LEARNING SIMILARITY SPACE

Abdurrahman Carkacioglu and Fatos-Yarman Vural

Middle East Technical University, Department of Computer Engineering, Ankara/TURKEY

ABSTRACT

In this study, we suggest a method to adapt an image retrieval system into a configurable one. Basically, original feature space of a content-based retrieval system is nonlinearly transformed into a new space, where the distance between the feature vectors is adjusted by learning. The transformation is realized by Artificial Neural Network architecture. A cost function is defined for learning and optimized by simulated annealing method. Experiments are done on the texture image retrieval system, which use Gabor Filter features. The results indicate that configured image retrieval system is significantly better than the original system.

1. INTRODUCTION

Most of the available image retrieval systems are designed by using a fixed set of features and a similarity metric that restricts the performance and human preferences in a specific task. On the other hand, it is well known that the design of a generic feature space, which is linearly separable, is almost impossible in many practical problems.

Roughly glanced though the literature published recently, most of the works, use signal and/or image processing feature extraction methods with or without preprocessing step in order to represent a given image. It is expected that extracted features are computationally feasible, reduce the problem data without discarding valuable information and represent the original image successfully. The effectiveness of the representation space is determined by how well patterns, in our case images, from different classes can be separated [2]. Nevertheless, there does not exist a single best representation for a given image [3].

After having selected the `right` set of features, and having characterized an image as a point in a multidimensional vector space, researchers make some assumption about the metric of the space [2],[4]. Typically, feature space is assumed to be Euclidean [4]. After selecting the metric of the space, a distance function is defined, such as Euclidean, Mahalanobis or City Block, in order to measure the distance between the feature vectors. The smaller the distance, the more similar the images to the query. Mathematically, it is expected that feature vectors of similar images are close to each other and this closeness is measured by a distance function.

The similarity metric is critical for content-based image retrieval systems [4],[5]. Unfortunately, image similarity computed by existing mathematical metric is not always consistent with the human perception. For example Euclidean distance may not effectively preserve the perceptual similarity, due to subjectivity of perceived similarity with respect to the related task and database [6]. Moreover, similarity measure based on the nearest neighbor criterion in the feature space is unsuitable in many cases [3]. This is particularly true when the image features correspond to low-level image attributes such as texture, color or shape.

Figure 1, shows some problematic examples of features from two different classes [8]. In such cases, using Euclidean distance for the nearest neighbor search might retrieve patterns without any perceptual relevance to the original query [9].

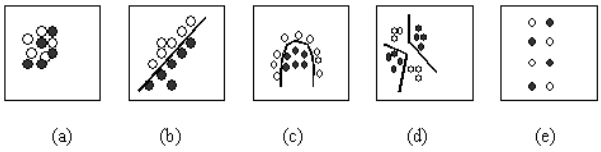

Figure 1. Possible problematic feature space in 2-d. Circles and squares are the feature values of two different classes. (a) Features are inadequate to distinguish the different classes. (b) Features are highly correlated. (c) Decision boundary is curved. (d) Distinct subclasses exist in the data. (e) Feature space is too complex.

In order to overcome the bottleneck of Euclidean metric discussed above, some efforts have spent in the context of image retrieval [9]. Ma and Manjunath [7] present a learning based approach to retrieve the similar image patterns by using self-organizing maps to get coarse labeling, followed by fine-tuning process using learning vector quantization. Santini and Jain [4] develop a similarity measure based on fuzzy logic. Minka and Picard report a system, which learns grouping of similar images from positive and negative examples provided by the users during query sessions [6]. Guo, Zhang and Li [5] defines a new metric called distance-from-boundary by the use of Support Vector Machines to measure image similarities. The basic idea is that a non-linear boundary separates images from the dissimilar ones.

In this study, we mainly concentrate on texture based image retrieval systems, that query the image database by example, where the user does not have any particular target in mind, but selects an image or draws a sketch and asks to retrieve similar images. Thus, the basic operation is ordering a portion of image database with respect to a similarity metric [1].

Most of the image retrieval systems are non-configurable in which the retrieval process does not depend on the content of the database. We propose a new method that enables us to make the system configurable without changing the underlying feature extraction mechanism. This task is achieved by the nonlinear transformation of the feature vectors. The transform domain is called "similarity space", where associated distance between feature vectors is trainable. As a transformation scheme Artificial Neural Network is employed. Simulated annealing is used to optimize a predefined cost function. Experiments on the Brodatz Album, indicate better performance in the similarity space compared to the original feature space.

