Improving Multiscale Recurrent Pattern Image Coding with Enhanced Dictionary Updating Strategies

Nuno M. M. Rodrigues, Eduardo A. B. da Silva, Murilo B. de Carvalho,
Sérgio M. M. de Faria, Vitor M. M. da Silva

***Abstract*— The Multidimensional Multiscale Parser (MMP) is a lossy multidimensional signal encoder, that uses an adaptive dictionary for approximating the original signal using multiscale recurrent pattern matching. In previous work we have shown the efficiency of MMP for image coding and we have also described new techniques to improve its performance, using predictive coding (MMP-Intra) and innovative strategies for reducing the dictionary redundancy. The combination of these methods for image coding achieves much better results than the state-of-the-art JPEG2000 and H.264/AVC Intra image encoders for text and compound images, but for smooth natural images it still presents small losses.**

In this work we present a new technique to improve the dictionary adaptation process of the MMP-Intra, based on enhanced updating techniques. Experimental results have showed that, when combined with dictionary growth control methods, this technique achieves consistent image quality gains for all image types. Furthermore we present some methods that eliminate the intrinsic substantial increase of the computational complexity associated with more rapidly growing dictionaries, without compromising the final quality of the decoded image.

***Index Terms*— Multidimensional Multiscale Parser, MMP-Intra, multiscale recurrent pattern matching, adaptive dictionary, image coding.**

I. INTRODUCTION

In this paper we present recent results in the development of the MMP-Intra image coding method [1]. MMP-Intra combines the Multidimensional Multiscale Parser (MMP) algorithm [2] with predictive coding techniques, like the ones defined by the H.264/AVC video coding standard for Intra-frame coding.

MMP-Intra, as the original MMP, uses approximate block matching with scales and an adaptive dictionary, that stores pixel blocks that are used to encode predicted residues or the original image patterns. Previous studies demonstrated that MMP-Intra allows for relevant gains over the original MMP for image coding applications. In [3] a new dictionary design technique for MMP-Intra was presented, that resulted in further gains in the coding results of MMP-Intra.

For non-low pass images, like text and compound (text and graphics) images, MMP-Intra improves the excellent performance of the original MMP, achieving impressive gains (up to 5 dB) over the state-of-the-art, transform-quantisation based image encoders, like JPEG2000 [4] and H.264/AVC Intra-frame image encoder [5]. For smooth natural images, where the performance of the original MMP is inferior to that of the JPEG2000 and H.264/AVC encoders, the use of MMP-Intra increases the decoded images' PSNR by up to 2 dB. This closes the gap between the performance of MMP-Intra and that of the best transform-based encoders to less than 0.5 dB. This comparative performance makes MMP-Intra an universally efficient image encoder, making it a viable alternative to transform-based image encoders.

Recent studies on MMP-Intra, described in this paper, focus on new ways to improve the approximation power of the dictionary, particularly for smooth images, with the objective of improving on the performance of the top image encoders also for smooth images, while maintaining or increasing the present coding performance advantage for non-smooth images.

To achieve this, new dictionary adaptation strategies were devised that allow for a quicker and more efficient adaptation of the dictionary patterns. Experimental results, presented in the final sections of this paper, show that the proposed method allows for a consistent gain in the MMP-Intra performance for all image types.

The faster growing rate of the dictionary has the immediate effect of increasing the algorithm's complexity. Along with a more elaborate discussion of this topic, we propose a method that efficiently reduces this additional complexity without any performance losses. In fact, our simulations demonstrated that this complexity reduction method even introduced some minor performance gains for some image types. Finally, a new context adaptive dictionary segmentation method based on the new updating strategies is also discussed.

Section II presents a short summary of the state-of-the-art of MMP-based methods. A short introduction of each of these methods is followed by a brief presentation of some previous results in dictionary growth control techniques. Section III presents the proposed techniques for dictionary adaptation and computational complexity reduction, along with the experimental results for each proposed method. Finally, some closing remarks and conclusions are presented in section IV.

