OPTIMAL PREDICTORS FOR THE DATA COMPRESSION OF DIGITAL ELEVATION MODELS USING THE METHOD OF LAGRANGE MULTIPLIERS.

M. Lewis & D.H. Smith
Department of Mathematics and Computing, The University of Glamorgan, Pontypridd, Mid-Glamorgan, CF37 1DL., U.K.

Tel.U.K. 443 480480 ext. 2268., Fax: U.K. 443 480553.,
Email : MLEWIS@uk.ac.glam

ABSTRACT

Square or rectangular grids are extensively used for digital elevation models (DEM's) because of their simplicity, their implicit topology and their minimal search time for applications. However, their inability to adapt to the variability of the terrain results in data redundancy and excessive storage requirements for large models. One approach to mitigating this is the use of data compression methods. One such method, based on probabilities, is that of Huffman encoding which gives error-free data compression. The key idea is the use of a model that predicts the data values. The method of Lagrange Multipliers for minimisation of the root mean square prediction error has been applied to local geometric predictors and compared with the least-squares fitting of quadratic and bilinear surface patches. The measure of goodness was the average entropy derived from the differences between the actual and predicted elevations. The lower the entropy, the better is the prediction method. An optimal 8-point predictor proved better than the fitting of polynomial surfaces and gave about a 4% to 7% improvement on a simple triangular predictor.

INTRODUCTION.

The data redundancy inherent in regular grid digital elevation models (DEMs) can be removed by the use of data compression techniques. One common approach is to step through the data values in some predefined order and to make a prediction of the current value from the previous values. The difference between the predicted integer value and the actual value is added to a string of prediction errors, which is encoded using a variable length coding technique such as Huffman encoding [2]. Error free recovery of the original data can be obtained by a reversal of the method. Kidner & Smith [3] proposed a simple triangular predictor for use before Huffman encoding. In this paper, we consider a number of alternative prediction methods.

A MEASURE OF COMPRESSION PERFORMANCE

In general, an estimate of the maximum amount of compression achievable in an error-free encoding process can be made by dividing the number of bits needed to represent each terrain height in the original source data by a first-order estimate of the entropy of the prediction error data. Since there is in general a large degree of redundancy in the source data, an accurate prediction process causes a reduction in the entropy value due to the probability density function of the prediction errors being highly peaked at zero and having a relatively small variance.The mathematical definition of entropy is:

$$H = -\sum_{1}^{N} p(i) \log_2 p(i) \qquad (1)$$

where H is the entropy and p(i) is the probability of each data value. So by estimating the entropy, one can determine how efficiently information can be encoded.

A METHOD OF MINIMISATION.

The method of Lagrange Multipliers for determining maxima and minima of a function S(x,y,z) subject to a constraint condition $\emptyset(x,y,z)=0$; consists of the formation of an auxiliary function:-

$$G(x,y,z) \equiv S(x,y,z)+\lambda \Phi(x,y,z) \qquad (2)$$

subject to the conditions that $\partial G/\partial x=0$, $\partial G/\partial y=0$, $\partial G/\partial z=0$ and $\partial G/\partial \lambda=0$, which are necessary conditions for a relative maximum or minimum. The parameter λ, which is independent of x,y and z, is called the Lagrange multiplier.
This method has traditionally been used in geostatistical estimation techniques such as kriging [1].

PREDICTION OF TERRAIN ELEVATION DATA.

Given a square or rectangular grid of points {(i,j): i=0,1,...,N; j=0,1,...,M} we will let Z denote Z(i,j), the point being predicted and take Z_1= Z(i-1,j), Z_2 = Z(i-1,j-1), Z_3= Z(i,j-1). We use the values of Z_1, Z_2, Z_3 in a predictor of the form:

$$\text{pred}(Z)= \text{Nearest Integer} \{ \mu_1 Z_1 + \mu_2 Z_2+\mu_3 Z_3 \}. \qquad (3)$$

The greatest compression will be achieved if the entropy of the set of values { Z - pred(Z) ; i=1,2,...N; j=1,2,...M } is minimised. Although the form of the expression for entropy makes minimisation difficult, we can attempt the minimisation indirectly as follows:
(1) Assume that the mean error $\{Z - (\mu_1 Z_1 + \mu_2 Z_2+\mu_3 Z_3)\}$ is zero;
(2) Subject to this constraint, minimise the squares of the errors

$$S(\mu_1,\mu_2,\mu_3) = \Sigma(Z-\mu_1 Z_1 - \mu_2 Z_2- \mu_3 Z_3)^2 \qquad (4)$$

where the summation is over all terrain height values i=1,2,...,N; j=1,2,...,M.

