Visual Communications and Image Processing, SPIE'94, Chicago

Fractal transform coding of color images

Bernd Hürtgen, *Paul Mols*, and *Stephan F. Simon*

Institut für Elektrische Nachrichtentechnik
RWTH, Rheinisch-Westfälische Technische Hochschule Aachen
52056 Aachen, Germany, Tel.:+49–241–80 7683, Fax:+49–241–8888–196
Email: huertgen@ient.rwth-aachen.de

ABSTRACT

The paper reports on investigations concerning the application of block oriented fractal coding schemes for encoding of color images. Correlations between the different color planes can be exploited for the aim of data compression. For this purpose the similarities between the fractal transform parameters of one block location but different color planes are regarded in a blockwise manner. Starting-point is the fractal code for one block in the dominant color plane which serves as prediction for the code of the corresponding block in the other planes. Emerging from this prediction the depending codes can be derived by a successive refinement strategy. Since the fractal code for the dominant color plane and the refinement information for determining the code for the other planes can be represented with fewer bits compared to the independently calculated ones, a coding gain can be achieved.

Keywords: fractal coding, attractor coding, still image coding, color image coding

1. INTRODUCTION

During the last years a novel coding and modeling scheme for natural signals has been developed which is widely known as *fractal coding*. The basic idea for encoding of signals by use of fractal techniques is originated in various publications of Barnsley et al, e.g. [1, 2, 3]. A first implementation of an automatic encoding routine for gray-scale images at common compression ratios has been proposed by Jacquin [4, 5, 6]. A review on fractal image coding may be found in [7] and an excellent mathematical foundation of fractal signal modeling is contained in [8]. Recently some improvements and modifications have been published, e.g. [9, 10, 11, 12, 13], which made the fractal technique a challenging candidate for encoding of images especially if high compression is the issue.

In principle the encoding process consists in finding for each part of the image another part, which after some sort of translation, scaling, rotation, and mirroring fits best in the sense of some given distortion measure. Though the shape of the image parts is arbitrary, most authors use square blocks. The encoding operation results in a number of parameters which after quantization serve as fractal code for this particular block. The parameters of all blocks together then form the code for the entire signal. A coding gain can be achieved if all parameters can be represented with fewer bits than the signal itself.

Currently the investigations concerning fractal coding schemes nearly exclusively concentrate on gray-scale images though in order to present a complete encoding scheme also color information has to be included. Therefore color image coding based on fractal techniques demands for further research. There are only three sources on this topic known to the author, from which [14] and [15] are based on the *yard stick method* but which is discarded in this paper. The only block-oriented approach which also serves as basis for this paper is published in [16].

The physiology of the human eye motivates the trichromatic theory of color vision, which states that the color of light received by the human observer may be specified by a mixture of only three primary colors. These are usually chosen to be either red, green, and blue which are the primary colors of light, or yellow, magenta, and cyan which are the primary colors of pigments. Hence color images consist of three image planes each representing a different spectral area. Due to the physical effect of image generation these planes are highly correlated.

The purpose of this paper is to illustrate how these correlations can be used for data compression purposes within a fractal coding environment. It is proposed to exploit the dependencies between the fractal codes of the different color planes within the transform domain rather than treating them in the original domain. The results show that jointly encoding blocks of different color planes but of the same spatial location results in a significant decrease in bits necessary for representing the fractal code of the entire image.

For this purpose the main characteristics of all fractal parameters describing an image block are investigated. It is shown, that for the same block location in different color planes not only the pixel intensities but also the parameters of the fractal code are highly correlated. Hence only slight modifications of the fractal code for one color plane are necessary in order to describe the other planes with sufficient accuracy.

After presenting the necessary mathematical foundation of fractal coding in section 2, section 3 deals with encoding of images in general. After describing the application of fractal coding schemes to gray-scale images in topic 3.1, some important characteristics of the different fractal coefficients representing an image block are investigated. This leads to the problem of quantization which is addressed in 3.2. Emphasis is put on topic 3.3 which describes how correlations between the different color planes can be exploited within the fractal domain. The paper concludes with some simulation results and a prospect on future investigations in section 4.

