A Comparison of Two Image Compression Techniques for Softcopy Photogrammetry

Kurt Novak and Fayez S. Shahin

Abstract

This paper discusses the basics of digital image compression and explains the conceptual differences of various approaches. Two techniques, the standard JPEG algorithm and the newly developed Hierarchical Predictive Coding (HPC) approach, are described in detail. They are compared with respect to geometric quality, loss of information, compression ratio, and compression speed. JPEG yields better compression ratios; however, it is significantly slower than HPC.

Introduction

The advancements in softcopy photogrammetry and the development of digital photogrammetric workstations have lead to a tremendous demand for digital aerial and satellite imagery. These images can be obtained directly by using solid-state cameras and satellite sensors, or indirectly by scanning existing photographs. Despite all the advantages of digital images over film, the problem of image storage needs more attention. Single satellite (SPOT) scenes contain 6,000 by 6,000 pixels or 36 MBytes, and aerial images are often scanned with a resolution of 20 micrometres, which results in image sizes of 10,000 by 10,000 pixels or 100 MBytes of data. As digital images generally contain a significant amount of redundant information, these large storage requirements can be reduced by proper data compression.

Image data compression is concerned with minimizing the number of bits required to represent an image. Early applications of data compression can be found in the fields of telecommunication, data storage, and printing (Wallace, 1991). High-resolution television, satellite remote sensing, and facsimile transmission are among the major users of digital image compression. Storage of large quantities of digital image data is of major concern for archiving aerial photographs and satellite images, as well as medical and document scans. Other applications of compressed images include time critical data processing, in which case the speed of image analysis algorithms is improved by applying them to compressed data.

For the past few years the Joint Photographic Expert Group (JPEG) has been working toward establishing an international digital image compression standard for still images, both gray scale and color (Rabbani *et al.*, 1990; Wallace, 1991). Photogrammetric research in image compression has focused on the suitability of the JPEG algorithm for softcopy image analysis and remote sensing. Special attention was paid to geometric and radiometric degradations of reconstructed images after compression (Sarjakoski *et al.*, 1992; Nunes *et al.*, 1992).

Department of Geodetic Science and Surveying, The Ohio State University, 1958 Neil Avenue, Columbus, OH 43210.

K. Novak is presently with TransMap Corp., 1275 Kinnear Road, Columbus, OH 43212.

F.S. Shahin's present address is P.O. Box 86, Nablus, West Bank, Israel.

Image compression algorithms can be classified in lossless and lossy techniques. Lossless compression reduces the number of bits required to represent an image such that the reconstructed image is numerically identical to the original one on a pixel-by-pixel basis. On the other hand, lossy compression schemes allow degradations in the reconstructed image in exchange for a reduced bit rate. These degradations may or may not be visually apparent, and greater compression can be achieved by allowing more degradation.

The next part of this article describes theoretical aspects of image compression and some standard definitions, such as entropy and redundancy, the compression ratio, and the Huffman code. The third part of the paper deals with the classification of compression techniques. Then we discuss the JPEG algorithm and the Hierarchical Predictive Coding (HPC) approach. Finally, experiments are presented to show the advantages and disadvantages of both techniques with respect to geometric quality, loss of information, compression ratio, and compression speed.

Image Compression — Basic Concepts

This section explains basic terms and techniques used in digital image compression. It deals with the definition of entropy and redundancy, the compression ratio, and the Huffman coding technique.

Entropy and Redundancy

In most real-world images, the gray values of adjacent pixels are highly correlated. This means that a great deal of information about the gray value of a pixel could be obtained by inspecting its neighbors. Therefore, a large amount of redundant information is available in these images.

A measure of the information content of an image is the *entropy* (Lynch, 1985). It expresses the minimum number of bits necessary for the representation of an image without any loss of information. The entropy is a global measure of the correlation between gray values of neighboring pixels. For an n-bit image with M gray levels ($M = 2^n$), the entropy H can be computed as

$$H = -\sum_{i=0}^{M} p(g_i) \log_2 p(g_i) \tag{1}$$

where $p(g_i)$ is the percentage of occurrence of each gray value, which can be obtained from the image histogram. Most data sources, including digital images, have non-uniform gray value distributions. If these distributions were uniform, the entropy would have a maximum at

Photogrammetric Engineering & Remote Sensing, Vol. 62, No. 6, June 1996, pp. 695–701.

0099-1112/96/6206–695$3.00/0

$$H_{max} = \log_2 M \tag{2}$$

where M is the number of gray levels. Because real images seldom have uniform probability distributions, their entropy is always less than H_{max}. For a given digital image, the redundancy R is defined as follows (Lynch, 1985):

$$R = \log_2 M - H \tag{3}$$

The entropies and redundancies of the images used for our investigations are shown in Table 1.

