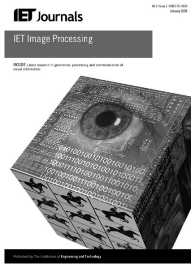
www.ietdl.org

Published in IET Image Processing
Received on 25th September 2012
Revised on 3rd February 2013
Accepted on 14th February 2013
doi: 10.1049/iet-ipr.2012.0538



ISSN 1751-9659

Lossy and lossless image encoding using multi-scale recurrent pattern matching

Danillo Bracco Graziosi[1], Nuno Miguel Rodrigues[2], Eduardo A.B. da Silva[3], Murilo B. de Carvalho[4], Sérgio Manuel Maciel de Faria[2]

[1]Instituto de Telecomunicações, Portugal and with PEE/COPPE, Univ. Fed. Rio de Janeiro, CP 68504, 21945-970 Rio de Janeiro, Brazil
[2]Instituto de Telecomunicações, Portugal and with ESTG, Inst. Pol. Leiria, Apartado 4163, 2411-901 Leiria, Portugal
[3]PEE/COPPE/DEL/Poli, Univ. Fed. Rio de Janeiro, CP 68504, 21945-970 Rio de Janeiro, Brazil
[4]TET/CTC, Univ. Fed. Fluminense, Campus da Praia Vermelha, 24210-240 Niterói, Brazil
E-mail: danillo@lps.ufrj.br

Abstract: In this study, the authors investigate the use of multi-scale recurrent pattern matching paradigm for lossless image compression. The multi-scale multidimensional parser (MMP) algorithm is a successful implementation of this paradigm for lossy image compression, and can naturally perform lossless compression since it was first derived from a Lempel–Ziv lossless scheme. However, neither its recently adopted coding tools had been adapted for lossless coding nor a thorough analysis of its performance had been carried out. In this work, the authors evaluate MMP's lossless compression capability, proposing modifications for some of its predictions modes, as well as the inclusion of an adaptive prediction mode based on least squares. The residual information is also coded with well-known techniques used in lossless compression. Experimental results for MMP show that the algorithm achieves a good performance for images such as computed generated graphics and scanned documents, whereas keeping a competitive performance for natural images. Since the algorithm's structure is exactly the same for lossless and lossy compression, the obtained results suggest that MMP is able to achieve a high compression performance for a wide range of images and rates, from lossy to lossless, without any prior analysis of the image to be coded.

1 Introduction

Lossless compression is mainly used in applications where compression artefacts are unacceptable, such as in the case of medical imaging, preservation of artwork or satellite images [1]. Several algorithms for lossless compression [1–3] were developed considering constraints in available memory and computational resources. With the increase in computational power and improved quality of terminals, together with the decrease in prices and the evolution of editing softwares, it makes sense to reevaluate the role of lossless coding, and once again consider its application in a broader scenario.

Most lossless algorithms compress an image using two independent steps [4]: image 'modelling' and image 'coding'. The image modelling process can be regarded as an inductive inference problem, where one needs to find an appropriate model class, order and parameters to suitably represent the data. For instance, in predictive coding, if the model is successful in guessing the next pixel's value, then the residual information will have a skewed probability distribution centred around zero, and can be compressed effectively in the coding step.

For the coding step a variable-length code such as Huffman code, Rice code, or Arithmetic code is usually used [1]. For such algorithms, codewords with high probability are represented with fewer bits, improving compression results for highly peaked probability distributions, such as the ones obtained from the prediction residues. For efficient implementation, some algorithms assume a distinct probability distribution, such as an exponential or generalised Gaussian distribution. Examples of lossless image encoders that use such models for their residue's statistics are lossless JPEG (JPEG-LS) [2] and minimum rate predictors (MRP) [5] algorithms. However, compression efficiency decreases when the model does not fit the actual probability distribution, which can occur if the prediction fails and residues have many values distant from zero.

One way to avoid assuming a specific model for the sources' statistics is to use adaptive methods based on pattern matching. Pattern matching algorithms rely on the fact that sequences tend to occur repeatedly on the source to be encoded, and thus the source can be efficiently compressed by detecting and encoding these repeated patterns using fewer bits. This can be accomplished by using dictionary entries for the occurring patterns, and adapting the dictionary codewords as the image is encoded.

Early versions of pattern matching lossless encoding algorithms were presented by Ziv and Lempel [6] and focused on text encoding, dealing with one-dimensional data. Improved dictionary updating techniques were adopted in [7, 8]. Pattern matching is not only used in lossless coding, but also in lossy compression algorithms, where approximate pattern matching is applied, also known as Lossy-Lempel–Ziv algorithms [9–11]. Other examples of pattern matching-based algorithms are the Grammar codes [12] and the 'deflate' algorithm, used in the PNG file format [13].

The multi-scale multidimensional parser (MMP) algorithm [14] was successfully used as a pattern matching-based coding algorithm for lossy image encoding. Its distinctive feature is the use of multi-scale transformations, enhancing the dictionary adaptability and also its compression performance. Recent developments, such as a hierarchical prediction step [15] and a flexible segmentation scheme [16], have improved MMP's compression efficiency for a broad range of different images types, in a wide range of compression rates. In this paper, we show the advantages of MMP and propose its use as a lossless image encoder. Results obtained for both lossy and lossless compression support the statement that MMP can be considered an efficient alternative for image compression for a broad range of rates and various kinds of images.

This paper is organised in the following manner: Section 2 briefly reviews the MMP algorithm. Section 3 proposes adaptations at MMP's image modelling stage for lossless compression. Section 4 analyses some techniques usually employed to improve the image coding stage of lossless compression algorithm and their performance in the MMP framework. Section 5 evaluates the performance for lossless compression of the proposed lossless MMP algorithm, as well as its performance as a lossy encoder. Section 6 presents the conclusions of this work.

2 Multidimensional multi-scale parser

The MMP algorithm was first applied in lossy image coding in [17], and since then it has been successfully used not only for image coding, but also for a varied class of signals, such as ECG [18], stereo images [19] and video coding [20].

The MMP algorithm uses a block-based approach, dividing the image into non-overlapping blocks, usually of size 16 × 16. Blocks are encoded in raster scan order, and each block may be further segmented. In earlier versions, the block segmentation consisted of dividing the block in a quadtree structure [17]; later on, it was substituted by a dyadic segmentation [14], where the segments are divided in half, alternatively in the vertical and horizontal directions. Results presented in [16] showed that gains in rate-distortion performance can be achieved by the use of a more flexible segmentation scheme, where the encoder explicitly sends the direction of segmentation information (horizontal or vertical) to the decoder. The resulting segmentation of the input block can be represented by a binary segmentation tree, as depicted in Fig. 1, where the nodes of the tree indicate a block segmentation, and the tree leaves represent the subblocks.