2. MATHEMATICAL FOUNDATION

Consider a signal $\mathbf{x} = (x_1, x_2, \cdots, x_n)^T$ consisting of n sample values $x_i;\ 1 \leq i \leq n;\ x_i \in \mathbb{R}$ as point in the n-dimensional vector space $\mathbb{R}^n$. With the definition of the *Euclidean norm*

$$||\mathbf{x}|| := \sqrt[+]{\langle \mathbf{x}, \mathbf{x} \rangle} = \sqrt{\sum_{i=1}^{n} |x_i|^2} \tag{1}$$

defined by the square root of the inner product $\langle \mathbf{x}, \mathbf{x} \rangle$ and by inducing a metric

$$\varrho(\mathbf{x}, \mathbf{y}) := ||\mathbf{x} - \mathbf{y}||,\ \forall\, \mathbf{x}, \mathbf{y} \in \mathbb{R}^n \tag{2}$$

the vector space becomes a normed metric space denoted by $(\mathbb{R}^n, \varrho)$. Transformations within this space can be described by a linear operator $\mathbf{A}$ for which a consistent *operator*- or *matrix norm* $||\mathbf{A}||$ is the so called *spectral*- or *Hilbert norm*, defined by

$$||\mathbf{A}||_{sp} := \sup_{\lambda \in \sigma(\mathbf{A}^T\mathbf{A})} \sqrt{|\lambda|} \tag{3}$$

which is the square root of the largest eigenvalue in magnitude of the matrix $\mathbf{A}^T\mathbf{A}$. Additionally for every linear operator $\mathbf{A}$ the *spectral radius* $r_\sigma(\mathbf{A})$ is defined by

$$r_\sigma(\mathbf{A}) := \sup_{\lambda \in \sigma(\mathbf{A})} |\lambda| \tag{4}$$

which is the magnitude of the largest eigenvalue of $\mathbf{A}$. Any norm $||\mathbf{A}||$ and spectral radius r_σ of a linear operator $\mathbf{A}$ are connected through the following equations:

$$\begin{aligned} r_\sigma(\mathbf{A}) &\leq ||\mathbf{A}|| \\ r_\sigma(\mathbf{A}) &= \lim_{j \to \infty} \sqrt[j]{||\mathbf{A}^j||}\ . \end{aligned} \tag{5}$$

2.1. Fractal encoding

All existing practical implementations of block-oriented fractal coding schemes emerge from an affine transformation which is capable of performing scaling, rotating, mirroring, and shifting operations in order to exploit the presupposed self-similarities of the signal. An affine transformation W of the entire signal $\mathbf{x}$ is defined by

$$W:\ \mathbb{R}^n \to \mathbb{R}^n \quad \Rightarrow\ \mathbf{x} \to \mathbf{A}\mathbf{x} + \mathbf{b} \tag{6}$$

consisting of a linear part $\mathbf{A}\mathbf{x}$ and an additive part $\mathbf{b}$. The transformation W is called *contractive* if there exists a constant factor $s < 1$, such that

$$\varrho\,(W(\mathbf{x}), W(\mathbf{y})) \leq s\ \varrho(\mathbf{x}, \mathbf{y})\ \ \forall\, \mathbf{x}, \mathbf{y} \in \mathbb{R}^n. \tag{7}$$

With the definition of the metric (2), the affine transformation (6), and the contractivity (7) we obtain the sufficient condition

$$\|\mathbf{A}\| \leq s < 1 \tag{8}$$

for contractivity in the sense of any norm.

The encoding process for a given signal $\mathbf{x}$ now consists in finding a contractive transformation $\mathbf{A}$ and a vector $\mathbf{b}$ such that the approximation error

$$\varrho(W(\mathbf{x}), \mathbf{x}) = \varrho(\mathbf{A}\mathbf{x}+\mathbf{b}, \mathbf{x}) \tag{9}$$

is as small as possible. Below it is shown that for reconstruction at the decoder only the knowledge of the transformation $(\mathbf{A}, \mathbf{b})$ is necessary. Therefore data compression can be achieved, if the transformation can be represented with fewer bits than the signal itself.

2.2. Fractal decoding

Banach's fixed point theorem gives us an idea how the decoding process works:

Let $W : \mathbb{R}^n \rightarrow \mathbb{R}^n$ be a contractive transformation and $(\mathbb{R}^n, \varrho)$ a metric space with metric ϱ. Then the sequence of signals $\{\mathbf{x}_k\}$ constructed by $\mathbf{x}_{k+1} = W(\mathbf{x}_k)$ converges for any arbitrary initial signal $\mathbf{x}_0 \in \mathbb{R}^n$ to the unique fixed point $\mathbf{x}_f \in \mathbb{R}^n$ of the transformation W, i.e.