The Compression Ratio

The general compression ratio ΔC is defined as the ratio of the number of bytes of the original image before compression and the number of bytes of the compressed image. The maximum compression ratio ΔC_{max} that can be achieved without any loss of information is defined as follows (Storer, 1988):

$$\Delta C_{max} = \frac{\log_2 M}{H} \tag{4}$$

For a digital image, we reach the maximum compression ratio when the image coding results in bits-per-pixel rates equal to the entropy (Storer, 1988). Thus, to achieve the maximum compression ratio, we first need to eliminate or reduce the correlations between pixels, and then to code these pixels as efficiently as possible.

Optimum methods encode independent (uncorrelated) pixels in a way that makes the average bits per pixel equal or close to the image entropy. This kind of coding will consequently achieve the maximum compression ratio and preserve all information. The first implementation of such a coding was described by Shannon *et al.* (1949), and an improved method was developed by Huffman (1952).

The Huffman Code

Huffman developed a procedure for encoding a statistically independent source in such a way as to produce the most efficient code. An efficient code is one that achieves bits-per-pixel rates close to the image entropy. A source can be defined as an information-generating process that emits a sequence of symbols chosen from a finite alphabet. For example, a computer performs its computations on binary data. These data may be considered as a sequence of symbols generated by a source with a binary alphabet composed of 0 and 1. In the case of digital images, an n-bit image can be viewed as being generated by a source (the CCD sensor) with an alphabet of 2^n symbols, which represent the possible gray values.

The Huffman coding method is based on the following principles:

- If the two least probable symbols of a source with an alphabet of size (a) are combined, a new source with ($a-1$) symbols is generated.
- If the code words for the reduced source are known, it can be shown that the code words of the original source are identical to those of the reduced source for all symbols that have not been combined. Furthermore, the code words of the two least probable symbols of the original source are formed by appending 0 or 1 to the right of the code word of the combined symbol in the reduced source.
- The Huffman code for a source with only two symbols consists of the code words 0 and 1.

To construct the Huffman code of a given source, the original source is reduced by repeatedly combining the two least probable symbols until a source with only two symbols is obtained. The Huffman code for this reduced source is known (0 and 1). Then the code words for the previous stage are found by appending a 0 or 1 to the code word corresponding to the two least probable symbols. This process is continued until the Huffman code of the original source is found.

Table 1. Entropies and Redundancies of the Three Images which Were Used for Evaluating the Compression Schemes [units: bits/pixel]

[bits/pixels]	Aerial Image	Road Image	Target Image
Entropy	5.79	3.36	2.75
Redundancy	2.21	4.64	5.25

The Huffman method will result in a variable length code. The length of a code word is inversly proportional to the probability of the symbol. This means that long code words are used for low probability symbols, and short code words for high probability symbols.

Classification of Image Compression Techniques

The following section takes a closer look at lossy and lossless compression techniques, as well as the algorithms and transformations involved. It is based on publications by Jain (1988) and Storer (1988).

Lossy Compression Techniques

In lossy compression techniques, degradations of gray values are allowed in the reconstructed image in exchange for a reduced bit rate as compared to lossless schemes. Typically, lossy compression algorithms consist of three steps:

(1) *Image Decomposition or Transformation.* In this step the image is transformed to a new domain, such as the frequency domain, in order to reduce the dynamic range of the signal (gray values) and to eliminate the correlations between the original gray values.
(2) *Quantization.* In this step the transformed pixel values are mapped onto a smaller, finite number of output levels, in order to reduce the number of possible output symbols. The type and degree of quantization impacts the quality of a lossy scheme.
(3) *Symbol Coding.* The resulting output sequence from the quantization process must be efficiently coded, by using methods such as Huffman coding.

The most common lossy compression techniques are based on predictive coding and on transform coding.

Lossy Predictive Coding

The predictive coding scheme employs the correlation between neighboring pixel values to determine the gray values of a digital image. Differential Pulse Code Modulation (DPCM) is the most common approach to predictive coding. Generally, DPCM works as follows: The predicted image is computed on a pixel-by-pixel basis by investigating the correlations between the gray values of a pixel and all its neighbors. The predicted image is subtracted from the original image to form a differential image in which pixel values are less correlated than in the original. The differential image is quantized and encoded (Rabbani *et al.*, 1990).