In [15], prediction was incorporated into the MMP algorithm. When prediction is successful, it generates residue signals that have a distribution function concentrated in a reduced set of values around zero. More regular patterns are thus generated and the probability of using patterns already present in the dictionary increases. The prediction schemes used in MMP are based on the H.264/AVC Intra modes [21], that use an explicit switched

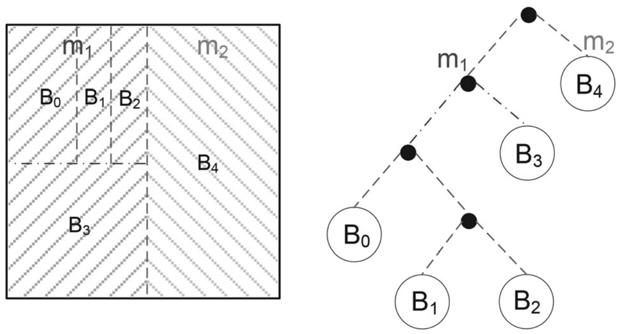


Fig. 1 *Flexible block segmentation with prediction-block segmentation can be arbitrarily performed in the horizontal or vertical directions*

prediction approach. Similar to the H.264/AVC standard, in MMP the prediction window size is adaptively chosen, and the prediction block is also segmented in the same flexible manner as explained previously. It is important to note that a block that is encoded without being further segmented should be entirely contained in a prediction window, and a prediction window may contain more than one such block. Fig. 1 shows a block with two prediction partitions, and further sub-block partitions of the residue information.

Instead of the traditional transform-quantise-encode procedure, MMP exploits an encoding paradigm based on dictionary search for its residue encoding. Block matching is performed between the residue blocks and the dictionary's codewords. The innovation of the MMP algorithm is the introduction of matches at different scales [14]. Through the use of scale transformations, MMP allows matching between blocks of different sizes. In [14], it is shown that multi-scale pattern matching can outperform ordinary pattern matching when used to lossly compress data from memoryless Gaussian sources.

The dictionary is updated with recently encoded patterns, inspired by the works of Ziv and Lempel [6]. Once the block is segmented, the dictionary indexes used to approximate each part of the block are concatenated and this new pattern is added to the dictionary. Note that by using recently encoded patterns, the dictionary adapts to the image statistics and no previous assumption is needed for its encoding.

In order to encode a block, one can choose many different combinations of prediction modes, segmentation flags and even dictionary indexes. Coding decisions are made based on a Lagrangian cost optimisation algorithm. One way to achieve a performance point on the convex hull of the operational rate-distortion curve is to use the segmentation tree structure with associated Lagrangian cost and perform a minimisation procedure similar to the one described in [22]. The cost of a tree, $\mathcal{T}$, representing a block's segmentation, can be determined by

$$J(\mathcal{T}) = D(\mathcal{T}) + \lambda R(\mathcal{T}) \quad (1)$$

where $D(\mathcal{T})$ is the distortion obtained when using the selected codewords to encode a block, and $R(\mathcal{T})$ is the rate necessary to send the prediction modes, segmentation flags and dictionary indexes associated to that specific node.

The elements of the output bitstream, that is, the segmentation flags, the prediction modes and the dictionary indexes, are losslessly encoded using a context-based adaptive arithmetic encoder. The context used for the

doi: 10.1049/iet-ipr.2012.0538

elements depends on the corresponding block size. This information does not need to be explicitly sent to the decoder since both encoder and decoder replicate the same segmentation structure.

A key factor for the performance of a pattern matching algorithm is the dictionary adaptation process. For a given stationary ergodic source, the performance of a pattern matching algorithm approaches the entropy of the source when given an infinite sequence [23]. However, for image encoding, we are far away from these conditions, so we have to employ techniques for a faster dictionary adaptation. A study on several techniques to enhance dictionary adaptation, and their use on MMP, can be found in [15].

The exact same algorithm can be used for lossless encoding, where only perfect matching is allowed in the dictionary index search routine, that is, λ in (1) is zero. The same procedure for prediction followed by residue coding via pattern matching used for lossy compression is directly applicable to lossless coding. The only restriction is the presence of a complete 1×1 dictionary, in order to enable the representation of any pattern without distortion. In case a perfect match is not possible in higher scale, the block is recursively divided up to the extreme case where every pixel is coded from the 1×1 dictionary.

Following the two step approach towards lossless encoding (modelling and encoding), we modified the MMP algorithm and tested several different predictors, and also implemented commonly used lossless residue coding techniques. The next sections will provide further details on the algorithms.

3 Improving the image modelling stage of MMP's lossless compression algorithm

We attempted to improve the MMP's image modelling stage using predictors more suitable for the lossless case. In the next subsections, we will compare the efficiency of explicitly sending the prediction information or implicitly deriving this information, and we will show two techniques applied at the pixel level that improved prediction, therefore reducing the overall bit rate for lossless compression.

3.1 *Implicit prediction mode*

The first predictive version of MMP [24] sends the prediction information explicitly to the decoder. This is not common in most lossless image compression algorithms, which use an implicit prediction stage. Examples are the context-based, adaptive lossless image codec (CALIC) [25], JPEG-LS [2] or edge directed prediction (EDP) [26] algorithms. Although such approaches avoid the overhead of sending the explicit prediction information, their prediction algorithms may not always be adequate for predicting the entire image. This fact motivated a study, presented below, in order to evaluate the explicit prediction approach employed in the MMP framework.

To evaluate the implicit prediction approach, we replaced the MMP explicit intra prediction step with the implicit prediction step of CALIC [25] (gradient adaptive prediction, GAP), JPEG-LS [2] (median edge detection, MED) and EDP [26] (least-square prediction, LSP). In other words, in this experiment, we employed the MMP algorithm to encode the residual image resulting from the prediction step. No prediction mode information was sent since both encoder and decoder followed the same procedure for obtaining the predictors, and this procedure was the same for the entire image. As MMP residue coding does not rely on any smoothness constraint, it was able to cope with the inefficiencies of using only one single predictor. We also provided results for coding the image using an MMP version without any prediction [14], and compared all those results with the compression rates that we obtained just by using the MMP-FP algorithm [16], which uses the intra prediction modes of the H.264/AVC algorithm without any optimisation for lossless coding.

We compared the following algorithms: MMP without any prediction [14] (MMP), using the prediction model used by JPEG-LS [2] (MMP-MED), by CALIC [25] (MMP-GAP), by EDP [26] (MMP-LSP), or the explicit INTRA prediction modes used in the MMP-FP [16]. These techniques with their specifics characteristics are presented on Table 1, which also summarises all the algorithms developed in the scope of this work and to be discussed on the next sections. The testset comprises smooth images and also scanned compound images, that is, scanned documents of images mixed with text, such as PP1205 and PP1209.

Table 2 shows the results of compression achieved with the MMP algorithm, when using different prediction models. From it, we can see that MMP-MED, MMP-GAP and MMP-LSP are suited for smooth images, and present higher compression results than when no prediction is used. However, for compound and computer generated images, the prediction cannot capture the texture of the image appropriately, and the algorithm that uses no prediction (MMP) outperforms the proposed methods, achieving higher compression gains. In addition, for the algorithms that use prediction, we can see that the compression obtained by the one that explicitly sends the prediction information (MMP-FP) was higher than the one obtained using just the implicit methods. Results for the explicit transmission of the prediction model have shown that it is efficient also for smooth images. We can then conclude that, on average, the explicit transmission of the prediction model is more effective than the implicit assumption of a certain prediction model. Therefore, in the remainder of this paper, we will restrict our investigation to the explicit prediction models.

