Lossless Image Compression using Pattern Matching

Marcelo S. Pinho

Abstract— **This work addresses the problem of lossless compression of graylevel images using pattern matching. Even though there are some image formats, such as TIFF and PNG, which use this technique to reduce the size of image files, practical results have shown that there are better compressors, such as JPEG-LS and lossless SPIHT. Results of different schemes based on pattern matching when compressing graylevel images are presented and it shows their weak performances. The aim of this work is to show that this low efficiency is in accordance with universal source coding theory.**

Index Terms— **image coding, lossless image compression, pattern matching coding, universal source coding**

Resumo— **Este trabalho aborda o problema da compressão sem perdas de imagens em tons de cinza, usando a técnica de casamento de padrões. Embora existam formatos de imagens (tais como o TIFF e o PNG) que utilizam esta técnica para reduzir o tamanho dos arquivos, resultados práticos mostram que existem compressores melhores, tais como o JPEG-LS e a versão sem perda do SPIHT. Neste trabalho são apresentados resultados da compressão de imagens em tons de cinza através de diferentes esquemas que utilizam casamento de padrões. Estes resultados mostram um fraco desempenho dos codificadores baseados nesta técnica. O objetivo deste artigo é mostrar que esta baixa eficiência está de acordo com o que era esperado pela teoria da codificação universal de fontes de informação.**

Palavras chave— **codificação de imagens, compressão sem perdas de imagens, codificação por casamento de padrões, codificação universal**

I. Introduction

Image compression is a subject which has been studied for more than two decades. In fact, the amount of information represented by images is increasing rapidly in communication networks and the design of efficient systems requires good image compressors. Since lossy schemes can achieve much better compression rates, the majority of work has focused on that topic. However, in some applications, such as medical imaging and remote sensing, a lossless system can be required.

The studies of pattern matching as a technique to solve the data compression problem go back to the work of Lempel and Ziv. Based on this technique, they introduced two different versions of source codes, which nowadays are known as $LZ77$ [25] and $LZ78$ [26]. The Lempel-Ziv encoders became popular because they have a low computational complexity and they are universal, that is, their performances converge to the source entropy with high probability if the data was drawn from an ergodic source [23], [26].

Manuscrito recebido em 6 de agosto de 2010; revisado em 7 de novembro de 2010.

M. S. Pinho (mpinho@ita.br) is with the Department of Electronic Engineering at Instituto Tecnológico de Aeronáutica - ITA. Praça Marechal Eduardo Gomes, 50 - São José dos Campos - SP - Brasil - 12228-900.

The problem of image compression is a typical application where universal source coding should be applied, since a good probability model is *a priori* unknown. Many studies about image compression using pattern matching were done in the past. In fact, this technique has been used to design lossy algorithms for graylevel images [1], [2], lossless schemes for bi-level images [9] and lossless compressors for graylevel images [7], [11], [16]. Furthermore some image formats, such as TIFF and PNG, make use of this technique to compress different kinds of images. However, the results obtained for lossless graylevel image compression are poor when compared to the results of other algorithms, such as JPEG-LS [19] and lossless SPIHT [13].

This work uses some results of universal source coding theory to improve the understanding of the performance of pattern matching techniques when it is applied to losslessly compress graylevel images. The text is organized as follows. Section II addresses theoretical results of universal source codes based on pattern matching. The subject of section III is the lossless image compression problem. Section IV presents practical performances of different schemes using pattern matching and uses the theory to better understand these results. Closing the work, section V presents the conclusion.