II. THE MMP-BASED ALGORITHMS

This section presents the MMP and MMP-Intra algorithms. An exhaustive description of these methods can be found

Nuno Rodrigues is with Instituto de Telecomunicações (IT) and Escola Superior de Tecnologia e Gestão, Inst. Politécnico de Leiria (ESTG) - Portugal, e-mail: nuno.rodrigues@co.it.pt. Eduardo da Silva is with PEE/COPPE/DEL/Poli, Univ. Fed. Rio de Janeiro, Brazil, e-mail: eduardo@lps.ufrj.br. Murilo de Carvalho is with TET/CTC, Univ. Fed. Fluminense, Brazil, e-mail: murilo@telecom.uff.br. Sérgio Faria is with IT and ESTG Leiria - Portugal, e-mail: sergio.faria@co.it.pt. Vitor Silva is with IT and DEEC, Universidade de Coimbra, Portugal, e-mail: vitor.silva@co.it.pt.

respectively in [2] and [1]. In section II-C we present a brief summary of the existing dictionary redundancy control techniques, that are relevant for the methods proposed in this paper. Detailed information about these methods can be found in [3].

A. The MMP algorithm.

MMP encodes each original image block by approximating it with a vector from an *adaptive dictionary* $\mathcal{D}$. This is done using *different scales*, meaning that blocks of different dimensions can be approximated by this procedure. These dimensions correspond to successive binary segmentations of an original square block, first in the vertical, then in the horizontal direction.

MMP can be summarised by the following main steps:
For each block of the original image, $\mathbf{X}^{l}$[1]:

1) find the dictionary element $\mathbf{S}_i^l$ that minimises the Lagrangian cost function of the approximation, given by: $J(\mathcal{T}) = D(\mathbf{X}^l, \mathbf{S}_i^l) + \lambda R(\mathbf{S}_i^l)$, where $D(.)$ is the sum of square differences (SSD) function and $R(.)$ is the rate needed to encode the approximation;
2) parse the original block into two blocks, $\mathbf{X}_1^{l-1}$ and $\mathbf{X}_2^{l-1}$, with half the pixels of the original block;
3) apply the algorithm recursively to $\mathbf{X}_1^{l-1}$ and $\mathbf{X}_2^{l-1}$, until level 0 is reached;
4) based on the values of the cost functions determined in the previous steps, decide whether to segment the original block or not;
5) if the block should not be segmented, use vector $\mathbf{S}_i^l$ of the dictionary to approximate $\mathbf{X}^l$;
6) else
   a) create a new vector $\mathbf{S}_{new}^l$ from the *concatenation* of the vectors used to approximate each half of the original block: $\mathbf{X}_1^{l-1}$ and $\mathbf{X}_2^{l-1}$;
   b) use $\mathbf{S}_{new}^l$ to approximate $\mathbf{X}^l$;
   c) use $\mathbf{S}_{new}^l$ to *update* the dictionary, making it available to encode future blocks of the image.

When applied recursively, this algorithm generates a binary segmentation tree for each original block, that is encoded using two binary flags ('0' for the tree nodes, or block segmentations and '1' for tree leafs, or unsegmented blocks).

This binary tree is encoded using a top-bottom preorder approach. In the final bit-stream, each leaf flag is followed by an index, that identifies the vector of the dictionary that should be used to approximate the corresponding sub-block. These items are encoded using an adaptive arithmetic encoder.

Unlike conventional vector quantisation (VQ) algorithms, MMP uses *approximate block matching with scales* and an *adaptive dictionary*.

The use of an adaptive dictionary is illustrated by the final step of the previous algorithm. Every segmentation of a block from level l originates the concatenation of two dictionary blocks of level $l-1$. The resulting block is used to update the dictionary, becoming available to encode future blocks of the image, independently of their size. This updating procedure for the dictionary uses only information that can be inferred by the decoder, since it is based exclusively in the encoded segmentation flags and dictionary indexes.

Block matching with scales allows for the matching of vectors of different lengths. In order to do this, MMP uses a separable scale transformation T_N^M to adjust the vectors' sizes before attempting to match them. For example, in order to approximate an original block $\mathbf{X}^l$ using one block $\mathbf{S}^k$ of a different scale of the dictionary, MMP first determines $\mathbf{S}^l = T_k^l[\mathbf{S}]$. Detailed information about the use of scale transformations in MMP is presented in [2].

