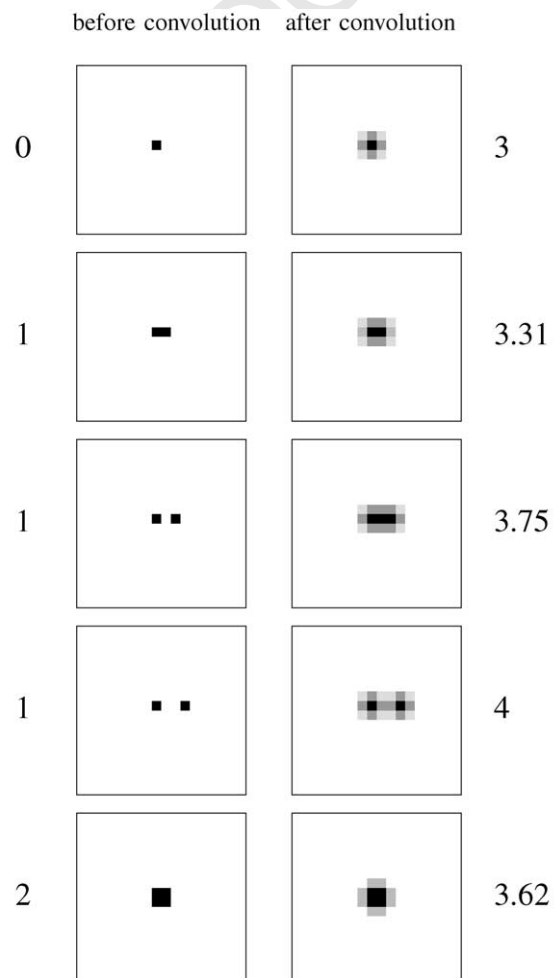
606 The present authors have made comparisons of various
 607 ways to compute the convolution, including a kernel model
 608 that takes the distances between the SOM weight vectors
 609 into account in a manner resembling the U -matrix technique
 610 (Ultsch & Siemon, 1990; Kraaijveld et al., 1995). In the
 611 experiments, the form of the convolution was forced to
 612 follow the form of the U -matrix, i.e. the span of the
 613 convolution was tuned inversely to the distances between
 614
 615
 616

617 the SOM units. That method also bears similarity to the
 618 smoothed data histogram approach (Pampalk et al., 2002) in
 619 which data points are not mapped one-to-one to their BMUs
 620 but spread into s closest map units in the feature space.
 621 However, it turned out that results are almost the same if the
 622 convolution is simply computed over the two-dimensional
 623 map surface, which is computationally much simpler
 624 (Koskela, Laaksonen, & Oja, 2002).

625 Let us next study the effects the convolution has on the
 626 surface entropy with a series of simple artificial examples
 627 with a convolution mask

$$628 \quad h = \begin{matrix} \frac{1}{16} & \frac{1}{8} & \frac{1}{16} \\ \frac{1}{8} & \frac{1}{4} & \frac{1}{8} \\ \frac{1}{16} & \frac{1}{8} & \frac{1}{16} \end{matrix}.$$

629 The left column of images in Fig. 2 shows a series of SOM
 630 surface value fields prior to convolution. The right image
 631 column displays the same surfaces after a convolution with
 632
 633
 634



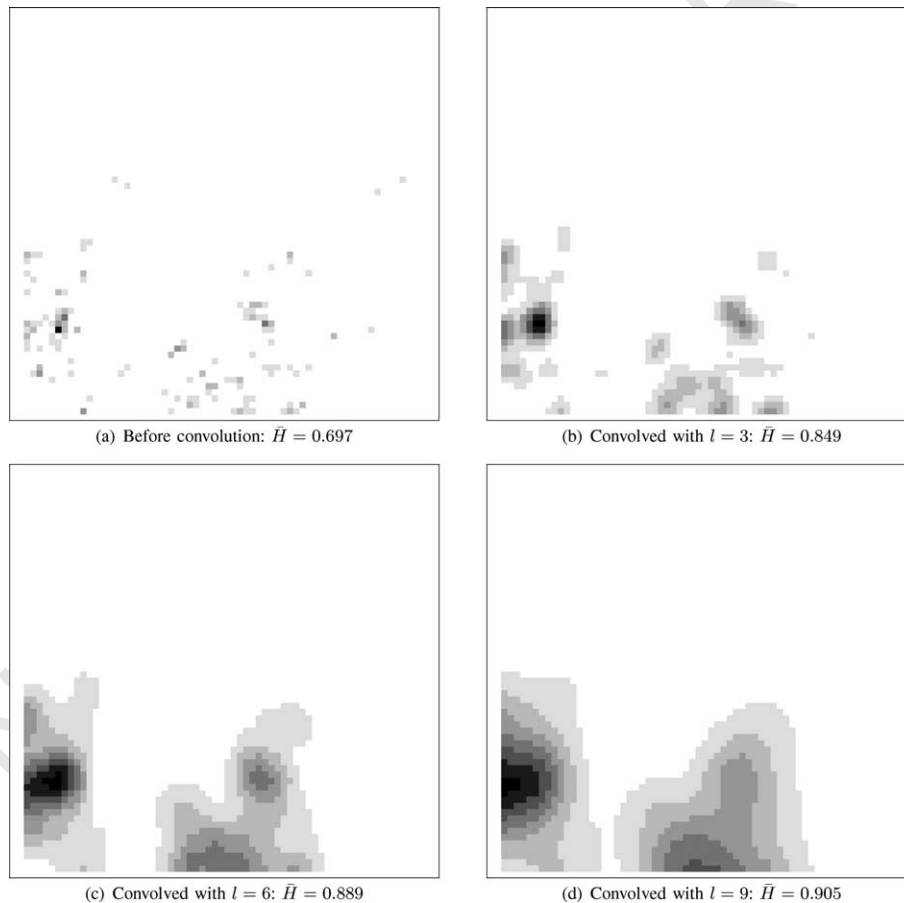
635 Fig. 2. Entropies of value fields before and after a convolution. The gray
 636 shades have been scaled in each image separately so that the darkest shade
 637 corresponds to the largest value in that particular image.
 638
 639
 640
 641
 642
 643
 644
 645
 646
 647
 648
 649
 650
 651
 652
 653
 654
 655
 656
 657
 658
 659
 660
 661
 662
 663
 664
 665
 666
 667
 668
 669
 670
 671
 672