It is important to note that when coding pixel-by-pixel in a raster scan order, only a half non-symmetric plane of the image, that is the reconstructed pixels from lines above the current location, is available for prediction. The intra-prediction modes use a blocking prediction structure. Therefore they are not constrained to a half non-symmetric

Table 1 MMP taxonomy, classifying the developed algorithms and indicating the techniques used in each encoding step

Name	Prediction mode	Residue encoding
MMP	no prediction	multi-scale block matching
MMP-FP	explicit INTRA	multi-scale block matching
MMP-FP/DPCM	explicit INTRA/DPCM	multi-scale block matching
MMP-MED	implicit MED	multi-scale block matching
MMP-GAP	implicit GAP	multi-scale block matching
MMP-LSP	implicit LSP	multi-scale block matching
MMP-HistRest	explicit INTRA	block matching + histogram restriction
MMP-ResRemap	explicit INTRA	block matching + residue remapping
MMP-ErrorFeed	explicit INTRA	block matching + error feedback

Table 2 Lossless compression results in bits-per-pixel

Images	MMP	MMP-MED	MMP-GAP	MMP-LSP	MMP-FP
Barbara	6.331	5.452	5.339	**4.451**	4.519
cameraman	6.582	4.933	4.905	4.788	**4.772**
gold	6.544	5.020	4.853	**4.805**	4.947
Lena	6.043	4.546	4.506	**4.220**	4.297
PP1205	**2.385**	3.421	4.745	5.715	2.809
PP1209	2.709	2.997	4.856	5.418	**2.692**
shapes	1.841	1.104	1.458	2.174	**1.056**
average	4.634	3.925	4.380	4.510	**3.585**

Best results are highlighted in bold

plane and can use reconstructed values in lines below the current position, in case they were coded in previous blocks. In addition, since lossless coding provides perfect reconstruction, the block prediction modes could also be modified to use prediction neighbourhoods having pixels adjacent to the position to be predicted, therefore enhancing the intra-prediction capability. In the next subsection, we propose the use of an enhanced intra-block prediction mode for lossless compression.

3.2 Enhanced intra-prediction modes

High-performance lossless image coding algorithms usually perform image encoding on a pixel-by-pixel basis, using very small prediction neighbourhoods. The H.264/AVC lossless video coding standard adopted a modification to its horizontal and vertical intra-prediction modes to enhance compression performance [27]. For these modes, adjacent pixels are used for prediction, instead of the ones at the border of the upper and left block edges. This results in better coding efficiency since the adjacent neighbouring value is usually a better prediction than a pixel located at the block edge.

For example, in the lossless H.264/AVC horizontal prediction, a row of pixels $[x_1, x_2, x_3, x_4]$ is predicted using the pixels in adjacent positions, forming the prediction vector $[x_0, x_1, x_2, x_3]$, instead of the prediction vector generated from only the rightmost pixel of the previous block, that is $[x_0, x_0, x_0, x_0]$. So, the residues to be encoded are

$$\begin{aligned} r_1 &= x_1 - x_0 \\ r_2 &= x_2 - x_1 \\ r_3 &= x_3 - x_2 \\ r_4 &= x_4 - x_3 \end{aligned} \tag{2}$$

In the lossless compression case, MMP also adopts this intra

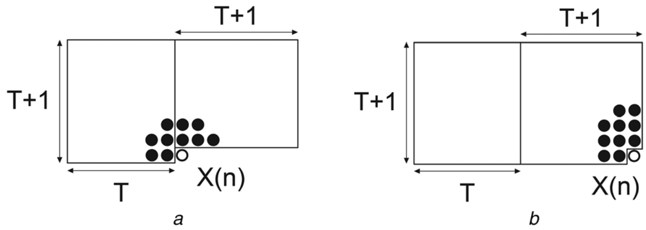


Fig. 2 *Least-square prediction*

a LSP training neighbourhood
b Modified LSP training neighbourhood for a border pixel

residual DPCM process for the horizontal and vertical prediction modes. With this modification, the algorithm tends to use larger block-sizes in the prediction step, therefore saving bits associated with the prediction information. Furthermore, such enhanced prediction favours the narrowing of the residue's histogram, which also helps the dictionary adaptation process [15].

3.3 Adaptive predictor for edges with arbitrary directions

The lossless H.264/AVC prediction, described above, works well for image features that have horizontal or vertical directions. For features with arbitrary directions, a method based on least-squares optimisation is more appropriate. In [28], an adaptive model based on least-squares minimisation was successfully added to the set of possible prediction modes used by the MMP encoder. We refer to it as the LSP. We have adapted such prediction mode, that has been previously used in [29], for block encoding and lossy compression. As shown in [28], its performance increases towards higher rates, so it is a natural candidate for use in lossless encoding.

In the LSP method, a filter produces an estimation of the the next pixel value $x(n, m)$, based on a linear combination of the N past samples. The coefficients of the filter (the weights of the linear combination) are chosen so that the mean-squared error between the actual and the predicted pixels is minimised.

Since the image's statistics vary spatially, the minimisation of the error is performed inside a window that slides through the image as the pixel position varies. This window is referred to as the analysis set or training window. Each pixel may be predicted using a different prediction filter, whose coefficients are chosen to minimise the prediction error for all the positions in the corresponding training window. An important issue with this predictor is the optimisation of the size and shape of the training windows, or, in other words, the optimisation of the appropriate model order and training neighbourhood.

The filter implements an Nth order Markovian model, and often the nearest pixels are used. A convenient choice of a training window is the double-rectangular window that contains $M = 2T(T+1)$ elements (see Fig. 2*a*). Let us define two indicator functions, $g(i)$ and $f(j)$. $g(i)$ provides the delta displacement between the position of a pixel inside the training window of size M and the position of the pixel to be filtered, indexed by i. $f(j)$ provides the delta displacement between the adjacent neighbouring pixels and the pixel to be estimated, in the Nth order Markovian model, indexed by j.

doi: 10.1049/iet-ipr.2012.0538

To express the training procedure using matrix notation, we arrange the pixels used in the training region in an $M \times 1$ column vector $\boldsymbol{y} = [X(\boldsymbol{n} - g(1))X(\boldsymbol{n} - g(M))]^{\mathrm{T}}$. As the prediction window slides through the M positions, the arrangement of the N adjacent neighbours of the local prediction support region may be arranged in vector, that stacked together form the $\boldsymbol{M} \times \boldsymbol{N}$ matrix

$$C = \begin{bmatrix} X(\boldsymbol{n} - f(1) - g(1)) & \cdots & X(\boldsymbol{n} - f(N) - g(1)) \\ \vdots & & \vdots \\ X(\boldsymbol{n} - f(1) - g(M)) & \cdots & X(\boldsymbol{n} - f(N) - g(M)) \end{bmatrix}$$

The prediction process is then equivalent to

$$C\boldsymbol{a} = \boldsymbol{y} \tag{3}$$

Since C is $M \times N$, and the size M of the training window is supposed to be larger than the window size N, the least-squares solution to this problem is obtained through the left pseudo-inverse [30], given by

$$\boldsymbol{a} = (C^{\mathrm{T}}C)^{-1}(C^{\mathrm{T}}\boldsymbol{y}) \tag{4}$$

In our simulations, using order $N = 10$ and training area $M = 112$, we obtained the best results for the majority of the tested images.

The neighbourhood and training area in [29], shown in Fig. 2*a*, assume a raster scan, so because of spatial causality only a non-symmetric half plane is available for prediction. Since in block coding this is not necessarily true for all the pixels inside the block, we modified the neighbourhood and training areas for the positions where the region of support of the predictors in the original training area would become non-causal. If the predictor mask of the pixel to be predicted is still inside a causal area, but its training area is not, then the training area is shifted to the left, until it is entirely inside the area solely composed of decoded pixels. However, if the support region for the predictor of the current position falls in the range where decoded pixels are not available (e.g. in the right border of the block), then the whole neighbourhood and training area are modified to use only pixels to the left of the position of interest, as shown in Fig. 2*b*.

