

Texture Feature Extraction and Classification by Combining Statistical and Neural Based Technique for Efficient CBIR

Siddhivinayak Kulkarni and Pradnya Kulkarni

School of Science, Information Technology and Engineering,
University of Ballarat,
Mount Helen, Victoria-3353, Australia
{S.Kulkarni, P.Kulkarni}@ballarat.edu.au

Abstract. This paper presents a technique based on statistical and neural feature extractor, classifier and retrieval for real world texture images. The paper is presented into two stages, texture image pre-processing includes downloading images, normalizing into specific rows and columns, forming non-overlapping windows and extracting statistical features. Co-occurrence based statistical technique is used for extracting four prominent texture features from an image. Stage two includes, feeding of these parameters to Multi-Layer Perceptron (MLP) as input and output. Hidden layer output was treated as characteristics of the patterns and fed to classifier to classify into six different classes. Graphical user interface was designed to pose a query of texture pattern and retrieval results are shown.

Keywords: Texture images, Multi Layer Perceptron, Classifier, image retrieval.

1 Introduction

Image databases are becoming very popular due to the large amount of images that are generated by various applications and due to the advancement in storage devices, image compression, scanning, networking etc. Retrieving the specific images based on their content has become an important research area for the last decade. These images are retrieved based on their content such as global colour, texture, shape as low level features.

An image database may contain thousands of textured images. The main problem user that user is facing of locating the images having similar texture pattern in the given query. More specifically, this problem is considered in two main parts: a. finding the images having the similar texture given in query and b. specifying a texture in query.

A good image retrieval system dealing with textures must provide solutions to both problems. In posing a query in terms of texture, it is not realistic to expect the user to draw a texture that he or she wants to retrieve. Therefore, all the textures which are extracted from the database are classified into different clusters and to pose a query in

terms of these textures. While retrieving the texture patterns, those are similar to the query, only that particular cluster has been considered. The similarity is calculated based on the query pattern and all the texture images which belong to the same class. This technique reduces the search only for that particular class and effectively reduces the searching time. This similarity is calculated based on weighted Euclidean distance and presented to the user. This is the effective way to express the texture query and getting the result based on that particular query.

In this paper, gray level single textured images are used to extract the texture features and construct a feature vector by using co-occurrence matrix for each textured image. These statistical based extracted features are used an input and output to the Multi Layer Perceptron and characertics was taken from hidden layer, which is then fed to classifier to classify these features into six different classes for efficient retrieval. The results obtained are very promising and some of the results are illustrated in this paper.

The rest of the paper is organised as follows: Section 2 gives the brief idea regarding analysis of prominent texture feature extraction techniques, Section 3 details the proposed research methodology for extracting texture features based on statistical-neural technique, Section 4 describes experimental results and analysis and the paper is finally concluded in Section 5.

2 Analysis of Prominent Texture Feature Extraction Techniques

In texture feature representations such as pixel neighborhood [1], a simple texture feature can be constructed by comparing suitable properties of current pixel with the properties of neighboring pixels. But the disadvantage of this technique is that these features are not very accurate as the feature vectors entirely depend upon the center pixel. In tamura features [2], all the six texture properties are visually meaningful so this texture representation becomes attractive in image retrieval. These properties of texture are easy to recognize by human but elusive when to be described quantitatively by a machine.

Markov random fields are attractive because they yield local and parsimonious texture descriptions. But MRF model use individual pixels based measurements and are hence not easily flexible to change in image resolution [3], [4], [5]. With SAR models, there are major difficulties in selecting the size of the dependent pixel neighborhood and the appropriate window size in which texture is regarded homogenous. MRSAR model was developed to overcome this problem. But MRSAR is computationally a very expensive set of features [6]. The wold decomposition model avoids the actual decomposition of images and tolerates a variety of in-homogeneities in natural data, making it suitable for use in large collections of natural patterns [7]. The statistical properties such as mean and variance are extracted from the wavelet sub-bands as texture representations. To explorer the middle band characteristics, tree structured wavelet transform is used to improve the classification accuracy. The wavelet transform when combined with other techniques such as Kohonen map achieves better results. Gabor transform provides an attractive

approach [8], which is well suited to texture classification and database retrieval. These are far superior compared to co-occurrence features and less sensitive to noise. But there are possibilities for either mistreatment or adaptation to suit specific data. This technique is not generally applicable for segmentation or image analysis.