In a lossy DPCM scheme, m pixels within a neighborhood of the current pixel are used to make a linear prediction of the pixel's gray value. The linear estimate is denoted by $\hat{x}_m$, and x_i are the original gray values: i.e.,

$$\hat{x}_m = \sum_{i=1}^{m} \alpha_i \, x_i \cdot \tag{5}$$

The α_i are the predictor coefficients. The prediction is usually rounded to the nearest integer. It is also necessary to clip the prediction to the range $[0, 2^n-1]$ for an n-bit image. The set of predictor coefficients may be fixed for all images (global prediction), or may vary from image to image (local prediction), or may even vary within an image (adaptive prediction). To compute the optimum predictor coefficients, the squared prediction error is usually minimized.

The difference image (e_m) is constructed by subtracting predicted gray values from the original ones: i.e.,

$$e_m = x_m - \hat{x}_m \cdot \tag{6}$$

Once the difference image is computed, it has to be quantized. A large portion of the compression achieved by a lossy DPCM is due to quantization. A quantizer is simply a staircase function that maps a large number of input values into a smaller number of output levels. Let e be the difference image and $p_e(e)$ be its histogram. A quantizer maps variable e into a discrete variable e_q that belongs to a finite set $\{r_i, i = 0, ..., N-1\}$. The range of values of e that map to a particular e_q are defined by a set of points $\{d_i, i = 1, ..., N\}$. The quantization rule states that if e lies in the interval $(d_i, d_{i+1}]$, it is mapped (quantized) to r_i, which also lies in the same interval.

A major limitation of the DPCM scheme is that the predictor and the quantizer are both fixed throughout the image. DPCM can be made adaptive in terms of the predictor or the quantizer or both. Adaptive prediction usually reduces the prediction error prior to quantization, because the prediction is dynamically modified according to the existing configuration of gray values. For example, by using adaptive predictors, the coefficients for a uniform area of gray values are different from those of edges. Therefore, adaptive prediction results in less quantization errors and a better quality of the reconstructed image. Generally, non-adaptive predictors perform poorly at gray-level edges.

Transform Coding

A general transform coding scheme involves subdividing a digital image into smaller blocks and performing a unitary transform on each block. A unitary transform can be defined as a reversible linear transformation whose kernel describes a set of discrete, orthonormal base functions. The goal of the transform is to eliminate the correlations between the original pixels of the block; this decorrelation generally results in the block energy being redistributed among a small set of transformation coefficients. An optimal transformation completely decorrelates the data by packing the most amount of energy in the fewest number of coefficients. Furthermore, an optimal transform should have a minimum number of operations. The most common transforms used in image compression are the Principal Component Transformation (PCT), the Discrete Cosine Transform (DCT), and the Walsh-Hadamard Transform (WHT) (Jain, 1988).

The transform coding scheme employs a strategy for the selection, quantization, and encoding of the transformed coefficients of each block. The two strategies most commonly employed are

- *Zonal Sample Selection.* Zonal sampling consists of retaining only those transformation coefficients that are located in a pre-specified zone in the transformed block; all other coefficients are set to zero. Generally, the lower frequency coefficients are retained while the higher frequency coefficients are discarded. Each retained coefficient is quantized and encoded. The major disadvantage of this sampling technique is that some of the coefficients outside of the coefficient retention zone may contain significant information, and their omission can result in significant reconstruction errors.
- *Threshold Sample Selection.* A threshold level is selected, and only the coefficients whose values are above the threshold are quantized and encoded; all others are discarded.

Lossless Compression Techniques

Some image compression applications require the reconstructed image to be numerically identical to the original image on a pixel-by-pixel basis. An example is medical imaging, where lossy compression schemes may compromise diagnostic accuracy. Lossless compression schemes are also important in remote sensing, where the spectral characteristics of images have to be preserved. As one might expect, the price to be paid for an error-free image is a much lower compression ratio as compared to lossy schemes (Rabbani *et al.*, 1990).

The primary difference between lossy and lossless schemes is the inclusion of quantization in lossy techniques. By quantization, the number of possible output symbols is reduced. The reduction of the number of output symbols at the quantization step leads to degradations in the reconstructed image in exchange for a higher compression ratio. For an 8-bit image, the maximum compression ratio that can be obtained using lossless schemes is 8 divided by the entropy of that image (see Equation 4).

The most common lossless scheme is Lossless Predictive Coding. It can be implemented by using the same steps as explained in the lossy scheme (Rabbani *et al.*, 1990) with the exclusion of the quantization step.

The JPEG Algorithm

The Joint Photographic Expert Group (JPEG) has proposed and developed an algorithm to serve as an international standard for the compression of continuous tone (gray-scale or color) images. The overall algorithm comprises three main components: the baseline system, the extended system, and the independent function (Wallace, 1991).

