# Module

# 6

# STILL IMAGE COMPRESSION STANDARDS

# Lesson
# 16
# Still Image Compression Standards: JBIG and JPEG

## Instructional Objectives

At the end of this lesson, the students should be able to:

1. Explain the need for standardization in image transmission and reception.
2. Name the coding standards for fax and bi-level images and state their characteristics.
3. Present the block diagrams of JPEG encoder and decoder.
4. Describe the baseline JPEG approach.
5. Describe the progressive JPEG approach through spectral selection.
6. Describe the progressive JPEG approach through successive approximation.
7. Describe the hierarchical JPEG approach.
8. Describe the lossless JPEG approach.
9. Convert YUV images from RGB.
10. Illustrate the interleaved and non-interleaved ordering for color images.

## 16.0 Introduction

With the rapid developments of imaging technology, image compression and coding tools and techniques, it is necessary to evolve coding standards so that there is compatibility and interoperability between the image communication and storage products manufactured by different vendors. Without the availability of standards, encoders and decoders can not communicate with each other; the service providers will have to support a variety of formats to meet the needs of the customers and the customers will have to install a number of decoders to handle a large number of data formats. Towards the objective of setting up coding standards, the international standardization agencies, such as International Standards Organization (ISO), International Telecommunications Union (ITU), International Electro-technical Commission (IEC) etc. have formed expert groups and solicited proposals from industries, universities and research laboratories. This has resulted in establishing standards for bi-level (facsimile) images and continuous tone (gray scale) images. In this lesson, we are going to discuss the highlighting features of these standards. These standards use the coding and compression techniques – both lossless and lossy which we have already studied in the previous lessons.

The first part of this lesson is devoted to the standards for bi-level image coding. Modified Huffman (MH) and Modified Relative Element Address Designate (MREAD) standards are used for text-based documents, but more recent standards like JBIG1 and JBIG2, proposed by the Joint bi-level experts' group (JBIG) can efficiently encode handwritten characters and binary halftone images. The latter part of this lesson is devoted to the standards for continuous tone images. We are going to discuss in details about the Joint Photographic Experts Group (JPEG) standard and its different modes, such as baseline (sequential), progressive, hierarchical and lossless. The more recent and advanced coding standard – the JPEG-2000 will be discussed in the next lesson (lesson-17).

## 16.1 Coding Standards for Fax and Bi-level Images

Consider an A4-sized (8.5 in x 11 in) scanned page having 200 dots/in. An uncompressed image would require transmission of 3,740,000 bits for this scanned page. It is however seen that most of the information on the scanned page is highly correlated along the scan lines, which proceed in the direction of left to right in top to bottom order and also in between the scan lines. The coding standards have exploited this redundancy to compress bi-level images. The coding standards proposed for bi-level images are:

**(a) Modified Huffman (MH):** This algorithm performs one-dimensional run length coding of scan lines, along with special end-of-line (EOL), end-of-page (EOP) and synchronization codes. The MH algorithm on an average achieves a compression ratio of 20:1 on simple text documents.

**(b) Modified Relative Element Address Designate (MREAD):** This algorithm uses a two-dimensional run length coding to take advantage of vertical spatial redundancy, along with horizontal spatial redundancy. It uses the previous scan line as a reference when coding the current line. The position of each black-to-white or white-to-black transition is coded relative to a reference element in the current scan line. The compression ratio is improved to 25:1 for this algorithm.

**(c) JBIG1:** The earlier two algorithms just mentioned work well for printed texts but are inadequate for handwritten texts or binary halftone images (continuous images converted to dot patterns). The JBIG1 standard, proposed by the Joint Bi-level Experts Group uses a larger region of support for coding the pixels. Binary pixel values are directly fed into an arithmetic coder, which utilizes a sequential template of nine adjacent and previously coded pixels plus one adaptive pixel to form a 10-bit context. Other than the sequential mode just described, JBIG1 also supports progressive mode in which a reduced resolution starting layer image is followed by the transmission of progressively higher resolution layers. The compression ratios of JBIG1 standard is slightly better than that of MREAD for text images but has an improvement of 8-to-1 for binary halftone images.