3 Research Methodology

This section deals with the retrieval of texture images in detail. To retrieve texture images, it is important to pre-process these images. This preprocessing includes the formation of a texture image database, extraction of texture features from these images, classifying these features in appropriate classes for retrieval. Research methodology is divided broadly into two sections: Section-I Texture images pre-processing and section-II Feature Extraction and Classification. Figure 1 shows the block diagram of the proposed technique.

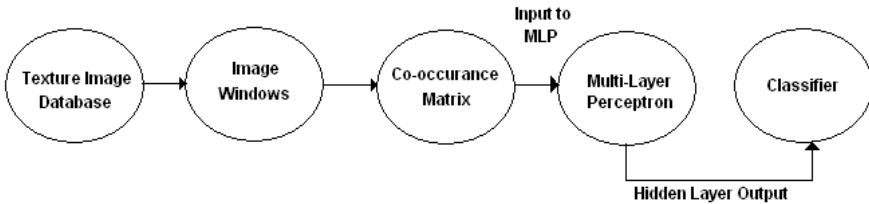


Fig. 1. Block diagram of proposed statistical-neural technique for texture image classification

3.1 Texture Image Database

Texture image database was created by downloading images from World Wide Web and consists of 500 texture images. Each image is re-arranged as 512 x 512 pixels. These images consist of textures of both statistical and structural natures. Structural textures are considered to be consists of texture primitives which are repeated systematically within the texture. In statistical texture usually no repetitive texture can be identified. These texture images mainly contain the texture of brick wall, wood, sky, grass, glass and fire. Each image is divided into 16 non-overlapping sub-images each 128×128 pixels in size, thus creating a database of (500×16) 8000 texture images.

3.2 Co-occurrence Matrix

Gray Level Co-occurrence Matrix (GLCM) is one of the texture feature extraction methods, estimates the image properties related to the second order statistics. It contains the information about grey levels (intensities) of pixels and their neighbours, at fixed distance and orientation. Each entry (i, j) in GLCM corresponds to the number of occurrences of the pair of gray levels i and j which are a distance d in direction θ apart in original image. Figure 2 shows the distances and orientations of pixel p for co-occurrence matrix.

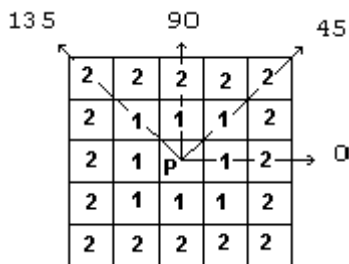


Fig. 2. Distances and directions of pixel p for co-occurrence matrix

In order to estimate the similarity between the different grey level co-occurrence matrices, Haralick [9] proposed 14 statistical features extracted from them. To reduce the computational complexity, only some of these features were selected. The description of four most relevant features that are widely used in the literature [10] [11] are given in Table 1.

Energy is a measure of textual uniformity of an image. Energy reaches its highest value when grey level distribution has either a constant or a periodic form. A homogenous image contains very few dominant grey tone transitions, and therefore the *P* matrix for this image will have fewer entries of larger magnitude resulting in larger value of energy feature. Also, energy feature have smaller value if *P* matrix contains large number of smaller entries.