Then $S(\mu_1,\mu_2,\mu_3)$ becomes:

$$\Sigma_{ij} Z^2+ \Sigma_{ij} \mu_1^2 Z_1^2+\Sigma_{ij} \mu_2^2 Z_2^2+\Sigma_{ij}\mu_3^2 Z_3^2 - 2\Sigma_{ij} Z Z_1\mu_1 - 2\Sigma_{ij} Z Z_2\mu_2 -$$
$$2\Sigma_{ij} Z Z_3\mu_3 + 2\Sigma_{ij} Z_1 Z_2\mu_1\mu_2 + 2\Sigma_{ij} Z_1 Z_3\mu_1\mu_3 + 2\Sigma_{ij} Z_2 Z_3\mu_2\mu_3 .$$

In order that $S(\mu_1,\mu_2,\mu_3)$ be minimised subject to the mean error being zero, we let

$$G(\mu_1,\mu_2,\mu_3) = S(\mu_1,\mu_2,\mu_3)+\lambda(\Sigma_{ij} Z- \mu_1\Sigma_{ij} Z_1 - \mu_2\Sigma_{ij} Z_2- \mu_3\Sigma_{ij} Z_3) \qquad (5)$$

and set the partial derivatives
$\partial G/\partial\mu_1, \partial G/\partial\mu_2, \partial G/\partial\mu_3, \partial G/\partial\lambda$ to zero; i.e.:

$$2\mu_1\Sigma_{ij} Z_1^2 - 2\Sigma_{ij} Z Z_1 + 2\Sigma_{ij} Z_1Z_2\mu_2 + 2\Sigma_{ij} Z_1Z_3\mu_3 - \lambda\Sigma_{ij} Z_1 = 0.$$

$$2\mu_2\Sigma_{ij} Z_2^2 - 2\Sigma_{ij} Z Z_2 + 2\Sigma_{ij} Z_1Z_2\mu_1 + 2\Sigma_{ij} Z_2Z_3\mu_3 - \lambda\Sigma_{ij} Z_2 = 0.$$

$$2\mu_3\Sigma_{ij} Z_3^2 - 2\Sigma_{ij} Z Z_3 + 2\Sigma_{ij} Z_1Z_3\mu_1 + 2\Sigma_{ij} Z_2Z_3\mu_2 - \lambda\Sigma_{ij} Z_3 = 0.$$

$$\Sigma_{ij} Z - \mu_1\Sigma_{ij} Z_1 - \mu_2\Sigma_{ij} Z_2 - \mu_3\Sigma_{ij} Z_3 = 0.$$

Define the coefficients C as:

$C_{11} = \Sigma_{ij}\ Z(i-1,j)^2$

$C_{12} = \Sigma_{ij}\ Z(i-1,j) \times Z(i-1,j-1)$

$C_{13} = \Sigma_{ij}\ Z(i-1,j) \times Z(i,j-1)$

$C_{22} = \Sigma_{ij}\ Z(i-1,j-1)^2$

$C_{23} = \Sigma_{ij}\ Z(i-1,j-1) \times Z(i,j-1)$

$C_{33} = \Sigma_{ij}\ Z(i,j-1)^2$

$C_{01} = \Sigma_{ij}\ Z(i,j) \times Z(i-1,j)$

$C_{02} = \Sigma_{ij}\ Z(i,j) \times Z(i-1,j-1)$

$C_{03} = \Sigma_{ij}\ Z(i,j) \times Z(i,j-1)$

$C_0 = \Sigma_{ij}\ Z(i,j)$

$C_1 = \Sigma_{ij}\ Z(i-1,j)$

$C_2 = \Sigma_{ij}\ Z(i-1,j-1)$

$C_3 = \Sigma_{ij}\ Z(i,j-1)$.