Note that the differential pulse-code modulation (DPCM) technique described previously also apply for the LSP predictor, that is, we can also use the original values of the neighbouring pixels in the training phase and for the pixel estimation. The use of the DPCM technique with intra prediction modes (horizontal and vertical modes) and the LSP mode in the MMP framework defines the MMP-FP/DPCM algorithm.

3.4 Experimental results

Table 3 shows in the first column the first-order entropy of selected smooth and compound images. The next two columns show the entropy of the images after applying the indicated prediction methods only. Finally, the last two columns show the compression achieved using the MMP-FP algorithm and the MMP-FP/DPCM algorithm.

Table 3 Image entropy

Images	First order	LSP	Intra modes only	MMP-FP	MMP-FP/ DPCM
Barbara	6.331	5.339	4.451	4.519	**4.352**
cameraman	6.582	4.905	4.788	4.772	**4.570**
gold	6.544	4.853	4.805	4.947	**4.505**
Lena	6.043	4.506	4.220	4.297	**4.212**
PP1205	**2.385**	4.745	5.715	2.809	2.736
PP1209	2.709	4.856	5.418	2.692	**2.567**
shapes	1.841	1.458	2.174	1.056	**0.916**
average	4.634	4.380	4.510	3.585	**3.408**

Best results are highlighted in bold

The prediction stage is more efficient for smooth images when using LSP or the H.264/AVC prediction modes, but these methods are not suitable to predict compound images. However, when using the MMP-FP to code the residue, a high-compression rate can be achieved when encoding both smooth and compound images. Consistent gains were also reported by applying the DPCM modification, achieving the best compression results for all the tested images. The results show a drop of more than 1 bpp when using the MMP encoder compared to the first-order entropy, showing the efficiency of MMP's prediction scheme. When applying the proposed enhancements in the prediction stage of the algorithm, we have a drop of 0.18 bpp on average as well. This led us to adopt the DPCM adaptation of the intra-prediction modes.

4 Improving the residue coding stage of MMP's lossless compression algorithm

For the residue coding, MMP uses pattern matching with an adaptive dictionary updating scheme, which has proven to be very efficient when lossly coding smooth and compound images. Nevertheless, unlike most lossless image compression algorithms, in MMP the context used for entropy coding is not related to the neighbouring residual information. Furthermore, techniques for improving the probability model used for entropy coding, such as histogram truncation or error feedback are not part of the MMP algorithm, since they are usually applied for pixel coding, and not for block coding. In the next subsections, we evaluate the use, in the MMP lossless encoder, of some techniques that are traditionally used in lossless image encoders, such as error remapping [31], histogram restriction [32] and context quantisation [25].

4.1 Histogram restriction

The finite dimensionality of the input alphabet limits the range of possible prediction residue sample values. Then it is possible to remove the probability attributed to these forbidden values once the prediction is known, and to fine-tune the residue probability that is going to be used for residue coding. This technique is successfully applied in many lossless encoders, such as CALIC [25] and MRP [5]. This is similar to using a residue probability conditioned to the predicted value, that is, the residue histogram is restricted to values conditioned on the prediction value. The conditional probability of a residue e, given the predicted

doi: 10.1049/iet-ipr.2012.0538

value x_{pred}, can be obtained in the following manner

$$\begin{aligned}\Pr\left(e|x_{pred}\right) &= \frac{\Pr\left(e, x_{pred}\right)}{\Pr\left(x_{pred}\right)} \quad \text{(Baye's rule)} \\ &= \frac{\Pr\left(x_{pred}|e\right) * \Pr(e)}{\Pr\left(x_{pred}\right)} \\ &= \frac{\Pr(e)}{\left(\left(\Pr\left(x_{pred}\right)\right)/\left(\Pr\left(x_{pred}|e\right)\right)\right)}\end{aligned} \tag{5}$$

Since $x \in [0, 255]$, the possible residual values are within the interval $[-x_{pred}, 255 - x_{pred}]$. Therefore, the probability of the prediction x_{pred} in (5) can be replaced by the probability of the error assuming values $-x_{pred} \leq e \leq 255 - x_{pred}$, and we have

$$\frac{\Pr(e)}{((\Pr(-x_{pred} \leq e \leq 255 - x_{pred}))/\Pr(-x_{pred} \leq e \leq 255 - x_{pred}|e))} \tag{6}$$

Once the residue value e is known, we have that $\Pr(-x_{pred} \leq e \leq 255 - x_{pred}|e) = 1$, and the conditional probability of (6) becomes

$$\frac{\Pr(e)}{\sum_{x=0}^{255} \Pr\left(x - x_{pred}\right)} \tag{7}$$

which indicates that the probability of the forbidden residues assumes a zero value and the remaining probabilities are rescaled, so that the sum of residue probabilities is equal to one.

The same principle was implemented for the MMP residue encoding. Given the block prediction value, the codewords of the dictionary that correspond to forbidden residue patterns assumed a zero value probability, sin order to use a probability distribution that takes into account only the possible residue patterns. The new adjusted probability distribution is used by the arithmetic encoder for entropy coding of the chosen dictionary codeword. Note, however, that the use of this technique requires the knowledge of the predicted value for residue encoding, which is not possible when using the DPCM technique, described in Section 3. Therefore this technique is mutually exclusive with the DPCM enhancement prediction method.

4.2 Residue remapping

The pixels of an 8 bit image can have 256 different values. The difference between prediction and the actual pixel can assume values between -255 and 255, which needs 9 bits for its representation. However, as already stated, the possible values assumed by the residue are found in the interval $[-x_{pred}, 255 - x_{pred}]$, that contains 256 values, and can be represented by 8 bits only. A common procedure used in lossless encoders, such as CALIC [25] or JPEG-LS [2] is the residue remapping. A typical residue remapping function for a residue $\delta = x - x_{pred}$ is as follows

$$\mathcal{M}^{-+}\left(x, x_{pred}\right) = \begin{cases} 2\delta & \text{if } 0 \leq \delta \leq \theta \\ 2|\delta| - 1 & \text{if } -\theta \leq \delta \leq 0 \\ \theta + |\delta| & \text{otherwise} \end{cases} \tag{8}$$

where $\theta = \min(x_{pred}, 255 - x_{pred})$.

Residue remapping reduces the alphabet size for entropy coding by merging the tails of distributions with their central part. For a highly peaked distribution centred at the predicted value, the remapping function does not alter the shape of the residue distribution. On the other hand, for a bimodal distribution, the remapping function will merge the probability tails, enhancing the probability of a value near the edge of the allowed interval. A revision of different remapping functions can also be found in [31]. In the case of the MMP algorithm, the residue remapping was applied to each coordinate of the multidimensional residue block after block prediction. The new remapped block is then used for pattern matching.

4.3 Error feedback

The error feedback technique, used among others in JPEG-LS [2] and CALIC [25] encoder, consists of adaptively adjusting the predicted value in accordance to the residue neighbourhood. The value that is used for updating is usually an average value of the neighbouring residues, and a context is also used to increase adaptation speed of the mean value.

For the MMP algorithm, the same principle can be applied. The average values of the residues are later used for updating the block prediction value. The block dimension is used as context, as well as the chosen prediction mode. For example, a horizontal prediction block value is updated with average values of residues obtained from the same block dimension after horizontal prediction.

