Multi-layered Image Representation: Application to Image Compression

F.G. Meyer [1], A.Z. Averbuch [2], J.O. Strömberg [3], and R.R. Coifman [4]
[1] Radiology and Computer Science, [4] Mathematics, Yale University, New Haven, CT
[2] School of Mathematical Sciences, Tel Aviv University, Tel Aviv 69978, Israel
[3] Dept. of Mathematics, Tromsö University, Norway
e-mail: francois.meyer@yale.edu

Abstract

We describe a new multi-layered representation technique for images. An image is encoded as the superposition of one main approximation, and a sequence of residuals. The strength of the multi-layered method comes from the fact that we use different bases to encode the main approximation and the residuals. The different bases complement each others: –some of the basis functions will give reasonable account of the large trend of the data, while others will catch the local transients, or the oscillatory patterns. By selecting different bases, we allow different features to be discovered in the image.

1 Introduction

The underlying assumption behind standard compression methods is that the basis (e.g. the DCT basis, or a wavelet basis) used for compression is well adapted to most images. If one wishes to obtain a very sparse representation of an image then it is critical that the functions in the basis bear a strong resemblance to the image being analyzed. If the geometric properties (shape, smoothness, periodicity, etc) of the basis functions match the corresponding characteristics of the data to be analyzed, then the coefficients in such a basis are meaningful, and provide rich information about the data being analyzed. Unfortunately basis functions that compactly code smooth images (such as wavelets with many vanishing moments) are ill suited to represent oscillatory patterns. Reciprocally, oscillatory analyzing functions such as the DCT can efficiently code textures, but they are ill equipped to represent smooth images. Our approach follows a completely different direction: – instead of forcing all images to adapt to one single basis, we use a collection of libraries of bases to represent a single image. The main contribution of this paper is a new paradigm for image representation and image compression. We describe a new multi-layered representation technique for images. An image is encoded as the superposition of one main approximation, and a sequence of residuals.

2 Libraries of bases

During each iteration of the algorithm, we encode the remainder R^i of the previous iteration. This is done by selecting the basis that is best adapted to R^i among a very large and highly redundant library $\mathcal{L}$ of waveforms [3]. Such libraries are composed of functions, or atoms, with different time frequency localization. The atoms provide an overcomplete representation: - there is not a unique decomposition of R^i over the library. Among all possible decompositions, one would like to pick up the most compact decomposition. In this work we will be using several libraries :

• **Local trigonometric functions** [1] provide an adaptive tiling of the spatial domain in terms of oscillating patterns. An image can be broken into blocks of different sizes within which a local Fourier expansion, or a DCT, is performed. Instead of abruptly cutting blocks in the image, we use *smooth orthogonal projectors* [1].

• **Wavelet packets** [7] make it possible to adaptively tile the frequency domain into different bands of arbitrary size. Wavelet packets have been used to characterize textures in images [2, 5, 8]. However a wavelet packet is always associated with two peaks in frequency that does not allow to selectively localize a unique frequency.

• **Brushlets** [6] are new families of steerable wavelet packets that adaptively segment the Fourier plane to obtain the most concise and precise representation of the image in terms of oriented textures with all possible directions, frequencies, and locations.

3 Cascade of compressions: the "multi-layered" paradigm

We consider a sequence of libraries of orthonormal bases:

$$(\mathcal{L}_i)_{i \in Z}\,. \tag{1}$$

In this work $\mathcal{L}_0$ contains only one single wavelet basis, and $\mathcal{L}_1$ can be the library of wavelet packets associated with some different QMFs, or a library of local cosine transforms. The choice of the libraries, and their arrangement