B. The MMP-Intra algorithm.

MMP-Intra is a successful combination of the original MMP algorithm with intra-frame prediction techniques, like the ones used in the H.264/AVC standard [5]. For each original block, $\mathbf{X}^l$, MMP-Intra determines the prediction block, $\mathbf{P}_m^l$, and the respective residue values, $\mathbf{R}_m^l$. The block with residual pixels is then encoded using MMP.

MMP-Intra uses essentially the same prediction modes used by H.264/AVC intra coded blocks [1]. Intra prediction is used *hierarchically* for blocks of dimensions 16×16, 16×8, 8×8, 8×4 and 4×4 (corresponding to levels 8 to 4 of the segmentation tree). This hierarchical prediction scheme, allied to the use of Lagrangian R-D cost functions, allows the encoder to optimise the block prediction, determining the best trade-off between the prediction accuracy and the additional overhead introduced by the prediction data.

When compared with the original MMP algorithm, MMP-Intra has to encode some additional information for the block prediction (namely the prediction mode and the block size used for the prediction). Using this information, the decoder is able to decode each image block, by finding the corresponding prediction block and adding it to the decoded residual block. Details about MMP-Intra can be found in [1].

C. Efficient Dictionary design for MMP-Intra

In MMP-Intra, as in MMP, each new block, $\mathbf{S}_{new}^{l'}$, that is created at a scale l' (by the concatenation of two vectors of level $l'-1$ of the dictionary) becomes available for all scales of $\mathcal{D}$. MMP-Intra uses an initial dictionary consisting of a few blocks with constant value. This highly sparse initial dictionary is obviously not efficient for coding images, but the updating procedure quickly adapts the patterns in the dictionary to the typical patterns that are being encoded.

Observations of the final dictionary sizes of the MMP-Intra process led to the conclusion that for high rates the final dictionary sizes are much larger than for lower target bit rates. Further studies have shown that the final number of blocks for each level of the dictionary is, by far, much larger than the total number of blocks that are actually used. This discrepancy grows with the target bit rate and exists for different image types and target compression ratios. This process has the disadvantage of increasing the average bit rate needed to transmit each index of the dictionary, that grows with the total number of indexes, compromising the method's performance.

[1]The superscript l means that the block $\mathbf{X}^l$ belongs to *scale* l or *level* l of the segmentation tree (with dimensions $(2^{\lfloor \frac{l+1}{2} \rfloor} \times 2^{\lfloor \frac{l}{2} \rfloor})$).

In [3], a new algorithm was proposed to limit the dictionary growth. The basic principle underlying this procedure is to avoid that new vectors, very close to those already available in the dictionary space, are used to update the dictionary. In order to achieve this, a test condition was added to the dictionary update procedure, to ensure that the quadratic distortion between each new block of level l, $\mathbf{S}^l_{new}$, and the ones already available in the dictionary is not inferior to a given threshold d. This introduces a "minimum distance condition" between any two vectors of each level of the dictionary.

The value of d must be carefully chosen. If this value is too small, the aim of controlling the dictionary growth will not be achieved, and if it is too large, the dictionary will lose its efficiency in approximating the images' patterns. In the case of MMP-Intra, the optimum value for d is a function of the target bit-rate and therefore of the parameter λ. Experimental tests lead to the following expression for $d(\lambda)$:

$$d(\lambda) = \begin{cases} 5 & if \quad \lambda \leq 15; \\ 10 & if \quad 15 < \lambda \leq 50; \\ 20 & otherwise. \end{cases} \qquad (1)$$

Equation 1 allows the encoder to automatically achieve a close to optimum R-D relation, for any given target bit-rate. Further details on how this equation was determined can be found in [3].

Reference [3] also introduces a context adaptive arithmetic encoder for the dictionary indexes With this method, the dictionary indexes are divided into groups, according to a context criterion, that, for MMP-Intra, is the original scale of the block. Using such procedure, each index is transmitted using one context symbol followed by an index, that chooses among the elements of that context. This further explores the statistical dependencies of the MMP symbols, generating gains in the arithmetic coding module.