3. TRANSFORMATION FROM FEATURE SPACE TO SIMILARITY SPACE

We search a space called "similarity space" where the selected distance measure such as Euclidean distance, between patterns can be adjusted to distinguish patterns from different classes and to assign visually similar patterns into the same class. Therefore, in such a space between class variances should be large enough whereas class variances should be as small as possible. As expected, mapping from the original feature space to similarity space is highly non-linear, subjective and task dependent.

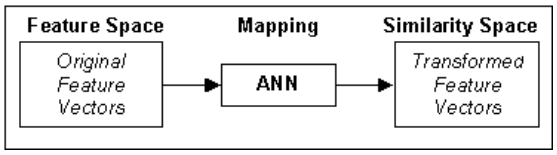

Figure 2. Block diagram of transformation

The nonlinear transformation is accomplished by the use of neural networks as indicated in Figure 2.

A typical ANN architecture, which can be used for such transformation, is shown in Figure 3. Since the outputs of ANN are bounded, i.e. between 0 to 1, similarity space is also bounded.

The number of input neuron is taken equal to the dimension of the original feature space. On the other hand, determining the number of hidden neuron and also output neuron (i.e. dimension of the similarity space) is not usually straightforward. The goal is to use as few neurons as possible for each layer. Note that reducing the number of output layer reveals to reducing the number of dimensions in the similarity space. Also, if the number of output neuron is chosen less than the number of input neuron, then the transformation works as a nonlinear dimension reduction system.

After selecting the number of hidden and output neurons, one can determine the similarity space algorithmically. This time standard backpropagation algorithm can be used for the training since the input versus output is known. Note that, Backpropagation algorithm minimizes the mean square error (MSE) between the generalized and the actual outputs. However the magnitude of the error does not clearly indicate how successful the ANN is separating the classes to be identified [10].

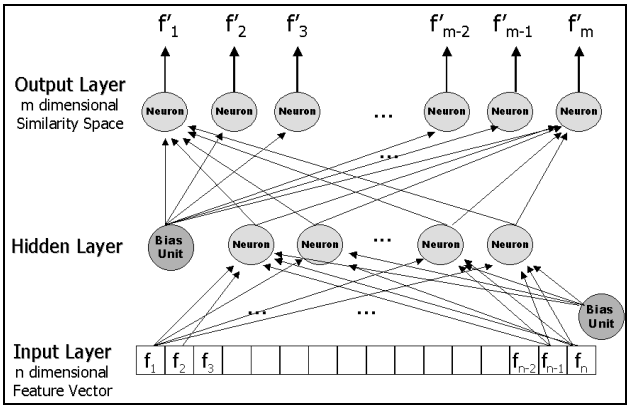

Figure 3. ANN architecture used for transformation.

Due to aforementioned problems, training should be handled as an unsupervised way. First a cost function must be defined in order to measure the goodness of the similarity space. Second, an algorithm that searches the optimal parameters of the cost function should be employed.

4. ANN TRAINING AS A GLOBAL OPTIMIZATION PROBLEM

Although various unsupervised learning algorithms such as hill climbing, genetic algorithms, and etc. are available to optimize the cost function, we cast the training of the ANN as a problem of global optimization and use the simulated annealing method.

Mathematically speaking, given a set S of feasible solutions and real valued cost function $g : S \to R$, global optimization may be formulated as the search for $s \in S$ such that $g(s) \le g(s') \ \forall s' \in S$. For neural networks, S is the space of connection weight vectors including bias terms and g is the cost function. Simulated annealing requires the notion of a neighborhood structure over S, where the neighborhood $N(s_c)$ of the current solution $s_c \in S$ is the set of new solutions s' that can be generated

from s_c. Typically, $N(s_c)$ consists of slight perturbations of s_c, e.g. a weight vector can be perturbed by adding a random vector in $[-e,+e]^d$, where d is the number of connection weights. Simulated annealing described in Figure 4 is an iterative algorithm that allows escape from local minima in the error surface by probabilistically accepting disimprovements, or "up-hill moves". Although downhill moves are allowed anytime, uphill moves are more likely during the beginning of the process when the temperature is high, and they become less likely at the end as the temperature becomes lower.

$s_c \leftarrow$ random solution in S
$T \leftarrow T_0$
Repeat I times
  Repeat J times
  Begin
    Choose $s' \leftarrow$ a random element from $N(s_c)$
    $\Delta = g(s') - g(s_c)$
    if ($\Delta \le 0$) then $s_c \leftarrow s'$
    else $s_c \leftarrow s'$ with probability $e^{-(\Delta / T)}$
  End
  $T \leftarrow T * k$ where $0 < k < 1$

Figure 4.Pseudo code for simulated annealing algorithm

5. TEXTURE IMAGE RETRIEVAL

Although, the nonlinear transformation schema proposed in the previous section can be used for any image retrieval and search problem, in this study we suffice to apply it to the texture image retrieval problem due to its popularity in the context of similarity based retrieval systems. At this point a cost function is needed to design an optimal transformation, which improves separability of the similarity space. There are many ways of defining the cost function depending on the nature of the problem and the characteristics of the images in the database. One may define the cost function, which considers the full ranking, i.e. for each query, the user can determine the order of retrieved subimages. In this study we only cluster the similar textures as close as possible and separate distinct classes as much as possible.

