# A NEW METHOD TO COMPRESS STEREO IMAGES USING MULTISCALE RECURRENT PATTERNS, WITHOUT THE USE OF DISPARITY MAP

*M. H. V. Duarte* [1], *M. B. Carvalho* [2], *E. A. B. da Silva* [3], *C. L. Pagliari* [4], *G. V. Mendonça* [5]

[1]Centro Federal de Educação Tecnológica do Ceará
[2]CTC/TET/Universidade Federal Fluminense
[3,5]PEE/COPPE/DEL/EE/Universidade Federal do Rio de Janeiro
[4]DEE/Instituto Militar de Engenharia
[1]heveline,[3]eduardo,[5]gelson@lps.ufrj.br,[2]murilo@telecom.uff.br,[4]carla@ime.eb.br

## ABSTRACT

In this paper we propose a new technique for compressing stereo images using multiscale recurrent patterns. The idea consists on representing blocks of the image using variable-sized elements of a dictionary. More specifically, blocks of the image are represented by contracted, expanded or displaced elements of the dictionary. The patterns learned along the reference image process are included in the dictionary. The novelty of this work is the absence of the disparity map in the stereo image coding process. That is, the burden of its calculation, the pre- and post-processing phases, the generation of the error images, as well as their coding and transmission are not necessary. In brief, the proposed method cuts off the load of the whole disparity estimation process yet presenting high-quality results at the decoder end.

## 1. INTRODUCTION

The practical use of stereoscopic video systems can bring realism to many applications such as 3D movies, medicine surgery, video-conferencing, multimedia, and remote operations, among others, due to the depth perception offered by these systems. Having an efficient method to compress the stereo images is one of the main challenges for an unrestricted use of stereoscopic systems. This is because stereo images require the transmission of, at least, double the amount of data used in monocular image coding systems, and we have to cope with limited channel bandwidth. An efficient encoding of stereo video systems requires exploiting the binocular redundancy between stereo images as well as the temporal redundancy between consecutive frames of each view [1]. The techniques used to code binocular images are similar to those used in monocular video coding. Usually, the reference image (the right-view or left-view) is encoded using a known method like MPEG-2. Next, the disparity map using both views is estimated [2]. The remaining view can be either motion estimated from the previous reference frame, or disparity estimated from the reference view. The resultant error images, the *Displaced Frame Difference*—DFD (stereo video), and the *Disparity Compensated Difference*— DCD (stereo pairs) are coded and transmitted. The main step to obtain the DCD is the disparity estimation process that tackles an ill-posed problem. The quality of the disparity map rules the amount of information carried by the DCD and the number of bits to code it. Nevertheless, accurate disparity maps demand a high bit rate to be transmitted. One solution is the use of block-based disparity estimators which may produce very inaccurate disparity maps because they are not, in many cases, blockwise constant. One possible way of tackling this problem is as in [3], where an hierarchical MRF (Markov Random Fields) model and selective overlapped block disparity compensation is employed. In this paper, we propose another alternative. It also uses a hierarchy of block sizes, but the disparity map does not need to be obtained. Our method is based on the concept that, since a stereo image pair involves two similar images, we can use a recurrent patterns method [4, 5, 6], where the learned patterns of the coding process of the reference image can be used to code the other image. We employ a new class of coders using multiscale recurrent patterns. A recently proposed method referred to as *Multidimensional Multiscale Parser*— MMP [7, 8] uses contractions and expansions of elements belonging to a dictionary to code each segment of an image. This method is efficient for monocular image coding, especially when the image is composed by pictures and text.

In MMP [7, 8], the image is initially divided in blocks of size $N \times N$. Each block is further segmented in smaller variable-sized blocks. These smaller blocks are then approximated by contracted/expanded versions of matrices in a dictionary. These contractions and expansions are per-

This work has been supported by CNPq, Brazil, grants nº 522381/95-2 and 143217/97-6 and FAPERJ, Brazil, grant nº 170.331/2000

formed using a procedure similar to the usual decimation and interpolation operations [9]. The output of MMP is composed by the dictionary indexes corresponding to each block, as well as information regarding the segmentation. This segmentation is specified by a binary tree that is encoded using a sequence of binary flags, in a top down fashion. The sequence of dictionary indexes and the sequence of binary flags are encoded by an arithmetic coder [10]. The segmentation tree is optimized in a rate-distortion sense by the use of a pruning algorithm operating from the bottom (leafs) of the tree to the top (root). The MMP algorithm starts with a small initial dictionary that is updated as the input data is encoded, by the inclusion of concatenations of previously coded blocks. As the dictionary grows it is expected that it should contain elements more alike the previously coded blocks, decreasing the number of tree splittings. Consequently, a smaller number of indexes and flags are generated, lowering the bit rate and improving the compression.

