Three-pass Lossless Image Compression

Mohammad Fathi, Hassan Taheri, M. A. Fasihi
Department of Electrical Engineering
Amirkabir University of Technology
htaheri@aut.ac.ir

Abstract:
Context-based lossless image compression schemes use 180° type modeling contexts in one pass scanning an image. In this paper, a lossless image compression (LIC) scheme is introduced and for capturing high-order dependencies, statistical features and spatial configuration of an image, three-pass scheme with 360° type modeling contexts is proposed. Linear and nonlinear relationships between pixels in a context are used for context determination. Also the three-pass algorithm is applied to JPEG-LS standard, and for context determination, local gradients of intensity capturing the level of activity (smoothness, edginess) surrounding a pixel, is employed. For images with small dimensions three-pass schemes and for large dimensions one-pass schemes result in higher compression ratios.

Keywords: Image Compression, JPEG-LS, Prediction.

1. Introduction

An image compression scheme consists of two components, an encoder and a decoder. The encoder takes an uncompressed image and encodes it into a more compact format. The decoder performs the opposite actions of the encoder; it takes the encoded image and attempts to reconstruct the original uncompressed image. This process will either be lossless, near lossless or lossy, which will be determined by the particular needs of the user. The two methods of lossless and near-lossless image compression both deal with the value or brightness of each individual pixel.

Lossless compression guarantees that the value of each pixel in the reconstructed image will match its corresponding original value. When near-lossless compression is used, the reconstruction process may introduce errors in the reconstructed values, the maximum magnitude of these errors can usually be limited by the user. Rather than deal with the value of every pixel, lossy compression schemes attempt to determine visually important components of an image. By maintaining visually important information, the reconstructed image has a similar "look" to the original. By throwing away visually unimportant information, the level of compression can be improved. Recent lossless compression schemes have mostly been composed using a two-stage process involving prediction and coding.

1.1 Prediction

It is assumed that an image is scanned using a raster-scan technique, that is to say the pixels are scanned from left to right, top to bottom. Therefore, at any time instant t it is assumed that all the previous pixels $x_0x_1\ldots x_{t-1}$ have been scanned. Both the encoder and the decoder scan the image in the same way. Therefore, if the encoder is encoding x_t, then it can assume that

the decoder has decoded the pixels $x_0x_1\ldots x_{t-1}$. The common scanning method used by both the encoder and decoder allows the value of x_t to be predicted in the same way by the encoder and decoder. The pixels that are used for this prediction cause the predictor to predict in a certain way, thus they are referred to as the causal neighborhood.

1.2 Static Prediction

The first step is to use a fixed prediction function that predicts the value of the current pixel. An example of a prediction function is the Median Adaptive Predictor, is presented below [1].

$$\hat{x} = \begin{cases} \min(N,W) & \text{if} \quad NW \geq \max(N,W) \\ \max(N,W) & \text{if} \quad NW \leq \min(N,W) \\ N+W-NW & \text{otherwise} \end{cases} \qquad (1)$$

Another way of expressing this predictor is to say it returns the median of three *N, W, N+W-NW*. Its adaptive nature and low complexity give a good tradeoff between accuracy and speed.

1.3 Adaptive Correction

Once a predicted value is obtained from the static predictor(s), a scheme can perform adaptive correction on that prediction. This stage is more open to the implementers to experiment, however there are a few common methods. The most straightforward method is to correct the predicted value by the current mean predicted value. This tends to involve "learning" the behavior of the prediction errors over time. Depending on the state of knowledge at that particular time, the encoder and decoder can correct the predicted value based on previous observed behavior.

2. TSGD Model

It is a widely accepted observation [2] that the global statistics of residuals from a fixed predictor in continuous-tone images are well modeled by a TSGD (two-sided geometric distribution) centered at zero. According to this distribution, the probability of an integer value *e* of the prediction error is proportional to $\theta^{|e|}$, where $\theta \in (0,1)$ controls the two-sided exponential decay rate. However, it was observed that [3] a dc offset is typically present in *context-conditioned* prediction error signals.