III. The new Dictionary Updating Procedure

The aim of the original dictionary updating procedure, using scale transformations in the MMP and MMP-Intra methods, is to create new patterns that will become available to approximate future blocks in the image. It is reasonable to assume that one block, that was used to approximate one region of an image, has a fair probability of being useful in the approximation of future regions of the same image. This self similitude property of digital images justifies the dictionary update procedure. This concept is expanded in MMP for blocks of different dimensions, through the use of scale transforms.

As was explained in the previous sections, for each determined approximation of an image block, we insert, at the most, one new block at each scale of the dictionary. The new dictionary adaptation methods, studied in this work, update the MMP dictionary with more than one block for each approximation.

The main idea behind this procedure is to provide the MMP dictionary with a richer set of patterns, that continue to relate to the originally created block and increase the approximation power of the dictionary. The higher cardinality of the dictionary has the immediate effect of increasing its

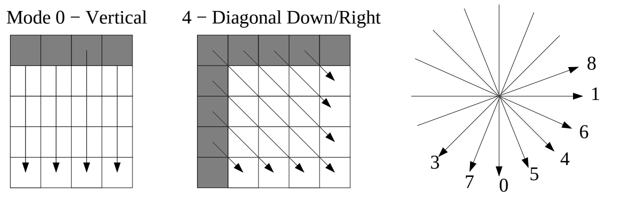

Fig. 1. Examples of some of the prediction modes and the eight direction used for the prediction in MMP-Intra.

entropy, so the inclusion of more blocks in the dictionary can have an adverse effect on the final coding performance. This means that the new blocks used in the updating procedure most be carefully chosen, in order to avoid a decay on the algorithm's performance. In the following discussion we show that, among other important aspects, the use of the previously presented distortion control between the dictionary words is one important factor for the efficiency of the proposed method.

In the next section we describe the possibilities for the generation of new patterns based on the original reconstructed block, that were used in this study. After this we present some details on the implementation of the tests and their experimental results. We also introduce a method to further control the dictionary size, that decreases the computational complexity of the algorithm and results in no performance losses, and, in some cases, even some quality gains. Finally, we discuss a new context adaptive dictionary segmentation method, based on the new updating strategies.

A. The Generation of the New Dictionary Patterns

In this work we considered three main possibilities for creating the new patterns, used in the dictionary update step: the use of *geometric transforms* to determine the new blocks; the use of *displaced* versions of the image residue and the use of the block with the *symmetric values*.

Let $\hat{\mathbf{X}}^l(x,y)$ be the approximation of the original image block, $\mathbf{X}^l(x,y)$, determined by MMP-Intra. This block corresponds to the sum of the original predicted block, $\mathbf{P}^l(x,y)$, with the reconstructed residue block, $\hat{\mathbf{R}}^l(x,y)$, i.e.,

$$\hat{\mathbf{X}}^l(x,y) = \mathbf{P}^l(x,y) + \hat{\mathbf{R}}^l(x,y). \qquad (2)$$

In the original MMP-Intra method, only $\hat{\mathbf{R}}^l(x,y)$ is used to update all scales of the dictionary, through the use of scale transforms.

In MMP-Intra, the prediction modes used to determine the image residues are mainly directional modes, that try to predict image patterns that occur in the image with different orientations. This can be observed in figure 1, where these spatial relations become evident. For example, the horizontal and vertical prediction modes can be related trough the use of a 90^o rotation. This fact motivated the use of *geometric transforms* in the dictionary updating procedure. In this case, up to four new versions of the originally determined block, $\hat{\mathbf{R}}^l(x,y)$, are inserted in the dictionary, resulting from rotated (by 90^o and -90^o) and mirrored (horizontally and vertically) versions of $\hat{\mathbf{R}}^l(x,y)$.

The self similitude properties of a digital image may occur at any position. Nevertheless, the original MMP and MMP-Intra encoders only explore the self similitude between blocks that are located at the set of points defined by the block segmentation described in section II. This means, for example, that one 8×8 block can only be used to match a image block with upper left corner pixel coordinates given by $(8.i, 8.j)$, where $i = 0, ..., N_{lines}/8$ and $j = 0, ..., N_{rows}/8$.

