

# CONTENT-BASED TEXTURE IMAGE RETRIEVAL USING THE LEMPEL-ZIV-WELCH ALGORITHM

Leonardo Vidal Batista, Moab Mariz Meira

*Departamento de Informática - DI / UFPB*

*Cidade Universitária - 58.051-900*

*João Pessoa - PB - Brasil*

Nicomedes L. Cavalcanti Júnior

*Centro de Informática - CIn / UFPE*

*Av. Prof. Luiz Freire, s/n, Cidade Universitária - 50.740-540*

*Recife - PE - Brasil*

Keywords: Content-based Image Retrieval, Texture classification, Lempel-Ziv-Welch.

Abstract: This paper presents a method for content-based texture image retrieval using the Lempel-Ziv-Welch (LZW) compression algorithm. Each texture image in the database is processed by a global histogram equalization filter, and then an LZW dictionary is constructed for the filtered texture and stored in the database. The LZW dictionaries thus constructed comprise a statistical model to the texture. In the query stage, each texture sample to be searched is processed by the histogram equalization filter and successively encoded by the LZW algorithm in static mode, using the stored dictionaries. The system retrieves a ranked list of images, sorted according to the coding rate achieved with each stored dictionary. Empirical results with textures from the Brodatz album show that the method achieves retrieval accuracy close to 100%.

## 1 INTRODUCTION

Keyword annotation is the most traditional image retrieval paradigm. In this approach, the images are first annotated manually by keywords. They can then be retrieved by their corresponding annotations. However, there are three main difficulties with this approach, i.e., the large amount of manual effort required in developing the annotations, the differences in interpretation of image contents, and the inconsistency of the keyword assignments among different indexers (Faloutsos et al., 1993; Flickner et al., 1995). As the size of image repositories increases, the keyword annotation approach becomes infeasible. To overcome the difficulties of the keyword-based approach, an alternative mechanism, content-based image retrieval (CBIR) was proposed in the early 1990's. CBIR consists in using visual features, which are image primitives, such as color, texture, and shape features, as the image index. This approach has the advantage of automatic feature extraction.

The explosive growth of digital image technologies in the last years brings the necessity to investigate and develop new search tools to efficiently locate pictorial information. Conventional search tools generally allow exclusively textual queries. Most Internet search tools specifically designed for images

looks for textual relevant information about image content by analyzing the filename of the graphic files, 'META' tags and ALT attributes of the 'IMG' tags in the HTML code of the pages, terms near the images in the pages, nature and orientation of sites and providers, etc. Conventional image databases generally stores textual information along with the images, thus allowing textual queries. However, it is known that textual queries, notably in the scope of multimedia databases, tend to present a great number of irrelevant results while failing to present many relevant results. Many efforts have been made to develop CBIR tools, by extracting and analyzing pictorial features such as shapes, colors and textures (Zibreira, 2000).

CBIR has attracted great research attention, ranging from government (Jain, 1993; Jain et al., 1995) and industry (Bach et al., 1996; Dowe, 1993; Flickner et al., 1995), to universities (Huang et al., 1996; Ma and Manjunath, 1999; Mandal et al., 1997; Pentland et al., 1996; Smith and Chang, 1997). Even ISO/IEC has defined MPEG-7 (ISO/IEC et al., 1997a; ISO/IEC et al., 1997b; ISO/IEC et al., 1997c) to encompass a standard multimedia content description interface. Many CBIR systems, both commercial (Bach et al., 1996; Dowe, 1993; Faloutsos et al., 1993; Flickner et al., 1995) and academic have been developed recently.

In Flickner's and Addi's works (Flickner et al., 1995; Addis et al., 2002), tools which approach some of the relevant aspects of CBIR are presented. These tools were applied to construct a museum and gallery web applications respectively which allow search for artworks by selecting predominant colours from a palette or by sketching shapes on a canvas.