In this paper we propose a variation of MMP that includes displacements of previously coded blocks in the dictionary, efficiently exploiting the redundancy between the two views of the stereo pair while avoiding the use of disparity maps. Initially we used the same dictionary to encode both views.

The organization of the paper is as follows. Section 2 starts with a description of the method *Multidimensional Multiscale Parser* - MMP. Next, we describe the modifications implemented to MMP focusing the exploitation of stereo image pairs characteristics. Sections 4 and 5 present, respectively, the experimental results and conclusions.

## 2. DESCRIPTION OF THE MMP

The MMP represents a new class of compression algorithms. It is based on matching multiscale recurrent patterns. It tries to represent an input block using contracted or expanded versions of processed blocks. The MMP dictionary is adaptive, it learns patterns occurred in image blocks already processed.

The MMP algorithm has basically four stages:

- Obtaining the costs for each segment of the block;
- Obtaining the optimal segmentation tree;
- Coding flags and indexes of the optimal segmentation tree;
- Updating the dictionary.

Initially the image is divided in blocks of size $N \times N$. Each block is processed in order to find the best approximation for it. This approximation is formed by expansions and contractions of elements of a dictionary. The largest possible element has size $N \times N$, and it can be segmented in smaller elements. The segmentation can be represented by a binary tree. An example of binary tree corresponding to a segmentation procedure is shown in figure 1. The root of the tree corresponds to the largest element (of size $N \times N$). Each node of the tree, if divided, can generate two children nodes. The division can be made horizontally (as in figure 1) or vertically. The segmentation can proceed until the element is the size of a pixel.

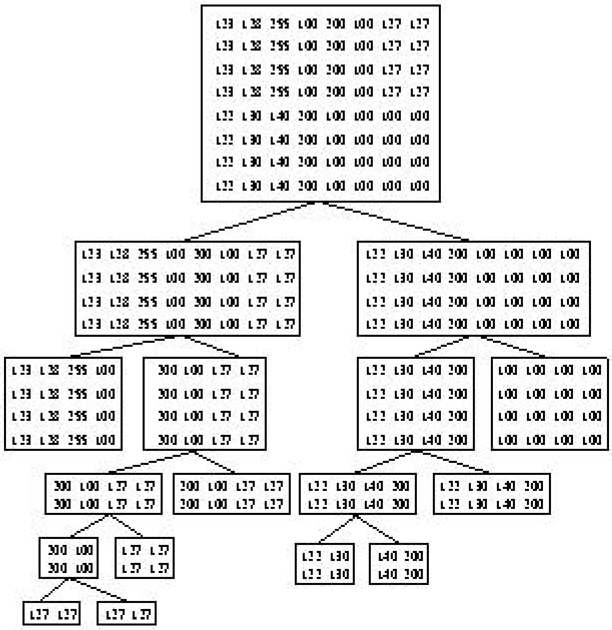


**Fig. 1**. An example of a binary segmentation tree, representing the best segmentation for a block $N \times N$, where $N = 8$.

Each block corresponding to a leaf (a node without a child) is represented by an index denoting an element of the dictionary. The dictionary elements can be expanded or contracted using a scale transformation [9], preserving the block dimension, in order to match the size of the target block. Whenever two leafs of the tree corresponding to two children of the same node are encoded, the resulting element from their concatenation is included in the dictionary. The segmentation tree is obtained by an iterative procedure that attempts to minimize the Lagrangian cost $J$ [11] given by:

$$J = D + \lambda R$$

where $D$ is the distortion, $R$ is the rate, and $\lambda$ is a constant (Lagrange multiplier). The distortion is given by the mean square error between the original and decoded image blocks. The rate is the number of bits per pixel used to transmit the indexes of the elements of the dictionary as well as the segmentation tree.