673 this mask. The left and right value columns of Fig. 2 show
 674 the corresponding entropy values. One can see how the
 675 entropy measure is unable to make a difference between the
 676 three nonconvolved two-unit distributions. On the other
 677 hand, it is evident from the entropy values obtained after the
 678 convolution that the nearer the two peaks are to each other,
 679 the more the distribution resembles the one-peak distri-
 680 bution of the first row. Even for the distribution on the last
 681 row, where the nonconvolved entropy is the largest, the
 682 convolved entropy is smaller than that of either of the two-
 683 peak cases.

684 One may note that the entropy of the convolution mask h
 685 itself can also be computed to yield $H = 3$. This value might
 686 then further be used as a subtracter to ‘correct’ the post-
 687 convolution entropy values in the right column of Fig. 2. As
 688 the result, the convolved entropy values would seem to be
 689 sums of two terms, one standing for the spread of the
 690 original distribution and the other, constant value, for the
 691 shape of the convolution mask. Unfortunately this is true
 692 only for very compact distributions such as those seen in
 693 Fig. 2. In the other extreme, when the distribution is even
 694 over the whole SOM surface, the convolution really does
 695 not change the entropy value at all.

729 The exact relationships between (i) the spatial compact-
 730 ness of a pattern class; (ii) the entropy of its distribution on
 731 the SOM; and (iii) the increase of the entropy in the
 732 convolution still remain concealed. There are numerous
 733 reasons for this. Most importantly, the compactness and
 734 shape of a pattern class should first be defined in terms of the
 735 pattern space, as depicted in Fig. 1. Only then would we be
 736 able to study how the subsequent nonlinear projections
 737 preserve these characteristics and to what extent interclass
 738 impurities are being introduced as an unavoidable byprod-
 739 uct. Following this line, the two-fold role of the selected
 740 SOM size needs to be regarded. The smaller the map,
 741 the denser the class distributions are forced to be, but they
 742 will then also inevitably overlap. When the SOM size is
 743 increased, the distributions become less overlapped, while
 744 simultaneously becoming more fragmented.

745 An example illustration with real data is again obtained
 746 from the CBIR application of Corel images, using the
 747 sunsets image class and a 64×64 -sized SOM trained with
 748 the *Scalable Color* descriptor. Fig. 3 visualizes the original
 749 distribution of BMU hits and a series of distributions with
 750 increasing size of the convolution mask. The SOM surface
 751 convolution has been performed as two consecutive
 752



727 Fig. 3. Distribution of the sunsets class on a 64×64 -sized SOM trained with the *Scalable Color* descriptor (a) before and (b)–(d) after convolution with
 728 triangular masks of size 3, 6, and 9 map units, respectively.

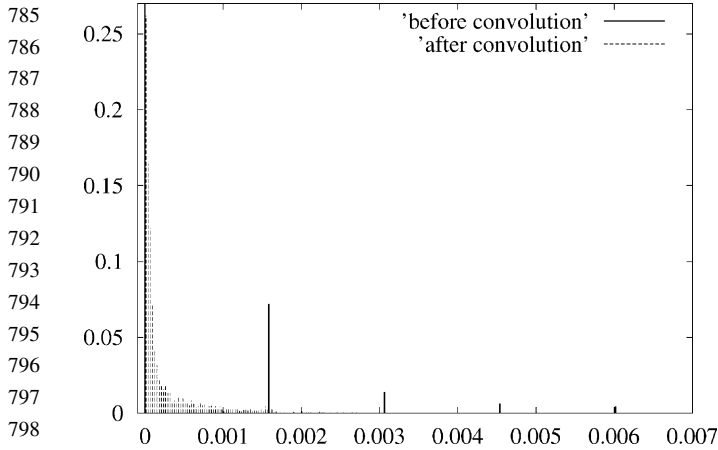


Fig. 4. Histograms of SOM surface values before and after a convolution. The sunsets class has been distributed on a 64×64 -sized SOM trained with the *Scalable Color* descriptor.