Table 1. Features extracted from grey level co-occurrence matrix

Energy	$\sum_i \sum_j P_d^2(i, j)$
Entropy	$\sum_i \sum_j P_d(i, j) \log P_d(i, j)$
Contrast	$\sum_i \sum_j (i - j)^2 P_d(i, j)$
Inverse Difference Moment	$\sum_i \sum_j \frac{P_d(i, j)}{ i - j ^2}, i \neq j$

Entropy measures the disorder of an image and it achieves its largest value when all the elements in *P* matrix are equal. When the image is not textually uniform many GLCM elements have a very small value, which implies the entropy is very large. Entropy is inversely proportional to GLCM energy.

Contrast measures the difference between the highest and lowest values of a contiguous set of pixels. In other words, Contrast is a difference moment of the *P* and it measures the amount of local variations in an image. Low contrast images features low special frequencies.

Inverse Difference Moment (IDM) measures image homogeneity. This parameter achieves its largest value when the most of the occurrences in GLCM are concentrated near the main diagonal. IDM is inversely proportional to GLCM contrast.

3.3 MLP as Feature Extractor and Classifier

Multi Layer Perceptron is proposed to extract more features of these texture images. For input and output, features (energy, entropy, contrast and inverse difference moments) are used and output of the hidden layer is taken as the characteristics of texture patterns. This MLP learns the same pattern to provide typical features at the hidden layer. The network is trained using a supervised learning algorithm. The feature vector is fed to another MLP feature classifier.

This stage describes the capability of the Multi-layer Perceptron for the classification tasks and the implementation of a trainable MLP and presents a suitable kind of MLP structure for classification of texture data. In order to discuss classification capabilities of the MLP, it is necessary to define what “classification” means in the ANN context. Here it is defined in the broadest possible sense as the process which gives an output when a pattern is presented to the input of the network.

An MLP texture feature classifier is used to classify the textures in appropriate classes. It has p inputs which are the same as the number of hidden units in an MLP feature extractor. The output of the hidden layer that was obtained from MLP was used as input to the classifier. There were 6 texture classes, so the number of outputs was 6.

4 Experimental Results and Analysis

The objective of these experiments is to illustrate that the proposed texture feature extraction provides a powerful tool to aid in image retrieval. The image data set used in these experiments was texture images downloaded from the Internet. The experiments were conducted separately for feature extraction using co-occurrence matrix, feed them to MLP and classifying these texture patterns. After segmenting the images into 16 sub-images, the four texture features based on co-occurrence matrix is extracted and applied to MLP.

4.1 Training of Multi-Layer Perceptron

The experiments were conducted in two stages, firstly the training of the MLP and secondly the training of the classifier. Before training of an MLP, different training parameters were used. The total number of images used for the experiments was 500. After segmenting the images into 16 non-overlapping sub-images, the first 12 sub-images were used for training and the last 4 sub-images were used for testing. The number of training and testing sub-images were 6000 (500 x 12) and 2000 (500 x 4). In the case of the MLP, the same patterns were applied to the output. The values of the learning rate (η) and momentum (α) were 0.9 and 0.8 respectively.

The auto-associator was trained different numbers of hidden units and iterations to improve feature extraction. It was very important to train the auto-associator properly so that the classification of these features became an easy task. Table 2 shows the effect of varying the hidden units and iterations on RMS error.

Table 2. Effect of varying the number of hidden units and iterations on RMS error

Hidden units	Iterations	RMS error
4	1000	0.00576
6	1000	0.00312
8	1000	0.00077
10	5000	0.00064
12	10000	0.00011

4.2 Training of the Classifier

The classifier was trained after obtaining the output from the hidden layer from the MLP. The hidden layer output was given as input to the classifier. The output of the hidden layer of an MLP depends upon the number of units in the hidden layer and the number of training pairs. The number of inputs to the classifier is the same as the number of hidden units used to train the MLP.

4.3 Classification Results for the Testing Set

Table 3 shows the results obtained by the feature classifier on the testing set. The experiments were conducted by altering the number of units in hidden layer and the number of iterations for training. The best classification rate (92.65%) was obtained with number of 16 hidden units and 7500 iterations. Table 3 shows that as the number of hidden units increased, the classification rate also increased. The number of testing pairs was 2000.