The *baseline system* algorithm compresses 8 bits-per-pixel images and operates only in sequential mode. In this sequential mode, the image is processed from top to bottom in a single pass by compressing the first row of data, followed by the second row, and continuing until the end of the image is reached. Accordingly, the baseline system is a simple and efficient algorithm that is adequate for most image coding applications. The *extended system* adds capabilities that allow the baseline system to satisfy a broader range of applications. One of its applications supports 12 bits-per-pixel input in addition to 8 bits-per-pixel. Finally, the *independent function* is included for applications requiring lossless compression.

The JPEG baseline system starts with dividing the original image into 8- by 8-pixel blocks. Each block is independently transformed using the Discrete Cosine Transformation (DCT). The forward DCT for a block of 8 by 8 pixels is defined as follows:

$$F(u, v) = \frac{1}{4} C(u)C(v) \sum_{j=0}^{7} \sum_{k=0}^{7} f(j, k) \cos\left[\frac{(2j+1)\, u\pi}{16}\right] \cos\left[\frac{(2k+1)\, v\pi}{16}\right] \tag{7}$$

where

j, k are spatial coordinates in the block domain;
$f(j,k)$ are gray values of the pixels of the block;
u, v are coordinates in the frequency domain; and
$C(u)$, $C(v)$ is $1/\sqrt{2}$ for $u, v = 0$, and 1 otherwise.

The DCT has become by far the most widely used transformation for image compression due to its simplicity and fast implementation. Unlike other algorithms, it is image independent. Figure 1 shows an 8- by 8-pixel block of gray values. This block was then transformed using Equation 7; the 64 forward DCT coefficients are displayed in Figure 2.

In the next step, all transformation coefficients are normalized (weighted) by applying a user defined normalization

100	100	100	100	150	100	110	130
100	100	101	100	105	100	100	100
101	101	103	104	107	108	108	108
97	97	100	100	105	106	108	108
100	100	110	110	110	100	110	99
110	110	110	113	114	112	112	110
100	100	100	100	120	120	120	110
98	99	100	120	120	120	120	90

Figure 1. The original image block of 8 by 8 gray values ($f(j, k)$).

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Figure 3. The JPEG-75 quantization table ($Q(u, v)$).

array that is fixed for all blocks. Each component of the normalization array $Q(u,v)$ is an 8-bit integer that determines the quantization step size. Larger values correspond to larger quantization steps. Generally, the components of this array are computed such that small values (small quantization steps) are associated with low frequencies, and large values (large quantization steps) are associated with high frequencies. The level of compression of an image can be modified by changing this array, e.g., by scaling it by a constant. Thus, different normalization arrays will yield different compression ratios.

The normalized coefficients are then uniformly quantized by rounding to the nearest integer. The resulting normalized and quantized coefficient $FNQ(u,v)$ is given by

$$FNQ(u,v) = \text{nearest integer} \frac{F(u,v)}{Q(u,v)} \tag{8}$$

where

$F(u,v)$ is the coefficient after DCT at pixel (u, v), and $Q(u,v)$ is the normalization factor from the quantization table.

The 75 percent quantization table, which was defined by JPEG, is shown in Figure 3.

The upper-left coefficient in the DCT block is referred to as the DC coefficient, while the remaining coefficients are called the AC coefficients. The quantization of the AC coefficients produces many zeros, especially at higher frequencies. To take advantage of these zeros, the 2D array of normalized and quantized DCT coefficients is formatted into a 1D vector using the zigzag ordering scheme shown in Figure 4. This zigzag ordering rearranges the coefficients in approximately decreasing order. Many coefficients towards the end of the array are zero.

Next, the *DC prediction* is performed, which is a prerequisite for the coding of the DC coefficient. It is done by subtracting the DC term of the previous block from the DC term of the current block. DC prediction produces a differential DC coefficient, which is typically small due to the high correlation of neighboring DC coefficients. Each differential DC coefficient is encoded by a variable length code (VLC), such as the Huffman code. It represents the number of significant bits in the DC term followed by a variable length integer (VLI) representing the value itself.

In a similar fashion, AC coefficients are coded with alternating VLC and VLI codes. The VLC table, however, is a two-dimensional table which is indexed by a composite 8-bit value. The lower 4 bits of the 8-bit value, the column index, is the number of significant bits of a non-zero AC coefficient. The higher order 4 bits, the row index, is the number of zero coefficients which precede the non-zero AC coefficient.