one-dimensional convolutions of a triangular window form

$$w_l[n] = \frac{l - |n|}{l}, \quad n = -l, -l + 1, \dots, l. \quad (9)$$

One may note that the convolutions in Fig. 2 correspond to mask size $l=2$. In Fig. 3 the window sizes have been $l=3, 6, \text{ and } 9$. It can be seen in all images that the studied class is concentrated in a certain area of the SOM of the feature in question, but, on the other hand, it is split into two or more separate regions. The larger the convolution window is, the smoother is the overall shape of the distribution due to

Table 2
Normalized entropies of convolved SOMs with the *Color Structure* (CS), *Scalable Color* (SC), *Edge Histogram* (EH), *Homogenous Texture* (HT), and *keyword* (KW) descriptors, respectively

SOM size	All	Faces	Cars	Planes	Sunsets	Horses	Tr. signs
<i>CS</i>							
4×4	0.997	0.936	0.982	0.830	0.780	0.945	0.990
16×16	1.000	0.949	0.984	0.899	0.837	0.953	0.981
64×64	1.000	0.946	0.977	0.905	0.848	0.942	0.940
256×256	1.000	0.926	0.954	0.869	0.835	0.908	0.848
<i>SC</i>							
4×4	1.000	0.945	0.997	0.960	0.861	0.989	0.960
16×16	1.000	0.950	0.993	0.950	0.894	0.944	0.965
64×64	1.000	0.944	0.980	0.928	0.889	0.917	0.943
256×256	1.000	0.930	0.949	0.883	0.867	0.882	0.861
<i>EH</i>							
4×4	0.994	0.947	0.986	0.825	0.734	0.969	0.856
16×16	0.999	0.937	0.950	0.882	0.826	0.976	0.879
64×64	0.999	0.927	0.931	0.893	0.835	0.969	0.834
256×256	0.999	0.903	0.908	0.868	0.842	0.934	0.785
<i>HT</i>							
4×4	0.995	0.955	0.972	0.933	0.823	0.984	0.895
16×16	0.999	0.949	0.979	0.935	0.879	0.983	0.929
64×64	0.999	0.938	0.971	0.926	0.889	0.977	0.895
256×256	1.000	0.923	0.946	0.890	0.880	0.938	0.808
<i>KW</i>							
4×4	0.996	0.797	0.913	0.856	0.826	0.922	0.818
16×16	0.999	0.851	0.853	0.800	0.844	0.902	0.799
64×64	0.996	0.803	0.764	0.710	0.822	0.809	0.752
256×256	0.992	0.772	0.713	0.653	0.783	0.751	0.657

the vanishing of the details. The selection of a proper size for the convolution mask can thus be identified as a form of the general *scale-space problem*, and different sizes will certainly be optimal for different purposes, e.g. for visualization and for classification.

Fig. 4 displays two histograms of SOM surface values obtained again, with the sunsets class and the 64×64 -sized SOM of the *Scalable Color* descriptor as in Fig. 3. The first histogram is calculated from the nonconvolved ‘raw’ value field (Fig. 3(a)) and it peaks strongly in two locations, corresponding to zero and one images being mapped to a particular BMU. The first peak (zero images) actually rises well above the scale of the figure, to the value 0.9. There are also weaker but still visible peaks for two, three, and four image cases. The second histogram has been obtained after convolution ($l=6$, Fig. 3(c)) and it can be seen to be much smoother, as it is the result of a low-pass filtering operation on the discrete value field. Especially, there is a notable fraction of values that are slightly greater than zero.

Similarly as in Table 1, Table 2 shows the normalized entropies of all images and the six image classes with the five feature extraction methods and four SOM grids, this time calculated after convolving the discrete value fields with triangular masks. The size of the used mask depends on the size of the SOM: symmetric triangular masks with $l=2, 4, 6, \text{ and } 8$ map units were used for the $4 \times 4, 16 \times 16, 64 \times 64, \text{ and } 256 \times 256$ sized SOMs, respectively.

From the resulting normalized entropies in Table 2, it can be seen that the convolution further increases the entropies

of the distributions, so that when using all images the normalized entropy approaches the theoretical maximum of one. For the image classes, the normalized entropy values continue to express the discriminating abilities of the features with the used image classes, now taking also the SOM topology into account. For sunsets and traffic signs, being the easiest classes for the features, the normalized entropies are overall the lowest. In this setting, the normalized entropies on SOMs of different sizes are rather similar and the inverse proportionality with respect to SOM size present in Table 1 is not observed here since we used larger convolution windows with larger SOMs. Overall, the convolution spreads the dense clusters and partially fills the empty gaps between them, causing thus increased entropy.

6. Multiple feature extractions

In some application areas it is possible to use more than one feature extraction method in parallel. Our example case, content-based image retrieval, is such an area. In CBIR, three different feature categories are generally recognized: color, texture, and shape features. Each of them is useful in CBIR by its own right, and it is wise not to combine them all in one descriptor. In addition, within each category there exist various different feature extraction techniques that complement each other.

