ESANN 2012 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 25-27 April 2012, i6doc.com publ., ISBN 978-2-87419-049-0. Available from http://www.i6doc.com/en/livre/?GCOI=28001100967420.

Texture Classification Based on Symbolic Data Analysis

Carlos W.D. de Almeida[1] and Renata M.C.R. de Souza[1] and Ana Lúcia B. Candeias[2]

1- Federal University of Pernambuco, Center of Informatics (CIn)
Recife, Brazil. E-mail: cwda@cin.ufpe.br and rmcrs@cin.ufpe.br

2- Federal University of Pernambuco, Department of Cartographic Engineering
Recife, Brazil. E-mail: analucia@ufpe.br

Abstract. This article presents a hybrid approach for texture-based image classification using the gray-level co-occurrence matrices (GLCM) and a new Fuzzy Kohonen Clustering Network for Symbolic Interval Data (IFKCN). The GLCM matrices extracted from an image database are processed to create the training data set using IFKCN algorithm. The IFKCN organizes and extracts prototypes from processed GLCM matrices. The experimental results demonstrate that the proposed method is encouraging with an average successful rate of 97.39%.

1 Introduction

The perception of texture is believed to play an important role in the human visual system for recognition and interpretation. Several authors have worked in finding descriptors and features for texture identification. Existing features and techniques for modeling textures include Gabor transforms [1], affine adaption [2] and some invariant feature descriptors such as Zernike moments [4]. Haralick [5] suggested the use of gray-level co-occurrence matrices (GLCM) to extract texture features from an image.

Efficient organization and indexing of objects are steps of paramount importance in texture classification. Cluster analysis is the organization of a collection of patterns into clusters based on similarity In clustering analysis, the items to be grouped are usually represented as a vector of quantitative or qualitative measurements where each column represents a variable. In practice, however, this model is too restrictive to represent complex data since to take into account variability and/or uncertainty inherent to the data, variables must assume sets of categories or intervals, possibly even with frequencies or weights. These kinds of data have been mainly studied in Symbolic Data Analysis (SDA) [6]. The aim of SDA is to provide suitable methods for managing aggregated data described by multi-valued variables, where the cells of the data table contain sets of categories, intervals, or weight distributions [6].

The focus of the paper is to show the effectiveness of a hybrid approach in handling and classification from a texture database, which is characterized by: (*i*) a new texture image descriptor based on Gray Level Co-occurrence Matrix (GLCM) [5] and Symbolic Data Analysis (SDA) [6] (*ii*) a new method Fuzzy Kohonen Clustering Network for Symbolic Interval Data (IFKCN).

2 The Proposed Approach for Texture Classification

Four main modules are the building blocks of the proposed method: (*i*) Texture Descriptor, (*ii*) Symbolic Data Analysis, (*iii*) Clustering, and (*iv*) Classification. Our ap-

ESANN 2012 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 25-27 April 2012, i6doc.com publ., ISBN 978-2-87419-049-0. Available from http://www.i6doc.com/en/livre/?GCOI=28001100967420.

proach has a learning stage where the texture descriptor module receives images from a database and applies the GLCM method.

Gray level co-occurrence matrix (GLCM) [5] describes the relative frequencies with which two pixels separated by a distance d under a specified angle occur on the image. In this work, we will use a set of offsets sweep through 180 degrees (ie 0, 45, 90, and 135 degrees) and $d = 1$, obtaining four GLCM matrices. These angles are essential to obtain a feature vector invariant to rotation. Computed these matrices, we get an enormous amount of information on the texture. It is totally impractical to store such information. To reduce the dimensionality of the data, we extract symbolic data and obtain a reduced representation of data. Then, the GLCM matrices are pre-processed in order to obtain input data for the clustering or classification modules. In the Clustering module, the IFKCN algorithm organises and extracts prototypes from the processed matrices, which ends the learning stage. The classification module receives a pre-processed query image and compares it with the prototypes (representations of clusters) obtained in the clustering module. The final result is a list of images belonging to a few number of clusters considered to be the nearest to the user's query.

2.1 Texture Descriptor Module

A Gray level co-occurrence matrix [5] $P(i,j)$ describes the relative frequencies with which two pixels separated by a distance d under a specified angle Θ occur on the image, one with graytone i and the other with graytone j. Such matrices of graytone spatial dependence frequencies depend on the angular relationship between neighboring pixels and on the other distance between them. The GLCM can be defined as:

$$P^{\Theta}(i,j) = Pr(I(p_1) = i \wedge I(p_2) = j \wedge \|p_1 - p_2\| = d) \tag{1}$$

where P is the probability, p_1 and p_2 are positions in the gray scale image I. In this work, we will use a set of offsets sweep through 180 degrees (ie 0, 45, 90, and 135 degrees), obtaining four GLCM matrices. The algorithm proceeds as follows:

TEXTURE DESCRIPTOR ALGORITHM SCHEMA

1: Compute for each imagem Img_s
2: **for** $s = 1$ to S **do**
3: Compute the GLCM (P_s^{Θ}) from image Img_s using the angles: $\Theta = \{0^\circ, 45^\circ, 90^\circ, 135^\circ\}$.
4: **end for**

2.2 SDA Module

This module transforms the GLCM matrices P^{Θ} in a single GLCM matrix composed of symbolic interval data PI. Then, we re-extract symbolic interval data from the matrix PI in order to obtain a vector of interval data X, in turn, will constitute the input data for the clustering module. For each position in the gray scale matrix $P^{\Theta}(i,j)$, we extract the minimum and maximum probabilities over all angles Θ. This way, we create a new variable $PI(i,j)$ in order to capture the variability of the probability over different values of Θ. Problems with choosing $[min; max]$ can arise when these extreme

ESANN 2012 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 25-27 April 2012, i6doc.com publ., ISBN 978-2-87419-049-0. Available from http://www.i6doc.com/en/livre/?GCOI=28001100967420.

values are in fact outliers. The extraction of the minimum and maximum probabilities is computed as:

$$PI(i,j) = \begin{cases} \textbf{if } max(P^{\Theta}(i,j)) == 0 \textbf{ then} \\ \quad PI(i,j) = [0;0] \\ \textbf{else} \\ \quad PI(i,j) = [min(P^{\Theta}(i,j)) > 0; max(P^{\Theta}(i,j))] \\ \textbf{end if} \end{cases} \tag{2}$$

SDA Module assumes that the interval matrix PI is composed by n items or individuals (rows) that are described by p interval-type variables (columns). For a given number n of interval data $x_i = [a_i, b_i]$ $(i = 1, ..., n)$, the extraction of a vector (X) from a matrix (PI) is defined by the following equation:

$$x_i = \begin{cases} \textbf{if } max\{b_{ij} \mid j = 1, ..., p\} == 0 \textbf{ then} \\ \quad x_i = [0,0] \\ \textbf{else} \\ \quad x_i = [min\{a_{ij} \mid j = 1, ..., p\} > 0, max\{b_{ij} \mid j = 1, ..., p\}] \\ \textbf{end if} \end{cases} \tag{3}$$

SDA ALGORITHM SCHEMA

```
Given a GLCM matrix P^Θ, do
for i = 1 to I do
    for j = 1 to J do
        Compute the PI(i,j) finding the minimum and maximum probabilities over all
        angles P^Θ(i,j) (Θ = 0°, 45°, 90°, 135°), using the Equation 2.
    end for
end for
for i = 1 to I do
    Find the minimum and maximum boundary in the PI matrix for the jth variable (j = 1
    , ..., p), using the Equation 3.
end for
```

2.3 Clustering Module

The classical Fuzzy Kohonen Clustering Network (FKCN) [7] is a batch clustering algorithm that combines the ideas of fuzzy membership values for learning rates, the parallelism of the Fuzzy C-Means (FCM), and the structure and self-organizing update rules of the Kohonen Clustering Network (KCN) [8].

Let $\Omega = \{1, ..., n\}$ be a set of n objects indexed by k and described by p symbolic interval variables $\{\mathbf{x}_k^1, ..., \mathbf{x}_k^p\}$ indexed by j. A symbolic interval variable X [6] is a correspondence defined from Ω in $\mathbb{R}$ such that for each $k \in \Omega$, $X(k) = [a,b] \in \Im$, where $\Im = \{[a,b] : a, b \in \mathbb{R}, a \leqslant b\}$ is the set of closed intervals defined from $\mathbb{R}$. Each pattern k is represented as a vector of intervals $\mathbf{x}_k = (x_k^1, ..., x_k^p)$, where $x_k^j = [a_k^j, b_k^j] \in \Im$. This method looks for a partition of Ω into c clusters $\{P_1, ..., P_c\}$ indexed by i. A prototype $\mathbf{g}_i$ of cluster P_i be also represented as a vector of intervals $\mathbf{g}_i = (g_i^1, ..., g_i^p)$, where $g_i^j = [\alpha_i^j, \beta_i^j] \in \Im$. For a partition of Ω in c clusters $\{P_1, ..., P_c\}$ and a corresponding set of prototypes $\{g_1, ..., g_c\}$ such that an adequacy criterion J

ESANN 2012 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 25-27 April 2012, i6doc.com publ., ISBN 978-2-87419-049-0. Available from http://www.i6doc.com/en/livre/?GCOI=28001100967420.