$$\mathbf{x}_f = W(\mathbf{x}_f) = \mathbf{A}\mathbf{x}_f + \mathbf{b}\,. \tag{10}$$

The reconstruction error $\varrho(\mathbf{x}_f, \mathbf{x})$ between the original signal $\mathbf{x}$ and its fractal reconstruction $\mathbf{x}_f$ is then bounded by

$$\varrho(\mathbf{x}_f, \mathbf{x}) \leq \frac{1}{1 - \|\mathbf{A}\|}\, \varrho(W(\mathbf{x}), \mathbf{x})\,. \tag{11}$$

The contractivity condition (8) is sufficient but not necessary for the convergence of the iteration process. For affine transformations a sufficient and necessary condition can be formulated by using the spectral radius of the transformation matrix. Emerging from any arbitrary initial signal $\mathbf{x}_0$ the decoder iteratively applies the transformation (6). For the *k*-th iterate then follows

$$\mathbf{x}_k = \underbrace{W(W(\cdots W(\mathbf{x}_0)))}_{\text{k times}} = W^{\circ k}(\mathbf{x}_0) = \mathbf{A}^k\mathbf{x}_0 + \left(\sum_{i=0}^{k-1} \mathbf{A}^{\mathbf{i}}\right)\mathbf{b}\,. \tag{12}$$

If and only if the spectral radius satisfies $r_\sigma(\mathbf{A}) < 1$, then $\lim\limits_{k\to\infty} \mathbf{A}^k = \mathbf{0}$ and $\lim\limits_{k\to\infty} \sum\limits_{i=0}^{k-1} \mathbf{A}^i = (\mathbf{I} - \mathbf{A})^{-1}$ with the identity $\mathbf{I}$ and the null matrix $\mathbf{0}$ [17]. Hence the sequence $\{\mathbf{x}_k\}$ converges to the fixed point of the transformation

$$\mathbf{x}_f = \lim_{k\to\infty} \mathbf{x}_k = (\mathbf{I} - \mathbf{A})^{-1}\mathbf{b} = \mathbf{A}x_f + \mathbf{b}. \tag{13}$$

Due to the fact that not the original signal itself, but a fixed point of a contractive transformation which is close to the signal, is encoded, fractal coding sometimes is termed *attractor coding*.

3. FRACTAL IMAGE CODING

3.1. Encoding of gray-scale images

In contrast to common transformations, e.g. DCT, whose coding gain emerges from the correlations of adjacent samples, fractal coding schemes mainly exploit some sort of *long range correlations* also termed *self-similarities* within the signal. For practical reasons fractal coding schemes operate in a block-oriented manner which allows describing them as vector quantization with signal dependent codebook.

The image $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T \in \mathbb{R}^n$ consisting of n pixels $x_1, x_2, \ldots, x_n$ is segmented into $N_R = n/n_R$ non-overlapping image blocks $\mathbf{x}_i$ with n_R elements. Then for each of these blocks one codebook entry $\mathbf{y}_j$ from a set of N_D entries is selected which after scaling with α_{ij} and adding an offset $b_i\mathbf{1}$ minimizes some given distortion measure

$$\varrho(\mathbf{x}_i, \hat{\mathbf{x}}_i) = \varrho(\mathbf{x}_i, \alpha_{ij}\mathbf{y}_j + b_i\mathbf{1}). \tag{14}$$

The codebook is generated from the image $\mathbf{x}$ itself by use of a codebook construction matrix $\mathbf{C}$. If $\mathbf{F}_j$ denotes the 'fetch-operation' of the codebook entry $\mathbf{y}_j = \mathbf{F}_j\mathbf{C}\mathbf{x}$ from the codebook and $\mathbf{P}_i$ the 'put-operation' of the modified codebook entry $\hat{\mathbf{x}}_i = \alpha_{ij}\mathbf{y}_j + b_i\mathbf{1}$ into the approximation $\hat{\mathbf{x}}$, the mapping process for the entire image $W : \mathbb{R}^n \to \mathbb{R}^n$ may be formulated by

$$\hat{\mathbf{x}} = \sum_{i=1}^{N_R} \mathbf{P}_i(\alpha_{ij}\mathbf{F}_j\mathbf{C}\mathbf{x} + b_i\mathbf{1}) = \left\{\sum_{i=1}^{N_R} \mathbf{P}_i\alpha_{ij}\mathbf{F}_j\mathbf{C}\right\}\mathbf{x} + \sum_{i=1}^{N_R}\mathbf{P}_i b_i\mathbf{1} = \mathbf{A}\mathbf{x} + \mathbf{b} = W(\mathbf{x}) \tag{15}$$

This represents an affine transformation consisting of a linear part $\mathbf{A}$ and a non-linear offset $\mathbf{b}$ which together form the *fractal code* $(\mathbf{A}, \mathbf{b})$ for the approximation of the original signal.