This offset is due to integer-value constraints and possible bias in the prediction step. Thus, a more general model, which includes an additional *offset parameter μ,* is appropriate. Letting μ take noninteger values, the two adjacent modes often observed in empirical context-dependent histograms of prediction errors are also better captured by this model. We break the fixed prediction offset into an integer part R (or "bias"), and a fractional part s (or "shift"), such that $\mu = R - s$, where $0 \leq s < 1$. Thus, the TSGD parametric class $P(\theta,\mu)$, for the residuals of the fixed predictor at each context, is given by:

$$\begin{cases} P_{(\theta,\mu)}(\in) = C(\theta,s)\theta^{|\in - R + s|}, \in = 0,\pm1,\pm2,\ldots \\ C(\theta,s) = (1-\theta) \Big/ (\theta^{1-s} + \theta^{s}) \end{cases} \qquad (2)$$

Where, $C(\theta,s)$ is a normalization factor. The bias R calls for an integer adaptive term in the predictor. In the sequel, we assume that this term is tuned to cancel *R*, producing average residuals between distribution modes located at 0, -1. Consequently, after daptive prediction, the model of (2) reduces to:

$$\begin{cases} P_{(\theta,\mu)}(\in) = C(\theta,s)\theta^{|\in + s|}, \in = 0,\pm1,\pm2,\ldots \\ 0 < \theta < 1, 0 \leq s < 1 \end{cases} \qquad (3)$$

The model of (3) is depicted in fig.1.

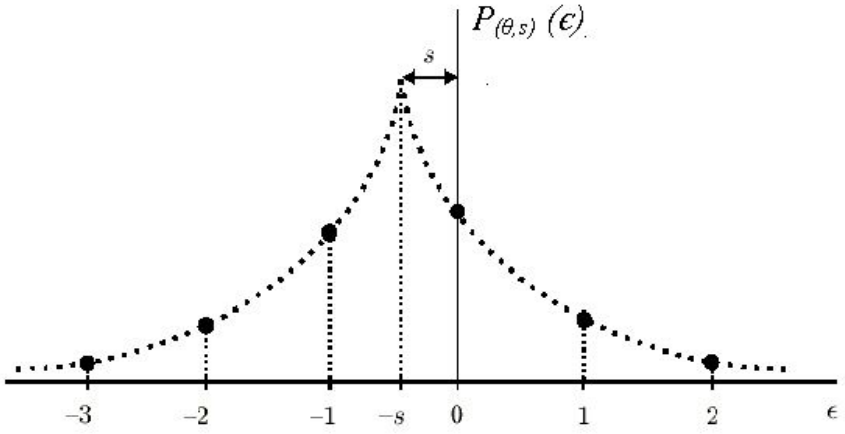

Figure 1: Two-sided geometric distribution

The TSGD centered at zero corresponds to *s=0*, and, when *s=1/2* is a bi-modal distribution with equal peaks at –1 and 0. This reduced range for the offset is matched to the Golomb codes [4,5], whose structure enables simple calculation of code word of any given sample, without recourse to the storage code tables, as would be the case with unstructured, generic Huffman codes. In an adaptive mode, a structured family of codes further relaxes the need to dynamically updating code tables due to possible variations in the estimated parameters [6].

2.1 Bias Estimation

In principle, *maximum-likelihood* (ML) estimation of R in (2) would dictate a bias cancellation procedure based on the *median* of the prediction errors incurred so far in the context by the fixed predictor (1). However, storage constraints rule out this possibility. Instead, an estimate based on the *average* could be obtained by just keeping a count N of context occurrences, and a cumulative sum B of fixed prediction errors incurred so far in the context. Then, a *correction value R* could be computed as the rounded average (4) and added to the fixed prediction $\hat{x}_{MED}$, to offset the prediction bias.

$$R = \left\lceil \frac{B}{N} \right\rceil \quad (4)$$

3. Context Determinations and Modeling in JPEG-LS Standard

The context that conditions the encoding of the current prediction residual for pixel *x* in JPEG-LS is built out of the following differences.

c	*b*	*d*
a	*x*	

Figure 2: JPEG-LS context

$$g_1 = d - b; \quad g_2 = b - c; \quad g_3 = c - a; \quad (5)$$

These differences represent the local gradient, thus capturing the level of activity (smoothness, edginess) surrounding a sample, which governs the statistical behavior of prediction errors. By symmetry, g_1, g_2 and g_3 influence the model in the same way. Since further model size reduction is obviously needed, each difference $g_i, i = 1,2,3$ is quantized into a small number of approximately *equiprobable*, connected regions by a quantizer $k(.)$ independent of i. This quantization aims at maximizing the *mutual information* between the current sample value and its context, an information-theoretic measure of the amount of information provided by conditioning context on the sample value to be modeled.