measuring the fitting between the clusters and their prototypes is locally minimized. This criterion J is based on a squared Euclidean distance and is defined as:

$$J = \sum_{i=1}^{c}\sum_{k=1}^{n}(u_{ik})^m \phi(\mathbf{x}_k - \mathbf{g}_i) = \sum_{i=1}^{c}\sum_{k=1}^{n}(u_{ik})^m \sum_{j=1}^{p}\left[(a_k^j - \alpha_i^j)^2 + (b_k^j - \beta_i^j)^2\right] \quad (4)$$

The algorithm sets an initial partition and convergence when the criterion J reaches a stationary value representing a local minimum. IFKCN controls learning rates and size of the neighborhoods changing the value of weight exponents with time as follows:

$$m_t = m_0 - t * \frac{m_0 - 1}{t_{max}} \quad (5)$$

where m_0 is the initial value of weight exponent greater than one and t_{max} is the iteration limit. The membership degree u_{ik} $(k = 1, ..., n)$ of each pattern k in each cluster P_i, minimizing the clustering criterion J under $u_{ik} \geqslant 0$ and $\sum_{i=1}^{c} u_{ik} = 1$, is updated according to the following expression:

$$u_{ik,t} = \left[\sum_{h=1}^{c}\left(\frac{\sum_{j=1}^{p}\left[(a_k^j - \alpha_{i,t-1}^j)^2 + (b_k^j - \beta_{i,t-1}^j)^2\right]}{\sum_{j=1}^{p}\left[(a_k^j - \alpha_{h,t-1}^j)^2 + (b_k^j - \beta_{h,t-1}^j)^2\right]}\right)^{\frac{2}{m_t-1}}\right]^{-1} \quad \texttt{for } (i = 1, ..., c) \quad (6)$$

The fuzzified learning rate θ is defined as:

$$\theta_{ik,t} = (u_{ik,t})^{m_t} \quad (7)$$

The prototype $\mathbf{g}_i = (g_i^1, ..., g_i^p)$ of class P_i $(i = 1, ..., c)$, which minimizes the clustering criterion J, has the bounds of the interval $g_i^j = [\alpha_i^j, \beta_i^j]$ $(j = 1, ..., p)$ updated according to the following expression:

$$\alpha_{i,t}^j = \alpha_{i,t-1}^j + \frac{\sum_{k=1}^{n}(\theta_{ik,t})(a_k^j - \alpha_{k,t-1}^j)}{\sum_{k=1}^{n}(\theta_{ik,t})} \quad \texttt{and} \quad \beta_{i,t}^j = \beta_{i,t-1}^j + \frac{\sum_{k=1}^{n}(\theta_{ik,t})(b_k^j - \beta_{k,t-1}^j)}{\sum_{k=1}^{n}(\theta_{ik,t})} \quad (8)$$

IFKCN-FD ALGORITHM SCHEMA

```
Fix the number of clusters c. Select ε > 0. Set t_max and m_0 > 1.
Initialize the prototype vector g_i and membership degree u_ik (k = 1,...,n) (i = 1,...,c) of
  the pattern k belonging to cluster P_i (i = 1,...,c) such that u_ik ≥ 0 and Σ_{i=1}^c u_ik = 1.
for t = 1 to t_max do
    for k = 1 to n do
        Calculate the learning rate (Equation 5), the fuzzy membership (Equation 6) and the
         fuzzified learning rate (Equation 7).
        Update all cluster centroids interval (Equation 8).
        Compute E_t = ||v_t − v_{t−1}||^2.
        if E_t ≤ ε then
            Stop.
        else
            next t.
        end if
    end for
end for
```

ESANN 2012 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 25-27 April 2012, i6doc.com publ., ISBN 978-2-87419-049-0. Available from http://www.i6doc.com/en/livre/?GCOI=28001100967420.

2.4 Classification Module

This module receives a pre-processed image query and return to the user the classe considered to be the most similar to his/her query. After the training phase, it is possible to use the IFKCN to construct a classifier in which each prototype represents one class type. If labelled data are available, this information can be used to assign each prototype a label. The IFKCN is labelled based on votes between the labels according input data vectors and only uses the one which is most frequent. Finally, class label of each original data vector is the label of the corresponding *Best Matching Units* (BMUs) [8].

A BMU is the winning node of the IFKCN, that is, the prototype more similar to the query. The search procedure considers at least the first BMU. It is possible that more than one of the BMUs have to be considered in order to classify.