The equations reduce to:-

$$\mu_1C_{11}+\mu_2C_{12}+\mu_3C_{13}+(-\lambda/2)C_1 = C_{01}$$

$$\mu_1C_{12}+\mu_2C_{22}+\mu_3C_{23}+(-\lambda/2)C_2 = C_{02}$$

$$\mu_1C_{13}+\mu_2C_{23}+\mu_3C_{33}+(-\lambda/2)C_3 = C_{03}$$

$$\mu_1C_1+\mu_2C_2+\mu_3C_3+(-\lambda/2).0 = C_0.$$

Once all the coefficients C have been calculated, the problem reduces to solving 4 linear equations in 4 unknowns. The solution vector $[\mu1,\mu2,\mu3,-\lambda/2]$ gives the fitting coefficients. The term involving the Lagrange multiplier (λ) is not required in the prediction process.

The above equations have been set up for predicting a value for a point Z(i,j) based on 3 nearby terrain heights. The same method can be used for predicting a value for the same terrain height from 8 neighbouring heights with solution vector $[\mu_1,...,\mu_8,-\lambda/2]$ giving the fitting coefficients to Z(i-1,j), Z(i-1,j-1), Z(i,j-1), Z(i-2,j), Z(i-2,j-1), Z(i-2,j-2), Z(i-1,j-2) and Z(i,j-2), which are shown as points **1...8** in Fig. 1.

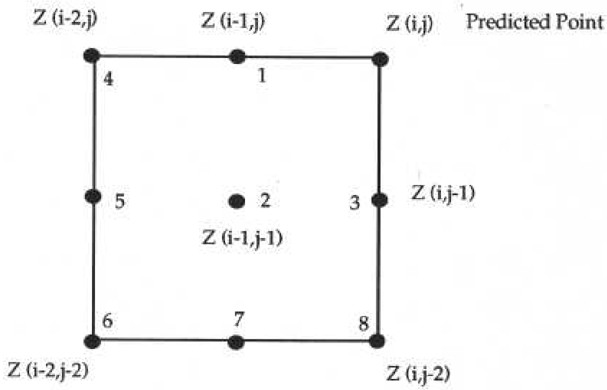


Fig. 1.

APPLICATION TO DIGITAL ELEVATION MODEL DATA.

In this section we will compare the entropy values for our 3-point and 8-point predictors with the triangular predictor of Kidner and Smith [3] which is given by pred(Z)= Z(i-1,j) - Z(i-1,j-1) + Z(i,j-1). Given a square or rectangular grid the first row and column (or first two rows and columns for an 8-point predictor) are excluded and the points are scanned column by column starting from Z(1,1) (or Z(2,2) for an 8-point predictor). For each point the prediction is calculated and the error pred(Z)-Z is recorded. From the frequencies of these errors, the probabilities of the errors and hence the entropy can be calculated.

We will use two British Ordnance Survey 401x401 Digital Elevation Model grids consisting of points sampled at 50 metre intervals accurate to the nearest metre. Source data is held on disk as 2-byte integers (16 bits). ST06 is an area of South Wales containing sea and land areas to the south and west of Cardiff and ST08 covering the Taff and Rhondda Valleys centred near the town of Pontypridd. The terrain profiles consist of both smooth and sharp changes in topology, i.e. deep valleys and rounded hills in ST08 and areas with smoother gradients but containing coastal cliffs in ST06. The original data was rounded to units of 2 metres as the elevation range then allowed all elevations to be represented in 8 bits for convenient comparison.

The entropies, values of S^2 and μ-values are given in Table 1. For the triangular predictor, the 3-point predictor and the 8-point predictor, the entropies are 1.3910, 1.3581, 1.3042 bits per elevation for ST06 and 2.3689, 2.3465 and 2.2577 bits per elevation for ST08. These values should be compared with the 8 bits per elevation of the original data.

Optimum Predictors for a Digital Elevation Model.

Predictor	Data Set	(S^2)	Coefficients (μ -values)	Entropy (bits/elevation)
Triangular [3]	ST06	$9.7596x10^4$	1,-1, 1	1.3910
3pt	ST06	$8.0336x10^4$	0.8267 - 0.6967 0.8703	1.3581
8pt	ST06	$6.9809x10^4$	0.8237 - 0.4168 0.9829 - 0.1169 - 0.0866 0.0824 - 0.0773 - 0.1912	1.3042
Triangular [3]	ST08	$3.10961x10^5$	1,-1, 1	2.3689
3pt	ST08	$2.87193x10^5$	0.9380 -0.8610 0.9230	2.3465
8pt	ST08	$2.46358x10^5$	1.0370 -0.5736 0.9236 -0.2214 -0.0313 0.0585 -0.0263 -0.1666	2.2577

Table 1.