Texture is a fundamental attribute used by the human visual system and computer vision systems for segmentation, classification and interpretation of scenes (Porat and Zeevi, 1989). There has been a great interest in the development of texture-based pattern recognition methods in different areas, such as remote sensing (Dell'Acqua and Gamba, 2003; Augusteijn et al., 1995), image-based medical diagnosis (Southard and Southard, 1996), industrial automation (Kumar and Pang, 2002) and biometric recognition (Jain et al., 2004; He et al., 2004).

Although intuitively recognized by the human visual system, texture is not easy to characterize formally. The problem resides in the intrinsic difficulty to define what is most relevant to characterize texture, as the answer depends on subjective perceptual considerations and on particular applications. Therefore, texture feature extraction and modeling tends to be a difficult and application-driven task. A popular yet rather vague definition states that textures are spatial patterns formed by more or less accurate repetitions of some basic subpatterns (Baheerathan et al., 1999).

Modern lossless data compression algorithms have been applied to pattern recognition problems, due to their ability to construct accurate statistical models, in some cases with low computational requirements (Bell et al., 1990). A solid theoretical foundation for using LZ78 (Ziv and Lempel, 1978) and other dictionary-based compression algorithms (Bell et al., 1990) for pattern classification is well established (Ziv, 1988; Ziv and Merhav, 1993).

A potential problem associated with lossless dictionary-based compression algorithms for image retrieval is the fact that these methods search exact matches in the dictionary for strings in the message to be compressed. A precise dictionary constructed for a given texture class may present a poor performance when compressing a new sample from the same class, if this new sample presents gray-level deviations caused by digitization noise or illumination changes.

Degraded performance, even when gray-level deviations are subtle, indicates that the constructed model may not be able to adequately describe the new texture sample, and consequently classification accuracy may also degrade.

Two possible solutions to this problem are:

1. Adoption of a lossy dictionary-based compressor (Finamore and Leister, 1996), less sensitive to small, spurious gray-level deviations;

2. Reduction of these deviations by means of image processing techniques, prior to the use of a lossless dictionary-based compression algorithm.

Histogram equalization is a well-known non-linear operation that generates an approximately uniform distribution of gray-levels over the available range (Bovik, 2000). Histogram equalization tends to map to the same value multiple gray levels that have similar values, thus reducing the small gray level deviations that tends to cause mismatches in the searching stage of lossless dictionary-based compressors. The operation also decreases the probability that an image retrieval system discriminates texture classes by average gray level or variance, instead of by its textural properties (Randen and Husy, 1999). This allows a more precise evaluation of the capabilities of the method to discriminate texture attributes.

Image retrieval based on universal data compression models have several potential advantages over classical methods: since there is no feature selection, no information is discarded - the models describe the classes as a whole (Frank et al., 2000); no assumptions about the probability distributions of the classes are required; the adaptive model construction capability of compression algorithms offers a uniform way to work with different types of sources (Ojala et al., 2002); the similarity rule is very simple (Teahan and Harper, 2001).

This paper proposes a new content-based image retrieval method for texture images using histogram equalization and the Lempel-Ziv-Welch (LZW) lossless compression algorithm (Welch, 1984). The rest of this paper is organized as follows. Section 2 presents some fundamental concepts; section 3 presents the LZW algorithm; section 4 describes the proposed method; section 5 presents the empirical evaluation of the proposed search tool; and section 6 presents a discussion of the results and the concluding remarks.

## 2 ENTROPY AND MARKOV MODELS

Let  $S$  be a stationary discrete information source that generates messages over a finite alphabet  $A = \{a_1, a_2, \dots, a_M\}$ . The source chooses successive symbols from  $A$  according to some probability distribution that, in general, depends on preceding symbols. A generic message is modeled as a stationary stochastic process  $\mathbf{x} = \dots x_{-2}x_{-1}x_0x_1x_2\dots$ , with  $x_i \in A$ . Let  $\mathbf{x}^n = x_1x_2\dots x_n$  represent a message of length  $n$ . Since  $|A|=M$ , the source can generate  $M^n$  different messages of length  $n$ . Let  $x_i^n$ ,  $i = 1, 2, \dots, M^n$  denote the  $i^{\text{th}}$  of these messages, according to some sorting order, and assume that the

source follows a probability distribution  $P$ , so that message  $x_i^n$  is produced with probability  $P(\mathbf{x}_i^n)$ .