(d) **JBIG2:** This is a more recent standard proposed by the Joint bi-level Experts Group. It uses a soft pattern matching approach to provide a solution to the problem of substitution errors in which an imperfectly scanned symbol is wrongly matched to a different symbol, as frequently observed in Optical Character Recognition (OCR). JBIG2 codes the bit-map of each mark, rather than its matched class index. In case a good match cannot be found for the current mark, it becomes a token for a new class. This new token is then coded using JBIG1 with a fixed template of previous pixels around the current mark. The JBIG2 standard is seen to be 20% more efficient than the JBIG1 standard for lossless compression.

## 16.2 Continuous tone still image coding standards

A different set of standards had to be created for compressing and coding continuous tone monochrome and color images of any size and sampling rate. Of these, the Joint Photographic Expert Group (JPEG)'s first standard, known as JPEG is the most widely used one. Only in recent times, the new standard JPEG-2000 has its implementations in still image coding systems. JPEG is a very simple and easy to use standard that is based on the Discrete Cosine Transform (DCT).

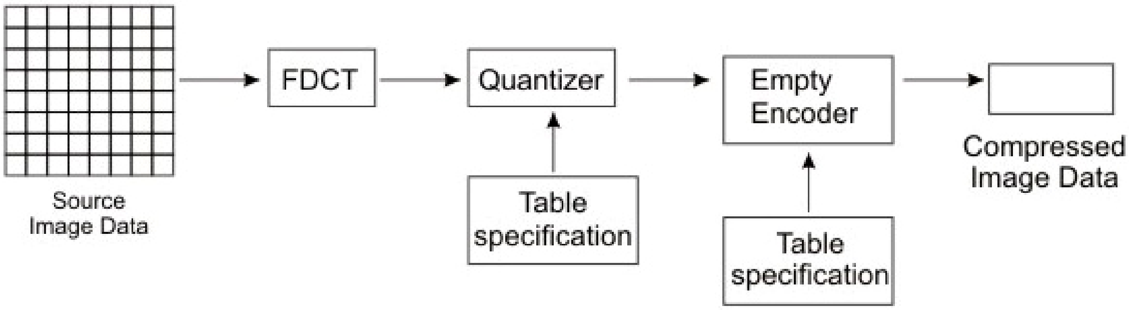

***Fig. 16.1:*** *JPEG Encoder*

Fig.16.1 shows the block diagram of a JPEG encoder, which has the following components:

(a) **Forward Discrete Cosine Transform (FDCT):** The still images are first partitioned into non-overlapping blocks of size 8x8 and the image samples are shifted from unsigned integers with range $\left[0,2^{p}-1\right]$ to signed integers with range $\left[-2^{p-1},2^{p-1}\right]$, where $p$ is the number of bits (here, $p=8$). The theory of the DCT has been already discussed in lesson-8 and will not be repeated here. It should however be mentioned that to preserve freedom for innovation and customization within implementations, JPEG neither specifies any unique FDCT algorithm, nor any unique IDCT algorithms.

The implementations may therefore differ in precision and JPEG has specified an accuracy test as a part of the compliance test.

**(b) Quantization:** Each of the 64 coefficients from the FDCT outputs of a block is uniformly quantized according to a quantization table. Since the aim is to compress the images without visible artifacts, each step-size should be chosen as the perceptual threshold or for "just noticeable distortion". Psycho-visual experiments have led to a set of quantization tables and these appear in ISO-JPEG standard as a matter of information, but not a requirement.

The quantized coefficients are zig-zag scanned, as described in lesson-8. The DC coefficient is encoded as a difference from the DC coefficient of the previous block and the 63 AC coefficients are encoded into (run, level) pair.

**(c) Entropy Coder:** This is the final processing step of the JPEG encoder. The JPEG standard specifies two entropy coding methods – Huffman and arithmetic coding. The baseline sequential JPEG uses Huffman only, but codecs with both methods are specified for the other modes of operation. Huffman coding requires that one or more sets of coding tables are specified by the application. The same table used for compression is used needed to decompress it. The baseline JPEG uses only two sets of Huffman tables – one for DC and the other for AC.