4.4 Experimental results

The techniques described in the previous sections are expected to adjust the probability value used for the coding stage, making the encoder adapt faster to the residue's local statistic. For example, in context quantisation, a pixel neighbourhood generates a context that is used for coding the next residue, which is likely to have a similar value to its neighbours. The histogram restriction will eliminate the unlikely residual values, and will eventually enhance the probability of the actual residue. Error remapping will try to merge different probability distributions into one single distribution, making error encoding more efficient. It is important to note that most of these techniques were shown to be successful on residues determined on a pixel-by-pixel basis.

These enhancement techniques have been tested in the MMP framework. However, the tested techniques did not provide any consistent gains over the MMP-FP algorithm using the DPCM enhancement on the intra prediction modes, as can be seen in Table 4. The results presented here show little to no improvement with the added techniques. For smooth images, such as Barbara and Lena, the prediction stage is effective and patterns present in the dictionary tend to be similar and to be clustered around zero. Therefore we can conclude that an improvement in the coding stage does not have potential for increasing the

doi: 10.1049/iet-ipr.2012.0538

Table 4 Lossless compression results for several different residue encoding techniques, given in bits-per-pixel

Images	MMP-HistRest	MMP-ResRemap	MMP-ErrorFeed	MMP-FP/DPCM
Barbara	4.427	4.423	4.587	**4.352**
cameraman	4.631	4.640	4.707	**4.570**
gold	4.711	4.833	4.914	**4.505**
Lena	4.347	4.227	4.317	**4.212**
PP1205	**2.696**	2.706	2.717	2.736
PP1209	2.623	2.724	2.678	**2.567**
shapes	1.040	0.970	**0.907**	0.916
average	3.496	3.503	3.547	**3.408**

Best results are highlighted in bold

coding gain. For non-smooth images, the presented technique might present some gains, but they are not enough to justify the adoption of the tested techniques over the DPCM technique. Since these are mutually exclusive, the DPCM enhancement technique was adopted instead of the residue enhancement techniques.

The use of a multi-scale adaptive dictionary limits the performance of such techniques. The dictionary in multiple scales can be regarded as a joint encoding of residues of several pixels [33], so the spatial statistical dependencies are already exploited by the encoding procedure. Also the adaptive updating of the dictionary only introduces patterns that obey to the restriction of predicted values, which are patterns close to zero. Since the scale adaptive block-matching is one of the key features of MMP's performance, we maintained the multi-scale structure and allowed the dictionary adaptation to handle the image encoding step, instead of using encoding techniques based on a pixel-by-pixel approach. Therefore no entropy adaptation technique was adopted by the MMP lossless algorithm, and the arithmetic encoder in the lossless version is exactly the same as in the lossy version. This provides the advantage of the encoding procedures for lossless and lossy coding being equal.

5 MMP encoding results

In this section, results for the proposed MMP-based lossless compression algorithm are presented along with its lossy coding results. A comparison with other algorithms considered as state-of-the-art in their respective fields is also presented.

The test set used for the simulations also incorporates many different types of images, from smooth natural images to artificial and compound images obtained by scanning documents. These images have been chosen to allow the assessment of the algorithms' compression performance for a wide class of images. All the tested images and MMP source code are available online [34].

5.1 Results for lossless compression

In the lossless case, we chose three different algorithms to compare with MMP: JPEG-LS [4], MRP [5] and PNG [1]. JPEG-LS [2] is a benchmark for lossless compression and uses implicit prediction and adaptive Golomb codes for residue encoding to efficiently compress the image in a single pass. The used encoder is available at http://www.hpl.hp.com/loco [35]. It allows an overall good compression performance for smooth images with relatively small computational cost. MRP [5] uses explicit prediction and adaptive residue encoding on a multiple-pass optimisation routine. Although computationally intensive, this algorithm produces one of the best compression rates reported for a large set of images, outperforming most other lossless algorithms. Our results were obtained with the optimisation flag for the predictors set, using the encoder available at http://www.itohws03.ee.noda.sut.ac.jp/matsuda/mrp/mrp-05.tar.gz [36]. PNG [1] uses a combination of prediction and pattern matching, which produces relatively good results, especially for compound images when compared with other algorithms. It was selected because, as MMP, it is dictionary-based, resembling the LZ77 algorithm [6]. However, unlike MMP, which employs a block-based approach, PNG [1] employs a pixel-based approach. We used the PNG [1] software implementation available at http://www.libpng.org/pub/png/apps/pnmtopng.html [37], and the results were further optimised with the 'pngcrush' tool [38].

Tables 5 and 6 show the compression results, in bits per pixel, for all four encoders.

Table 5 Lossless compression for smooth images

Image	JPEG-LS	MRP	PNG	MMP
airplane	3.817	**3.591**	4.220	3.943
baboon	6.037	**5.663**	6.210	6.028
balloon	2.904	**2.579**	3.238	2.830
Barb	4.691	**3.815**	5.199	4.352
Barb2	4.686	**4.216**	5.132	4.659
camera	4.314	**3.949**	4.674	4.570
couple	3.699	**3.388**	4.061	3.915
goldhill	4.477	**4.207**	4.662	4.505
Lena	4.607	**4.280**	4.888	4.489
Lennagrey	4.238	**3.889**	4.577	4.212
noisesquare	5.683	**5.270**	5.671	5.365
peppers	4.513	**4.199**	4.887	4.500
average	4.472	**4.087**	4.785	4.447

Results in bits-per-pixel. Best results are highlighted in bold

Table 6 Lossless compression for compound and artificial images

Image	JPEG-LS	MRP	PNG	MMP
pp1205	4.472	4.039	**2.550**	2.736
pp1209	4.616	3.903	2.785	**2.567**
scan_002	3.429	**3.029**	3.729	3.471
scan_004	2.370	**2.041**	2.404	2.224
scan_006	3.202	**2.844**	3.037	2.943
shapes	1.214	**0.685**	1.154	0.916
average	3.217	2.757	2.610	**2.476**

Results in bits-per-pixel. Best results are highlighted in bold

In the case of smooth images, the residue after prediction shows a probability distribution that is well fitted by an exponentially decaying model. JPEG-LS [2] and MRP [5] use this assumption, in order to capture the residues structure using models such as generalised Gaussian models and to obtain good compression results for smooth images, as shown in Table 5. PNG [1], on the other hand, does not explore this smoothness property and cannot compress such images as efficiently. Since PNG [1] codes its residue using a dictionary approach, despite being asymptotically optimum, it takes too long to adapt its dictionary to the image statistics. MMP also uses a dictionary approach for residue encoding. However, as shown in Table 5, its compression performance is compatible to the ones of JPEG-LS [2] and MRP [5], because of its faster dictionary adaption to the image statistics [15].

Algorithms that model the residue using a specific probability function like the exponential distribution in JPEG-LS [2], usually have a good performance for smooth images. The exponential assumption fits relatively well the residue distribution after prediction for these types of images. However, in the case of compound images, the smoothness assumption does not hold, and thus the prediction is not effective. Therefore, for compound images, the residues distribution does not match an exponentially decaying function. Then, the compression efficiency of MRP [5] and JPEG-LS [2] for some compound images can be lower than the one of pattern matching algorithms like PNG [1], which does not assume any distribution for the residual data, and adapts itself as the image is being coded. In Table 6, we can see that, similar to PNG [1], MMP is also able to achieve competitive compression performance for compound images.

The MMP prediction step, combined with flexible partitioning, is able to leverage its dictionary adaptation process and provide a competitive performance for smooth images and compound images. Our results suggest that MMP has a good overall performance when considering a wide range of image types, being able to rapidly adapt to any type of image's characteristic without compromising the coding efficiency.