Let

$$G_n(P) = -\frac{1}{n} \sum_{i=1}^{M^n} P(\mathbf{x}_i^n) \log_2 P(\mathbf{x}_i^n) \quad (1)$$

$G_n(P)$  decreases monotonically with  $n$  (Shannon, 1948) and the entropy of the source is:

$$H(P) = \lim_{n \rightarrow \infty} G_n(P) \text{ bits/symbol.} \quad (2)$$

An alternative formulation for  $H(P)$  uses conditional probabilities. Let  $P(\mathbf{x}_i^{n-1}, a_j)$  be the probability of sequence  $\mathbf{x}_i^n = \mathbf{x}_i^{n-1} a_j$  ( $\mathbf{x}_i^{n-1}$  concatenated with  $x_n = a_j$ ) and let  $P(a_j | \mathbf{x}_i^{n-1}) = P(\mathbf{x}_i^{n-1}, a_j) P(\mathbf{x}_i^{n-1})$  be the probability of  $x_n = a_j$  conditioned on  $\mathbf{x}_i^{n-1}$ . The entropy of the  $n^{\text{th}}$  order approximation to  $H(P)$  (Shannon, 1948) is:

$$F_n(P) = - \sum_{i=1}^{M^n} \sum_{j=1}^M P(\mathbf{x}_i^{n-1}, a_j) \log_2 P(a_j | \mathbf{x}_i^{n-1}) \text{ bits/symbol.} \quad (3)$$

$F_n(P)$  decreases monotonically with  $n$  (Shannon, 1948) and the entropy of the source is:

$$H(P) = \lim_{n \rightarrow \infty} F_n(P) \text{ bits/symbol.} \quad (4)$$

Eq. 4 involves the estimation of probabilities conditioned on an infinite sequence of previous symbols. When finite memory is assumed the sources can be modeled by a Markov process of order  $n-1$ , so that  $P(a_j | \dots x_{n-2} x_{n-1}) = P(a_j | x_1 \dots x_{n-1})$ . In this case,  $H(P) = F_n(P)$ .

Define the *coding rate* of a coding scheme as the average number of bits per symbol the scheme uses to encode the source output. A *lossless compressor* is a uniquely decodable coding scheme whose goal is to achieve a coding rate as small as possible. The coding rate of any uniquely decodable coding scheme is always greater than or equal to the source entropy (Shannon, 1948). Optimum coding schemes have a coding rate equal to the theoretical lower bound  $H(P)$ , thus achieving maximum compression.

For Markov processes of order  $n-1$ , optimum encoding is reached if and only if symbol  $x_n = a_j$  occurring after  $\mathbf{x}_i^{n-1}$  is coded with  $-\log_2 P(a_j | \mathbf{x}_i^{n-1})$  bits (Bell et al., 1990; Shannon, 1948). However, it may be impossible to accurately estimate the conditional distribution  $P(\cdot | \mathbf{x}_i^{n-1})$  for large values of  $n$ , due to the exponential growth of the number of different contexts, which brings well-known problems, such as context dilution (Bell et al., 1990).

### 3 THE LZW ALGORITHM

Even though the source model  $P$  is generally unknown, it is possible to construct a coding scheme based upon some (possibly implicit) probabilistic model  $Q$  that approximates  $P$ . The better  $Q$  approximates  $P$ , the smaller the coding rate achieved by the coding scheme.

In order to achieve low coding rates, modern lossless compressors rely on the construction of sophisticated models that closely follows the true source model. *Statistical compressors* encode messages according to an estimated statistical model for the source. *Dictionary-based compressors* replace strings of symbols from the message to be encoded with indexes into a dictionary of strings, which is generally adaptively constructed during the encoding process. When *greedy parsing* is used, at each step the encoder searches the current dictionary for the longest string that matches the next sequence of symbols in the message, and replaces this sequence with the index of the longest matching string in the dictionary.