The procedure to minimize the Lagrangian cost is as follows:

Let $J_P$ be the cost of a parent node whose children nodes costs are $J_L$ and $J_R$, where L and R correspond respectively to the nodes on the left and on the right of the parent node of the binary tree. We compare the cost $J_P$ to the sum of the costs $J_E + J_D$. If the sum $J_E + J_D$ of the costs of the children nodes is smaller than the cost of the parent node $J_P$, then the tree should not be pruned. that is, in this case, it is better to use the two smaller blocks than the bigger one to represent the input. If we decide not to prune, the cost associated to the parent node $J_P$ must be replaced by the smaller value $J_E + J_D$ before we resume the optimization procedure of the tree. The pruning procedure runs until all the tree has been traversed. The segmentation tree is transmitted to the decoder using a sequence of binary flags. The sequence of dictionary indexes and the sequence of binary flags are encoded by an arithmetic coder [10].

The dictionary is initialized with 1x1 matrices containing all pixel values in the range [MinPixel, MaxPixel], where MinPixel is the smallest pixel value found in the input image and MaxPixel is the greatest. As the encoding proceeds, the dictionary is updated whenever the blocks corresponding to the two children of a parent node are already encoded. Then, the two children nodes representations are concatenated and included in the dictionary. As the dictionary grows it is expected that it should contain elements more alike the previously coded blocks, decreasing the number of tree splittings. Consequently, a smaller number of indexes and flags are generated, lowering the bit rate and improving the compression. The decoder starts with the same initial dictionary as the encoder. As it receives the flags defining the tree and the elements corresponding to the tree leafs, it starts reconstructing the received image. The dictionary updating process at the decoder should be equivalent to that at the encoder. Whenever the decoder receives two terminal nodes they are concatenated and the resulting node is included in the dictionary. The decoding is a much faster operation than the encoding process, as the optimization of the segmentation tree is not necessary.

## 3. STEREO MMP: AN IMPLEMENTATION BASED ON MMP, EXPLOITING THE CHARACTERISTICS OF STEREO IMAGE PAIRS

The stereo MMP is an algorithm based on MMP, whose purpose is to directly encode a stereo image pair, composed by a reference image and a target image. It exploits its stereo characteristics, without explicitly evaluating the DCD. Here we used the left-view as the reference image, and the right-view as the target image.

The stereo MMP is based on two main points. One is the inclusion in the dictionary of elements corresponding to the displacement of previously coded blocks of the reference image, and the otherwise is the usage of variable-sized blocks, a basic property of MMP. If the cost of using a large block is greater than the cost of using two smaller blocks, a segmentation is performed. Therefore, a block in the target image can be represented as accurately as needed.

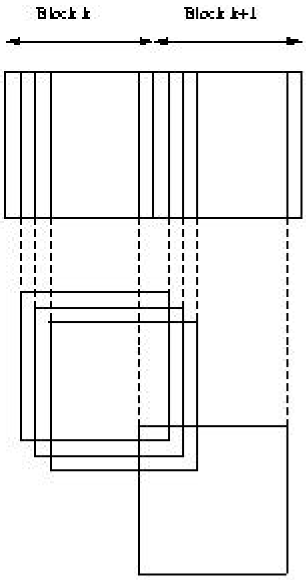


**Fig. 2**. Displaced elements obtained from two neighbor reconstructed blocks of a row of blocks (ROB) of the reference image. These displaced elements are inserted in the dictionary and probably will be used for coding the corresponding ROB of the target image.

In the stereo MMP, the inclusion of displaced elements in the dictionary replaces the disparity estimation process. These displaced elements are obtained directly from the reconstructed reference image, composed by the concatenations of previously encoded blocks. For example, for $N \times N$ blocks, a block whose upper-leftmost pixel is at position $(N * i, N * j + d)$, with $i$ and $j$ integers, corresponds to a dictionary element displaced d pixels to the right in relation to the one at $(N * i, N * j)$. These displaced elements are obtained from a window sliding horizontally over two neighboring blocks, as shown in figure 2. These displaced blocks are generated and inserted in the dictionary before processing the corresponding row of blocks (ROB) of the target image.

When the choice of an element results in a high cost, the decision process that specifies the best segmentation tree decides to split the block. This is equivalent, in DCD-based methods, to obtain the disparity using smaller blocks, yielding more accurate disparity estimates. Elements corresponding to half-pixel displacements were also included to improve the resolution of the estimation. Parallel-camera conditions are assumed so that disparities are purely horizontal, decreasing the number of displaced elements to generate. In addition, this number is limited by a search window given by the minimum and maximum disparity values for the image. These disparity values can be determined prior to coding. These values are sent to the decoder in order to compose the dictionary with the same elements present in the coding process.