To overcome this limitation, *displaced* versions of the reconstructed block can be use to update the dictionary. Because MMP-Intra encodes the *predicted residues* and not the original image blocks, the patterns used to update the dictionary are given by:

$$\hat{\mathbf{R}}^l_{\delta_x,\delta_y}(x,y) = \hat{\mathbf{X}}^l(x+\delta_x, y+\delta_y) - \mathbf{P}^l(x+\delta_x, y+\delta_y). \quad (3)$$

In our tests we've used displacement steps that correspond to one half or one quarter of the block dimensions and maximum displacement windows that have the same dimensions as the block $\hat{\mathbf{R}}^l_{\delta_x,\delta_y}(x,y)$. This means that, for a block of size $2^m \times 2^n$, we include in the dictionary all blocks $\hat{\mathbf{R}}^l{}_{\delta_x,\beta\delta_y}(x,y)$, with

$$\begin{cases} \alpha = 0, ..., s-1 \\ \beta = 0, ..., s-1 \\ \delta_x = \lfloor \frac{2^m}{s} \rfloor \\ \delta_y = \lfloor \frac{2^n}{s} \rfloor \\ s = 2, 4 \end{cases} \quad (4)$$

Notice that, due to the sequence by which the image blocks are processed, it is possible that at a certain time, one or several of the displaced blocks $\hat{\mathbf{R}}^l{}_{\delta_x,\beta\delta_y}(x,y)$ cannot be determined, because the corresponding prediction and reconstructed blocks are not yet available. Naturally, at each iteration, only the available blocks are used to update the dictionary.

Because MMP-Intra encodes predicted residues, the reconstructed residue blocks have positive, null and negative values. This means that we can use the block with the *symmetric values* to update the dictionary, i.e., at each dictionary adaptation step we also include the block $-\hat{\mathbf{R}}^l(x,y)$ in the dictionary. This block tends to have the same directional structure of the original block, meaning that it has the potential of being useful to encode future residues determined with a similar prediction mode, or even with a different one.

The use of the extra blocks, $\hat{\mathbf{R}}^l_{New}(x,y)$, created using any of the presented techniques, increases the growing rate of the dictionaries. This has direct implications on the growth of the redundancy among dictionary elements, that can reach a level where the increased entropy, caused by the additional elements, overcomes the potential gains achieved by the better approximation capabilities of the new dictionaries, causing performance losses. To avoid this, the distortion control between the dictionary blocks, described in section II-C, is applied when the new blocks $\hat{\mathbf{R}}^l_{New}(x,y)$ are used to update the dictionary.

B. Experimental Tests and Results

Experimental tests were performed using the previously proposed methods, for three types of test images: smooth images (represented by the well known image LENA, available at [6]), text image PP1205 and compound (text and grayscale) image PP1209. Images PP1205 and PP1209 were scanned, respectively, from pages 1205 and 1209 of the *IEEE Transactions on Image Processing*, volume 9, number 7, July 2000 and are available for download at [7].

Due to the limited available space, image PP1209 was chosen to present the experimental results, because it accurately represents, for most cases, the general behaviour of the methods. Figure 2 presents the results for the original MMP-Intra, MMP-Intra with distortion control and context segmentation of the dictionary (MMP-Intra DicLim) and for each one of the new updating strategies, with and without distortion control and context segmentation.

The experimental results lead to the following conclusions, when each of the proposed methods was used independently:

- *Geometric Transforms*: when used without the distortion control (see section II-C), this method achieved equivalent results for image LENA, but lead to a decreased performance for the text and compound images. When combined with the distortion control, the results are comparable to those of the original MMP-Intra and, for some cases, even produced some slight improvements. This demonstrates the power of the redundancy control algorithm and its efficiency when the new updating techniques are used.
- *Displaced versions*: when used with shift steps equal to half the block dimensions (i.e. $s = 2$ in eq. 4), this method is able to achieve gains over the original method for image LENA of about 0.1 to 0.2 dB, irrespective of whether distortion control is activated or not. For the text and compound images, this method only allows for quality improvements when the distortion control is activated. When this is not the case, the final quality remains equivalent to that of the original method.
  When used with shift steps that are equal to one quarter of the block dimensions (i.e. $s = 4$ in eq. 4) the results are similar to the ones achieved by the previous technique. In this case, the better approximation power of the dictionary is generally compensated by the larger overhead needed to encode the indexes of the extra blocks, that also have a negative effect on the computational complexity of the method.
- *The block with the symmetric values*: when the distortion control is not used, this method causes severe losses for the text and compound images and equivalent results for smooth image LENA. The described losses are dramatically reduced when the distortion control is activated, but the PSNR performance in this case remains equal or inferior to the on of the original case.