5.2 Results for lossy compression

In this section, we will review some of MMP's results for lossy coding, already presented in [28], but providing a deeper analysis on the influence of the LSP prediction mode. It is important to note that the LSP prediction mode is also used in the lossless case, and that the evaluated lossy algorithm is the same as the one used for lossless encoding. Instead of minimising the rate for zero distortion, we make it lossy by minimising the Lagrangian cost $D + \lambda R$.

For the lossy case, we selected for comparison two algorithms that use the transform-quantise-encode paradigm and are considered as state-of-the-art: JPEG2000 [39] and H.264/AVC Intra [40]. JPEG2000 applies a wavelet transform and quantises its coefficients using the EBCOT algorithm, being considered a reference for wavelet-based encoders. H.264/AVC FRExt 'high profile' has been reported to give excellent rate-distortion performance for

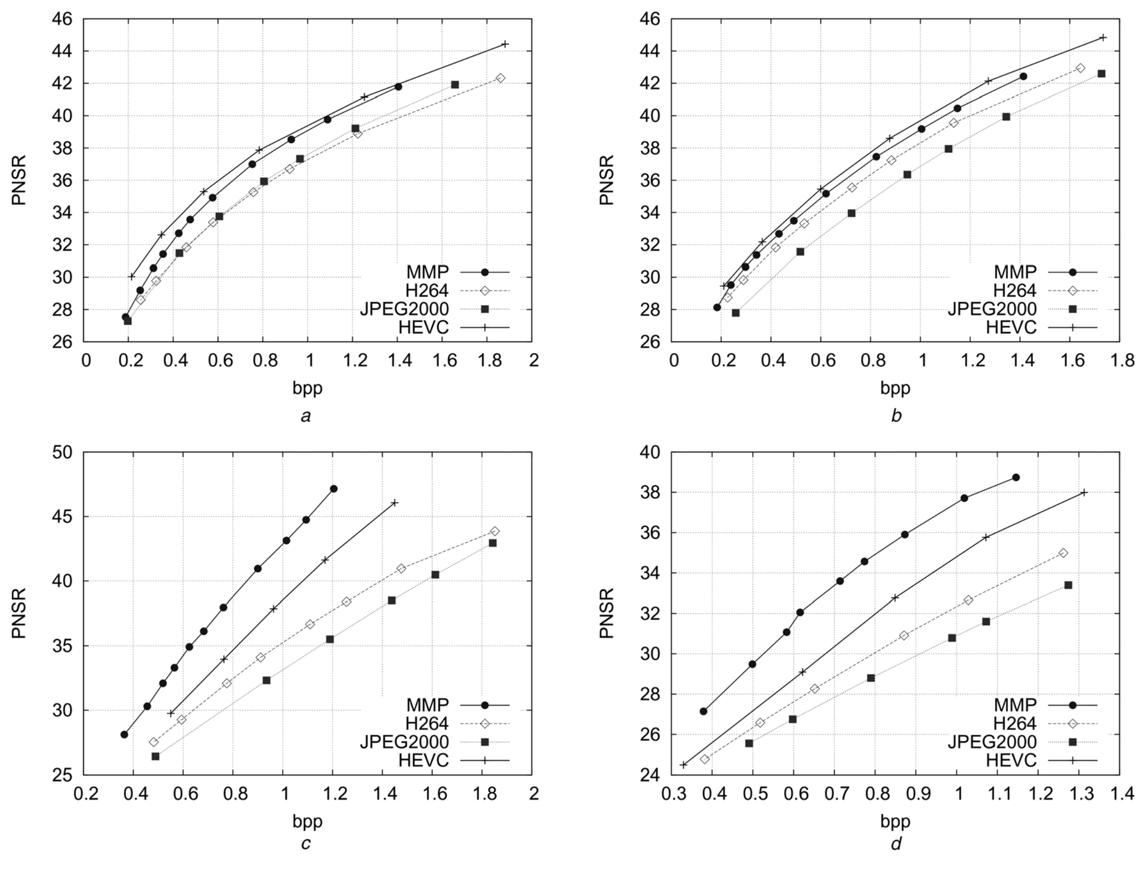


Fig. 3 *Rate-distortion curves for several images*

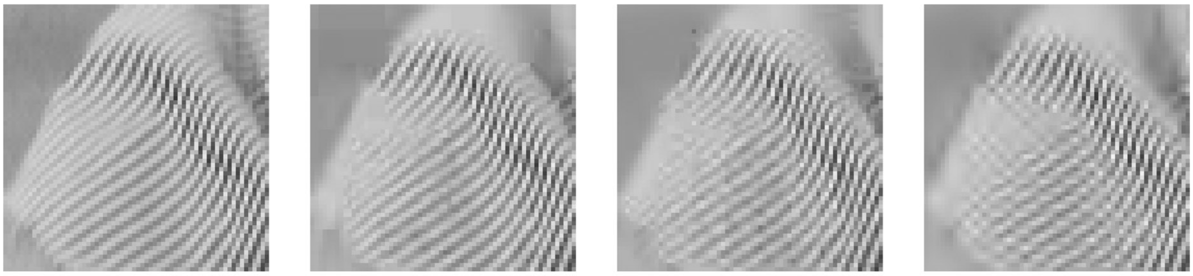
a Barbara
b Cameraman
c SCAN_0004_TOP
d PP1205

doi: 10.1049/iet-ipr.2012.0538

image encoding [41]. It segments an image into non-overlapping blocks and uses the predictions that inspired MMP's prediction step, followed by a transform-quantisation-encode operation. For completeness, we also provide results with the new high-efficiency video coding algorithm (HEVC [42]), with the remark that HEVC uses block sizes larger than the 16×16 block sizes of the other methods.

Fig. 3 shows the PSNR curves for the three selected algorithms with various types of images, from smooth to compound. For all tested images, MMP has outperformed both JPEG2000 and H.264/AVC algorithms from low-to-high rates, and have a comparable RD performance with HEVC for high bitrates. For the specific case of compound images, where the assumption of image smoothness does not hold, the MMP algorithm is able to obtain PSNR results 2 dB higher than those of HEVC, H.264/AVC and JPEG2000. For smooth images, the benefit of the LSP prediction is noticeable, and we obtain PSNR performances that are 1–2 dB higher than H.264/AVC, specially for higher rates, where the training of the LSP predictors improves because of a more accurately reconstructed neighbourhood.