Dictionary-based compressors with greedy parsing, such as LZW, are highly popular because they combine computational efficiency with low coding rates. It has been proved that each dictionary-based compressor with greedy parsing has an equivalent statistical coder that achieves the same compression (Bell et al., 1990). In dictionary-based coding, the dictionary embeds an implicit statistical model for the source.

The initial LZW dictionary contains all possible strings of length one. The LZW algorithm finds the longest string, starting from the first symbol of the message, which is already present in the dictionary. This string is coded with the index for the matching string in the dictionary, and the string is extended with the next symbol in the message,  $x_i$ . The extended string is added to the dictionary and the process repeats, starting from  $x_i$  (Bell et al., 1990).

LZW achieves optimum asymptotic performance for Markov sources, in the sense that its coding rate tends to the entropy of the source as the length of the message to be coded tends to infinity (Savari, 1997). It means that LZW algorithm learns a progressively better model for the source during encoding, and a perfect model for the source is learned when an infinite message has been coded. In practice, since the messages to be compressed are finite, LZW only learns an approximate model for the source.

### 4 THE PROPOSED METHOD

The accurate models built by modern lossless compressors can be used to characterize texture. Any effi-

cient model-based lossless compressor could be used, but LZW algorithm was chosen due to its good compromise between coding efficiency and computational requirements (Bell et al., 1990).

#### 4.1 Model Learning and Storage

Consider a database containing  $N$  texture samples  $\mathbf{t}_i$ ,  $i = 1, 2, \dots, N$ . The samples are  $n \times n$  images extracted from histogram-equalized images. In the model learning stage, the LZW algorithm compresses sample  $\mathbf{t}_i$  following the horizontal scanning order shown in Figure 1.a, and the resulting dictionary  $H_i$  is stored in the database as a model for the horizontal structure of  $\mathbf{t}_i$ ,  $i = 1, 2, \dots, N$ . The LZW algorithm then compresses  $\mathbf{t}_i$  following the vertical scanning order shown in Figure 1.b, and the resulting dictionary  $V_i$  is stored in the database as a model for the vertical structure of  $\mathbf{t}_i$ ,  $i = 1, 2, \dots, N$ .

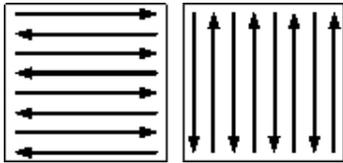


Figure 1: Scanning orders. (a) Horizontal; and (b) Vertical.

#### 4.2 The Retrieval Stage

In the retrieval stage, LZW operates in static mode. In this mode, one of the dictionaries generated in the model learning stage is used to encode a query sample, and no new strings are added to the dictionary during the encoding process.

A  $n \times n$  query sample  $\mathbf{x}$  is coded by the LZW algorithm with static dictionary  $H_i$ , following the horizontal scanning order shown in Figure 1.a., and the corresponding coding rate  $h_i$  is registered,  $i = 1, 2, \dots, N$ . Then the LZW algorithm with static dictionary  $V_i$  encodes  $\mathbf{x}$ , following the vertical scanning order shown in Figure 1.b, and the corresponding coding rate  $v_i$  is registered,  $i = 1, 2, \dots, N$ . As in the previous stage, all samples are extracted from histogram-equalized images.

Let

$$r_i = \frac{h_i + v_i}{2} \quad (5)$$

Query sample  $\mathbf{x}$  is considered more similar to texture  $\mathbf{t}_i$  than to texture  $\mathbf{t}_j$  in the database if  $\mathbf{r}_i < \mathbf{r}_j$ ,  $i, j = 1, 2, \dots, N$ . The rationale is that if  $\mathbf{x}$  is more similar to  $\mathbf{t}_i$  than to any other texture in the database (according to the texture models), the dictionaries  $H_i$  and  $V_i$  embeds the model that best describes its horizontal and vertical structure, thus yielding the smallest coding rates.