3.2. Quantization of the transform parameters

For a fractal based transmission and storage system the fractal code $(\mathbf{A}, \mathbf{b})$ must be represented by a number of bits which should be as small as possible. For the sake of simplification the fractal code is treated in a blockwise manner with each block code regarded independently from all others, so no inter-block DPCM is performed. The block code itself consists of four different parts, the scaling coefficient α_{ij}, the index for the codebook entry $j(i)$, the type of isometric transformation which has to be applied to the codebook entry $\mathbf{y}_j$, and the gray-scale offset b_i. The first three parameters for all blocks are contained in the transformation matrix $\mathbf{A}$ and the gray scale offset in the vector $\mathbf{b}$. Since α_{ij} and b_i are real valued, a quantization step is necessary as outlined below.

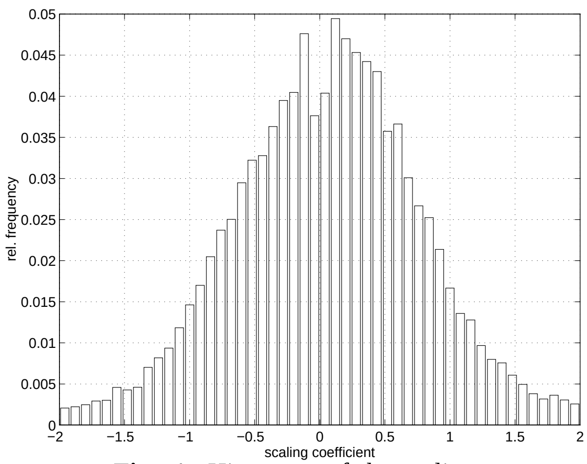

Fig. 1. Histogram of the scaling coefficient α_{ij} for $\alpha_{\max} = 2.0$.

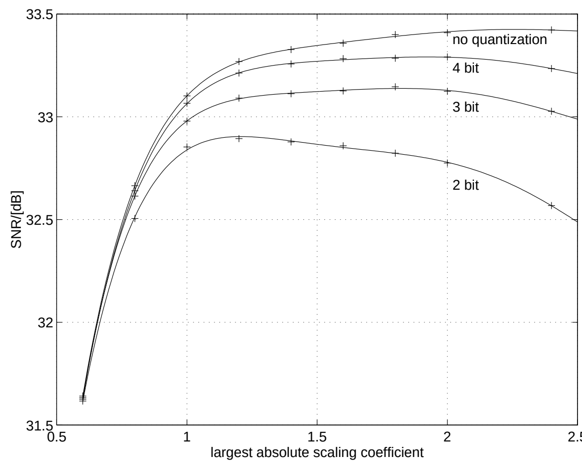

Fig. 2. Dependency of signal to noise ratio from $\alpha_{\max}$ and the quantization of the scaling coefficient.

Fig. 1 shows the Gaussian like histogram of the scaling coefficient derived from a large number of test images. Practical implementations restrict the scaling coefficients to the range $-\alpha_{\max} \le \alpha_{ij} \le \alpha_{\max}$; $\alpha_{\max} > 0$. For $0 < \alpha_{\max} < 1$ convergence of the reconstruction process can be guaranteed without any additional computational effort. Computer simulation showed that also for larger scaling coefficients a convergent reconstruction is obtained in most cases, but then a complex eigenvalue analysis of the transformation matrix $\mathbf{A}$ is necessary in order to decide whether the reconstruction converges or not. Details about the convergence properties of fractal coding schemes may be found in [18, 19]. The quantization of the scaling coefficients is rather uncritical since only this codebook entry $\mathbf{y}_j$ together with its appropriate *quantized* scaling coefficient is selected which fits best in the sense of the used distortion measure. In practical implementations 2, 3 or 4 bit have proven to be sufficient for the quantization of the scaling coefficients. As can be seen in Fig. 2 the number of bits spent for the scaling coefficients also influences the optimal bound $\alpha_{\max}$ ranging from 1.2 to 2.5.