II. Universal Source Coding and Pattern Matching

Universal source coding is an important and challenging problem in information theory which has been studied for a long time [5]. Its aim is to compress data generated by a source with completely or partially unknown statistics. Let $X_1^\infty = X_1 X_2 \ldots$ be the source output, which is a semi-infinite sequence of random variables which can assume values in a finite set A, according to a probability measure p. The set A is called the source alphabet and p is the source probability measure. It is well known that the rate $H(X_1^n)/n$ is a lower bound for the compression rates of uniquely decodable binary codes, where $H(X_1^n)$ denotes the entropy of X_1^n in bits. Furthermore, if p is known, there are many algorithms (e.g. arithmetic algorithm and Huffman algorithm) that can be used to build encoders that achieve this rate except by a roundoff error [4]. That is, if p is given, it is easy to present an encoder C which maps any sequence x_1^n into a codeword $C(x_1^n)$ (finite sequence of bits) such that

$$\frac{H(X_1^n)}{n} \leq \frac{E[|C(X_1^n)|]}{n} \leq \frac{H(X_1^n)}{n} + O\left(\frac{1}{n}\right), \qquad (1)$$

where $E[|C(x_1^n)|]$ is the expected value of the codeword length, $|C(x_1^n)|$ (in bits). If the source entropy H exists, then the rate of C converges to H when n grows indefinitely.

In many practical situations an encoder C is used to compress data produced by an unknown source or produced by a group of different sources. In such cases, C must be good enough to compress sources with different probability measures. Pattern matching became a popular technique in data compression because it solves this problem. In fact, if X_1^∞ is an ergodic process, in [26] it was proved that

$$\frac{|LZ78(X_1^n)|}{n} \to H, \text{ a.s.} \tag{2}$$

and in [23] it was shown that

$$\frac{|LZ77(X_1^n)|}{n} \to H, \text{ in probability} \tag{3}$$

The above results must be interpreted carefully. These results show that for n sufficiently large the compression rate is close to the source entropy with high probability. However, in practice, n is bounded and the rate can be far from the lower bound. Therefore, it is very important to know how fast the rate converges to the entropy. Usually the speed of convergence is measured by using the concept of redundancy. Let S be a source which produces X_1^∞ as output. The average redundancy of C with respect to (w.r.t.) the source S is defined as

$$R_{C,n}(S) = \frac{E[|C(X_1^n)|] - H(X_1^n)}{n}. \tag{4}$$

The objective of a universal source code is to compress efficiently any source belonging to a class of sources. Therefore, a good average redundancy w.r.t. a specific source S is not a proof of efficiency for a universal code. There are different methods to evaluate the performance of universal codes. One of them, which is frequently used, is the minimax redundancy of a class of sources. Let Ψ denote a class of sources and let Γ be the set of all uniquely decodable binary codes. The minimax redundancy of the class Ψ is defined as

$$R_n(\Psi) = \min_{C\in\Gamma} \max_{S\in\Psi} R_{C,n}(S). \tag{5}$$

The problem of finding the redundancy of a given class of sources has been studied by many authors [6], [17], [24]. One of the most famous results establishes that if Ψ is the class of Markov sources,

$$R_n(\Psi) = \Theta\left(\frac{\log n}{n}\right), \tag{6}$$

see [6]. Even though universal source codes which achieve that redundancy for Markov sources are known [12], [21], the computational complexity of those codes increases with the number of states of the Markov model and it can be difficult to use them in some practical applications. Pattern matching codes are an interesting alternative with low complexity. Furthermore, if the memory of the source is not bounded, as pointed out in [8], pattern matching may be the unique practical solution to estimate the source probabilities.

Since $LZ77$ and $LZ78$ were introduced, many efforts were done to compute their redundancies. Almost twenty years later, it was shown that for Markov sources, the redundancy of $LZ78$ is $O(1/\log n)$ [14] and the redundancy of $LZ77$ is $O(\log\log n/\log n)$ [15]. Therefore, the pattern matching codes do not achieve the best redundancy for Markov sources.

III. Lossless Image Compression

Typically, a practical scheme used to losslessly compress digital images has two steps. The first one is an image transform which aims to reduce the statistical dependence between pixels. The second step is an entropy code which is built considering that the transformed image can be modeled as a Markov source with a reduced number of states depending on a few number of neighbor pixels.

The use of an image transform would be unnecessary if a suitable probability model for the image sample, I, was known and if a good probability measure estimate (based on this model) could be extracted from I. However, in general, it is hard to find a reasonable probability model such that a good probability measure can be estimated efficiently. Actually, the transformed image, T, is useful because good practical results have been obtained when it is modelled as the output of a Markov source with a reduced number of states.