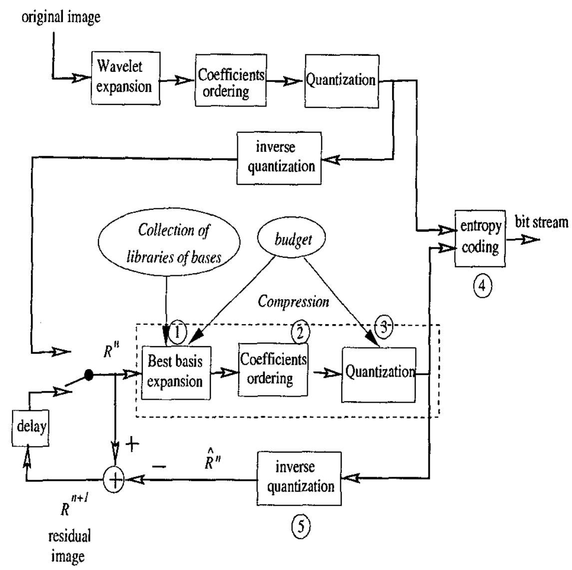

Figure 1: Block diagram of the multi-layered compression algorithm. In the first pass of the algorithm, the switch is turned towards the original image (i.e the image that we want to compress). In the subsequent refinement passes the switch is turned towards the residual image.

are parameters of the algorithm, and can be adjusted for large classes of images.

Let I be an image. We first compress I over the library $\mathcal{L}_0$, using at most b_0 bits. Let $\hat{R}_0$ be the result of the compression, after decompression. $\hat{R}_0$ is an approximation of the original image I:

$$I = \hat{R}^0 + R^1 \tag{2}$$

where R^1 is the approximation error. Let $\{\Phi_j^0,\ j \in E_0\}$ be the basis from $\mathcal{L}_0$ that was selected to compress I:

$$\hat{R}^0 = \sum_{j \in E_0} q_j^0 \Phi_j^0 \tag{3}$$

where $\{q_j^0,\}_{j \in E_0}$ are the quantized coefficients, and E_0 is the set of indices of the basis functions that constitute the best basis. At this point we refine the approximation of I by calculating an approximation of the residual R^1. This is achieved by compressing the residual with a budget of b_1 bits. In order to discover different features in the image, we use a different library: $\mathcal{L}_1$. We select the best basis $\{\Phi_j^1,\ j \in E_1\}$ from the library $\mathcal{L}_1$:

$$R^1 = \hat{R}^1 + R^2 \tag{4}$$

with

$$\hat{R}^1 = \sum_{j \in E_1} q_j^1 \Phi_j^1 \tag{5}$$

We note that we can reconstruct an approximation of I:

$$\hat{I}^1 = \hat{R}^0 + \hat{R}^1 \tag{6}$$

where $\hat{I}^1$ is an image that can be compressed with the budget of $b_0 + b_1$ bits. Figure 1 shows the main loop of the algorithm. Let us assume that we have carried the approximation of I up to step $n-1$, and let us explain how to calculate the term n of the approximation. Let R^n be the residual of the approximation at step $n-1$. We select the best basis $\{\Phi_k\}_{k \in E_n}$ from the library $\mathcal{L}_n$, where E_n is the set of indices of the basis functions that constitute the best basis. We expand R^n into the best basis $\{\Phi_k\}_{k \in E_n}$, and we compress R^n using the budget b_n:

$$R^n = \hat{R}^n + R^{n+1} \tag{7}$$

where $\hat{R}^n$ is the compressed residual (after decompression):

$$\hat{R}^n = \sum_{j \in E_n} q_j^n \Phi_j^n. \tag{8}$$

Finally we can reconstruct an approximation of I using the $n+1$ compressed residual images:

$$\hat{I}^n = \sum_{k=0}^{n} \sum_{j \in E_k} q_j^k \Phi_j^k \tag{9}$$

where $\hat{I}^n$ is an image that can be compressed with the budget of $\sum_{k=0}^{n} b_k$ bits. Note that the coefficients of the non-linear approximation are not the inner products with the basis functions. They are the quantized versions of the coefficients of the non-linear approximation. We can stop the iterations when the residual is noisy and does not contain any apparent structure. This noisy part constitutes the incoherent noise that was present in the image. At this point no more structure that can be pulled out of the residual image.This can be made explicit by looking at the statistical distribution of the coefficients of the expansion. After the final compression iteration, one might still want to save the final residual. This functionality is fully available in the multi-layer algorithm.