The second real valued parameter within the fractal code is the vector $\mathbf{b}$ containing the gray-scale offsets b_i for each block within the image. Its histogram is depicted in Fig. 3. Depending on the desired reconstruction quality three to eight bits have been found to be sufficient for its representation.

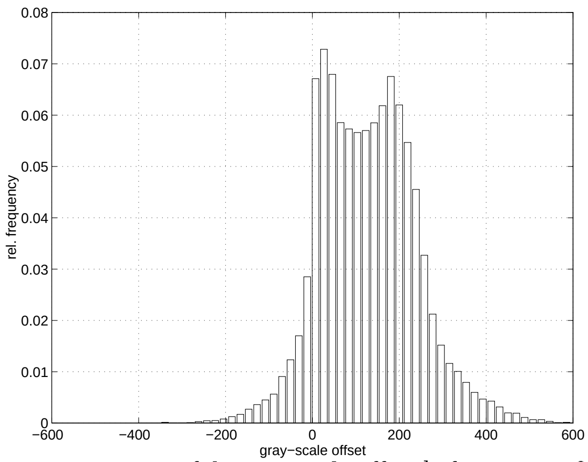

Fig. 3. Histogram of the gray-scale offset b_i for $\alpha_{\max} = 2.0$.

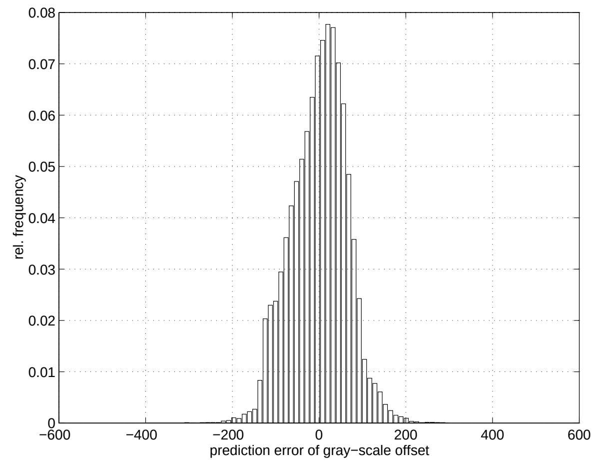

Fig. 4. Histogram of the prediction error $\triangle b_i = b_i - \hat{b_i}$ for $\alpha_{\max} = 2.0$.

The correlation coefficient between scaling and offset parameter is in the range 0.7 ... 0.95 depending on the image and the allowed scaling coefficients. This indicates that some additional savings in terms of bits necessary for representing the fractal code can be obtained, if those correlations are exploited. Fig. 5 shows that a prediction $\hat{b_i} = c_1 \alpha_{ij} + c_2$ for the offset, which is linear in the scaling coefficient, might be suited for these purpose. Here c_1, c_2 are the parameters describing the regression line as plotted in Fig. 5.

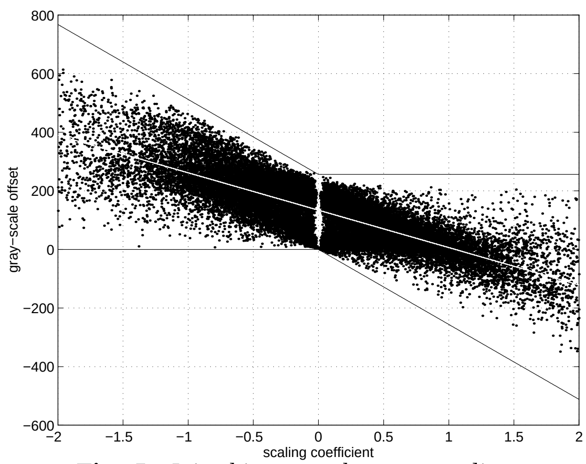

Fig. 5. Joint histogram between scaling coefficient α_{ij} and gray-scale offset b_i.

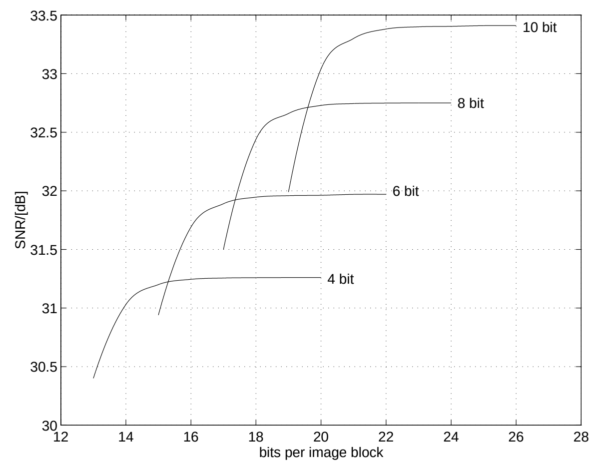

Fig. 6. Signal to noise ratio vs bits/image block. The codebook for each block contains 2^4, 2^6, 2^8, or 2^{10} entries, represented by 4, 6, 8, or 10 bits respectively.