From these observations we can conclude that the overall best results, considering all images types and all compression ratios, are achieved when the displaced blocks with $s = 2$ are used.

After evaluating the performance of each method independently, several tests were performed where the described methods for creating new dictionary patterns were combined. Two combinations were achieved results that, when all the test images are considered, are equivalent to those resulting from the isolated use of block displacements with $s = 2$. These combinations were: a) the geometric transforms combined with block displacements using $s = 2$ and b) the combination

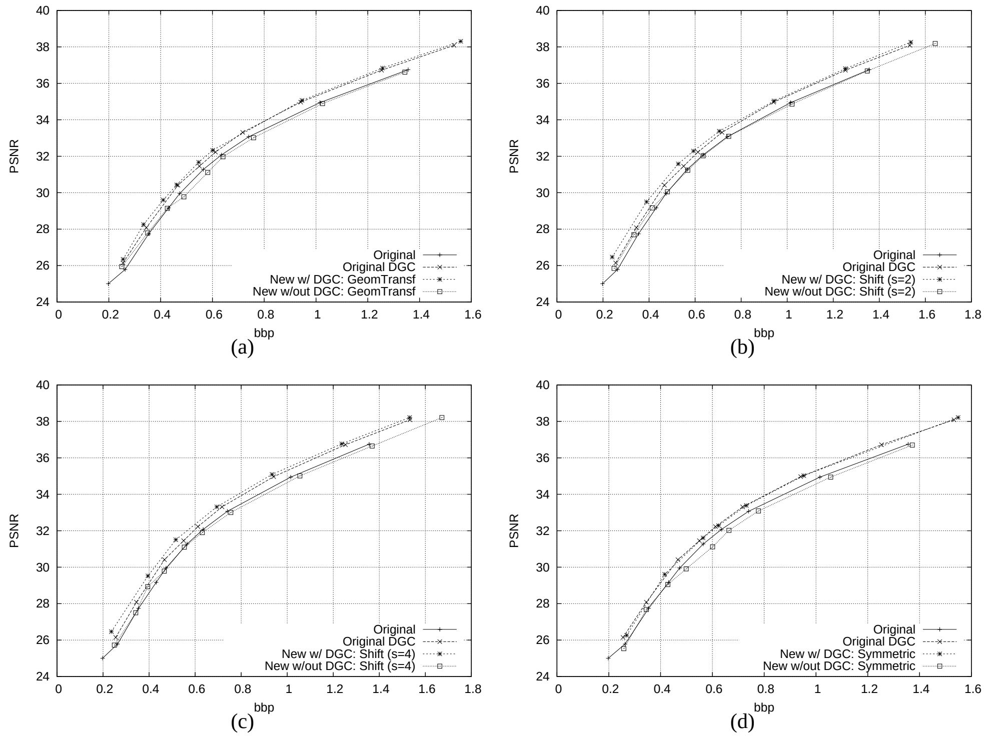

Fig. 2. Results for MMP-Intra (Original), MMP-Intra with dictionary growth control (DGC), MMP-Intra with new dictionary update strategies (New) with and without DGC, for image PP1209 and using: a) Geometric transforms; b) Displacements, s=2; c) Displacements, s=4; d) Symmetric block.

of all updating strategies. Considering the very close results and the increased computational complexity introduced by the extra dictionary blocks created by the combination of several methods, we conclude that, at this stage, the simple use of the displaced blocks, with $s = 2$, is a all-round better option.

C. Reducing the Computational Complexity

Being an approximate pattern matching based scheme, most of the computational complexity of MMP-Intra is related to the calculations of the sum of square differences used to determine the block of the dictionary that represents the best match for a given image block. In our implementation, a significant reduction of the number of operations needed for this process is achieved through the use of fast search methods [8]. Nevertheless, the complexity of the encoder still closely depends on the number of blocks present in each dictionary.