3.1 Toy example

Figures 2-7 illustrate the algorithm. Fig. 2 shows the original image *Barbara*. We show in Fig. 3 the result of the first iteration: the smooth regions of the image have been described with a few wavelet coefficients (compression ratio= 254). Fig. 4 shows the residual: what is left when we subtract the smooth regions $\hat{R}^0$ out of the original image. The residual R^1 contains all the edges, as well as the textured regions. The wavelet compressed image $\hat{R}^0$ constitutes the first part of the compressed image. During the second pass of the algorithm, we compress the first residual R^1 with an adapted local cosine basis (compression factor = 36.61).

Figure 2: Original image

Figure 3: The first layer $\hat{R}^0$ is a very sparse representation of the smooth trends in the image. This first layer was obtained by compressing the original image in a wavelet basis, by a factor of 254.

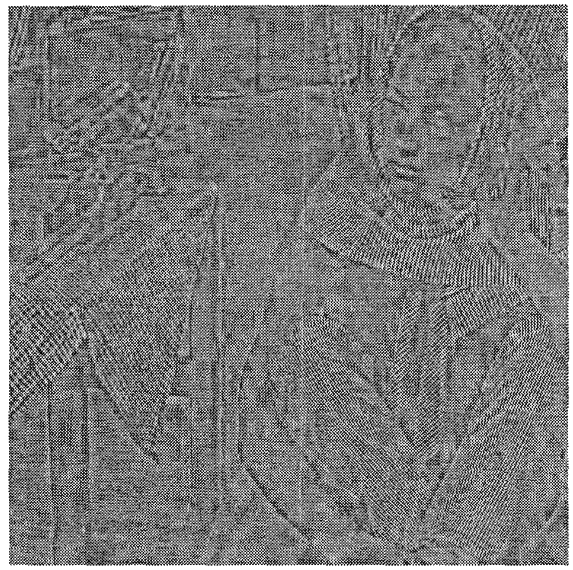

Figure 4: First residual part $R^1 = I - \hat{R}^0$

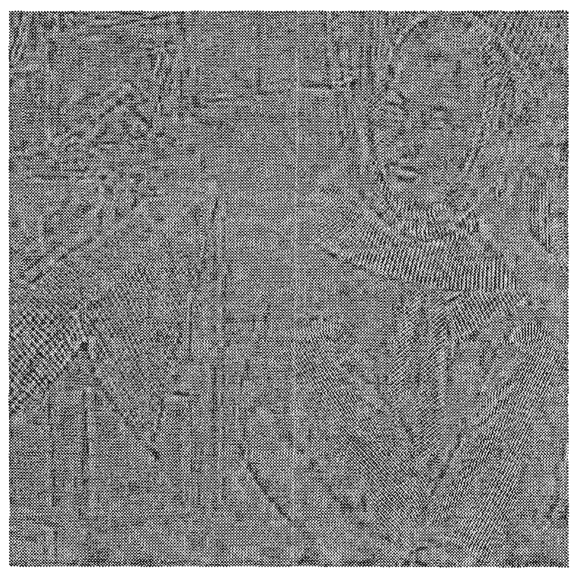

Figure 5: The second layer $\hat{R}^1$ is a sparse representation of the texture in the image. It was obtained by compressing the first residual image using an adapted local cosine basis by a factor of 36.61.