We used the same dictionary to encode the left and the right views. The statistics of the usage of the dictionary elements by type show that, when the target image is being coded, the displaced elements tend to be used more often than the others (initial, concatenated, contracted or expanded). Therefore, the dictionary was split in two: one containing the displaced elements, and the other containing all the other types. A flag was used to signal which dictionary was selected. This approach leads to a better estimation of the probability of the dictionary indexes, improving the coding efficiency.

## 4. EXPERIMENTAL RESULTS

The stereo MMP was tested in two versions: the first, that we called StereoMMP_1dic, uses just one dictionary containing all the types of elements (initial, concatenated, expanded, contracted and displaced) to encode both views, and the other, that we called StereoMMP_2dic, uses two dictionaries: one containing initial, concatenated, expanded and contracted elements and another dictionary containing just the displaced elements. The former dictionary is always used to encode the reference image. For encoding the target image (right view) we also use this dictionary, but we have the option to use the latter dictionary (containing only displaced elements), when this offers a smaller cost. A flag signals which dictionary has been choosen. The cost of this flag is also included in the optimization. We can see that, for all the stereo image pairs tested, the stereo MMP using two dictionaries performs better than the one using just one.

The two versions of stereo MMP were run to compress the stereo image pairs CORRIDOR [1], AQUA, SAXO [2] and MAN [3]. Figure 3 shows the rate x PSNR (*Peak Signal to Noise Ratio*) performance results for the pair CORRIDOR. The improvement obtained with the use of two dictionaries is larger for the target image (right view, in figure 3 (b)). This is expected because displaced elements tend to be frequent for the target image. The figure 4 shows the rate x PSNR performance results for the pairs AQUA, SAXO and MAN.

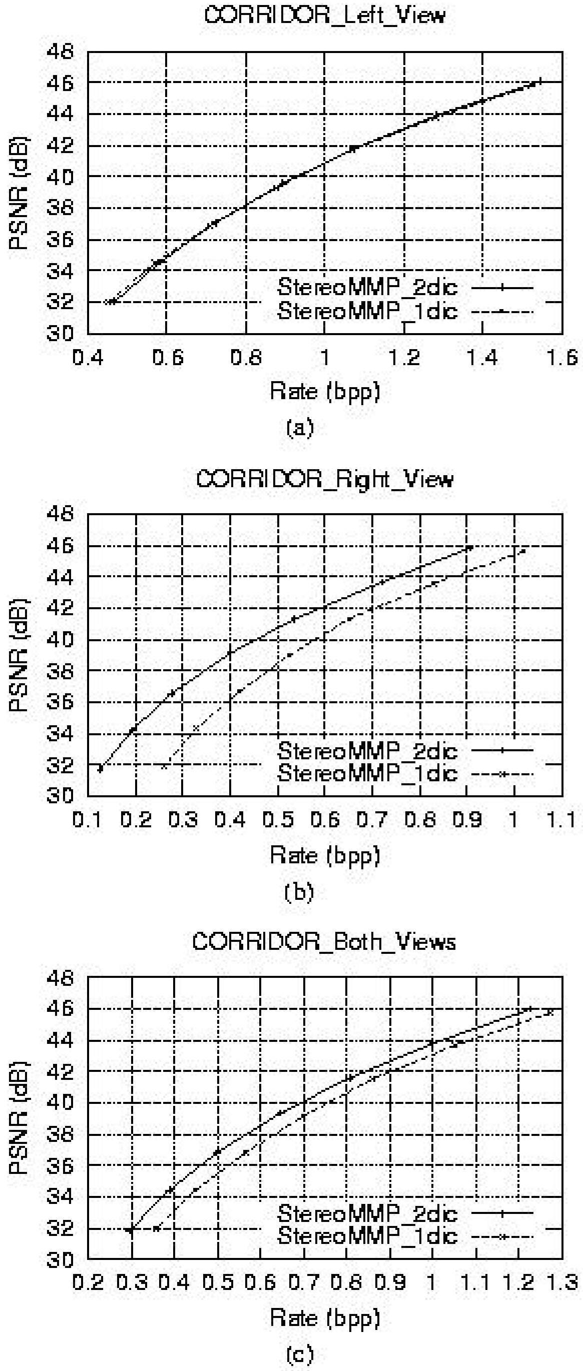
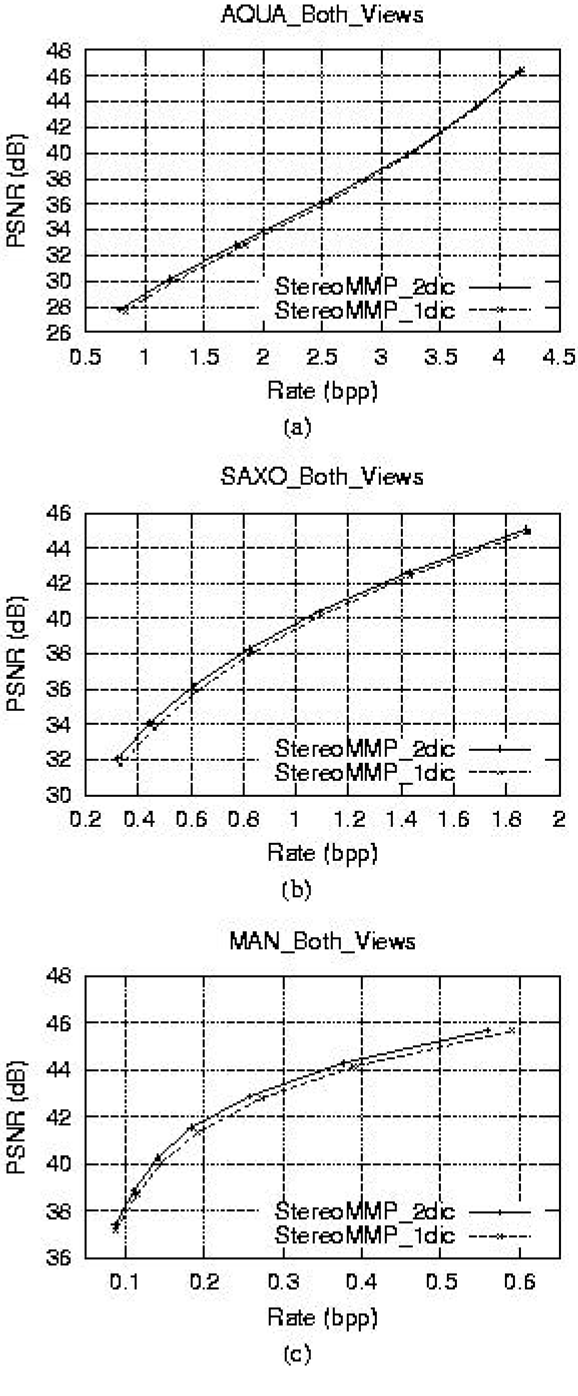