Basically the best algorithms for lossless image compression can be classified in two groups. The first one is composed by algorithms which use an adaptive predictor to compute the transformed image T (which is the prediction error in this case). CALIC [22] and JPEG-LS [19] are two examples of algorithms in this group. The compressors of the second group make use of an integer transform based on signal decomposition in orthogonal (or biorthogonal) functions, such as the Discrete Cosine Transform (DCT) and a Discrete Wavelet Transform (DWT). Examples of algorithms which use this kind of transform are SPIHT [13] and the lossless version of the JPEG2000 standard [18]. In general, an adaptive prediction is used in lossless and near lossless schemes and an integer transform is useful to design progressive compressors (i.e., a lossless/lossy scheme).

After the transformation (prediction or an integer transform), T is compressed by using an entropy code based on a probability measure which is estimated considering that it is the output of a Markov source with a low number of states. The markovian model is used because it was observed that the pixels of T still have a statistical dependence. Furthermore, the number of states must be low since the size of image samples, in general, is not large enough to permit a good probability estimation based on a different model (which is known as the context dilution problem). Golomb and arithmetic codes are some examples of entropy codes which are used in many lossless image compressors [13], [19].

IV. Lossless Image Compression using Pattern Matching

Since the problem of lossless image compression can be seen as a typical application of universal source coding theory, many efforts were done to apply pattern matching to compress digital graylevel images [7], [11], [16]. Even though this technique has been successfully used to compress text, the attempts to use it in graylevel image compression led to poor results when compared to the performances of other schemes. This section presents the practical performance of several schemes to losslessly compress graylevel images, based on pattern matching. The practical results were obtained using three

Fig. 1. Group of natural images

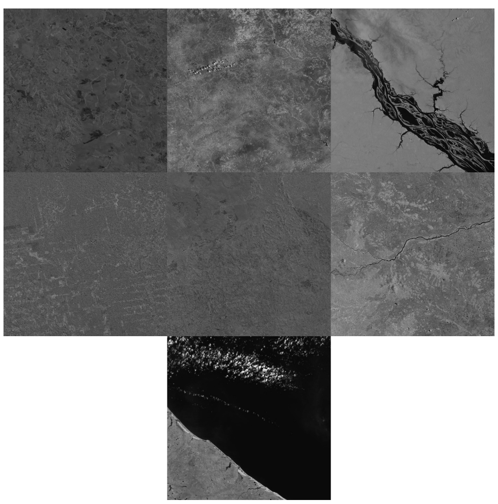

Fig. 2. Group of images from Landsat 5

different groups of images. The first one is composed by seven Natural Images: *Airplane, Baboon, Barbara, Lena512, Peppers, Sailboat, Tiffany*, which are available at *www.cipr.rpi.edu* (*homepage of the Center for Image Processing Research - Rensselaer Polytechnic Institute*, USA) and it is illustrated in Figure 1. The second group is constituted by seven images obtained from the Thematic Mapper of Landsat 5, which were provided by the Brazilian National Institute for Space Research. The images of this second group is presented in Figure 2. The composition of the last group is four Computerized Tomography Images and four Magnetic Resonance Images, which were also available at *www.cipr.rpi.edu*, see Figure 3. Practical results are analyzed in the light of universal source coding theory and this analysis is the main contribution of this work.