In principle, the number of regions into which each context difference is quantized should be adaptively optimized. However, the low complexity requirement dictates a fixed number of "equiprobable" regions. To preserve symmetry, the regions are indexed:
$q_i=k(g_i)=\{-T,\ldots-1,0,1,\ldots T\}\ k(g_i)=-k(-g_i)\ i=1,2,3,$
for a total $(2T+1)^3$ of different contexts. A further reduction in the number of contexts is obtained after observing that, by symmetry, it is reasonable to assume that:

$$prob\{e_t=\Delta|C_t=[q_1, q_2, q_3]\} = prob\{e_t= -\Delta|C_t=[-q_1, -q_2, -q_3]\} \quad (6)$$

Where C_t represents the quantized context triplet and $q_i = k(g_i), i = 1,2,3$. Hence, if the first nonzero element of C_t is negative, the encoded value is $-e_{t+1}$, using context $-C_t$. This is anticipated by the decoder, which flips the error sign if necessary to obtain the original error value. By merging contexts of "opposite signs," the total number of contexts becomes $((2T+1)^3+1)/2$.

For JPEG-LS, $T=4$ was selected [7], resulting in 365 contexts. This number balances storage requirements (which are roughly proportional to the number of contexts) with high-order conditioning. To complete the definition of the contexts in JPEG-LS, it remains to specify the boundaries between quantization regions.

For an 8-bit/sample alphabet, the default quantization regions are:

$$\{0\},\pm\{1,2\},\pm\{3,4,5,6\},\pm\{7,8,\ldots 20\},\pm\{e|e\geq 21\}$$

However, the boundaries are adjustable, except that the central region must be $\{0\}$.

4. Three-pass Interlaced Predictive Coding Scheme

The majority of the current lossless image compression methods code the pixels in the raster scan order. As a result, the contexts available for image modeling cannot spatially enclose the modeled pixels. At any moment, only the pixels at the top and to the left of a pixel being coded are known to the both encoder and decoder so that they can be used in modeling and prediction. We call such a spatial configuration of modeling context the 180° type, since the modeling pixels only form a semicircle around a modeled pixel. Many image features, such as the intensity gradient, edge orientation, and textures, can be better modeled in a completely enclosing context. We call a spatial configuration of modeling context that completely surrounds a modeled pixel the 360° type.

Intuitively, we desire an image-independent order of traversing pixels that can provide *adjacent*, *enclosing* modeling contexts for a maximum number of pixels. Three-pass interlaced sampling scheme is suitable in terms of overall compression gains. The encoding, and accordingly the decoding, of an image is done in three passes. Each pass uses an interlaced sampling of the original image. Denote a continuous-tone image of width W and height H by:

$I(i, j), 0 \leq i \leq W, 0 \leq j \leq H$.

The first pass encodes a sub sampled $W/2 \times H/2$ image denoted by μ , with the following relation between I and μ :

$$\mu(i, j) = \left\lfloor \frac{I(2i,2j) + I(2i+1,2j+1)}{2} \right\rfloor$$
$$0 \leq i \leq W/2, 0 \leq j \leq H/2 \quad (7)$$

$\mu(i, j)$ is average intensity of two diagonally adjacent pixels. This sub sampling is designed to benefit prediction and the context modeling in the subsequent passes. The $W/2 \times H/2$ image μ is encoded in raster scan sequence using a 180° type context. For instance, the prediction context may consist of previously encoded values

$\mu(i-1, j), \mu(i-1, j-1), \mu(i, j-1), \mu(i+1, j-1)$

as shown in Fig.3.

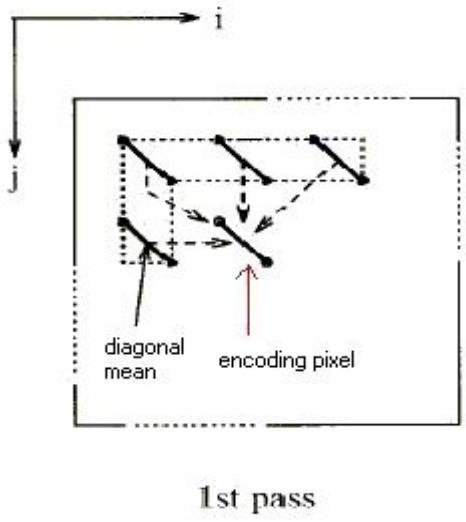

Figure 3: first pass

The prediction function is median estimation. The second pass uses sub-sampled image as the prediction contexts to encode *NH/2* pixels:

$$I(2i, 2j),\ I(2i+1, 2j+1),\quad 0 \le i \le W/2,\ 0 \le j \le H/2 \qquad (8)$$

Namely, the first pass codes the same pixels involved in the diagonal means. But in the second pass, individual pixel values will be resolved from the corresponding diagonal means. Again, the second pass proceeds in raster scan sequence. First, consider the encoding of *I(2i, 2j)*. Fig.4 is a snapshot of the second pass in which the x marks the pixel being currently coded, and the diagonals are the two-pixel means from the first pass.

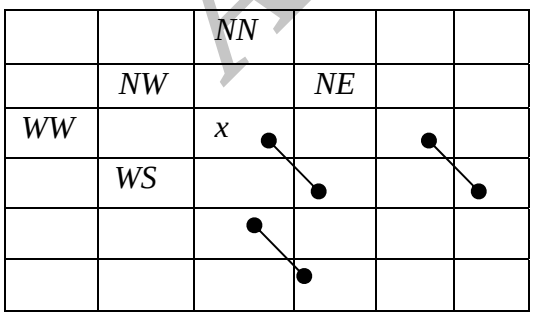

Figure 4: second pass

As shown by Fig.4, at this stage the sub sampled image μ from the first pass is available to the right and bottom of the current pixel, and the previously coded pixels in the second pass are available to the left and top. They provide 360° type contexts surrounding *I(2i, 2j)*. Specifically, we use a prediction context consisting of the values known to both the encoder and decoder: two-pixel means μ(*i, j*), μ(*i+1, j*), and μ(*i, j+1*), to the right and bottom of *I(2i, 2j)*, and pixels *I(2i-1, 2j+1), I(2i-1, 2j-1), I(2i+1, 2j-1), I(2i-2, 2j)* and *I(2i, 2j-2)* to the left and top of *I(2i, 2j)*, as marked in Fig. 4.

Once *I(2i, 2j)* is reconstructed, the decoder can set $I(2i+1, 2j+1) = 2\mu(I, j) - I(2i, 2j)$ without receiving any information on *I(2i+1, 2j+1)*.

The third pass encodes the remaining half of the original image. Namely, pixels interlaced in the checkerboard pattern, *I(2i, 2j+1)* and *I(2i+1, 2j)*. The prediction contexts available to the third pass are spatially enclosing and adjacent to the modeled pixels. If the third pass of the image is also done in the raster scan order, then as illustrated by Fig.5 an unknown pixel *x* in the third pass can use a 360° type context consisting of all of its four-connected neighbors, and two of its eight-connected neighbors.

NW	*N*	*NE*
W	*x*	*E*
	S	

Figure 5: third pass

5. Context Modeling and Quantization in LIC Scheme

In each of the tree passes, by selecting a context for every pixel, the value of it was predicted. Let x be the pixel to be coded, and $x_1, x_2, ...x_K$ be the values of pixels in a modeling context that surrounds the pixel *x*. For the modeling/prediction contexts with K=4, 9 and 6, for pass 1, 2, and 3 respectively, to minimize the code length $-\log P(x|x_1, x_2, ..x_K)$ of x, we would like to maximize the conditional probability $P(x|x_1, x_2, ..x_K)$. Suppose that the

pixels have an intensity resolution of Z bits. Then there are 2^{ZK} different contexts. Because of the excessive memory requirement to store the conditional probabilities for all possible contexts, and of prohibitive computational cost to estimate the probabilities, by using quantization possible contexts should be decreased. The simple technique of modeling the unknown pixel x as a linear combination of $x_1, x_2, ...x_K$, $\hat{x} = \sum_{k=1}^{K} a_k x_k$, as for the three passes above provides an efficient way of removing smoothness related redundancy based on unquantized contexts. By this technique, we convert the modeling problem of maximizing $P(x|x_1, x_2, ..x_K)$ to the one of maximizing $P(e|x_1, x_2, ..x_K)$, where $e = x - \hat{x}$. A simple linear predictor may fail to remove all the correlations between adjacent samples if the sample values have a nonlinear relationship in a given context. In order to facilitate context modeling of errors, we quantize the context $x_1, x_2, ...x_K$ that yields the prediction $\hat{x}$ to a binary number $t = t_K t_{K-1} ... t_1$ of bits, where

$$t_k = \begin{cases} 0 & if \quad x_k \geq \hat{x} \\ 1 & if \quad x_k < \hat{x} \end{cases} \qquad (9)$$