Let us denote by $P=(P_0, P_1, \dots, P_{K-1})$ and $Q=(Q_0, Q_1, \dots, Q_{K-1})$ the probability distributions on two equal-sized SOM surfaces, as explained in Section 4, obtained from the same data with two different feature extractions. Then the question of the independence of the features arises. As entropies $H(P)$ and $H(Q)$ measure the distributions of the single feature vectors, *mutual information* $I(P, Q)$ can be used for studying the interplay between them

$$I(P, Q) = \sum_{i=0}^{K-1} \sum_{j=0}^{K-1} R_{ij} \log \frac{R_{ij}}{P_i Q_j}, \quad (10)$$

where R_{ij} is the estimated joint probability, i.e. the number of images whose P -type feature vector is mapped to the P -type SOM's unit i and Q -type feature vector is mapped to the Q -type SOM s unit j , divided by N . Mutual information is always nonnegative and attains its minimum value of zero if the two features are statistically independent, i.e. if $R_{ij} = P_i Q_j$ for all i, j .

The dependency of the mutual information on the SOM size can to some extent be canceled, e.g. by dividing it with the smaller of the two entropies

$$\bar{I}(P, Q) = \frac{I(P, Q)}{\min\{H(P), H(Q)\}}. \quad (11)$$

yielding a measure $\bar{I}(P, Q) \in [0, 1]$ denoted here as *normalized mutual information*. The normalized mutual information thus equals to one if the denser one of the two distributions P and Q does not contain any information not present in the sparser one of the two. One could also have chosen to normalize $I(P, Q)$ by H_{\max} , which would have resulted in the normalized values being slightly smaller in general.

Table 3 illustrates the normalized mutual information. The pairwise normalized mutual information of the four studied MPEG-7 descriptors and the *keyword* feature are shown in two SOM sizes (4×4 and 16×16 map units). In addition, a normalized version (each vector component normalized independently to zero mean, unit variance) of the *Homogenous Texture* descriptor was used to train separate SOMs. With the larger SOMs of previous experiments (64×64 and 256×256 map units), the mutual information measure is as such not usable as the number of images sharing a common BMU becomes small and the joint probability distribution R_{ij} sparse. Therefore, when using larger SOMs with respect to the number of available data items, information about the spatial configuration of the data on the SOMs should be taken into account in this type of considerations. Unfortunately, it is not straightforward to calculate the mutual information of two features in that manner from the convolved map surfaces.

The results in Table 3 show that the SOMs trained with the unnormalized and normalized versions of the *Homogenous Texture* feature (HT and nHT) have by far the largest values for mutual information as can be well expected. Of the separate features, the two color-based features, *Color Structure* (CS) and *Scalable Color* (SC) have the largest value on both SOMs, which again was to be expected. Additionally, the mutual information of *Edge Histogram* (EH) and *Homogenous Texture* (HT) is high on the smaller SOM, but not so much on the larger SOM with more resolution. In general, those visual features that represent different aspects of the visual scene, e.g. color and shape, have a low value of mutual information, indicating

Table 3
Normalized Mutual Informations Of Different Features On 4×4 And 16×16 Sized SOMs

	4×4-sized SOM					16×16-sized SOM				
	SC	EH	HT	nHT	KW	SC	EH	HT	nHT	KW
CS	0.16	0.077	0.057	0.057	0.032	0.33	0.17	0.19	0.18	0.17
SC		0.016	0.019	0.018	0.020		0.13	0.15	0.14	0.17
EH			0.14	0.14	0.031			0.21	0.21	0.16
HT				0.64	0.013				0.62	0.15
nHT					0.017					0.15

1009 that they are close to independent. The pair with the smallest
1010 normalized mutual information is *Edge Histogram* and
1011 *Scalable Color*. These features could thus be used together
1012 to produce the most independent joint information. The
1013 *keyword* feature is the most neutral one in the sense that its
1014 mutual information values with different types of visual
1015 features are all nearly equal. The semantic content of the
1016 images expressed in terms of keywords thus does not seem
1017 to favor any visual feature type more than the others.

1020 7. Bayesian decision estimation

1022 If one knows the a priori probabilities and probability
1023 densities of the object classes, one can use the Bayesian
1024 decision rule to make optimal classification for object \mathbf{x}_j

$$1026 \text{class} = \arg \min_c P(C|\mathbf{x}_j), \quad (12)$$

1028 where the posterior probability can be computed by Bayes
1029 rule

$$1031 P(C|\mathbf{x}_j) = \frac{\rho(C)P(\mathbf{x}_j|C)}{P(\mathbf{x}_j)}. \quad (13)$$

1033 There $\rho(C)$ is the a priori probability of the class C and
1034 $P(\mathbf{x}_j|C)$ the probability distribution of the class C for object
1035 \mathbf{x}_j . This distribution is discrete as there are only a finite
1036 number of objects. The denominator has no influence on the
1037 maximization over classes and can be dropped. One should
1038 note that the fact that we are dealing with discrete
1039 probability distributions and Voronoi regions of differing
1040 sizes does not invalidate the principle of Bayesian decision
1041 optimality.

1042 In practice, the images are given in terms of their
1043 features, which are distributed over the feature maps in
1044 SOM training. If there are available L feature vectors \mathbf{x}_{jk} ,
1045 $k=0, \dots, L-1$ for the object \mathbf{x}_j , and if the features are
1046 assumed mutually independent, the above expression can be
1047 evaluated as

$$1049 \rho(C)P(\mathbf{x}_j|C) = \rho(C) \prod_{k=0}^{L-1} P(\mathbf{x}_{jk}|C) \quad (14)$$

1052 to decide on the j th object's membership in class C . The
1053 probability densities $P(\mathbf{x}_{jk}|C)$ are obtained from the
1054 convolved distributions over the separate feature maps of
1055 the images belonging to class C . In the CBIR setting,
1056 assuming independence for different image features such as
1057 color, texture, and shape is quite realistic, as indicated by the
1058 small mutual information measures given in Section 6.