Due to the restriction of the scaling coefficients and the range of allowed gray-scale values, also the offset parameters are constrained restricted. Its permissible range as function of the actual scaling coefficient is outlined in Fig. 5. Instead of quantizing and encoding the gray-scale offset independently from the scaling coefficient the prediction $\hat{b_i}$ serves as an estimate for b_i and therefore only the prediction error $\triangle b_i = b_i - \hat{b_i}$ needs to be transmitted. The advantage of this procedure can be seen easily from the histogram of the offset parameter b_i (Fig. 3) and the prediction error $\triangle b_i$ (Fig. 4). Since the distribution of the prediction error has a significantly smaller variance, it can be encoded with fewer bits compared with direct encoding of the offset parameter.

Additionally an index $j(i)$ for the selected codebook entry y_j must be included in the fractal code. The required number of bits is determined by the size of the codebook which typically contains about 2^4–2^{10} entries and significantly influences the available reconstruction quality. Moreover the codebook entry might be isometrically transformed in one of eight ways. Since all types of isometric transformations are approximately selected with same probability a three bit long fixed length code is additionally needed. Summarizing, one block can be represented by 11–25 bits, depending on the desired image fidelity. The available reconstruction quality in terms of the SNR for a typical test image and for various choices of the bit allocation is displayed in Fig. 6.

3.3. Application to color images

All statements carried out in the previous topics dealt with encoding of gray-scale images or images consisting of only one image plane. In order to present a complete encoding scheme for images, also the treatment of "multispectral images" should be considered. The following paragraphs describe the extension of the ordinary fractal coding scheme to a scheme capable of encoding color images. We restrict our investigations to RGB and YUV images, but the algorithm may easily be extended to other color spaces.

In the area of computer vision color images mostly are associated with three different sets of data each describing a different spectral domain of the same scenery. The color image itself is the composition of these sets. This is due to the fact that the color of light received by the human observer may be specified by a mixture of only three so-called primary colors. In the RGB space the primary colors are chosen to be red, green, and blue. In the YUV space the Y component contains the information about the intensity of the light whereas the U and V components contain the color information. For natural images the three image planes are highly correlated. In the RGB color space the correlation coefficient between two of the planes is about 0.78 for blue-red, 0.89 for red-green and 0.94 for green-blue, respectively. This indicates that significant improvements in terms of compression and/or reconstruction quality can be expected if those correlations are exploited.

The presented algorithm is based on the proposal published in [16]. It is shown that exploiting the correlations between the fractal codes of the different color planes for one block location results in a significant decrease in bits necessary for representing the fractal code for the entire image. For this purpose the parameters of the transform domain rather than the samples of the original domain are treated. It is shown that for the same block location in different color planes not only the pixel intensities but also the parameters of the fractal code are highly correlated. Hence only slight modifications of the fractal code for one color plane are necessary in order to describe the other planes with sufficient accuracy.

For the RGB color space the principal encoding procedure may be described exemplarily as follows: First the green or master component is encoded independently from the other ones which yields the master code $(\mathbf{A}, \mathbf{b})_G$. Green has been chosen for the master component because it possesses the largest correlation coefficient to the other components. This is also confirmed by a slightly better reconstruction quality as is reported in [16]. Initially the master code is also assigned to the slave components red and blue, so that $(\mathbf{A}, \mathbf{b})_{R,B} = (\mathbf{A}, \mathbf{b})_G$. This is of course an insufficient description for some blocks of the slave components, so that for those blocks a successive modification of the encoding parameters is performed until a sufficient reconstruction quality for all color planes is achieved. Then both slave codes $(\mathbf{A}, \mathbf{b})_{R,B}$ are only variations of the master code. The transition functions T_R, T_B with $(\mathbf{A}, \mathbf{b})_{R,B} = T_{R,B}((\mathbf{A}, \mathbf{b})_G)$ describing the modifications of the master code in order to get the slave code are much less complex and therefore may be stored with fewer bits than the slave codes themselves. Storing the master code $(\mathbf{A}, \mathbf{b})_G$ and the two transition functions T_R, T_B is much more efficient than storing all three codes $(\mathbf{A}, \mathbf{b})_{R,B,G}$ independently from each other. The two slave components are treated equally since there is no advantage in regarding the third component as function of the second one.

step	codebook entry	type of isometry	scaling coefficient	gray-scale offset
1	old	old	old	old
2	old	old	old	new
3	old	old	new	new
4	old	new	new	new
5	new	new	new	new

Tab. 1. Order of recalculating the different encoding parameters for the slave components.