Local Coordinates (x,y).

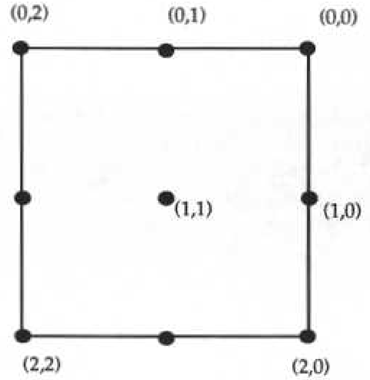


Fig. 2.

For each surface patch used to fit to a set of six points, we can work within a local coordinate system as in Fig. 2. This has the advantage that the matrix only has to be inverted once. The point to be predicted will simply have the coordinates (0,0) and so (0,0) is substituted into F(x,y). Then the difference between the value of the predicted height using this method and the actual height value is calculated and rounded to the nearest integer. This procedure is repeated over the whole terrain data matrix and stored as a string of difference values.

As for the Lagrange multiplier method, the average entropy of the resulting string of corrections is calculated. The results are presented in Table 2.

PREDICTION BY FITTING QUADRATICS THROUGH SETS OF TERRAIN HEIGHTS.

The least squares fit of a quadratic function defined by $a+bx+cx^2$ through three sets of three points in a west-east, south-north and a south-west to north-east direction was done to predict the point Z(i,j). (These are illustrated by ray 1, ray 2 and ray 3 in Fig. 3). This method would enable the capture of surface convexity or concavity. The actual predicted value was taken either as the median value of the three quadratics or as the average predicted value of the three quadratics. The resulting prediction errors in both cases were used to calculate the entropy as in the case of a least-squares surface fit.

A similar procedure was followed of retaining the first row and column of terrain height values together with the second and third row and column as prediction errors using the triangular predictor [3]. The errors arising from both the mean and the median prediction for the three quadratics are stored separately. In each case the entropy for these errors is calculated.

In this case we can either substitute in points to calculate the coefficients directly or minimise the least squares error as before. We do this for each ray separately.

$$L_n=\Sigma_{i=1..3}\ (Z_i - a - bx - cx^2)^2 \text{ where } Z_i = \text{height } (x_i). \tag{9}$$

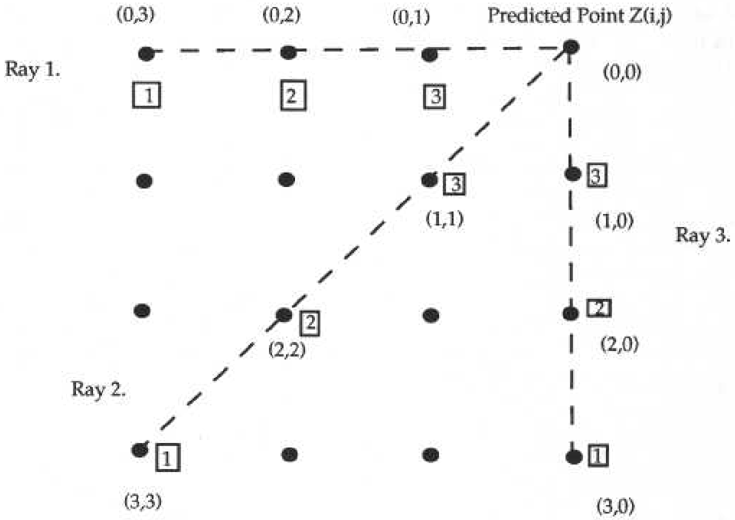


Fig. 3.

In matrix form this becomes:

$$\begin{bmatrix} 3 & \sum_i x_i & \sum_i x_i^2 \\ \sum_i x_i & \sum_i x_i^2 & \sum_i x_i^3 \\ \sum_i x_i^2 & \sum_i x_i^3 & \sum_i x_i^4 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} \sum_i z_i \\ \sum_i x_i z_i \\ \sum_i x_i^2 z_i \end{bmatrix}$$

The solution vectors [a,b,c] give the fitting coefficients for each quadratic equation used to determine the predicted height. In this case, we have used three rays with each ray consisting of three contributing terrain height values.