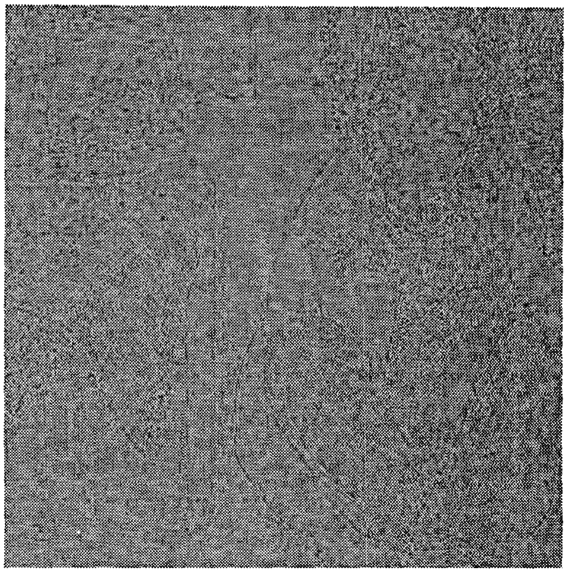

Figure 6: Second residual part $R^2 = R^1 - \hat{R}^1$

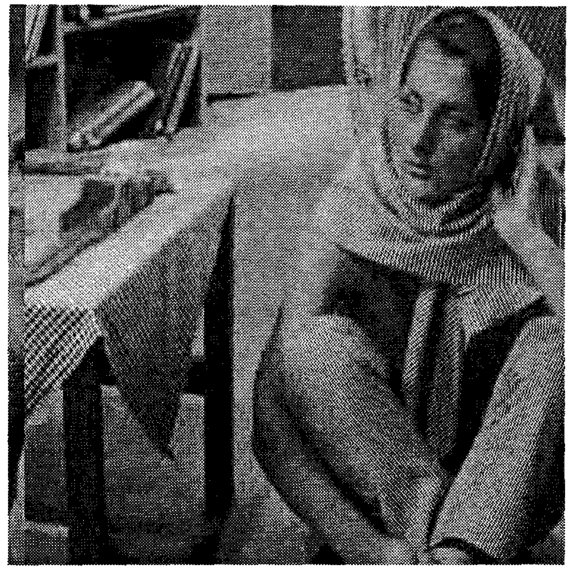

Figure 7: Barbara, Multi-layer: compression ratio: 32.

The result of the compression, $\hat{R}^1$, is shown in Fig. 5, and the error, R^2, is shown in Fig. 6. $\hat{R}^1$ constitutes the second layer. We note that most of the strong textures, as well as the edges have been removed from the image, and are encoded into this second layer. Note that the final residual R^2 is visually almost random noise, i.e. we do not perceive any structure inside the image. When $\hat{R}^0$, $\hat{R}^1$, and $\hat{R}^3$ are added together, we obtain an image which is compressed by a factor of 32, as shown in Fig. 7.

4 Frequency ordering and quantization

Once the best basis has been selected, the residual R^i is represented by a set of coefficients. Before quantization we organize the coefficients in such a way that we can take advantage of the expected decay of the coefficients. We use the same ordering, and the same quantization methods for wavelet libraries, and local trigonometric libraries. Our scanning method exploits the fact that if an image is smooth, then the amplitude of the coefficients decreases as the frequency of the basis function increases. This result is certainly true if we use local Fourier or cosine bases. In the case of wavelet packet we have a similar result [5]. Once the coefficients are ordered, we quantize them with a uniform scalar quantizer. The quantizer step is the same for all coefficients. Once the coefficients are quantized, a stream of bits is generated by bitplane encoding the absolute value of the quantized coefficients. If a coefficient is not quantized to zero, then its sign is also encoded. The bit stream contains long stretches of zeros and thus we use a zero-runlength coder to encode the stream. The parameters of the quantization are optimized in order to reach an exact budget. The best basis geometry is described by a quadtree. The number of bytes required to encode the geometry is always is small fraction of the total budget.