**Fig. 3**. Rate x PSNR for stereo MMP using just one dictionary including all the elements types (initial, concatenated, contracted, expanded and displaced) to encode both views, and stereo MMP using two dictionaries, one including all the types of elements, except displaced ones, and another including only displaced elements, applied to the stereo image CORRIDOR. (a) left-view, (b) right-view, and (c) both views.

[1] Computer Vision and Pattern Recognition Group, University of Bonn

[2] CCETT: Centre Commun d'Etudes de Télédiffusion et Télécommunications (test sequences shot and distributed under RACE DISTIMA European Project), France.

[3] HHI: Heinrich-Hertz-Institut, Berlin, Germany.



(a)

(b)

(c)

**Fig. 4**. Rate x PSNR for stereo MMP using just one dictionary including all the elements types (initial, concatenated, contracted, expanded and displaced) to encode both views, and stereo MMP using two dictionaries, one including all the types of elements, except displaced ones, and another including only displaced elements, applied to both views of the stereo images (a) AQUA, (b) SAXO, and (c) MAN.

The rate x PSNR performance results of stereo MMP using two dictionaries, including also lower rates, are shown in figures 5 and 6, for CORRIDOR and AQUA respectively. The results of [3] are also shown for comparison. We have that, for rates above 0.5 bps, the stereo MMP outperforms the MRF-based hierarchical block matching (HQBM) of [3]. Although for rates below 0.5 bps, HQBM performs better, note that, for those rates, the quality of the pair AQUA is very poor (PSNR < 26 dB).