The results for our representative test data were as follows: In ST06, the entropy taking the mean value of the ray predictions was 2.1692 bits and taking the median value 2.5885 bits. In ST08, the respective entropy values were 3.9579 and 4.2243 bits per elevation.

PREDICTION BY BILINEAR SURFACE FITTING.

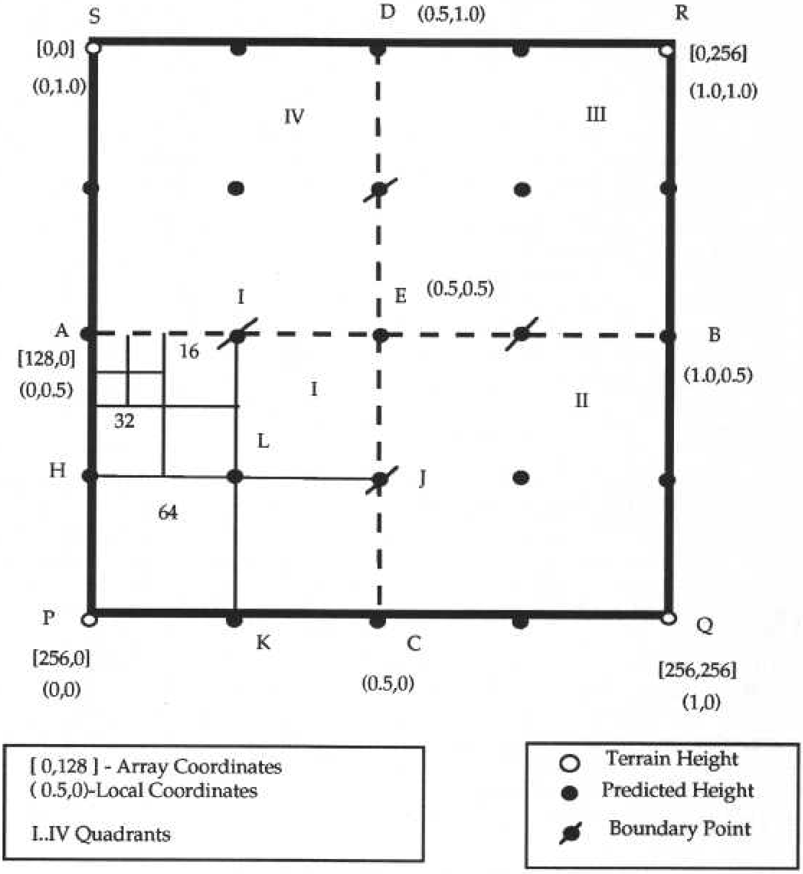
In this method, a bilinear surface defined by the equation $a+bx+cy+dxy$ is used to fit through four points P,Q,R,S and used to predict the five points A,B,C,D,E and the differences between the actual and predicted values stored in a matrix (see Fig.4). The procedure is repeated on the square P,C,E,A using the true height values to predict the points H,I,J,K,L in the diagram. When a predicted point is on the boundary of two separate surface patches, the mean value of the combined prediction error or correction value is taken. The initial array of terrain heights is a 256x256 array (257x257 height values) and is recursively sub-divided into quadrants of 128x128, 16 of 64x64 and 64 of 32x32 and so on. During this recursive subdivision, the above prediction process is applied to each quadrant where on each subdivision the four mid-points and centre point are predicted by fitting an increasingly finely grained bilinear surface. Once again, each surface patch is defined by a local coordinate system i.e. if P,Q,R,S have coordinates (0,0), (1,0), (1,1), (0,1) respectively then each predicted point in each quadrant at each sub-division will have the local x,y coordinates as illustrated by A,B,C,D,E of (0,0.5), (1,0.5), (0.5,1), (0.5,1) and (0.5,0.5).The four coefficients of each bilinear surface used to define each surface patch are calculated by a system of 4 linear equations formulated by the least squares minimisation procedure described above or can be done by direct substitution where each predicted point is:

$$P+(Q-P)x+(S-P)y+(P-Q-S+R)xy. \tag{10}$$

The algorithm terminates when the recursion has produced 4^7 smaller patches of 3x3 elevation points. Since our test-data, ST06 and ST08, are fixed sized arrays of 401x401 points, the recursive sub-division algorithm is run on 4 overlapping tiles of 257x257 points with origins at coordinates (0,0), (0,144), (144,0), (144,144) for the maximum terrrain data matrix coverage. In ST06 the entropy values were 2.0919, 1.8117, 2.9287, 2.7503 bits for the overlapping segments. The corresponding values for ST08 were 3.6112, 3.8811, 3.5570 and 3.6358 bits respectively.