Table 3. Results of classification of texture features on testing set

Hidden units	Iterations	RMS error	Classification rate (2000)	Classification rate [%]
16	500	0.0393	1469	73.45
16	1000	0.0670	1503	75.15
16	5000	0.0094	1742	87.10
16	7500	0.0034	1853	92.65

4.4 Graphical User Interface for Texture Query

Graphical user interface was designed to pose the query in terms of texture feature of the image. The user can select the texture from the six textures used for experiments and submit the query.

The Figure 3 shows the results obtained after selecting the texture as “Brick.” The confidence factor for each image is the actual output obtained after classifying the extracted features on testing set. All the images of the brick texture are retrieved and top 14 images are shown in Figure 3. The first image appearing at the top left corner has the confidence factor 0.999962, which is highest among all other brick texture images. The confidence factor of the images goes on decreasing from left to right and from top to bottom.



Fig. 3. Graphical Results of the query: brick texture

The query was tested for all the textures and found satisfactory results. All the images for each particular query were retrieved.

5 Conclusion

For texture image retrieval, a novel technique for extracting the texture features from the images was investigated in this paper. Statistical Co-occurrence based texture features are used as input and output to the MLP and output of the hidden layer was treated as detailed characteristics of the texture patterns. These texture features are classified into six different prominent classes to retrieve texture images based on query. The performance of MLP was evaluated on real world texture images. The results obtained from the classifier showed that statistical-neural feature extractor to be a promising feature extractor using only single hidden layer. For the classifier, the highest classification rate was obtained as 92.65% which was significant. This

classification rate can be improved by varying number of hidden units in hidden layer and number of iterations. As texture patterns are classified into specific classes, it is efficient for image retrieval to compare the images in that particular class. Future research combines texture and shape features for retrieving images based on multi-modal features in an image.

References

1. Smith, J.: Integrated Spatial and Feature Image Systems: Retrieval, Analysis and Compression, PhD Dissertation, Columbia University (1997)
2. Tamura, H., Mori, S., Yamawaki, T.: Texture Features Corresponding to Visual Perception. *IEEE Transactions on Systems, Man and Cybernetics* 8(6), 460–473 (1978)
3. Gimelfarb, G., Jain, A.: On Retrieving Textured Images from an Image Database. *Journal of Pattern Recognition Society* 29(9), 1461–1483 (1996)
4. Andrey, P., Tarroux, P.: Unsupervised Segmentation of Markov Random Field Modeled Textured Images using Selectionist Relaxation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20, 252–262 (1998)
5. Chellappa, R., Chatterjee, S.: Classification of Textures using Gaussian Markov Random Fields. *IEEE Transactions on Acoustics, Speech and Signal Processing* 33(4), 959–963 (1985)
6. Mao, J., Jain, A.: Texture Classification and Segmentation using Multiresolution Simultaneous Autoregressive Models. *Journal of Pattern Recognition* 25(2), 173–188 (1992)
7. Rao, A., Lohse, G.: Towards a Texture Naming System: Identifying Relevant Dimensions of Texture. In: *Proceedings of IEEE Conference on Visualization, San Jose, USA*, pp. 220–227 (1993)
8. Daubechies, I.: The Wavelet Transform, Time-Frequency Localisation and Signal Analysis. *IEEE Transactions on Information Theory* 9(36), 961–1005 (1990)
9. Haralick, R., Shanmugam, K., Dinstein, I.: Textural Features for Image Classification. *IEEE Transactions on System, Man and Cybernetics* 6, 610–621 (1973)
10. Partio, M., Cramariuc, B., Gabbouj, M., Visa, A.: Rock Texture Retrieval using Gray Level Co-occurrence Matrix. In: *Nordic Signal Processing Symposium, Norway* (2002)
11. Gonzalez, R., Woods, R.: *Book on Digital Image Processing* (1993)