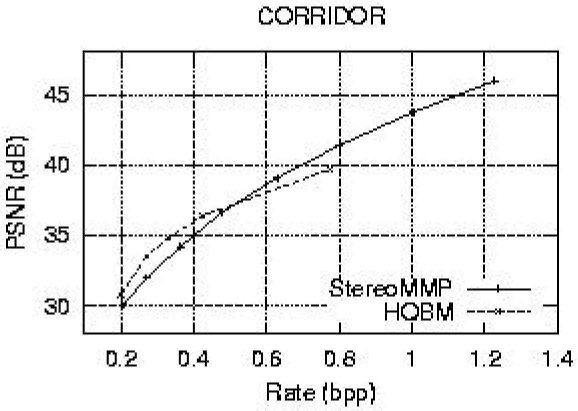


**Fig. 5**. Rate x PSNR for stereo MMP and HQBM [3] with the stereo image CORRIDOR (ROOM in [3]).

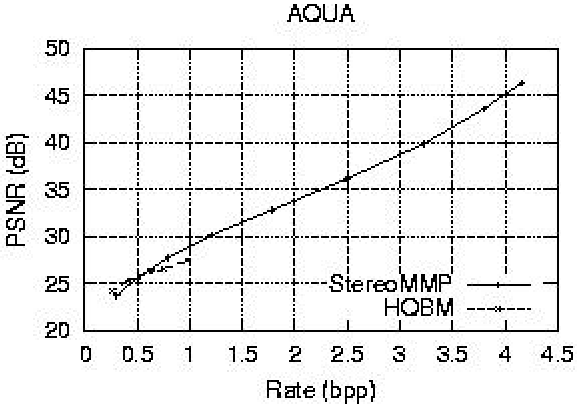


**Fig. 6**. Rate x PSNR for stereo MMP and HQBM [3] with the stereo image AQUA.

## 5. CONCLUSIONS

In this paper we have presented a new method for compressing stereo image pairs, based on multiscale recurrent patterns. The two views are encoded taking in account the similarity between then, without the necessity of calculating the disparity map. The algorithm avoids several well-know drawbacks of stereo codecs, presenting efficient alternatives and producing high-quality reconstructed images.

We have developed an efficient coder based on this principle. The analysis of its performance has shown that the inclusion of displaced elements in the dictionary is an effective way of encoding the target images. The algorithm performed very well for image pairs obtained using parallel camera conditions. In addition, its structure allows its adaptation to non-parallel camera geometries. For example, besides the dictionary of displaced blocks used to encode the target frame, one could add dictionaries of elements displaced and distorted taking into account the non-parallel geometry, thus increasing the match probabilities. In brief, the proposed method is quite promising, opening a new avenue in stereo image coding.

## 6. REFERENCES

[1] M.G. Strintzis and S. Malasiotis, "Object-based coding of stereoscopic and 3d image sequences: A review," *IEEE Signal Processing Magazine, Special Issue on Stereo and 3D Imag- ing (Invited paper)*, vol. 16, no. 3, pp. 14–28, May 1999.

[2] O. Faugeras, *Three-Dimensional Computer Vision: A Geometric Viewpoint*, MIT Press, Cambridge, Massachusetts, 1993.

[3] W. Woo, A. Ortega, and Y. Iwadate, "Stereo image coding using hierarquical mrf model and selective overlapped block disparity compensation," in *1999 IEEE International Conference on Image Processing*, Kobe, October 1999.

[4] J.Ziv and A. Lempel, "Compression of individual sequences via variable-rate coding," *IEEE Transactions on Information Theory*, vol. it-24, no. 5, pp. 530–536, September 1978.

[5] M. Atallah, Y. Genin, and W. Szpankowski, "Pattern matching image compression: algorithmic and empirical results," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, pp. 618–627, 1999.

[6] M. Alzina, W. Szpankowski, and A. Grama, "2d-pattern matching image and video compression," *IEEE Transactions on Image Processing*, 2001.

[7] M. B. Carvalho, E. A. B. da Silva, and W. A. Finamore, "Rate distortion optimized adaptive multiscale vector quantization," in *2001 IEEE International Conference on Image Processing*, Thessaloniki, October 2001, vol. II, pp. 439–442.

[8] M. B. Carvalho, E. A. B. da Silva, and W. A. Finamore, "Multidimensional signal compression using multiscale recurrent patterns," *Signal Processing - Special Issue on Image and Video Coding Beyond Standards*, 2002, To appear.

[9] M. Vetterli and J. Kovačević, *Wavelets and Subband Coding*, Prentice Hall PTR, Englewood Cliffs, New Jersey, 1995.

[10] T. C. Bell, J. G. Cleary, and I. H. Witten, *Text Compression*, Prentice Hall, Englewood Cliffs, NJ, 1990.

[11] K. Ramchandran and A. Ortega, "Rate-distortion methods for image and video compression," *IEEE Signal Processing*, vol. 15, no. 6, pp. 23–50, November 1998.