1059 A typical scenario in a CBIR setting is that there are no
1060 explicit classes given for the images. Instead, we are
1061 interested in only two classes given by the user: the images
1062 which are *relevant* vs *nonrelevant* to the present query,
1063 respectively. The relevance is totally determined by the
1064 user, who has in mind a certain type of image that she is

1065 looking for from the database. In the search technique called
1066 *query by example*, the CBIR system suggest or presents a
1067 number of images to the user at each query round, and the
1068 user is expected to evaluate their relevance to her current
1069 retrieval task. This information is then fed back to the
1070 retrieval system. A straightforward way to implement this is
1071 to ask the user to pick up those images that are relevant
1072 to her from the set of returned images. The other images are
1073 then assumed nonrelevant. With this information used
1074 as *relevance feedback*, the system is able to incrementally
1075 fine-tune the selection so that more and more relevant
1076 images will be shown at consecutive query rounds.

1077 Let us consider relevance feedback from the point of
1078 view of Bayesian decision theory. In the above Bayesian
1079 decision rule, we now ask what is the relevance vs.
1080 nonrelevance of any given image Note that relevance and
1081 nonrelevance are not mutually exclusive nor complemen-
1082 tary; by the way the distributions are convolved over the
1083 map surfaces, some images have nonzero probabilities of
1084 both relevance and nonrelevance, and there may be images
1085 for which both probabilities are zero, at least in early stages
1086 of the query process. Estimates for the above probabilities
1087 can be obtained from the relevance feedback information
1088 received earlier from the user. The prior probability $\rho(C)$
1089 can be estimated from the count of relevant objects found
1090 that far, divided by the total count of retrieved objects. That
1091 value is in information retrieval literature commonly
1092 referred to as the *precision* of the retrieval. The estimate
1093 is, however, overly biased toward large values as a
1094 functioning CBIR system should always exceed random
1095 browsing in accuracy. In our experiments with the PicSOM
1096 system using visual low-level features and semantic (high-
1097 level) image classes, the precision has most often been
1098 around 10–15 times the a priori probability in the beginning
1099 of an iterative query.

1100 The question asked in relevance feedback is which
1101 images to choose next for the user. In the most typical CBIR
1102 setting, such images should have maximal probability of
1103 relevance and minimal probability of nonrelevance. Instead
1104 of maximizing the posterior probability of object \mathbf{x}_j with
1105 respect to class C , as in (12), we should now find the *object*
1106 that gives maximum probability of relevance

$$1108 \text{object index} = \arg \max_j P(C_{\text{rel}}|\mathbf{x}_j), \quad (15)$$

1110 where C_{rel} is the class of objects known to be relevant. At
1111 the same time, the probability of the nonrelevance class
1112 C_{nonrel} should be minimized. A reasonable objective
1113 function, to be maximized over the index j , is then

$$1116 q_j = P(C_{\text{rel}}|\mathbf{x}_j) - \alpha P(C_{\text{nonrel}}|\mathbf{x}_j), \quad (16)$$

1118 where the parameter α can be used to adjust the
1119 weighting between the two terms. Let us write this
1120

in terms of the a priori probabilities and class distributions

$$q_j = \frac{\rho(C_{\text{rel}})P(\mathbf{x}_j|C_{\text{rel}}) - \alpha\rho(C_{\text{nonrel}})P(\mathbf{x}_j|C_{\text{nonrel}})}{P(\mathbf{x}_j)}, \quad (17)$$

where the denominator is the class-independent probability of image \mathbf{x}_j . In practice, this function will be maximized at the peak regions of $P(\mathbf{x}_j|C_{\text{rel}})$ where $P(\mathbf{x}_j|C_{\text{nonrel}})$ is small. Thus we may assume that for these images, the density $P(\mathbf{x}_j)$ is constant and it may be dropped from the expression.

Another simplifying assumption is that $\alpha = \rho(C_{\text{rel}})/\rho(C_{\text{nonrel}})$. Then what remains is just the difference between the class-conditional distributions. As in (14), both of these can be expressed as products of the corresponding relevance/nonrelevance distributions over the separate independent feature maps, and we obtain

$$q_j = \prod_{k=0}^{L-1} P(\mathbf{x}_{jk}|C_{\text{rel}}) - \prod_{k=0}^{L-1} P(\mathbf{x}_{jk}|C_{\text{nonrel}}) \quad (18)$$

as a *qualification* value or *score* for image j . The conditional probability densities $P(\mathbf{x}_{jk}|C_{\text{rel}})$ and $P(\mathbf{x}_{jk}|C_{\text{nonrel}})$ are obtained from the convolved feature distributions of the images marked by the user as relevant and nonrelevant, respectively.