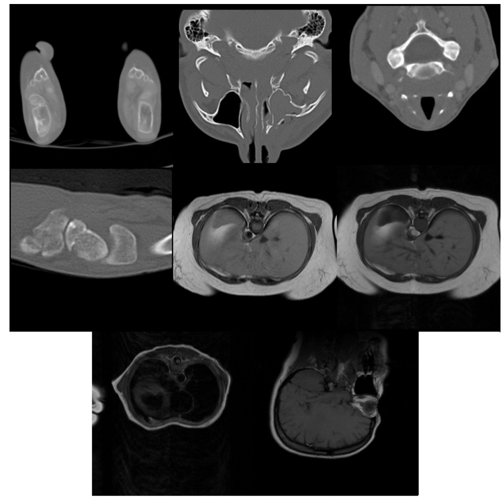

Fig. 3. Group of medical images

The most trivial manner to compress digital images using pattern matching is applying the algorithms directly. Even though it is not obvious from their description, pattern matching algorithms as any universal source code are also probability estimators [10]. Since the number of pixels is small the conditional probabilities can not be efficiently estimated and thus the results obtained by this way are not competitive. In fact, if the Natural Images are scanned by the Hilbert-Peanno procedure [11], the most popular version of $LZ78$, which is commonly called LZW [20], achieves 6,68 bpp (bits per pixel) on average. Comparing this bit rate to the bit rate of JPEG-LS (4,50 bpp on average) and to the bit rate of lossless SPIHT (4,51 bpp on average) it can be noticed that this scheme is not efficient.

In general, the convergence of the rate of a universal source code depends of the source alphabet cardinality. Therefore, in some applications, an improvement can be obtained if the image bit planes are compressed independently (even though this procedure neglects the statistical dependence of bit planes). If the pattern matching algorithm is applied in the image bit planes, its performance improve slightly (6,22 bpp). However, its result is still much worse than JPEG-LS and lossless SPIHT bit rates.

An alternative way to improve the performance is to compress the transformed image T (obtained from an Integer DWT, from an Integer DCT or from a predictor) employing a pattern matching algorithm. Table 1 presents the performance of LZW when it is used to compress independently the bit planes of the prediction error generated by JPEG-LS. The results presented in that Table are the average of the bit rates (in bits per symbol) for each group of images. Results of JPEG-LS and the sum of the bit planes first order entropy estimations are also presented in Table 1. From these results, it can be noticed that the LZW results are significantly worse than the results of JPEG-LS. Furthermore, it is interesting to observe that LZW results are still greater than the first order entropy estimation, for all groups of images.

Table 2 presents the practical results of LZW when used

TABLE I
LZW RESULTS (IN BPP) WHEN APPLIED IN BIT PLANES OF JPEG-LS PREDICTION ERROR

Group of Images	LZW	Entropy	JPEG-LS
Natural Images	4.89	4.72	4.50
Landsat Images	4.35	4.13	3.96
Medical Images	3.24	3.23	2.61

to compress the bit planes of the S+P transform. To make the analysis easier, the sum of the bit planes first order entropy estimations and the results of lossless SPIHT are also included in Table 2. In this case, the results of the pattern matching algorithm are significantly worse than the results of lossless SPIHT and they are also greater than the first order entropy estimation.

TABLE II
LZW RESULTS (IN BPP) WHEN APPLIED IN BIT PLANES OF S+P TRANSFORM

Group of Images	LZW	Entropy	SPIHT
Natural Images	5.08	4.93	4.51
Landsat Images	4.55	4.38	3.89
Medical Images	3.53	3.41	2.57

The performance of LZW when compressing the bit planes of an Integer DCT was measured using the transform proposed in [3]. Table 3 presents the practical results and the bit planes first order entropy estimations. It is interesting to notice that the LZW results are better than the entropy estimations. However, the difference is around 1,7% in the worst case (which is the group of Medical Images). In fact, a simple second order entropy estimation, which is 4.09 bits per symbol in the average for Medical Images, is enough to achieve a better rate than LZW.

TABLE III
LZW RESULTS (IN BPP) WHEN APPLIED IN BIT PLANES OF AN INTEGER DCT

Group of Images	LZW	Entropy
Natural Images	5,66	5,69
Landsat Images	4,85	4,91
Medical Images	4,12	4,19

Although the performance presented in Tables 1, 2 and 3 were obtained using the algorithm LZW, different pattern matching algorithms achieve similar results. For example, the standard PNG, which uses a algorithm based on $LZ77$ to compress a prediction error, achieves a bit rate of 4.9 bits per symbol in the mean for the group of Natural Images. It is interesting to notice that LZW produces a similar bit rate when used to compress the prediction error generated by JPEG-LS, see Table 1.