## 5 EXPERIMENTAL RESULTS

The complete Brodatz album (Brodatz, 1966), obtained from a public archive, was used to evaluate the performance of the proposed method. All 112 textures have  $640 \times 640$  pixels, with 8 bits/pixel. In the experiments, each Brodatz texture is taken as a single class. This corpus is the same used by (Xu et al., 2000), thus allowing direct comparison with another CBIR system from the literature. Some examples of the Brodatz textures are shown in Figure 2. Notice that some of Brodatz images are so irregular that they cannot be considered as texture images, according to most accepted definitions for texture. These highly irregular images have a negative effect on recognition rate (Xu et al., 2000).

In order to assess the effect of histogram equalization, tests were made with and without applying this operation before model learning and retrieval. In this section and in the next one, the proposed method with and without histogram equalization will be identified as CBIR-LZW-HE and CBIR-LZW, respectively. Each Brodatz texture was partitioned in  $128 \times 128$  non-overlapping subimages, which were taken as texture samples. Nine of these samples were stored in the database along with the corresponding LZW dictionaries constructed as describe in section 4.1, and the other ones were used as query samples for testing the accuracy of the method. Therefore, the database contains 112 texture classes, hereafter identified as  $C_i$ ,  $i = 0, 1, \dots, 111$ , and each class  $C_i$  has nine samples, identified as  $s_{ij}$ ,  $i = 0, 1, \dots, 111$ ,  $j = 0, 1, \dots, 8$ , for a total of  $9 \times 112 = 1008$  texture samples. Given a query sample  $\mathbf{x}$  from class  $C_i$ , retrieval of any one of the nine samples  $s_{ij}$ ,  $j = 0, 1, \dots, 8$ , from the same class is considered successful.

A number of accuracy measures are used in the literature. In order to directly compare the results of the proposed method with those of (Xu et al., 2000), the adopted accuracy measure was the *average recognition rate*. Consider that for each query the system returns the  $c$  most similar texture samples from the database, according to the LZW models. For a query sample from class  $C_i$ , let  $h(c)$  be the number of samples from  $C_i$  in the  $c$  retrieved samples. Recognition rate,  $R(c)$ , is then defined as the ratio between  $h(c)$  and the total number of samples from classes  $C_i$  stored in the database:

$$R(c) = \frac{h(c)}{9} \quad (6)$$

The average recognition rate,  $AR(c)$ , is the average value of  $R(c)$  for all test queries. From the definition, it follows that  $AR(c)$  does not decrease as  $c$  increases and, since there are 1008 texture samples in the test database,  $AR(c) = 1$  for  $c = 1008$ . More efficient retrieval mechanisms tends to achieve  $AR(c) = 1$  for

smaller values of  $c$  than less efficient mechanisms. In the ideal case for a database with 9 samples/class,  $AR(9) = 1$ .

Results are shown in Table 1. Figure 2 presents as an example a query which resulted in a low  $AR(9)$ , and Figure 3 presents the whole Brodatz texture from which this query samples was taken. Figure 4 compares the performance of the proposed method with M2, which achieved the best performance among the four variations of the method proposed by (Xu et al., 2000).

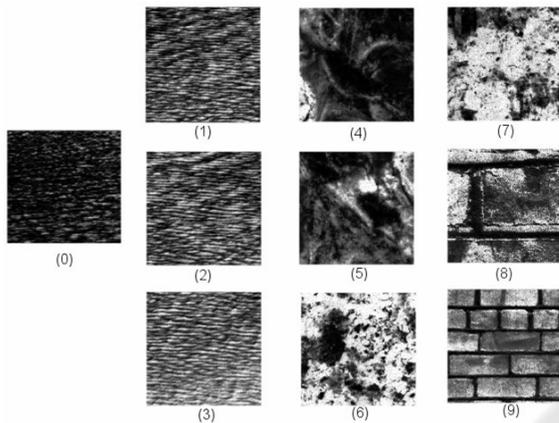


Figure 2: A query resulting in  $AR(9) = 0.3333$  with CBIR-LZW-HE. Sample 0 is the query sample, sample 1 is the first retrieved sample, sample 2 is the second retrieved sample, and so on.