In practice, a problem arises from the fact that, especially in the early stages of the query process, the probability distributions are quite sparse and only a few of the map units have nonzero probabilities for either relevance or nonrelevance on any given feature map. Then the products of the estimated probabilities are mostly zero and this objective function is useless. To solve this problem, we have in practice replaced the products of featurewise probabilities by sums

$$q'_j = \sum_{k=0}^{L-1} [P(\mathbf{x}_{jk}|C_{\text{rel}}) - P(\mathbf{x}_{jk}|C_{\text{nonrel}})] \quad (19)$$

Then, it is sufficient if an image is estimated as highly relevant on one of the feature maps, even if the relevance may be zero on some other map.

A probabilistic interpretation for the score function (19) is that we are now looking at each feature separately. An image is considered relevant if its probability of relevance is high and probability of nonrelevance is low on *either* feature 0 or feature 2 or ... or feature $L-1$. Thus we still assume the features independent, as in (18), but do not look at the total probability of the images. This is quite reasonable; it is easy to imagine queries in which, e.g. the color has no semantic meaning at all while the shape is very important.

Eq. (19) is exactly the core of the PicSOM CBIR system. Its applicability has been experimentally validated in various studies including Laaksonen et al. (2002), and Rummukainen et al. (2003), where PicSOM's retrieval performance was found to be at least on the same level as two other CBIR systems, one implemented by ourselves

and based on vector quantization and the other being the publicly available GIFT system (Squire et al., 2000). A key finding in all performed experiments has been that PicSOM's retrieval accuracy can be increased by adding new features in the system, i.e. by increasing L in (19). This fact holds—but to smaller extent—even in the case when the added feature is highly correlated with one or more existing features in the sense of large mutual information defined by Eq. (11).

In relevance feedback, another problem to be solved is how to update the densities of the relevant and nonrelevant classes after each query round. In practice, this can be done in the PicSOM system simply by adding the hits caused by the new relevant and nonrelevant samples to the map units, convolving them with the mask used, and renormalizing the distributions to unit sums. This process can also be given a Bayesian interpretation in a similar fashion as in Cox, Miller, Minka, Papathomas, and Yianilos (2000) or Vasconcelos and Lippman (1999). Similarly as in Cox et al. (2000), let us assume that the user has in her mind a certain image \mathbf{x}_{rel} that she is looking for. The query is started in such a way that the CBIR system suggests to the user a set of images \mathcal{D}_0 . If \mathbf{x}_{rel} happens to be among them, the query stops. Otherwise, the user indicates a subset of them, \mathcal{R}_0 , as relevant, meaning that they somehow resemble the desired image or belong to the same category by some criterion that the user has in her mind. The system shows another set of images \mathcal{D}_1 , the user chooses the relevant ones \mathcal{R}_1 from them, etc.

Let us denote the history of the query up to the $t-1$ 'st round by $\mathcal{H}_{t-1} = (\mathcal{D}_0, \mathcal{R}_0, \mathcal{D}_1, \mathcal{R}_1, \dots, \mathcal{D}_{t-1}, \mathcal{R}_{t-1})$. The optimal images to be shown to the user at the next round t are those images \mathbf{x} that maximize the current probability of relevance $P(\mathbf{x} = \mathbf{x}_{\text{rel}}|\mathcal{H}_{t-1})$. A recursive update for this probability can be obtained by writing (Cox et al., 2000)

$$P(\mathcal{R}_t|\mathbf{x} = \mathbf{x}_{\text{rel}}, \mathcal{H}_{t-1}, \mathcal{D}_t)P(\mathbf{x} = \mathbf{x}_{\text{rel}}|\mathcal{H}_{t-1}, \mathcal{D}_t) \quad (20)$$

$$= P(\mathcal{R}_t, \mathbf{x} = \mathbf{x}_{\text{rel}}|\mathcal{H}_{t-1}, \mathcal{D}_t) \quad (21)$$

$$= P(\mathbf{x} = \mathbf{x}_{\text{rel}}|\mathcal{H}_{t-1}, \mathcal{D}_t, \mathcal{R}_t)P(\mathcal{R}_t|\mathcal{H}_{t-1}, \mathcal{D}_t) \quad (22)$$

$$= P(\mathbf{x} = \mathbf{x}_{\text{rel}}|\mathcal{H}_t)P(\mathcal{R}_t|\mathcal{H}_{t-1}, \mathcal{D}_t). \quad (23)$$

In a deterministic CBIR system, the set of displayed images \mathcal{D}_t is completely determined once the query history \mathcal{H}_{t-1} is given. Therefore, \mathcal{D}_t can be dropped from the conditional probabilities. One thus ends with an iterative update rule similar to the one obtained in Cox et al. (2000)

$$P(\mathbf{x} = \mathbf{x}_{\text{rel}}|\mathcal{H}_t)P(\mathbf{x} = \mathbf{x}_{\text{rel}}|\mathcal{H}_{t-1}) \frac{P(\mathcal{R}_t|\mathbf{x} = \mathbf{x}_{\text{rel}}, \mathcal{H}_{t-1})}{P(\mathcal{R}_t|\mathcal{H}_{t-1})}. \quad (24)$$

The denominator does not depend on \mathbf{x} , but is just a normalizing factor summing the term in the numerator over all \mathbf{x} . The important factor in the update is the term

1233 $P(\mathcal{R}_t | \mathbf{x} = \mathbf{x}_{\text{rel}}, \mathcal{H}_{t-1})$. At any point \mathbf{x} , given that it is the
 1234 relevant image that the user is looking for, the closer the
 1235 marked images in \mathcal{R}_t are to \mathbf{x} , the higher is this probability.
 1236 Thus, qualitatively, the probability will be increased in the
 1237 vicinity of \mathcal{R}_t but not elsewhere. The same effect is in
 1238 PicSOM obtained by the simple method of adding the hits
 1239 caused by \mathcal{R}_t to the distribution, convolving, and renorma-
 1240 lizing. Exactly the same is done for the distribution of
 1241 nonrelevant images, and thus both distributions get tuned in
 1242 the query process.