The Hierarchical Predictive Coding Scheme

Hierarchical Predictive Coding (HPC), as described in this section, was developed by the authors based on a compression scheme for digital video sequences (Florida Atlantic University, 1992). The algorithm starts by creating a scale level representation of the original image (image pyramid). It is derived by smoothing the gray values of the original image to a predefined, lower resolution. For instance, a square image of 1024 by 1024 pixels is decomposed into the following five scale levels: 1024 by 1024, 512 by 512, 256 by 256, 128 by 128, and 64 by 64. The image pyramid is generated by convolving higher resolution images with the Gaussian operator and reducing the pixel size by half.

853.0	-26.8	-19.4	11.8	6.1	-3.8	-9.6	15.5
-10.4	-1.0	10.6	-11.1	15.8	-17.3	1.9	11.2
11.5	-11.9	-10.5	11.5	9.4	-10.0	-7.0	7.9
16.9	-1.3	3.2	-5.1	11.9	-12.4	-2.0	12.4
0.6	-5.8	-10.5	-0.7	7.5	-0.9	-6.5	7.7
9.6	-19.1	7.7	4.3	5.4	-12.9	1.5	6.8
14.8	1.2	-2.8	-1.5	5.6	-1.3	-2.0	-1.1
3.6	-4.2	-2.9	-0.1	1.0	-0.7	-1.8	6.1

Figure 2. Forward DCT coefficients of the block shown in Figure 1 ($F(u, v)$).

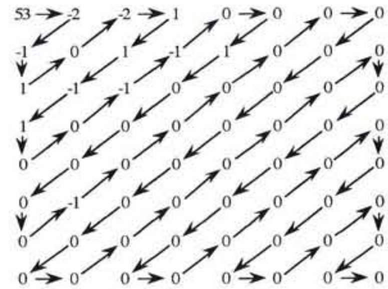

Figure 4. The quantized and normalized DCT coefficients $FNQ(u, v)$ are ordered according to the shown scheme.

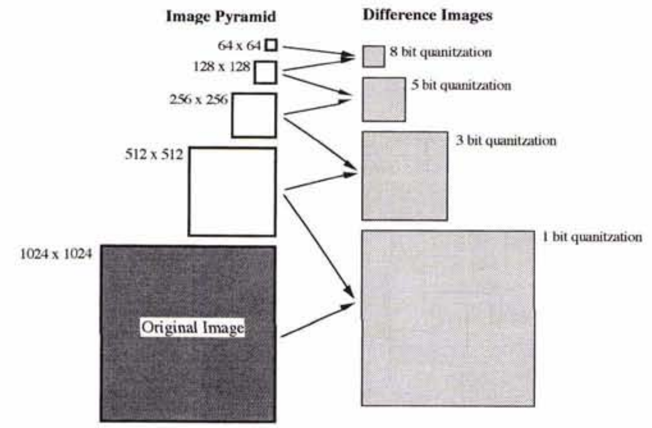

Figure 5. Graphic representation of the HPC-8.5.3.1 scheme.

Difference images are computed between all levels. These differences represent the high frequency components of the previous level of the image pyramid. The original image can be reconstructed from its representation at the coarsest level (e.g., 64 by 64 pixels), and all the difference images. The pixel values of the difference images are usually small; therefore, they can be coded and stored by a lower number of bits. This would result in a lossless compression of the original image. However, in order to achieve higher compression ratios, the number of bits of the difference images are reduced by quantization, which leads to image degradation and lossy compression.

The HPC algorithm consists of the following steps (see Figure 5):

- The image pyramid is created by smoothing the original image sequentially with a Gaussian operator of size 3 by 3. This results in a scale level representation of the original image.
- The difference images at all levels are computed. For instance, an original image of 1024 by 1024 pixels is replaced by one 64- by 64-pixel image and four difference images.
- The values of the highest level image (64 by 64) and the first difference image (128 by 128) are stored as 8-bit integers.
- The values of the 256/128 difference image are stored and coded as 5-bit integers.
- The values of the 512/256 difference image are stored and coded as 3-bit integers.
- The values of the 1024/512 difference image are stored and coded as 1-bit integers.

Using this representation (HPC-8.5.3.1), a compression ratio of 3.6 can be achieved. For higher compression ratios, the difference images are represented by fewer bits. However, the degradation of the reconstructed image increases. In our experiments, we reached a compression ratio of 4.5 by quantizing the difference images in the following way: 128: 8 bit; 256: 3 bit; 512: 3 bit; and 1024: 1 bit. This will be referred to as HPC-8.3.3.1 in the next section.