Intuitively, t represents high-order spatial structures of the modeling context. Besides spatial texture patterns, the variability of $x_1, x_2, ...x_K$ also shapes $P(e|x_1, x_2, ..x_K)$. Clearly, the variance of the conditional probability $P(e|x_1, x_2, ..x_K)$ strongly correlates to the smoothness of the image around the modeled pixel x. To model this correlation with a small number of parameters and at a small computational cost, we define a so-called error strength discriminator to be:

$$\Delta = \sum_{k=1}^{K} w_k |x_k - \hat{x}| \qquad (10)$$

Now the problem is changed from estimating $P(e|x_1, x_2, ..x_K)$ to estimating $P(e|\Delta)$. To prevent the problem of context dilution in estimating, we quantize Δ to L levels. In practice, $L = 8$ is found to be sufficient. Since Δ is a random variable, it requires only scalar quantization. Globally optimal quantization of Δ via dynamic programming is practical [8]. By combining, via Cartesian product, the quantized error strength discriminator Δ of L levels and the 2^K quantized texture patterns of (9), we finally quantize the 2^{ZK} contexts into $L2^K$ contexts. They are called quantized contexts denoted by $C(d,t)$, $0 \leq d < L$, $0 \leq t < 2^K$.

5.1 Context-based, Adaptive Error Modeling

After quantization of contexts, adaptive error correction value, which is bias R in TSGD model, is calculated. For every context $C(d,t)$, with a count $N(d,t)$ of context occurrence and a cumulative sum $B(d,t)$ of fixed prediction errors incurred so far in the context, the correction value is obtained as below:

$$\bar{e}(d,t) = \frac{B(d,t)}{N(d,t)} \qquad (11)$$

Final prediction of x is $\dot{x} = \hat{x} + \bar{e}(d,t)$, and entropy error is $\in = x - \dot{x}$. In decoder after estimation and calculation of $\dot{x}$, the original value of x is decoded as follow:

$$\begin{cases} \dot{x} = \hat{x} + \bar{e}(d,t) \\ x = \dot{x} + \in \end{cases} \qquad (12)$$

5.2 Optimal Context Quantization

To defining the Δ, Ideally, we would like to determine the coefficients w_k such that: $\Delta = \sum_{k=1}^{K} w_k |x_k - \hat{x}|$ is the least-squares estimator of $|\in|$ (the magnitude of the error being entropy coded). But this cannot be done because the Δ quantizer has to be fixed in order to compute $\in = x - \dot{x}$, where $\dot{x} = \hat{x} + \bar{e}(d,t)$, $d = Q(\Delta)$. The next best thing is to make Δ

the least-squares estimator of $|e|$, where $|e| = |x - \hat{x}|$. The standard linear regression is used to determine the coefficients w_k. Specifically, given a predictor $\hat{x} = \sum_{k=1}^{K} a_k x_k$ corresponding to one of the three passes, training set S of $|e| = |x - \hat{x}|$ and $|x_k - \hat{x}|, (1 \le k \le K)$, is collected. Then w_k, $(1 \le k \le K)$, are chosen by linear regression to minimize

$$\sum_{S} (|e| - \Delta)^2 = \sum_{S} (|x - \hat{x}| - \sum_{k=1}^{K} w_k |x_k - \hat{x}|)^2 \quad (13)$$

over the training set. If time complexity is not of concern, the linear regression can be used to optimize Δ for each input image. But on a single image basis, the compression benefit usually does not justify the optimization cost. Instead, weights w_k should be optimized for classes of images off-line. In real-time coding process, suitable fixed weights w_k are used. Once Δ is optimized over a training set, we can then optimize the Δ quantizer. The quantization criterion is to minimize the conditional entropy of the errors based on $P(\in |Q(\Delta))$. In an off-line design process, we get a set of $(\in, \Delta)$ pairs from training images, and use the standard dynamic programming technique to choose $0 = q_0 < q_1 < q_2 \ldots q_{L-1} < q_L = \infty$ to partition the error terms $\in = x - \dot{x}$ into L ranges:

$$S_d = \{\in | q_d \le \Delta < q_{d+1}\} \quad (14)$$

Such that :

$$-\sum_{\in} P(\in) \log P(\in | q_d \le \Delta < q_{d+1}) \quad (15)$$

is minimized. As in the determination of Δ, the design of optimal quantizer should be done off-line over a training set.