1243 8. Conclusions

1244
 1245 In this paper, we have shown how distributions of feature
 1246 vectors calculated from objects of mutual semantic
 1247 similarity can be studied on the SOM surfaces. We
 1248 demonstrated that the entropy of the distribution character-
 1249 izes quantitatively the compactness of an object class. The
 1250 compactness in turn—even though not exactly defined—is
 1251 an intuitive indicator of the success of the feature extraction
 1252 and SOM training phases for that particular class. The more
 1253 compact the distribution is, the smaller overlap with other
 1254 classes can be expected in visualization and other uses of the
 1255 SOM. More informative entropy values were obtained if the
 1256 SOM surface with the class distribution was low-pass
 1257 filtered prior to the calculation of the entropy. In that
 1258 fashion, vector distributions which are unimodal are favored
 1259 whereas fragmented and multimodal distributions are
 1260 punished.

1261
 1262 When studying two different SOMs created with
 1263 different feature extraction methods, we showed that
 1264 the mutual information of the distributions could be used
 1265 to identify both the most similar and the most
 1266 uncorrelated pair of features. In our example application
 1267 of content-based image retrieval, this quantitatively
 1268 confirmed the expected result that, e.g. two color features
 1269 are mutually more correlated than, e.g. a color feature
 1270 and a texture feature.

1271
 1272 The described techniques can be utilized in selecting an
 1273 effective set of features in various application areas. The
 1274 content-based image retrieval application turned out to be
 1275 an eligible field for these considerations. There exists a vast
 1276 number of different feature extraction methods for images
 1277 and other visual data, and the proposed method can be used
 1278 as an efficient way of comparing these features and the
 1279 SOMs produced with them. Instead of having to run
 1280 extensively actual retrieval sessions with the CBIR system,
 1281 a direct measure based on the ability of the feature
 1282 extraction to discriminate images belonging to a certain
 1283 set of semantic similarity or relevancy from other images
 1284 was obtained. Mutual information can also be used to select
 1285 the subset of the feature extraction methods with the most
 1286 independent features.

1287 In addition, it was shown here how the distributions on
 1288 SOM surfaces can be given a Bayesian interpretation and

used for choosing either the most probable class for a data
 item, or the most likely data item belonging to a given class.
 This duality opens new perspectives for the use of SOMs in
 data exploration and visualization and is an essential part of
 the PicSOM CBIR system.

Acknowledgements

This work was supported by the Academy of Finland in
 the projects Neural methods in information retrieval based
 on automatic content analysis and relevance feedback and
 New information processing principles, the latter being part
 of the Finnish Centre of Excellence Programme.

References

- Bishop, C. M., Svensén, M., & Williams, C. K. I. (1998). GTM: the generative topographic mapping. *Neural Computation*, 10(1), 215–234.
- Castelli, V., & Bergman, L. D. (Eds.). (2002). *Image databases—Search and retrieval of digital imagery*. New York: Wiley.
- Cover, T. M., & Thomas, J. A., (1991). *Elements of information theory*. Wiley series in telecommunications. New York: Wiley.
- Cox, I. J., Miller, M. L., Minka, T. P., Papatthomas, T. V., & Yianilos, P. N. (2000). The Bayesian image retrieval system, PicHunter: theory, implementation and psychophysical experiments. *IEEE Transactions on Image Processing*, 9(1), 20–37.
- Del Bimbo, A. (1999). *Visual information retrieval*. Los Altos, CA: Morgan Kaufmann Publishers.
- Duda, R. O., & Hart, P. E. (1973). *Pattern recognition and scene analysis*. New York: Wiley.
- Heskes, T. (1999). Energy functions for self-organizing maps. In E. Oja, & S. Kaski (Eds.), *Kohonen Maps* (pp. 303–315). Amsterdam: Elsevier.
- Kohonen, T. (2001). *Self-organizing maps* (3rd ed) *Springer series in information sciences*. Vol. 30. Berlin: Springer.
- Kohonen, T., Oja, E., Simula, O., Visa, A., & Kangas, J. (1996). Engineering applications of the self-organizing map. *Proceedings of the IEEE*, 84(10), 1358–1384.
- Koikkalainen, P. (1994). Progress with the tree-structured self-organizing map. In A. G. Cohn, *11th European conference on artificial intelligence* (pp. 211–215). European Committee for Artificial Intelligence (ECAI), Wiley.
- Koikkalainen, P., & Oja, E. (1990). Self-organizing hierarchical feature maps. In: *Proceedings of international joint conference on neural networks, San Diego, CA, USA* (Vol. II) (pp. 279–284).
- Koskela, M., & Laaksonen, J., (2003). Using long-term learning to improve efficiency of content-based image retrieval. In: *Proceedings of third international workshop on pattern recognition in information systems (PRIS 2003)*, Angers, France (72–79).
- Koskela, M., Laaksonen, J., & Oja, E., (2002). Implementing relevance feedback as convolutions of local neighborhoods on self-organizing maps. In: *Proceedings of international conference on artificial neural networks, Madrid, Spain* (pp. 981–986).
- Kostiainen, T., & Lampinen, J., (2001). Self-organizing map as a probability density model. In: *Proceedings of the international joint conference on neural networks (IJCNN 2001)*, Washington DC, USA (pp. 394–399).
- Kraaijveld, M. A., Mao, J., & Jain, A. K. (1995). A nonlinear projection method based on Kohonen's topology preserving maps. *IEEE Transactions on Neural Networks*, 6(3), 548–559.