The weak performance of pattern matching algorithm in this specific application is frequently reported in the literature. However, the published papers do not link this poor performance to universal source coding theory. From theoretical results of pattern matching codes it could be concluded that this type of code is not effective for memoryless sources or for Markov sources with a small number of states, as pointed out in section 2. In fact, these results are related to the problem of context dilution. Since pattern matching algorithms build a large context dictionary, the estimation of conditional probabilities is diluted, leading to a suboptimal convergence rate for these classes of sources. However, it is important to point out that even though the context dilution argument is more intuitive, the redundancy rate is the quantitative measure of the inefficiency of these algorithms for those classes of sources.

Tables 1 and 2 show that pattern matching schemes do not achieve even the estimations of the first order entropies, which could be almost reached easily by the use of any source code based on the frequency rate of bits. In the case of the Integer DCT proposed in [3], reported in Table 3, LZW is slightly better than the first order entropy, but is worse than the second order entropy, which is also easy to be reached by the use of any source code based on the pair of bits frequency rate. Furthermore, if the performances of JPEG-LS and lossless SPIHT were compared to the first order entropy estimation of the transformed image, T, it would be noticed that these algorithms results are not much lower than the estimative. In fact, the average of the estimatives for JPEG-LS prediction errors is 4.71, and the estimative for S+P transform is 4.76 in the mean for the group of Natural Images. This kind of results would be expected if the image transform could be modeled as the output of a Markov source with a small number of states. Therefore, considering that the Markovian is a good model, the results presented here are in accordance with source coding theory. On the other hand, if the aim of the analysis is to find a good model for the image transform, these results can be seen as an evidence that the Markovian is a reasonable model and that models with larger number of states may be usefulness.

Finally, it is important to mention that a more realistic model could be obtained by the use of non stationary processes. In fact, the algorithms used in SPIHT and in JPEG-LS do not consider that the image transform is stationary. For example, in the probability estimation of JPEG-LS, the pixels encoded more recently have a larger weight than the pixels encoded in the remote past. However, the improvement obtained by the use of this kind of technique is very limited. Actually, from the results of first order entropy estimation of T, which is obviously based on a stationary model, it can be noticed that this improvement is less than 10%. Furthermore, since the bit rate of pattern matching algorithms is worse than the first order entropy (or than the second order entropy), it is clear that the non stationarity is not the reason of their poor performances.

V. CONCLUSION

This work presented a performance analysis of pattern matching when losslessly compressing graylevel images on the light of universal source coding theory. Practical works on the area of image compression have shown that pattern matching achieves weak results when compared to other schemes, even though it is used in some image formats. Practical results for three different groups of images using the pattern matching technique over an prediction error, an Integer Multiresolution Transform or an Integer DCT were presented, confirming the weak performance of this technique. In general, the transformed images are modeled as a data generated by a Markov source with few states. Based on results of source coding theory, this correspondence shows that the weak performance would be expected if the transformed images are really well modeled by Markovian source with a low number of states. Furthermore, the results presented here can also be seen as another evidence that this model is useful in image compression. Finally, it is important to mention that pattern matching is an extremely useful tool in data compression and it may be efficient even in different problems concerning image processing. However, in lossless graylevel image compression there are better solutions.

REFERENCES

[1] M. Alzina, W. Szpankowski, and A. Grama. 2d-pattern matching image and video compression: theory, algorithms, and experiments. *IEEE Transactions on Image Processing*, 11:318–331, 2002.

[2] M. B. Carvalho, E. A. B. da Silva, and W. Finamore. Multidimentional signal compression using multiscale recurrent patterns. *Elsevier Science Signal Processing*, 82:1559–1580, 2002.

[3] S. Chokchaitam, M. Iwahashi, and S. Jitapunkul. A new unified lossless/lossy image compression based on a new integer dct. *IEICE Transactions on Information*, (7):1598–1606, 2005.

[4] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley, New York, 1991.

[5] L. D. Davisson. Universal noiseless coding. *IEEE Transactions on Information Theory*, 19:783–795, 1973.