In detail the process works as follows: The green component is encoded just the same way as an ordinary gray-scale image. Then the fractal code of the green component is blockwise assigned to the slave components red and blue. If some error criterion is met the master code is also suitable for describing the slave components with sufficient accuracy and encoding of this particular block is finished after step one as is indicated in Tab. 1. If the reconstruction error is to large a recalculation of the gray-scale offset is performed (step 2). If still no sufficient approximation is obtained the remaining transform parameters which are the scaling coefficient (step 3), the type of isometric transformation (step 4), or even the codebook entry itself (step 5) are recalculated.

In the same way as described above for the RGB space the encoding procedure can also be performed on the basis of the YUV color space. Naturally the luminance component Y is chosen to be the master component whereas the chrominance signals U and V are the slave components. Tab. 2 depicts for the test image "lena" up to which step the slave code must be refined in order to get a sufficient reconstruction quality for the RGB and YUV color space. One can see that for both color spaces a recalculation of the type of isometric transformation (step 4) without also determining a new codebook entry is very rare. Therefore this step may be omitted so that if no sufficient approximation is obtained after determining a new scaling coefficient, a new codebook entry together with an appropriate isometric transformation is selected. Hence, two bits are sufficient for determining the type of slave transformation.

component	step 1	step 2	step 3	step 4	step 5
Y	-	-	-	-	4096
U	272	3803	19	0	2
V	179	2649	231	14	32
G	-	-	-	-	4096
R	281	3088	294	31	402
B	1120	2407	100	14	455

Tab. 2. Number of slave blocks for which a refinement of the fractal code has to be performed.

4. IMPLEMENTATION, RESULTS AND CONCLUSIONS

The reconstruction quality is essentially determined by the number and size of blocks within the image. A variable block size has proven advantageous in comparison to a fixed size. As is reported in [10] a coding gain up to 3 dB can be expected if a variable block segmentation is applied. For the sake of a small segmentation overhead a quadtree partitioning has been chosen which allows to adjust the number and size of image blocks within a wide range. So the first step of the encoding process is to determine the number of blocks together with an appropriate block partitioning. If the number of available bits is prescribed, a resulting average number of bits per image block can be calculated.

Secondly the size of the codebook is fixed. According to Fig. 6 high reconstruction quality requires a large codebook but on the other hand for high compression a smaller one proves optimal.

No difference between the color components has been made for the scaling coefficients since their quantization turned out to be uncritical as mentioned above. This is not the case for the gray-scale offset. Here the larger portion of available bits is assigned to the luminance or green component. The optimal bit allocation for the gray-scale offset is depicted in Tab. 3

overall bits for gray-scale offset	RGB space	YUV space
16	R-5/G-6/B-5	Y-6/U-5/V-5
15	R-5/G-5/B-5	Y-6/U-5/V-4
14	R-5/G-5/B-4	Y-6/U-4/V-4
13	R-4/G-5/B-4	Y-5/U-4/V-4
12	R-4/G-4/B-4	Y-5/U-4/V-3
11	R-4/G-4/B-3	Y-5/U-3/V-3
10	R-3/G-4/B-3	Y-4/U-3/V-3

Tab. 3. Bit allocation for the gray-scale offset.

	codebook entry	type of isometry	scaling coefficient	gray-scale offset	type of slave transform
master component	4-10	3	2-4	4-6	-
one slave component	0, 4-10	0, 3	0, 2-4	0, 3-6	2
			compression 4x4 blocks	**compression 8x8 blocks**	**compression 16x16 blocks**
min number of bits per block		17	21:1	90:1	361:1
max number of bits per block		73	6:1	23:1	84:1

Tab. 4. Overall bit allocation for master and slave component. Zero bits for some of the encoding parameters of the slave component indicate that the corresponding parameter of the master component is taken.