CLASSIFICATION ALGORITHM SCHEMA

1: Compare $\mathbf{x}_j = (x_j^1, \ldots, x_j^p)$ with all the node vectors $(\mathbf{m}_d)$ $(d = 1, \ldots, M)$ of the IFKCN using the Euclidean distance.
2: Obtain an ordered list of all BMUs.
3: Determine the first BMUs in the IFKCN.
4: Extract the label associated with the selected BMU(s).
5: **if** the label data are not available **then**
6: Determine the next BMU in the IFKCN and repeat the step 4.
7: **end if**

3 Experimental Results

In order to assess the performance of the proposed hybrid approach, experiments with the Brodatz database [9], were carried out. In the experiments, each Brodatz texture constitutes a separate class. Each texture have 640×640 pixels, with 8 bits/pixel. The texture was partitioned in 32×32 subimages, taking 400 non-overlapping texture samples in each class. The samples were separated in two disjoint sets, one for training and the other for testing the classifier. This corpus is the same used by Li et al [10].

The evaluation is based by the *correct classification rate* (CCR) [10]. These measurements are estimated in the framework of a Monte Carlo experience with 30 random partitions of the training and test sets. This approach is compared with several classifiers in Li et al [10].

In this experiment, the effect of the training set size in the classifier accuracy was then assessed. A small fraction (from 1.25% to 10%) of the 400 subimages are used in training the classifiers, while the rest are used for testing. The training samples in each class was set to 1.25% (5 samples), 2.5% (10 samples), 3.75% (15 samples), 5% (20 samples), 6.25% (25 samples), 7.5% (30 samples), 8.75% (35 samples) and 10% (40 samples). For the proposed method, the number of clusters c was set to 30 (the same number of classes in database). Figure 1 summarizes the results of the proposed method, along with the results [10] for the single and fused SVM classifier, the Bayes classifiers using Bayes distance and Mahalanobis distance, and the LVQ classifier. These measurements are estimated in the framework of a Monte Carlo experience with 30 random partitions of the training and test sets. From this figure, we can observe

ESANN 2012 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 25-27 April 2012, i6doc.com publ., ISBN 978-2-87419-049-0. Available from http://www.i6doc.com/en/livre/?GCOI=28001100967420.

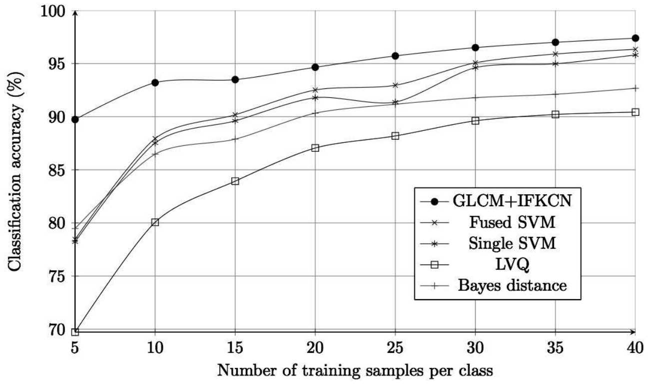

Fig. 1: Texture classification CCR rates

the superiority of the proposed classifier in comparision with the other ones.

4 Conclusions

In this paper, an approach for texture-based image classification using the gray-level co-occurrence matrices (GLCM) and FKCN for Symbolic Interval Data (IFKCN) methods is presented. To show the usefulness of the proposed methodology, an application with a benchmark data set was considered. The proposed classifier was compared with several classifiers according to the error rate of classification. The results demonstrated that our method outperformed the other ones.

References

[1] Anil K. Jain and Farshid Farrokhnia. Unsupervised texture segmentation using gabor filters. *Pattern Recognition*, 24(12):1167–1186, 1991.

[2] J. Zhang, M. Marsza, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *International Journal of Computer Vision*, 73(2):213–238, 2007.

[3] Dimitrios Charalampidis and Takis Kasparis. Wavelet-based rotational invariant roughness features for texture classification and segmentation. *IEEE Trans. Image Process.*, 11:825–837, 2002.

[4] A. Khotanzad and Y.H. Hong. Invariant image recognition by zernike moments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(5):489–497, 1990.

[5] Robert M. Haralick, K. Shanmugam, and Its'Hak Dinstein. Textural features for image classification. *IEEE Transactions on In Systems, Man and Cybernetics*, 3(6):610–621, 1973.

[6] Hans-Hermann Bock and Edwin Diday. *Analysis of Symbolic Data: Exploratory Methods for Extracting Statistical Information from Complex Data*. Springer, 2000.

[7] James C. Bezdek, Eric Chen-Kuo Tsao, and Nikhil R. Pal. Fuzzy kohonen clustering networks. In *Proc. of the First IEEE Conference on Fuzzy Systems*, San Diego, USA, 1992.

[8] T. Kohonen. *Self-Organizing Maps*. Springer-Verlag, 3rd edition edition, 2001.

[9] P. Brodatz. *Textures: A Photographic Album for Artists and Designers*. Dover Publications, 1966.

[10] Shutao Li, James T. Kwok, Hailong Zhu, and Yaonan Wang. Texture classification using the support vector machines. *Pattern Recognition*, 36(12):2883–2893, 2003.