6. Results

Results for implemented schemes, which are one and three pass schemes, JPEG-LS standard (one-pass) and three-pass compression applied to it, are shown in following table. The comparison factor among implemented schemes is compression ratio that is calculated by dividing entropy of the original image by entropy of the compressed image. Images have been selected from Dr.S.Barre's web page [9].

Classifying images into two groups does analyzing the results. One group consists of images with dimensions smaller than 512*512 pixels and another with dimensions equal or larger than 512*512 pixels. As shown in table 1, compression ratios obtained from three-pass compression schemes are higher than one-pass schemes for images in first group. For second group, one-pass schemes have higher ratios.

Because of small data set (pixels) at first group, suitable training of context models and context dilution doesn't take place in one-pass schemes. But three-pass schemes that use 360° type modeling contexts increase ability of isolation and resolution between contexts and obtain higher compression ratios than one-pass schemes. Instead, because of higher image scanning in three-pass schemes, they have larger compression time than one-pass schemes, which is not suitable for real time applications. But for images with small dimensions, it is acceptable.

By increasing image dimensions, enough data set (pixels) for training context models is provided. In this case, compression ratios of one-pass schemes are increased, so as for second image group, this schemes because of higher ratios and smaller compression time are more suitable than three-pass schemes.

Table 1: Comparsion of compression schemes

Images	Pixels	1-pass JPEG-LS	1-pass LIC	3-pass JPEG-LS	3-pass LIC
MRI	208*256	1.78	2.02	**2.86**	2.8
MR-angio	256*256	1.16	1.18	**2.34**	2.32
MR-Knee	256*256	1.76	2.39	**3.16**	2.94
MR-Abdomen	256*256	1.72	2.07	**2.9**	2.80
CR-Chest	440*440	2.11	2.7	**2.85**	2.62
CT-Abdomen	512*512	**2.82**	2.56	2.54	2.34
CT-Ankle	512*512	**2.84**	2.58	2.56	2.26
Colon	512*512	**2.86**	2.82	2.82	2.72
Hip	512*512	**2.9**	2.86	2.88	2.8
CR-Abdomen	1976*1576	**4.71**	4.46	3.93	3.48
Average CR	----------	2.46	2.56	**2.88**	2.70

Introducing the image compression scheme to decoder, needs encoding excessive data to compressed image, which causes to increase entropy. So the optimal case in selecting image compression schemes, is independency to type and dimensions of images. For this selection, average of compression ratios for each scheme is calculated. As shown, three-pass JPEG-LS scheme has largest average among others and is suitable for compression applications, independent from type and dimensions of images.

References

[1] M. J. Weinberger, J. Rissanen, and R. Arps, "Applications of universal context modeling to lossless compression of gray-scale images," IEEE Trans. Image Processing, vol. 5, pp. 575-586, Apr. 1996.

[2] A. Netravali and J. O. Limb, "Picture coding: A review," Proc. IEEE, vol. 68, pp. 366—406, 1980.

[3] G. G. Langdon Jr. and M. Manohar, "Centering of Context-Dependent Components of Prediction Error Distributions", proc. SPIE, vol.2028, PP. 21-27, Feb.1995.

[4] N. Merhav, G. Seroussi, and M. J. Weinberger, "Coding for sources with two-sided geometric distributions and unknown parameters," IEEE Trans. Inform. Theory 1998. Available as Technical Report No. HPL-94-111, Apr. 1998, Hewlett-Packard Laboratories.

[5] N. Merhav, G. Seroussi, and M. J. Weinberger, "Optimal prefix codes for two-sided geometric distributions", Available as Technical Report No. HPL-94-111, Apr. 1998, Hewlett-Packard Laboratories.

[6] D.E.Knuth, "Dynamic Huffman Coding", J. Algorithms, vol.6, pp.163-108, 1985.

[7] Information Technology – Lossless and Near-lossless compression of continuous-tone still images, 1999.ISO/IEC 14495-1, ITU Recommed. T.87.

[8] X. Wu, "Optimal Quantizatiom by Matrix-searching", J. Algorithms, vol. 12, Dec. 1991, pp.663-673.

[9] S. Barre, "Medical Image Samples", available at: http://www.baree.nom.fr/medical/samples.