Tab. 4 summarizes the various bit allocation possibilities which assign at least 17 and at most 73 bits to one image block consisting of three different image planes. This results in a compression factor depending on the size of the underlying image block rising from 6:1 for 4 by 4 blocks up to 361:1 for 16x16 image blocks. For typical natural test images an average compression factor of 20:1 to 60:1 can be obtained with reasonable fidelity. The signal to noise ratio representing the objective picture quality strongly depends on the type of image. It lies between 21–24 dB for the test image "baboon" and between 28 and 36 dB for the image "lena". In contrast to the results published in [16] encoding of the RGB components resulted in a slightly better reconstruction quality compared to the YUV components. An additional improvement may be obtained if the master component is permitted to change from block to block, so that for each block that component is regarded as master component which results in the best reconstruction quality. A comparison with the common JPEG algorithm resulted in advantages for the fractal compression scheme if high compression is desired and in slight advantages for JPEG if very good reconstruction is the issue.

5. REFERENCES

[1] M. F. Barnsley, V. Ervin, D. Hardin, and J. Lancaster, "Solution of an inverse problem for fractals and other sets," in *Proc. Natl. Acad. Sci. USA*, vol. 83, pp. 1975–1977, Apr. 1986.

[2] M. F. Barnsley and J. H. Elton, "A new class of markov processes for image encoding," *Advances in applied probability*, vol. 20, pp. 14–22, 1988.

[3] M. F. Barnsley, *Fractals Everywhere*. London: Academic Press Inc., 1988.

[4] A. E. Jacquin, "A novel fractal based block-coding technique for digital images," in *Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing ICASSP'90*, vol. 4, pp. 2225–2228, 1990.

[5] A. E. Jacquin, "Fractal image coding based on a theory of iterated contractive image transformations," in *Proceedings SPIE Visual Communications and Image Processing '90*, vol. 1360, pp. 227–239, 1990.

[6] A. E. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations," *IEEE Transactions on Image Processing*, vol. 1, pp. 18–30, Jan. 1992.

[7] A. E. Jacquin, "Fractal image coding: A review," *Proceedings of the IEEE*, vol. 81, pp. 1451–1465, Oct. 1993.

[8] L. Lundheim, *Fractal signal modelling for source coding*. PhD thesis, Universitetet I Trondheim Norges Tekniske Høgskole, 1992.

[9] J. M. Beaumont, "Advances in block based fractal coding of still pictures," in *Proceedings of the IEE colloquium "The Application of Fractal Techniques in Image Processing '90"*, Dec. 1990.

[10] B. Hürtgen, F. Müller, and C. Stiller, "Adaptive fractal coding of still pictures," in *Proceedings of the International Picture Coding Symposium PCS'93*, (Lausanne, Switzerland), p. 1.8, 1993.

[11] D. M. Monro, "A hybrid fractal transform," in *Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing ICASSP'93*, vol. 5, pp. 169–172, 1993.

[12] G. E. Øien, *L_2-optimal attractor image coding with fast decoder convergence*. PhD thesis, Universitetet I Trondheim Norges Tekniske Høgskole, 1993.

[13] S. Lepsøy, *Attractor image compression - Fast algorithms and comparisons to related techniques*. PhD thesis, Universitetet I Trondheim Norges Tekniske Høgskole, 1993.

[14] H. Yan and G. Filippoff, "Color image compression based on fractal geometry," in *Proceedings of the 2nd Singapore International Conference on Image Processing '92*, pp. 3–5, 1992.

[15] B. Goel and S. Kwatra, "A data compression algorithm for color images based on run-length coding and fractal geometry," in *Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing ICASSP'88*, pp. 1253–1256, 1988.

[16] R. D. Boss and E. W. Jacobs, "Studies of iterated transform image compression and its application to color and DTED," Tech. Rep. 1468, Naval Ocean Systems Center, San Diego, CA, Dec. 1991.

[17] E. Kreyszig, *Introductory functional analysis with applications*. Robert E. Krieger Publishing Company, 1989.

[18] B. Hürtgen and F. Müller, "Modelling of fractal coding schemes," in *Proceedings of the VIIth European Signal Processing Conference EUSIPCO'94*, vol. 1, (Edinburgh, Scotland), pp. 600–603, 1994.

[19] B. Hürtgen and S. F. Simon, "On the problem of convergence in fractal coding schemes," in *Proceedings of the IEEE International Conference on Image Processing ICIP'94*, vol. 3, (Austin, Texas, USA), pp. 103–106, Nov. 1994.