6. COST FUNCTION

Let, **A** represents a distance from a class center to the farthest point within the same class and **B** represents a distance from the class center to its nearest neighboring class center (See figure 5). Our aim is to minimize **A**, while maximizing **B** for each class. Therefore, a cost function can be defined as

$$g(\overline{w}) = E(\mathbf{B}-\mathbf{A}) \approx \frac{1}{\mathrm{N}} \sum_{\forall \mathrm{Class}} (\mathbf{B}-\mathbf{A}),$$

where N represents the number of classes, and $\overline{w}$ is a weight vector of the ANN.

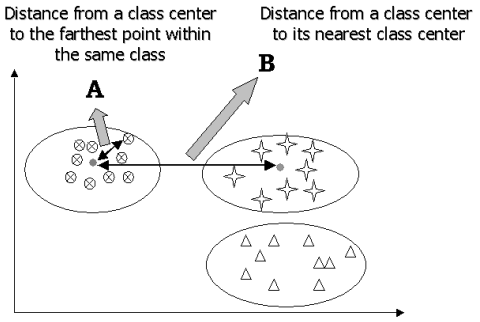

Figure 5. Sample similarity space in 2-d.

In some cases, an unbalanced increase in $\mathbf{B}-\mathbf{A}$ for one class, can significantly increases the cost function with the price of disturbing the separability of the rest of the classes. This situation may maximize the cost function, yet being very poor in well separated clustering. In order overcome this problem, each $\mathbf{B}-\mathbf{A}$ is scaled between 0 to 1 by the sigmoid function as shown in Figure 6.

$$\mathrm{Sgm(X)} = \frac{1}{1+\exp(-c*X)}$$

where c is a constant, which depends on the images in the database and dimension of the similarity space. Cost function can be redefined as follows

$$g(\overline{w}) = E(\mathrm{Sgm}(\mathbf{B}-\mathbf{A})) \approx \frac{1}{\mathrm{N}} \sum_{\forall \mathrm{Class}} \mathrm{Sgm}(\mathbf{B}-\mathbf{A})$$

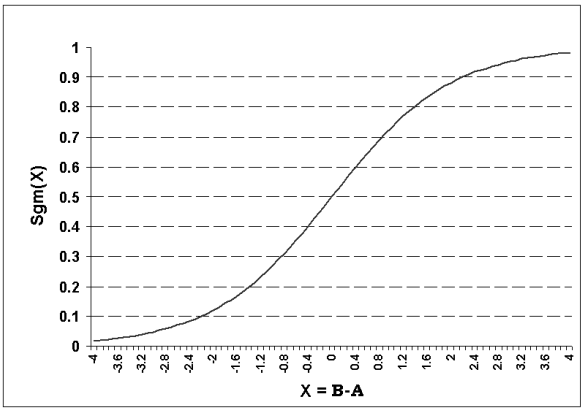

Figure 6. Sigmoid function, where c=1.0

7. EXPERIMENTS

Most of the studies in the literature use Brodatz Album, which consists of 112 images (112 Classes) of size 512x512 and 256 gray values. After dividing each image into 16 nonoverlapping subimages, total of 1792 images are obtained. The performance of the proposed descriptor is measured in terms of the average retrieval rate, which is defined as the average percentage number of patterns belonging to the same image as the query pattern in top 15

matches (self matches are excluded) [7],[11]. In another words, for each subimage the most similar 15 subimages are searched among 1791 subimages.

It is expected that, query and the retrieved subimages are the parts of the same image. The effectiveness of the representation is measured by the average retrieval rate.

One of the most popular feature extraction method is the Gabor Filters, reported by Manjunath and Ma [7] use Gabor filter dictionary, which contains four scales and six orientations. For each orientation and scales of Gabor Filter the relevant texture primitives are captured. Second order statistics of the Gabor Filter (4 scales * 6 orientation = 24 filter) responses of a given texture are used as a texture descriptor. Thus, an image is represented by 48 real numbers.