- 1345 Laaksonen, J., Koskela, M., Laakso, S., & Oja, E. (2001). Self-organizing
1346 maps as a relevance feedback technique in content-based image
1347 retrieval. *Pattern Analysis and Applications*, 4(2+3), 140–152.
- 1348 Laaksonen, J., Koskela, M., & Oja, E. (2002). PicSOM—self-organizing
1349 image retrieval with MPEG-7 content descriptions. *IEEE Transactions*
1350 *on Neural Networks, Special Issue on Intelligent Multimedia Proces-*
1351 *sing*, 13(4), 841–853.
- 1352 Lampinen, J., & Kostiainen, T. (2002). Generative probability density
1353 model in the self-organizing map. In U. Seiffert, & L. C. Jain, *Self-*
1354 *organizing neural networks—Recent advances and applications.*
1355 *Studies in fuzziness and soft computing* (Vol. 78) (pp. 75–94).
1356 Heidelberg: Physica-Verlag.
- 1357 Lampinen, J., & Oja, E. (1992). Clustering properties of hierarchical self-
1358 organizing maps. *Journal of Mathematical Imaging and Vision*, 2(2–3),
1359 261–272.
- 1360 Lew, M. S. (Ed.). (2001). *Principles of visual information retrieval*. Berlin:
1361 Springer.
- 1362 Luttrell, S. P. (1991). Code vector density in topographic mappings: scalar
1363 case. *IEEE Transactions on Neural Networks*, 2(4), 427–436.
- 1364 Manjunath, B. S., Salembier, P., & Sikora, T. (Eds.). (2002). *Introduction to*
1365 *MPEG-7: Multimedia content description interface*. New York: Wiley.
- 1366 MPEG (2003). MPEG-7 Overview vers. 9. ISO/IEC JTC1/SC29/WG11
1367 N5525.
- 1368 Pampalk, E., Rauber, A., & Merkl, D., (2002). Using smoothed data
1369 histograms for cluster visualization in self-organizing maps. In:
1370 *Proceedings of international conference on artificial neural networks*
1371 *(ICANN 2002), Madrid, Spain* (pp. 871–876).
- 1372 Pullwitt, D. (2002). Integrating contextual information to enhance SOM-
1373 based text document clustering. *Neural Networks*, 15(8–9), 1099–1106.
- 1374 Rui, Y., Huang, T. S., & Chang, S.-F. (1999). Image retrieval: current
1375 techniques, promising directions, and open issues. *Journal of Visual*
1376 *Communication and Image Representation*, 10(1), 39–62.
- 1377 Rummukainen, M., Laaksonen, J., & Koskela, M., (2003). An efficiency
1378 comparison of two content-based image retrieval systems, GIFT and
1379 PicSOM. In: *Proceedings of international conference on image and*
1380 *video retrieval (CIVR 2003), Urbana, IL, USA* (pp. 500–509).
- 1381 Smeulders, A. W. M., Worring, M., Santini, S., Gupta, A., & Jain, R.
1382 (2000). Content-based image retrieval at the end of the early years.
1383 *IEEE Transactions on Pattern Analysis and Machine Intelligence*,
1384 22(12), 1349–1380.
- 1385 Squire, D., Müller, W., Müller, H., & Pun, T. (2000). Content-based query
1386 of image databases: inspirations from text retrieval. *Pattern Recog-*
1387 *nition Letters*, 21(13–14), 1193–1198.
- 1388 Ultsch, A., & Siemon, H. P., (1990). Kohonen's self organizing feature
1389 maps for exploratory data analysis. In: *Proceedings of international*
1390 *neural network conference (INNC-90), Paris, France* (pp. 305–308).
- 1391 Vasconcelos, N., & Lippman, A., (1999). Learning from user feedback in
1392 image retrieval systems. In: *Advances in neural information processing*
1393 *systems 12: Proceedings of the 1999 conference (NIPS*99), Denver,*
1394 *CO, USA* (pp. 977–983).
- 1395 Vesanto, J., & Alhoniemi, E. (2000). Clustering of the self-organizing map.
1396 *IEEE Transactions on Neural Networks*, 11(3), 586–600.
- 1397 Zhou, X. S., & Huang, T. S. (2003). Relevance feedback for image
1398 retrieval: a comprehensive review. *Multimedia Systems*, 8(6), 536–544.
- 1399
1400