Experiments and Results

Both JPEG (baseline system) and HPC image compression and decompression software were implemented on a Sun SPARCstation 2 to investigate the advantages and disadvantages of the two techniques. The quality of the reconstructed images was evaluated by the following measures: visual loss of information, compression ratio, compression speed, and geometrical distortion of the image. The algorithms were tried out on images containing various levels of entropy, and should, therefore, show the influence of the image contents on the compression results. The following three images were used in our tests:

- A scanned, 8-bit aerial image of Chaco Canyon, New Mexico (Figure 6). The size of the image is 1024 rows by 1024 columns, and the number of bytes to store the original image is 1,048,576.
- An 8-bit road image captured by a digital camera mounted on a van (Figure 7). The size of the image is 1024 rows by 1280 columns, and the number of bytes to store the original image is 1,310,738.
- An 8-bit target image captured during the test-field calibration of a digital camera (Figure 8). The size of the image is

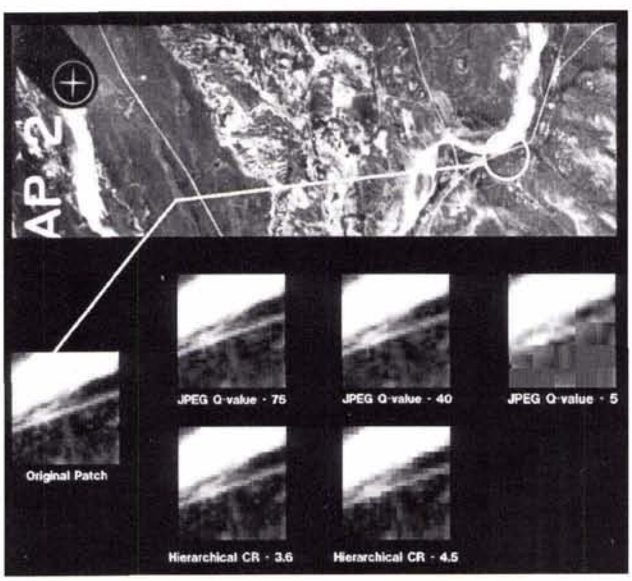

Figure 6. Compression results obtained by different algorithms and quantization levels of a scanned aerial photograph of Chaco Canyon, New Mexico.

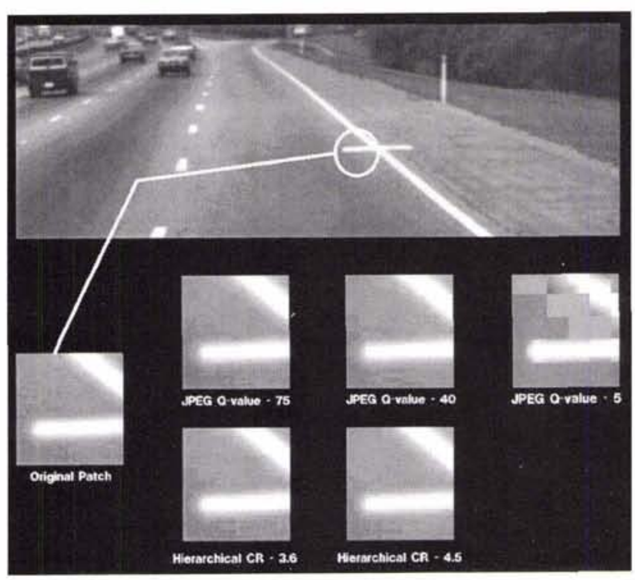

Figure 7. Compression results obtained by different algorithms and quantization levels of a high resolution digital road image collected from a driving van.

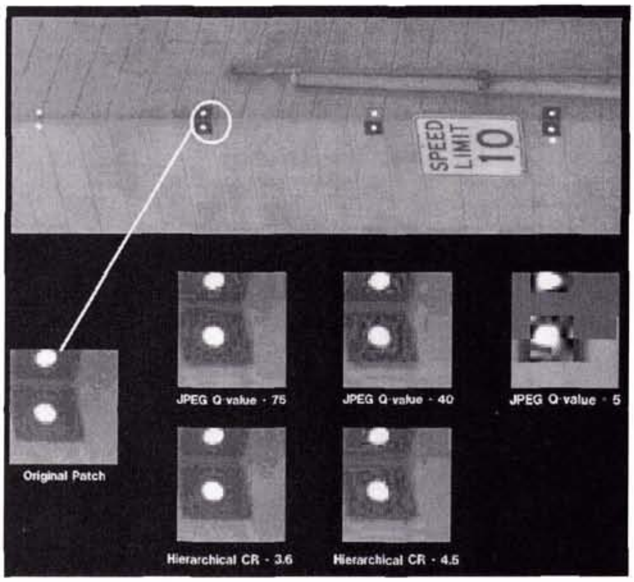

Figure 8. Compression results obtained by different algorithms and quantization levels of a digital calibration image containing control point targets.