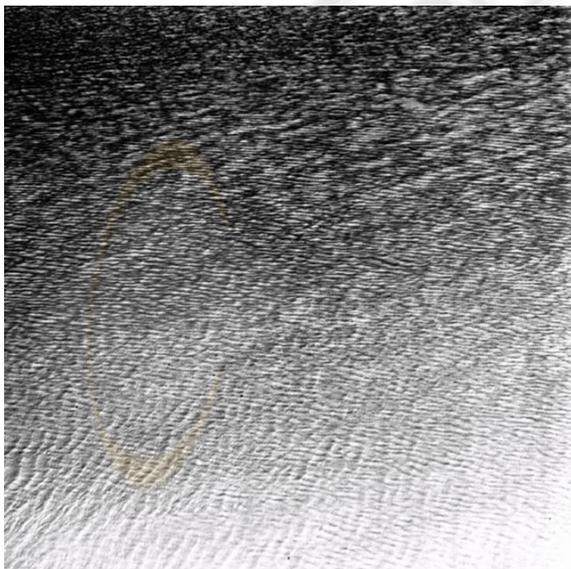


Figure 3: Texture D38, from which the query sample in Figure 2 was taken.

Table 1:  $AR(c)$  achieved by the proposed method, with and without histogram equalization, for various  $c$  (number of retrievals).

Sample size ( $n \times n$ )	CCR(%)	
	CLZW	CLZW-GHE
4 x 4	80.6	99.9
8 x 8	97.9	100
16 x 16	99.7	100
32 x 32	100.0	100

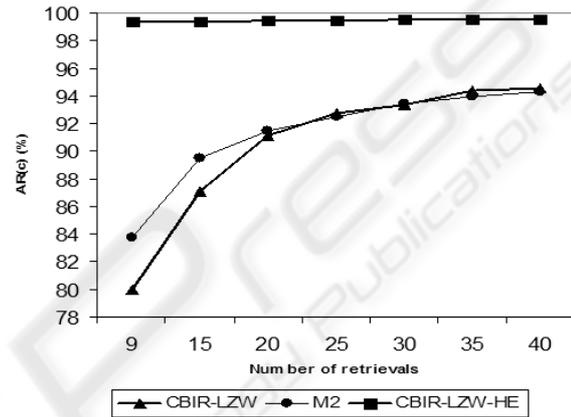


Figure 4:  $AR(c)$  achieved by the proposed method, with and without histogram equalization, and by the M2 method reported in literature.

## 6 DISCUSSION AND CONCLUSIONS

This paper proposed a new, simple and accurate content-based image retrieval method for texture images using histogram equalization and the LZW algorithm.

Table 1 shows that histogram equalization has a strong positive impact on performance. CBIR-LZW-HE achieved  $AR(9) = 0.9939$  (99.39%), while CBIR-LZW achieved  $AR(40) = 0.9459$  (94.59%).

Figure 4 compares the performance of the proposed method with M2 (Xu et al., 2000), over the same database. The comparison shows the superiority of CBIR-LZW-HE over M2. This superiority is still more remarkable in the most restrictive case ( $c = 9$ ), with CBIR-LZW-HE achieving  $AR(9)$  approximately 15 percent points above the value achieved by M2.

As stated before, many Brodatz images are so irregular that they cannot be considered as texture images, according to accepted definitions for texture. Xu's work (Xu et al., 2000) assessed the effects of this peculiarity of Brodatz album by dividing the textures in

two separate databases, one containing only visually homogeneous textures and the other one containing only visually inhomogeneous textures. By analyzing recognition rate over these two databases, they concluded that irregular images have a negative effect on recognition rate. On the other hand, CBIR-LZW-HE results, with  $AR(c)$  near 100%, even for small values of  $c$ , indicate that the method was to some extent robust to these irregularities.