[6] L. D. Davisson. Minimax noiseless universal coding for markov sources. *IEEE Transactions on Information Theory*, 29:211–215, 1983.

[7] W. A. Finamore and P. R. R. L. Nunes. A modified lempel-ziv and its application to errorless image compaction. *Proc. IEEE Int. Conf. Acoustic, Speach and Signal Proc.*, pages 14–17, 1991.

[8] Y. Hershkovits and J. Ziv. On sliding-window universal data compression with limited memory. *IEEE Transactions on Information Theory*, 44:66–78, 1998.

[9] Y. Jia and E. Yang. Context-dependent multilevel pattern matching for lossless image compression. *IEEE Transactions on Information Theory*, 49:3169–3184, 2003.

[10] G. G. Langdon Jr. A note on the ziv-lempel model for compressing individual sequences. *IEEE Transactions on Information Theory*, 29:284–287, 1983.

[11] A. Lempel and J. Ziv. Compression of two-dimentional data. *IEEE Transactions on Information Theory*, 32:2–7, 1986.

[12] J. Rissanen. A universal data compression system. *IEEE Transactions on Information Theory*, 29:656–664, 1983.

[13] A. Said and W. A. Pearlman. An image multiresolution representation for lossless and lossy compression. *IEEE Transactions on Image Processing*, 5:1303–1310, 1996.

[14] S. A. Savari. Redundancy of the lempel-ziv incremental parsing rule. *IEEE Transactions on Information Theory*, 43:9–21, 1997.

[15] S. A. Savari. Redundancy of the lempel-ziv string matching code. *IEEE Transactions on Information Theory*, 44:787–791, 1998.

[16] D. Sheinwald, A. Lempel, and J. Ziv. Two-dimensional encoding by finite-state encoders. *IEEE Transactions on Communications*, 38:341–347, 1990.

[17] P. Shields. Universal redundancy rates do not exist. *IEEE Transactions on Information Theory*, 39:520–524, 1993.

[18] D. S. Taubman and M. W. Marcelin. *JPEG2000: Image Compression Fundamentals, Standards and Practice*. Kluwer Academic Publishers, Boston, 2001.

[19] M. J. Weinberger, G. Seroussi, and G. Sapiro. The loco-i lossless image compression algorithm: principles and standardization into jpeg-ls. *IEEE Transactions on Image Processing*, pages 1309–1324, 2000.

[20] T. A. Welch. A technique for high-performance data compression. *IEEE Computer*, 17:8–19, 1984.

[21] F. M. J. Willems, Y. M. Shtarkov, and T. J. Tjalkens. The context-tree weighting method: basic properties. *IEEE Transactions on Information Theory*, 41:653–664, 1995.

[22] X. Wu and N. D. Memon. Context-based adaptive lossless image coding. *IEEE Transactions on Communications*, 45:437–444, 1997.

[23] A. D. Wyner and J. Ziv. The sliding-window lempel-ziv algorithm is asymptotically optimal. *Proc. IEEE*, 82:872–877, 1994.

[24] Q. Xie and A. R. Barron. Minimax redundancy for the class of memoryless sources. *IEEE Transactions on Information Theory*, 43:646–657, 1997.

[25] J. Ziv and A. Lempel. A universal algorithm for data compression. *IEEE Transactions on Information Theory*, 23:337–343, 1977.

[26] J. Ziv and A. Lempel. Compression of individual sequences via variable-rate coding. *IEEE Transactions on Information Theory*, 24:530–536, 1978.

Marcelo S. Pinho received the B.Sc., M.Sc. and D.Sc. degrees in electrical engineering from Pontifícia Universidade Católica do Rio de Janeiro, in 1994, 1996 and 2000, respectively.

In 1998, he was a visiting scholar in the Center for Image Processing Research of the Rensselaer Polytechnique Institute, NY, USA. Since April of 2002, he has been with the Department of Electronic Engineering at Instituto Tecnológico de Aeronáutica. His professional experience and research interests are in information theory, digital communication, and digital signal processing.