The proposed techniques are based on the inclusion of extra elements in the dictionary, that imply an additional growing rate of the number of dictionary elements and, inevitably, an increased computational complexity. This led to the study of new techniques, that allow for a reduction in the computational complexity of MMP-Intra, without compromising its efficiency. This was achieved by limiting the range of the dictionary scales that are updated through scaled versions of the original blocks.

In the original algorithm, when a new block $\mathbf{S}_{new}^{l'}$ is created from the concatenation of two vectors of level $l' - 1$, scaled versions of $\mathbf{S}_{new}^{l'}$ are used to update the dictionary at all used scales. Experimental results showed that scale transformations, for which the final scale is very different from l', create new blocks that are not closely related to the original patterns.

This observation was exploited by limiting the scales used in the dictionary update procedure. If a new block is created at an original scale l', we now update only the dictionaries corresponding to the scales in the range $max(0, l' - L_{low})$ to $min(l_{Max}, l' + L_{high})$. Several tests were performed, in order to determine the best values for the interval bounds L_{low} and L_{high}. The optimum values for these parameters varied slightly for different image types, but, as a general rule, the use of $L_{low} = L_{high} = 2$ allowed for a significant reduction on the computational complexity, without any noticeable loss in the final quality.

Figure 3 shows the results of using MMP-Intra with distortion control between the dictionary elements, context segmentation and the new updating procedure with displaced blocks with $s = 2$ (see eq. 4), for the original case (where all scales are used) and the simplified version of the algorithm, using $L_{low} = L_{high} = 2$. It is clear that the simplified version leads to no visible reduction in the final quality of the encoded images. In fact, for image PP1205, the limitation of the updated scales leads to some quality gains in the encoding method. For text images it is clear that scale transformations for which the final scale is very different from l' lead to patterns that are very unrelated to the typical image blocks.

Figure 3 also compares the results of the new method with the state-of-the-art, transform-quantisation based image encoders JPEG2000 [4] and H.264/AVC Intra encoder [5]. It is evident that the proposed techniques allow for MMP-Intra to achieve almost equivalent performances for the smooth image LENA, when compared with these encoders, specially for low to medium compression ratios. For other image types, like text and compound images, the proposed method increases even

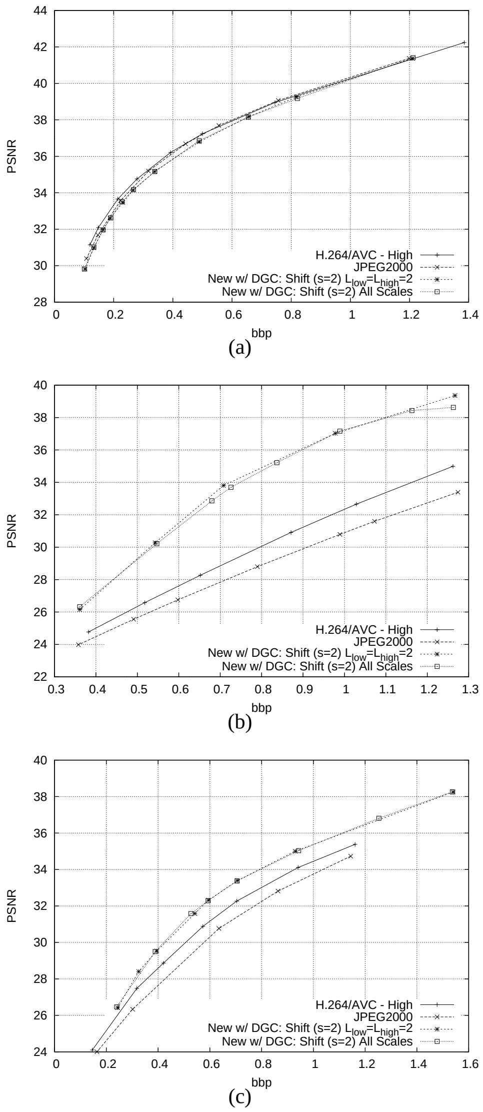

Fig. 3. Comparative results for JPEG2000, H.264 and MMP-Intra with the proposed method (New) using *displaced blocks, with s=2*. The cases when *all* scales and $L_{low} = L_{high} = 2$ are used are compared for images: a) LENA; b) PP1025; c) PP1209.

more the MMP-Intra's advantage over the standard encoders, reaching gains of up to 5 to 6 dB for the text image PP1205 and more than 1 to 2 dB for compound image PP1209.