Figure 2 presents an example query with a particularly low  $AR(9)$  (33, 33%). As can be seen in Figure 3, Brodatz texture D38, from which the query sample was taken, presents a strong non-uniform illumination variation, to the point of saturation in the lower right corner, making recognition a difficult task.

Future directions of research include assessing the robustness of the method to gray-scale, rotation and spatial-scale changes and investigating the use of lossy dictionary-based compressors. In fact, preliminary tests in this direction are already being conducted. First results indicate that histogram equalization makes the method robust to uniform gray-scale variations. Some developments are also being investigated in order to make the method invariant to rotation and non-uniform illumination variations, with promising results. It should be noticed that although invariance is an important feature in CBIR tools, in many practical applications (e.g. in industrial quality control by computer vision) images are acquired under strictly controlled conditions, and practically do not present scale, rotation or illumination changes.

The robustness of CBIR-LZW-HE to irregularities in the test database suggests that investigating the applicability of the method to content based non-textured images is also a very promising direction of research.

## REFERENCES

- Addis, M., Lewis, P., and Martinez, K. (2002). Artiste image retrieval system puts european galleries in the picture. *Cultivate Interactive*, (7).
- Augustejn, M. F., Clemens, L. E., and Shaw, K. A. (1995). Performance evaluation of texture measures for ground cover identification in satellite images by means of a neural network classifier. *IEEE Transactions on Geoscience and Remote Sensing*, 33(3):616–626.
- Bach, J. R., Fuller, C., Gupta, A., Hampapur, A., B. Horowitz, R. H., Jain, R., and Shu, C. F. (1996). The virage image search engine: An open framework for image management. In *Proc. SPIE Storage and Retrieval for Image and Video Databases IV*, San Jose, CA, USA.
- Baheerathan, S., Albreghsen, F., and Danielsen, H. E. (1999). New texture features based on the complexity curve. *Pattern Recognition*, 32(4):605–618.
- Bell, T. C., Cleary, J. G., and Witten, J. H. (1990). *Text Compression*. Englewood Cliffs: Prentice-Hall.
- Bovik, A. e. (2000). *Handbook of Image and Video Processing*. San Diego: Academic Press.
- Brodatz, P. (1966). *Textures: A Photographic Album for Artists and Designers*. New York: Dover.
- Dell'Acqua, F. and Gamba, P. (2003). Texture-based characterization of urban environments on satellite sar images. *IEEE Transactions on Geoscience and Remote Sensing*, 41(1):153–159.
- Dowe, J. (1993). Content-based retrieval in multimedia imaging. In *Proc. SPIE Storage and Retrieval for Image and Video Databases*.
- Faloutsos, C., Flickner, M., Niblack, W., Petkovic, D., Equitz, W., and Barber, R. (1993). Efficient and effective querying by image content. *Tech. Rep., IBM Res. Rep.*
- Finamore, W. and Leister, M. A. (1996). Lossy lempel-ziv algorithm for large alphabet sources and applications to image compression. volume 1, pages 235–238.
- Flickner, M., Sawhney, H., Niblack, W., Ashley, J., Huang, Q., Dom, B., Gorkani, M., Hafine, J., Lee, D., Petkovic, D., Steele, D., and Yanker, P. (1995). Query by image and video content: The qbic system. *IEEE Computer*.
- Frank, E., Chui, C., and Witten, I. H. (2000). Text categorization using compression models. In Storer, J. A. and Cohn, M., editors, *Proceedings of DCC-00, IEEE Data Compression Conference*, pages 200–209, Snowbird, US. IEEE Computer Society Press, Los Alamitos, US.
- He, X., Hu, Y., Zhang, H., Li, M., Cheng, Q., and Yan, S. (2004). Bayesian shape localization for face recognition using global and local textures. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(1):102–113.
- Huang, T., Mehrotra, S., and Ramchandran, K. (1996). Multimedia analysis and retrieval system (mars) project. In *Proc. 33rd Annu. Clinic Library Appl. of Data Processing Digital Image Access and Retrieval*.
- ISO/IEC, JTC1/SC29/WG11, and N1920 (1997a). Mpeg-7: Context and objectives (v.5).
- ISO/IEC, JTC1/SC29/WG11, and N1921 (1997b). Third draft of mpeg-7 requirements.
- ISO/IEC, JTC1/SC29/WG11, and N1922 (1997c). Mpeg-7 applications document.
- Jain, A. K., Ross, A., and Prabhakar, S. (2004). An introduction to biometric recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(1):4–20.
- Jain, R. (1993). Nsf workshop on visual information management systems. *SIGMOD Rec.*, 22(3):57–75.
- Jain, R., Pentland, A., and Petkovic, D. (1995). NSF-ARPA workshop visual inform. management syst.