DISCUSSION.

The results above show quite clearly that a small improvement can be made to the simple triangular predictor method for both the three-point and eight-point predictors by the minimisation method of Lagrange multipliers. Typical savings in the average entropy values varied between 4% and 7%. It was interesting to note that in both the 3-point and 8-point prediction methods, the significant coefficient (μ) values affecting the prediction of the terrain height always corresponded to points 1,2 and 3 with a smaller contribution from points 4,6 and 8 for the 8-point predictor. This seemed to support the rationale behind the triangular predictor [3]. Comparisons with a least squares fit of six points to predict the same point and different least squares fits of both a quadratic interpolation of neighbouring points and a bilinear surface interpolation have confirmed this. These surface fitting prediction methods failed to achieve a lower entropy value with the least squares fit being the best of the other three methods. Table 2 shows comparative entropy values for our data sets ST06 and ST08 for the different prediction methods described. For many data sets compression ratios above 4 or 5 are easily achievable using a error-free Huffman encoding algorithm with minor modification to the code given in [3] to include the calculation of the coefficients (μ-values) for either the 3-point or 8-point predictor.



Points A,B,C,D,E are predicted from P,Q,R,S. by fitting a bilinear surface a+bx+cy+dxy. The points H,I,J,K,L are predicted from surface fit to A,E,C,P. The process is repeated in quadrants II, III, IV to determine mid-points and centres of quadrants. Points I and J are predicted from 2 separate bilinear surface fits i.e point I from quadrants I and IV and point J from quadrants I and II. In such cases, the average correction (prediction error) is taken from the separate prediction estimates.

The smallest quadrant is a 3 x3. The total no. of squares is 16,384 (=4^7).

Figure 4.

COMPARISON BETWEEN PREDICTION METHODS		
Prediction Method	Entropy (bits/elevation)	
	ST06	ST08
Triangular Predictor [3]	1.3910	2.3689
3-point	1.3581	2.3465
8-point	1.3042	2.2577
Least Squares Surface Fit	2.0521	3.4159
Quadratic Fit (mean result)	2.1692	3.9579
Quadratic Fit (median result)	2.5885	4.2243
Bilinear Surface Fit (256x256)		
Origin at: (0,0)	2.0919	3.6112
(0,144)	1.8117	3.8811
(144, 0)	2.9287	3.5570
(144, 144)	2.7503	3.6358

Table 2.

CONCLUSIONS.

The 3-point and 8-point predictors are clearly superior to the other methods reported. With all predictors, the Huffman code gives an average code length slightly greater than the entropy. It is also necessary to store a code table, a look up table for efficient decoding, the coefficients for the optimal predictors and the first row and column(or first two rows and columns for the 8-point predictor) [3]. However for large models, the additional storage requirements are very small.

An alternative approach to terrain compression of a regular grid data is a transform technique - the two-dimensional discrete cosine transform (2D-DCT). Transform coding allows greater compression but is computationally intensive and gives some error on reconstruction of the data. Transform coding can be combined with Huffman encoding or Run Length Encoding to allow further compression. As an example, terrain data has been compressed by a factor of 18.75:1 with this method by adapting published algorithms [4]. In this case, the reconstructed data is about 75% error-free.

ACKNOWLEDGEMENTS.

This work has been undertaken with sponsorship by the Science and Engineering Research Council (SERC) in collaboration with the Defence Research Agency, Malvern, U.K.

REFERENCES.

[1] Dowd P.A. "A Review of Geostatistical Techniques for Contouring" in Fundamental Algorithms for Computer Graphics. Ed E.A. Earnshaw . NATO ASI Series Vol. F17, 1985, Springer-Verlag.

[2] Huffman D.A. "A Method for the Construction Of Minimum Redundancy Codes". Proc. IRE, Vol. 40, 1952. pp. 1098-1101.

[3] Kidner D.B. & Smith D.H. "Compression of Digital Elevation Models by Huffman Encoding"; Computers & Geosciences, Vol. 18, No. 8, pp. 1013-1034, 1992.

[4] Nelson M.R. The Data Compression Book. Prentice Hall. 1991.