1024 rows by 1280 columns, and the number of bytes to store the original image is 1,310,738.

Table 1 shows the entropy and the redundancy of the three test images.

The aerial image has the largest entropy, and the target image has the smallest entropy, because there are only a very few features in the latter image. Thus, the target image contains the most redundant information, while the aerial image contains the lowest amount of redundant information.

Compression Ratios

The three images were compressed using the JPEG algorithm at different quantization levels and the HPC technique using two different quantization thresholds. Table 2 shows the resulting compression ratios.

These results clearly show that a large amount of compression can be achieved when using lossy compression schemes. The compression ratio obtained by using lossless schemes is limited by the entropy of the image. For example, in the lossless compression case of the aerial image, the compression ratio is 1.3, which is approximately 8 divided by the entropy of the image (5.79 bits/pixel).

The reader should also notice the difference in compression ratios for different images at the same quantization value. For example, at JPEG quantization level 75, the aerial image was compressed only 5.9 times, while the target image was compressed approximately 43 times. This difference is due to the amount of redundancy available in the target image and the large information content in the aerial image. As the HPC scheme is not based on transform coding, the compression ratio is independent of the image contents (entropy).

Quality of the Reconstructed Image

After being compressed by a lossless scheme, the reconstructed image is identical to the original image on a pixel-by-pixel basis. Lossy schemes cause degradations of the reconstructed image in exchange for larger compression ratios. These degradations of the reconstructed image lead to radiometric and geometric distortions. We studied the radiometric as well as the geometric distortions caused by lossy schemes.

For evaluating *radiometric distortions*, the gray values of the reconstructed images were compared to the original image on a pixel-by-pixel basis. Table 3 presents the radiometric distortions (average and mean gray-value differences) of the images at different quantization levels.

These results show that the radiometric distortions depend to a limited degree on the level of quantization and, therefore, increase with the compression ratio. Thus, the radiometric quality of the reconstructed image deteriorates with larger compression. This is especially obvious for the JPEG algorithm; a block pattern appears in the reconstructed images if the quantization level was selected too low. The HPC scheme, on the other hand, causes a blur of the reconstructed image.

The *geometric distortions* are most important for softcopy photogrammetry as they change the locations of image points and, consequently, influence the accuracy of objects reconstructed by photogrammetric procedures. For the purpose of investigating geometric distortions of the reconstructed images, 2500 points were defined in the original image. In the reconstructed images, their positions were found using the least-square matching method. Table 4 presents the geometric distortions (in pixels) for different quantization levels of the aerial image.

By analyzing Table 4, one can conclude that geometric distortions for compression ratios below 5 are small and can be neglected. Thus, it is safe to use these images for precise photogrammetric analysis and point determination. Reconstructed images from compression levels above 10 can be used for tasks that do not require precise point determination, such as the extraction and interpretation of features.

Computing Time

As the sizes of digital images used in softcopy photogrammetry are very large – they range from anywhere between 4K by 4K (16 MBytes) to 10K by 10K pixels (100 MBytes) – computing time plays a major role in selecting a compression scheme. Computing time is independent of the quantization factor and the compression ratio.

The time required to compress and decompress a 1024 by 1024 (1-MByte) image on a Sun SPARCstation 2 using the JPEG

TABLE 2. COMPRESSION RATIOS DEPENDENT ON IMAGE TYPE AND QUANTIZATION VALUE

Technique	Quantization value(s)	Aerial Image	Road Image	Target Image
JPEG	100 (lossless)	1.3	2.4	2.9
JPEG	75 (lossy)	5.9	33.3	43.5
JPEG	40 (lossy)	12.5	40.0	62.5
HPC	8.5.3.1 (lossy)	3.6	3.6	3.6
HPC	8.3.3.1 (lossy)	4.5	4.5	4.5

TABLE 3. RADIOMETRIC DISTORTIONS OF RECONSTRUCTED IMAGES DEPENDENT UPON QUANTIZATION LEVELS. THE NUMBERS IN EACH COLUMN REPRESENT THE AVERAGE AND MEAN DIFFERENCES OF GRAY VALUES OF THE ORIGINAL AND THE RECONSTRUCTED IMAGES [UNITS: GRAY VALUES]