- Kumar, A. and Pang, G. K. H. (2002). Defect detection in textured materials using optimized filters systems. *IEEE Transactions on Man and Cybernetics, Part B: Cybernetics*, 32(5):553–570.
- Ma, W.-Y. and Manjunath, B. S. (1999). Netra: A toolbox for navigating large image databases. *Multimedia Systems*, 7(3):184–198.
- Mandal, M. K., Panchanathan, S., and Aboulnasr, T. (1997). Image indexing using translation and scale-invariant moments and wavelets. In *Storage and Retrieval for Image and Video Databases (SPIE)*, pages 380–389.
- Ojala, T., Pietikainen, M., and Maenpaa, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987.
- Pentland, A., Picard, R. W., and Sclaroff, S. (1996). Photobook: content-based manipulation of image databases. *Int. J. Comput. Vision*, 18(3):233–254.
- Porat, M. and Zeevi, Y. Y. (1989). Localized texture processing in vision: Analysis and synthesis in the gaborian space. *IEEE Transactions on Biomedical Engineering*, 36(1):115–129.
- Randen, T. and Husy, J. H. (1999). Filtering for texture classification: a comparative study. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(4):291–310.
- Savari, S. A. (1997). Redundancy of the lempel-ziv-welch code. In *Proc. DCC, Salt Lake City*, pages 191–200.
- Shannon, C. E. (1948). A mathematical theory of communication. *Bell Syst. Tech. J.*, 27:379–423.
- Smith, J. R. and Chang, S.-F. (1997). Visually searching the web for content. *IEEE MultiMedia*, 4(3):12–20.
- Southard, T. E. and Southard, K. A. (1996). Detection of simulated osteoporosis in maxillae using radiographic texture analysis. *IEEE Transactions on Biomedical Engineering*, 43(2):123–132.
- Teahan, W. J. and Harper, D. J. (2001). Using compression-based language models for text categorization. In *Workshop on Language Modeling and Information Retrieval*, pages 83–88.
- Welch, T. A. (1984). A technique for high performance data compression. *IEEE Computer*, 17(6):8–19.
- Xu, K., Georgescu, B., Meer, P., and Comaniciu, D. (2000). Performance analysis in content-based retrieval with textures. In *Proc. Int. Conf. Patt. Recog.*, volume 4, pages 275–278.
- Zibreira, C. (2000). *Descrição e Procura de Vídeo Baseadas na Forma*. Master thesis, Instituto Superior Técnico de Lisboa, Portugal.
- Ziv, J. (1988). On classification with empirically observed statistics and universal data compression. *IEEE Transactions on Information Theory*, 34(2):278–286.
- Ziv, J. and Lempel, A. (1978). Compression of individual sequences via variable-rate coding. *IEEE Transactions on Information Theory*, 24(5):530–536.
- Ziv, J. and Merhav, N. (1993). A measure of relative entropy between individual sequences with application to universal classification. *IEEE Transactions on Information Theory*, 39(4):1270–1279.