Figs. 4–6 show details of the original and recovered images after compression. Rates and nominal PSNR values are given in the labels of each figure.



a b c d

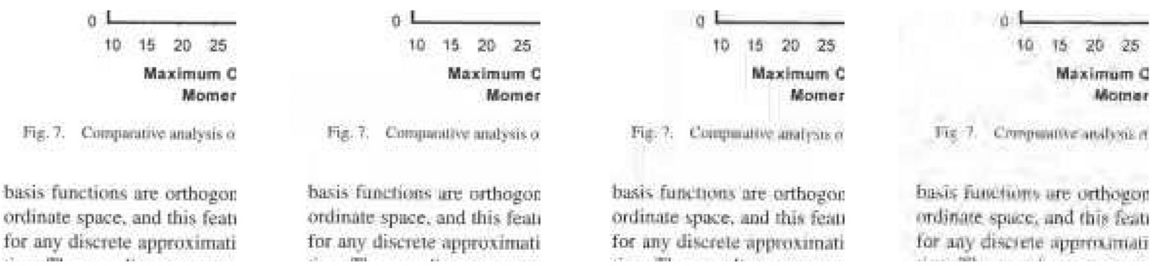
Fig. 4 *Encoding details for the texture in the cloth above Barbara's right shoulder, images encoded at 0.5 bpp*

a Original
b MMP – 33.56 dB
c H.264 – 31.84 dB
d JPEG2000 – 31.48 dB

a b

Fig. 5 *Results for the deblocking filter for Barbara image encoded at at 0.5 bpp*

The PSNR without deblocking is 33.55 dB and the PSNR with deblocking filter is 33.56 dB
a Without deblocking
b With deblocking



a b c d

Fig. 6 *Encoding details for a text extract from the Image SCAN_0004_TOP, at 0.5b pp*

a Original
b MMP – 32.08 dB
c H.264 – 28.76 dB
d JPEG2000 – 26.43 dB

For smooth images, algorithms such as JPEG2000 and H.264/AVC already have an excellent performance, but when sharp edges appear on the image, like those visible on the detail in Fig. 4*a*, these edges suffer from the high-frequency suppression caused by the quantisation process. For the JPEG2000-encoded image Fig. 4*d*, the ringing artefacts are visible and the fine texture of the cloth above the girl's shoulder is severely damaged. Also some parts of the texture are blurred. For the H.264/AVC compressed image in Fig. 4*c*, one can see a better subjective quality because of the deblocking filter, which prevents blurring effects on the fine textured areas. However, artefacts on the texture pattern can still be identified. Since the edge directions are not exactly the same as the directions defined by H.264/AVC predictions, there is a discontinuity of the lines from the texture, especially on the blocks' border. This is so because the prediction residues present high-frequency components that are eliminated by the quantisation process, generating the perceptible artefacts. For the MMP image in Fig. 4*b*, this effect is avoided by the LSP prediction, which adaptively adjusts to an arbitrarily-oriented edge direction, resulting in a much lower energy residue and a more efficient residue encoding. On Fig. 5*a*, one can note that MMP generates blocking effects at smooth parts of the image. This can be avoided by applying a deblocking filter, like the one proposed in [43] for the MMP algorithm. Fig. 5*b* shows that the image post-processed with the deblocking filter is indeed much smoother. Note also that the deblocked version has no PSNR loss compared to the original one.

For the compound images in Fig. 6, MMP gives the best subjective quality. Scanned images generally present localised high-frequency components for the letters and a smooth background. Document compression algorithms usually segment the image into smooth and textual areas and apply specific coding procedures to each area, but the scanning process tends to introduce artefacts on the background and on the fine text that makes the segmentation problem a non-trivial one. In the image encoded with MMP in Fig. 6*b*, we can see that the high-frequency details of the letters are still preserved, keeping the text readable. Since MMP uses pattern matching, it is especially suited for scanned documents, and thereby encodes the repeated letter patterns in a very efficient manner. For images encoded with H.264/AVC, as shown in Fig. 6*c*, the deblocking filter cannot effectively identify the fine structures and ends up blurring parts of the image. The image encoded with JPEG2000, in Fig. 6*d*, suffers from ringing artefacts and the text readability is compromised. For a more comprehensive study on a lossy document encoding version of MMP without the LSP prediction, we refer to [44].

6 Conclusions

In this paper, we evaluated the performance of MMP acting as a lossless encoder. We assessed several well-known lossless coding techniques and implemented them in the MMP framework such as lossless prediction modes and residue coding techniques. Our results showed that the MMP algorithm (enhanced using LSP and the DPCM mode) was able to achieve comparable performance to state-of-the-art lossless encoders for any type of image, without the need of a pre-analysis of the image or adaptation of its encoding procedure.

The experimental results also show that the same algorithm, when applied to lossy encoding, gives good PSNR performance for a wide range of image types, from smooth to textual images, outperforming in many cases encoders such as H.264/AVC Intra and JPEG2000, and having comparable performance as state-of-the-art encoders such as HEVC. Therefore MMP can be viewed as an universal tool for image encoding, that can be used for both lossy and lossless compression.

The coding approach introduced by the MMP algorithm, using a flexible prediction scheme followed by multi-scale pattern matching was shown to be an efficient tool for lossy and lossless encoding of a large variety of image classes. Naturally the adaptation of the dictionary comes at a cost. Since the dictionary is updated as the picture is being coded, the compression performance may not be optimal at the beginning of the coding process. However, the results presented in this work showed that MMP was able to still achieve a competitive performance, because of its fast multi-scale dictionary adaptation. Moreover, the scheme proved to be very efficient also when the modelling stage is not successful, such as in the cases of scanned documents. The residue coding step was able to cope with the misfit of the prediction stage and still achieve competitive compression results. Another important remark is the fact that the algorithm used was the same for lossy and lossless compression. Although many of the state-of-the-art lossless compression algorithms do not present a lossy counterpart and require a pre-coding stage for the analysis of a picture, MMP does its adaptation process while coding the picture, and the results presented here show that the method achieves competitive compression ratios for any required bit rate.

One remaining challenge is the computational complexity of MMP. Its coding procedure has a high-computational cost. Smart encoding techniques and parallel computation are some of the topics we are currently investigating, in order to achieve a significant reduction in encoding time.

All the code used for the simulations and the results are available online and can be found in [34].

7 Acknowledgments

This work has been supported by the Fundação Para a Ciência e Tecnologia (FCT), under project PTDC/EEA-TEL/66462/2006/, and Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), under the program PDEE (Programa de Doutorado no País com Estágio no Exterior), process 5443-09-1.

8 References

1 Sayood, K.: 'Lossless compression handbook' (Academic Press Series in Communications, Networking and Multimedia, 2003)

2 Weinberger, M.J., Seroussi, G., Sapiro, G.: 'The LOCO-I lossless image compression algorithm: principles and standardization into JPEG-LS', *IEEE Trans. Image Process.*, 2000, **9**, (8), pp. 1309–1324

3 Wu, X.: 'An algorithmic study on lossless image compression'. Proc. Data Compression Conf., Snowbird, USA, March 1996, pp. 150–159

4 Weinberger, M.J., Rissanen, J., Arps, R.: 'Applications of universal context modeling to lossless compression of gray-scale images', *IEEE Trans. Image Process.*, 1996, **5**, (4), pp. 575–586

5 Matsuda, I., Kaneko, T., Minezawa, A., Itoh, S.: 'Lossless coding of color images using block-adaptive inter-color prediction'. Proc. Int. Conf. Image Processing, San Antonio, Texas, USA, September 2007, pp. 329–332