Compression technique	Aerial Image	Road Image	Target Image
JPEG-75	2.65 / 3.06	1.32 / 1.52	0.47 / 0.54
JPEG-40	3.96 / 4.48	1.43 / 1.64	0.51 / 0.59
JPEG-5	8.26 / 9.80	2.05 / 2.36	1.72 / 1.98
HPC-8.5.3.1	1.60 / 1.81	1.58 / 1.78	1.55 / 1.79
HPC-8.3.3.1	1.62 / 1.89	1.61 / 1.80	1.60 / 1.80

Table 4. Geometric Distortions of the Reconstructed Aerial Image Dependent upon Different Quantization Levels. [units: pixel]

Distortions [comp. ratio]	JPEG-75 [5.9]	JPEG-40 [12.5]	JPEG-5 [42.3]	HPC-8.5.3.1 [3.6]	HPC-8.3.3.1 [4.5]
maximum x	0.292	2.654	17.036	0.308	1.303
maximum y	0.353	0.815	12.972	0.671	2.075
RMS in x	0.038	0.107	0.786	0.036	0.056
RMS in y	0.042	0.102	0.807	0.031	0.070

algorithm is about 20 seconds. The HPC algorithm needs 7 seconds for the same image size. The JPEG technique is considerably slower because it employs the discrete cosine transformation, which is very time consuming. The HPC scheme is faster because it is mostly based on convolutions. Therefore, it can be easily implemented on an array processor to perform real-time compression for digital image collection. As JPEG is the established standard for image compression, there are dedicated chips available that compress low resolution images (e.g., 640 by 480 pixels) in real time.

Conclusions

In this article, the popular JPEG compression scheme was compared to a new algorithm based on the Hierarchical Predictive Coding (HPC) technique. The HPC technique is faster and maintains a good geometric and radiometric image quality. However, we could only achieve small compression ratios. The JPEG technique, on the other hand, is rather slow if implemented in software, but it achieves high compression ratios that, in extreme cases, cause considerable distortions of the compressed image. Both radiometric and geometric distortions were evaluated, and it was found that, for compression ratios smaller than 5, these distortions are smaller than 1/10 of a pixel and can be neglected. Larger compression ratios may distort the images considerably and are not recommended for aerial softcopy photogrammetry. However, the reader should keep in mind that compression ratios obtained by JPEG heavily depend on the image contents (entropy), and, therefore, larger compression ratios may be adequate for machine vision and other close-range applications.

Image compression is an important topic in softcopy photogrammetry, as very large digital images (> 100 MBytes) are commonly used in photogrammetric workstations. Different compression schemes are already available in hardware and, as such, have become an integral part of softcopy stations. Most commercial compression hardware is based on the JPEG algorithm, which is the industry standard for image compression.

Image compression alone, however, will not solve all problems of handling large quantities of digital data. Databases need to be developed to manage digital images in a geographic information system (GIS), and make them easily and quickly accessible for the user. In a GIS, photogrammetric images must be tied to a geographic reference system and should contain the full orientation information, or they should be available as geocoded or ortho-rectified scenes which can be directly used as layers of a raster GIS.

References

Florida Atlantic University, 1992. *1992 R&D Program for Video Compression*, Year End Report, Communications Technology Center, Boca Raton, Florida.

Huffman, D.A., 1952. A Method for the Construction of Minimum Redundancy Codes, *Proceedings of the IRE*, pp. 1098–1101.

Jain, A., 1988. *Fundamentals of Digital Image Processing*, Prentice Hall, Englewood Cliffs, New Jersey.

Lynch, T., 1985. *Data Compression Techniques and Applications*, Lifetime Learning Publications, Belmont, California.

Nunes, P.R., A. Alcaim, and M. da Silva, 1992. Compression of Satellite Images for Remote Sensing Applications, *International Archives of Photogrammetry and Remote Sensing*, Volume 29, Comm. II.

Rabbani, M., and P. Jones, 1990. *Digital Image Compression Techniques*, SPIE Optical Engineering Press, Bellingham, Washington.

Sarjakoski, T., and J. Lammi, 1992. Compression of Digital Color Images by the JPEG, *International Archives of Photogrammetry and Remote Sensing*, Volume 29, Comm. II.

Shannon, E.E., and W. Hannon, 1949. *The Mathematical Theory of Communication*, University of Illinois Press, Urbana.

Storer, J., 1988. *Data Compression: Methods and Theory*, Computer Science Press, Rockville, Maryland.

Wallace, G., 1991. The JPEG Still Picture Compression Standard, submitted in December 1991 for publication in *IEEE Transactions on Consumer Electronics*.