We construct an ANN, which has 48 input, 24 hidden, and 12 output neurons, which corresponds to reducing the 48 dimensional space into a space of 12 dimension. The number of hidden and output neuron is chosen by trial and error. Inputs are the normalized Gabor Filter descriptor values. In order to train the ANN, for each image, we select 6 subimages randomly and run the simulated annealing codes in order to maximize the cost function. In order to avoid wasting computation time, we select T_0 such that average accepting uphill moves would be 0.8 at the beginning. Also I=1000, J=5000, k=0.994 and e=0.002 is selected. Scaling parameter c is chosen as 0.5.

The experiment was repeated 10 times. Figure 7 summarizes the results. Considering the average retrieval rates, it is clear that similarity space is more effectively represents the images comparing the normalized original feature space and due to the dimension reduction, *i.e.* $R^{48} \rightarrow [0,1]^{12}$, retrieving cost (space + computational time) is saved.

8. CONCLUSION

This study attacks the separability problem of feature space, designed for content-based image search and retrieval systems. In particular, the proposed method concentrates on the similarity metrics for texture descriptors. For this purpose, a nonlinear transformation scheme maps the original image into similarity space, where the patterns are better separated for distinct textures and closely clustered for similar textures. The nonlinear transformation is optimal with respect to a predefined cost function. Furthermore, it reduces the dimension of the features, in similarity space. Results indicate that the similarity space is more successful than the original space in retrieving the similar texture images.

Note that the nonlinear transformation, proposed in this study, is independent from the nature of the problem and can be applied to wide range of pattern recognition problems, such as face recognition, fingerprint recognition etc. By changing the cost function, various types of problem can be handled. The dimension reduction in the similarity space needs to be explored further.

TEST #	Training		Test		Train+Test	
	Original Space	Similarity Space	Original Space	Similarity Space	Original Space	Similarity Space
1	72.26	87.60	72.78	75.77	72.75	80.93
2	73.09	88.92	71.67	74.71	72.75	81.40
3	71.89	89.65	73.64	76.36	72.75	82.03
4	71.89	89.76	73.01	75.14	72.75	81.45
5	72.99	90.37	71.94	74.19	72.75	81.20
6	72.50	89.84	71.80	76.05	72.75	81.88
7	72.11	88.81	72.21	73.60	72.75	80.42
8	72.66	88.95	72.93	74.65	72.75	80.90
9	72.69	91.82	71.44	76.59	72.75	83.30
10	70.97	89.64	73.47	75.78	72.75	81.56
Avg.	72.30	89.54	**72.49**	**75.28**	**72.75**	81.51

Test : For each subimage, which is not used in the training phase, retrieve the most similar 9 subimage within 1120 (1120=10*112) image

Train+Test : For each image retrieve the most similar 15 image within 1792 image

Figure 7. Experimental results on Brodatz Album using Gabor Filter descriptor are shown. For each experiment, average retrieval rates are calculated.

9. REFERENCES

[1] Simone Santini, and Ramesh Jain, "Similarity Matching", Lecture Notes in Computer Science, p.571, 1996.

[2] Anil K. Jain, P.W. Duin, and Jianchang Mao, "Statistical Pattern Recognition: A Review",IEEE-PAMI,Vol.22,Jan. 2000

[3] Yong Rui and Thomas S. Huang, "Image Retrieval: Current Techniques, Promising Directions and Open Issues", Journal of Visual Com. and Image Rep., Vol.10, No.4, April 1999

[4] Simone Santini, and Ramesh Jain, "Similarity Measures", IEEE, PAMI, Vol.29, No.9, Sep. 1999

[5] Guodong Guo, Hong-Jiang Zhang, and Stan Z. Li, "Distance-From-Boundary as a Metric For Texture Image Retrieval", Proc. of ICASSP, May 2001

[6] David McG. Squire, "Learning a similarity-based distance measure for image database organization from human partitionings of an image set", 4. IEEE WACV, October 1998

[7] W.Y.Ma and B.S. Manjunath, "Texture-based retrieval from for browsing and retrieval of image data," IEEE-PAMI, Vol. 18, No. 8, Aug. 1996.

[8] R. O. Duda and P. E. Hart, "Pattern Classification and Scene Analysis", Wiley-Interscience, New York, 1973

[9] Guodong Guo, Stan Z. Li, and Kap Luk Chan, "Learning Similarity for Texture Retrieval", Proc. of the European Conf. On Computer Vision, June 2000.

[10] Robert Hochman and et.al., "Evolutionary Neural Networks: A Robust Approach to Software Reliability Problems", Proc.of the 8. Int. Symp. On Soft. Rel. Eng., Nov. 1997

[11] A. Carkacioglu and Fatos Y.Vural, "SASI: A New Texture Descriptor for Image Retrieval", ICIP October 2001, Greece