6 Ziv, J., Lempel, A.: 'A universal algorithm for sequential data compression', *IEEE Trans. Inf. Theory*, 1977, **23**, (3), pp. 337–343

doi: 10.1049/iet-ipr.2012.0538

7 Welch, T.A.: 'A technique for high performance data compression', *IEEE Comput.*, 1984, **17**, (6), pp. 8–19
8 Miller, V., Wegman, M.: 'Variations on a scheme by Ziv and Lempel', Combinatorial Algorithms on Words, NATO ASI Series, 1984, Vol. F12, pp. 131–140
9 de Carvalho, M.B., Finamore, W.A.: 'Lossy Lempel-Ziv on subband coding of images'. Proc. IEEE Int. Symp. on Information Theory, Trondheim, Norway, July 1994, pp. 415
10 Chan, C., Vetterli, M.: 'Lossy compression of individual signals based on string matching and one pass codebook design'. Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing, Detroit, USA, May 1995, pp. 9–12
11 Atallah, M., Genin, Y., Szpankowski, W.: 'Pattern matching image compression: algorithmic and empirical results', *IEEE Trans. Pattern Anal. Mach. Intell.*, 1999, **21**, pp. 21–618
12 Yang, E.H., Kieffer, J.C.: 'Efficient universal lossless data compression algorithms based on a greedy sequential grammar transform – Part I: without context models', *IEEE Trans. Inf. Theory*, 2000, **46**, (3), pp. 755–777
13 Randers-Pehrson, G.: 'PNG (Portable Network Graphics) Specification, Version 1.2'. 1999
14 de Carvalho, M.B., da Silva, E.A.B., Finamore, W.: 'Multidimensional signal compression using multiscale recurrent patterns', *Elsevier Signal Process.*, 2002, **82**, (11), pp. 1559–1580
15 Rodrigues, N.M.M., da Silva, E.A.B., de Carvalho, M.B., de Faria, S.M. M., Silva, V.M.M.: 'On dictionary adaptation for recurrent pattern image coding', *IEEE Trans. Image Process.*, 2008, **17**, (9), pp. 1640–1653
16 Francisco, N.C., Rodrigues, N.M.M., da Silva, E.A.B., *et al*: 'Multiscale recurrent pattern image coding with a flexible partition scheme'. Proc. IEEE Int. Conf. Image Processing, San Diego, USA, October 2008, pp. 141–144
17 de Carvalho, M.B., da Silva, E.A.B.: 'A universal multi-dimensional lossy compression algorithm'. Proc. IEEE Int. Conf. Image Processing, Kobe, Japan, October 1999, pp. 767–771
18 de Lima Filho, E.B., da Silva, E.A.B., de Carvalho, M.B., Pinage, F.S.: 'Universal image compression using multiscale recurrent patterns with adaptive probability model', *IEEE Trans. Image Process.*, 2008, **17**, (4), pp. 512–527
19 Duarte, M.H.V., de Carvalho, M.B., da Silva, E.A.B., Pagliari, C., Mendonça, G.V.: 'Stereo image coding using multiscale recurrent patterns'. Proc. IEEE Int. Conf. Image Processing, Rochester, USA, September 2002, pp. II661–664
20 Rodrigues, N.M.M., da Silva, E.A.B., de Carvalho, M.B., de Faria, S.M. M., Silva, V.M.M.: 'Improving H.264/AVC inter compression with multiscale recurrent patterns'. Proc. IEEE Int. Conf. Image Processing, Atlanta, USA, October 2006, pp. 1353–1356
21 Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG (ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6): 'Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC), Fidelity Range Extensions documents JVT-L047 (non-integrated form) and JVT-L050 (integrated form)', 2004
22 Sullivan, G.J., Baker, R.L.: 'Rate-distortion optimization for tree-structured source coding with multi-way node decisions'. Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing, San Francisco, USA, March 1992, pp. 393–396
23 Cover, T.M., Thomas, J.A.: 'Elements of information theory' (Wiley-Interscience, 1991, 99th edn.)
24 Rodrigues, N.M.M., da Silva, E.A.B., de Carvalho, M.B., de Faria, S.M. M., Silva, V.M.M.: 'Universal image coding using multiscale recurrent patterns and prediction'. Proc. IEEE Int. Conf. Image Processing, Genova, Italy, September 2005, pp. II-24520138
25 ISO/IEC JTC 1/SC 29/WC 1: 'A context-based, adaptive, lossless/ nearly-lossless coding scheme for continuous-tone images', 1995
26 Li, X., Orchard, M.T.: 'Edge directed prediction for lossless compression of natural images'. Proc. IEEE Int. Conf. Image Processing, Kobe, Japan, October 1999, pp. 58–62
27 Lee, Y.L., Han, K.H., Sullivan, G.J.: 'Improved lossless intra coding for H.264/MPEG-4 AVC', *IEEE Trans. Image Process.*, 2006, **15**, (9), pp. 2610–2615
28 Graziosi, D.B., Rodrigues, N.M.M., da Silva, E.A.B., de Carvalho, M. B., de Faria, S.M.M., Silva, V.M.M.: 'Improving multiscale recurrent pattern image coding with least-squares prediction mode'. Proc. IEEE Int. Conf. Image Processing, Cairo, Egypt, November 2009, pp. 2813–2816
29 Li, X., Orchard, M.T.: 'Edge-directed prediction for lossless compression of natural images', *IEEE Trans. Image Process.*, 2001, **10**, (6), pp. 813–817
30 Campbell, S.L., Meyer, C.D.: 'Generalized inverses of linear transformations' (Dover, New York, 1991, 1st edn.)
31 Rice, R.F.: 'Technical Report, JPL-79-22, JPL-83-17, and JPL-91-3: some practical universal noiseless coding techniques-parts I–III'. 1983
32 Wu, X., Memon, N.: 'Context-based adaptive lossless image coding', *IEEE Trans. Commun.*, 1997, **45**, (4), pp. 437–444
33 Rodrigues, N.M.M., da Silva, E.A.B., de Carvalho, M.B., de Faria, S.M. M., Silva, V.M.M.: 'Basis optimisation for multiscale recurrent pattern coding'. Proc. Conf. Mathematical Techniques and Problems in Telecommunications, Leiria, Portugal, September 2006
34 http://www.lps.ufrj.br/profs/eduardo/mmp/, accessed January 2013
35 http://www.hpl.hp.com/loco, accessed January 2013
36 http://www.itohws03.ee.noda.sut.ac.jp/matsuda/mrp/mrp-05.tar.gz, accessed January 2013
37 http://www.libpng.org/pub/png/apps/pnmtopng.html, accessed January 2013
38 http://pmt.sourceforge.net/pngcrush/, accessed January 2013
39 Taubman, D.S., Marcelin, M.W.: 'JPEG2000: image compression fundamentals, standards and practice' (Kluwer Academic Publishers, 2001)
40 Luthra, A., Sullivan, G.J., Wiegand, T.: 'Special issue on the H.264/ AVC video coding standard', *IEEE Trans. Circuits Syst. Video Technol.*, 2003, **13**, (7), pp. 557–559
41 Marpe, D., Wiegand, T., Gordon, S.: 'H.264/MPEG4-AVC fidelity range extensions: tools, profiles, performance, and application areas'. Proc. IEEE Int. Conf. Image Processing, Genova, Italy, September 2005, pp. 593–596
42 Pourazad, M.T., Doutre, C., Azimi, M., Nasiopoulos, P.: 'HEVC: the new gold standard for video compression', *IEEE Consumer Electron. Mag.*, 2012, **1**, (3), pp. 36–46
43 Rodrigues, N.M.M., da Silva, E.A.B., de Carvalho, M.B., de Faria, S.M. M., Silva, V.M.M.: 'Improving multiscale recurrent pattern image coding with deblocking filtering'. Proc. Third Int. Conf. E-business and Telecommunication Networks, Setúbal, Portugal, August 2006, pp. 224–236
44 Francisco, N.C., Rodrigues, N.M.M., da Silva, E.A.B., de Carvalho, M. B., de Faria, S.M.M., Silva, V.M.M.: 'Scanned compound document encoding using multiscale recurrent patterns', *IEEE Trans. Image Process.*, 2010, **19**, (10), pp. 2712–2724