5 Experiments

We have implemented the coder and decoder, and an actual bit stream is generated by the coder. Note that for all experiments we generated a compressed file with a size equal to the targeted budget. We present the results of the multi-layer compression algorithm, using the test image 512×512 "Barbara". This image is difficult to compress because it contains a mixture of large smooth regions, and long oscillatory patterns. Other results can be found in [4]. In order to evaluate the performance of our algorithm, we compared it to one of the best wavelet coder that was available to us: the SPIHT wavelet coder of Amir Said and William A. Pearlman [9]. Fig. 2 shows the original image Barbara. Fig. 8 shows the result of a compression of 32 using SPIHT, and Fig. 7 shows the result of the multi-layer coder for the same compression of 32. In this example we used two layers: – a first compression with wavelets, and a second compression with an adapted local cosine basis. We note that the texture on the pants of Barbara, as well as the texture of the table-cloth are very well preserved with the multi-layered representation. Furthermore, the multi-layer coder does not introduce unpleasant ringing artifacts. A quantitative comparison of SPIHT and the multi-layer coder is provided in table 1. The multi-layer coder clearly outperforms SPIHT on the image Barbara, both in terms of PSNR and visual quality.

Figure 8: Barbara, SPIHT: compression ratio: 32.

Barbara			
Rate (bpp)	Compression	SPIHT	Wavelet-Packets
1	8	36.41	36.58
0.67	12	33.40	33.76
0.5	16	31.39	31.97
0.4	20	30.10	30.70
0.333	24	29.13	29.73
0.286	28	28.27	28.94
0.25	32	27.58	28.27
0.20	40	26.65	27.25
0.154	52	25.79	26.19
0.125	64	24.86	25.37
0.10	80	24.25	24.61

Table 1: Coding results. 8bpp. 512x512 Barbara

6 Discussion and conclusion

One can interpret this new coding algorithm in terms of a multi-layered representation of the image: Each residual $\hat{R}^i$ constitutes one set – or layer – of features that share the same structural properties. A layer is defined with respect to the choice of a particular basis. For instance: (i) the wavelet basis defines the layer of smooth regions, (ii) the wavelet packet defines the layer of textured patterns, (iii) the local trigonometric transforms define the localized textured features, etc. The algorithm encodes iteratively all those features that have a strong correlation with the basis elements during this iteration. Because each layer has a sparse representation in the associated basis, the superposition of layers achieves a very compact The question of how many residuals, and how much budget (how many bits b_i) should be allocated to each residual R^i remains open. At the moment our approach consist in using two layers: wavelets, and wavelet packets or local trigonometric transforms. We use an exhaustive search to find the optimal allocation of the budget between the two layers. In the future we intend to address this problem with a comprehensive methodology, based on a variational approach, that will provide a clear understanding on how to adjust these parameters.

References

[1] P. Auscher, G. Weiss, and M.V. Wickerhauser, *Local sine and cosine bases of Coifman and Meyer and the construction of smooth wavelets*, Wavelets-A Tutorial in Theory and Applications, 1992, pp. 237–256.

[2] T. Chang and C.C.J. Kuo, *Texture analysis and classification with tree-structured wavelet transform*, IEEE Trans. on Image Processing **2, 4** (1993), 429–441.

[3] R.R. Coifman and Y. Meyer, *Size properties of wavelet packets*, Wavelets and their Applications (Ruskai et al, ed.), Jones and Bartlett, 1992, pp. 125–150.

[4] F.G. Meyer, A.Z. Averbuch, and R.R. Coifman, *Multilayered image compression*, submitted, 1998.

[5] F.G. Meyer, A.Z. Averbuch, and J-O. Strömberg, *Fast adaptive wavelet packet image compression*, submitted to IEEE Trans. on Image Processing, also available from http://noodle.med.yale.edu/ meyer/profile.html, 1998.

[6] F.G. Meyer and R.R. Coifman, *Brushlets: a tool for directional image analysis and image compression*, Applied and Computational Harmonic Analysis (1997), 147–187.

[7] Y. Meyer, *Wavelets: Algorithms and applications*, SIAM, 1993.

[8] K. Ramchandran and M. Vetterli, *Best wavelet packet bases in a rate-distortion sense*, IEEE Trans. on Image Processing (1993), 160–175.

[9] Amir Said and William A. Pearlman, *A new fast and efficient image codec based on set partioning in hierarchical trees*, IEEE Trans.on Circ.& Sys. for Video Tech. **6** (1996), 243–250.