D. Using the Block Origin as a Context for Arithmetic Coding

As was previously explained, MMP-Intra uses segmented dictionaries, according to a context criterion that corresponds to the original scale of each new block, with the objective of grouping blocks with similar characteristics in the same dictionary segments, originating probability distributions that allow for increased compression efficiency when an adaptive arithmetic encoder is used.

To further explore this factor, we implemented a version of the proposed method that used new contexts to segment the various dictionaries, based also on the origin of the block, that is, whether it is displaced, transformed, symmetric or original, that is used to update the dictionary. Considering the example where the dictionary is also updated using the geometric transformations, in the original method all blocks created at scale l' would be inserted in segment l' of all dictionaries. With the new version of the method, separate contexts are created to distinguish the blocks that were originally created by using the geometric transformations from the blocks that correspond to the original insertion algorithm.

This test intended to determine if the new context models have a beneficial effect in the performance of the arithmetic encoding of the dictionary's indexes. However, in this case the experimental results showed that this process does not lead to any quality improvements. Two main reasons can be found to justify this: first, the blocks that are used to encode the image don't originate mainly from one specific updating process and second, with the increased number of contexts, the method has no time to conveniently learn the distribution statistics. These two factors mean that the probabilistic distribution created by the new contexts causes no improvement in the arithmetic encoding of the dictionary indexes.

IV. CONCLUSIONS

In this paper we present a method for increasing the approximation power of the multiscale adaptive dictionary, used in the MMP-Intra image encoding algorithm. Several methods were investigated, that create new versions of the blocks used to update the MMP-Intra's dictionary.

Our experiments demonstrated that the use of the proposed method allows for consistent gains in the coding performance, for all types of tested images, at all compression ratios. The best results were achieved for the combination of the several methods or the isolated use of displaced blocks. An efficient scheme to reduce the computational complexity of the proposed method was also presented, that preserves or marginally increases the image quality gains, for some types of images.

The proposed method leads to consistent gains in the MMP-Intra's performance for all tested image types, at all compression ratios.

REFERENCES

[1] N. M. M. Rodrigues, E. A. B. da Silva, M. B. de Carvalho, S. M. M. de Faria, and V. M. M. Silva, "Universal image coding using multiscale recurrent patterns and prediction," *IEEE International Conference on Image Processing*, September 2005.
[2] M. de Carvalho, E. da Silva, and W. Finamore, "Multidimensional signal compression using multiscale recurrent patterns," *Elsevier Signal Processing*, no. 82, pp. 1559–1580, November 2002.
[3] N. M. M. Rodrigues, E. A. B. da Silva, M. B. de Carvalho, S. M. M. de Faria, V. M. M. Silva, and F. Pinagé, "Effi cient dictionary design for multiscale recurrent patterns image coding," *ISCAS 2006 IEEE International Symposium on Circuits and Systems*, May 2006.
[4] D. S. Taubman and M. Marcelin, *JPEG2000: Image Compression Fundamentals, Standards and Practice*. Kluwer Academic Publishers, 2001.
[5] J. V. T. J. of ISO/IEC MPEG & ITU-T VCEG (ISO/IEC JTC1/SC29/WG11 and I.-T. S. Q.6), *Draft of Version 4 of H.264/AVC (ITU-T Recommendation H.264 and ISO/IEC 14496-10 (MPEG-4 part 10) Advanced Video Coding)*, March 2005.
[6] http://sipi.usc.edu.
[7] http://www.estg.ipleiria.pt/∼nuno/MMP/.
[8] Z. Pan, K. Kotani, and T. Ohmi, "Improved fast search method for vector quantization using discrete walsh transform," *ICIP '04 - IEEE International Conference on Image Processing*, vol. 5, pp. 3177–3180, October 2004.