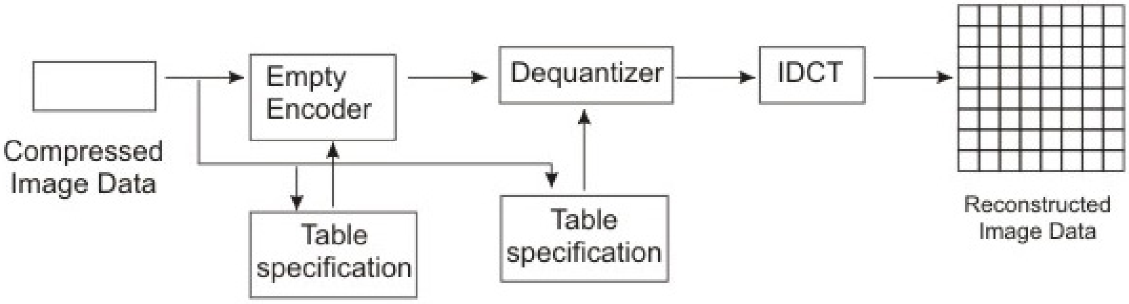

***Fig. 16.2 :*** *JPEG Decoder*

Fig.16.2 shows the block diagram of the JPEG decoder. It performs the inverse operation of the JPEG encoder.

## 16.3 Modes of Operation in JPEG

The JPEG standard supports the following four modes of operation:

- Baseline or sequential encoding

- Progressive encoding (includes spectral selection and successive approximation approaches).
- Hierarchical encoding
- Lossless encoding

**16.3.1 Baseline Encoding:** Baseline sequential coding is for images with 8-bit samples and uses Huffman coding only. In baseline encoding, each block is encoded in a single left-to-right and top-to-bottom scan. It encodes and decodes complete 8x8 blocks with full precision one at a time and supports interleaving of color components, to be described in Section-16.4. The FDCT, quantization, DC difference and zig-zag ordering proceeds in exactly the manner described in Section-16.2. In order to claim JPEG compatibility of a product it must include the support for at least the baseline encoding system.

**16.3.2 Progressive Encoding:** Unlike baseline encoding, each block in progressive encoding is encoded in multiple scans, rather than a single one. Each scan follows the zig zag ordering, quantization and entropy coding, as done in baseline encoding, but takes much less time to encode and decode, as compared to the single scan of baseline encoding, since each scan contains only a part of the complete information. With the first scan, a crude form of image can be reconstructed at the decoder and with successive scans, the quality of the image is refined. You must have experienced this while downloading web pages containing images. It is very convenient for browsing applications, where crude reconstruction quality at the early scans may be sufficient for quick browsing of a page.

There are two forms of progressive encoding: (a) spectral selection approach and (b) successive approximation approach. Each of these approaches is described below.

**16.3.2.1 Progressive scanning through spectral selection:** In this approach, the first scan sends some specified low frequency DCT coefficients within each block. The corresponding reconstructed image obtained at the decoder from the first scan therefore appears blurred as the details in the forms of high frequency components are missing. In subsequent scans, bands of coefficients, which are higher in frequency than the previous scan, are encoded and therefore the reconstructed image gets richer with details. This procedure is called spectral selection, because each band typically contains coefficients which occupy a lower or higher part of the frequency spectrum for that 8x8 block.

***Fig.16.3 Spectral Selection Approach***

Fig.16.3 illustrates the spectral selection approach. Here all the 64 DCT coefficients in a block are of 8-bit resolution and successive blocks are stacked one after the other in the scanning order. The spectral selection approach performs the slicing of coefficients horizontally and picks up a band of coefficients, starting with low frequency and encodes them to full resolution.

**16.3.2.2 Progressive scanning through successive approximation:** This is also a multiple scan approach. Here, each scan encodes all the coefficients within a block, but not to their full quantized accuracy. In the first scan, only the *N* most significant bits of each coefficient are encoded (*N* is specifiable) and in successive scans, the next lower significant bits of the coefficients are added and so on until all the bits are sent. The resulting reconstruction quality is good even from the early scans, as the high frequency coefficients are present from the initial scans.

Fig.16.4 illustrates the successive approximation approach. The organization of the DCT coefficients and the stacking of the blocks are same as before. The successive approximation approach performs the slicing operation vertically and picks up a group pf bits, starting with the most significant ones and progressively considering the lower frequency ones.

***Fig.16.4 Successive Approximation Approach***

**16.3.3 Hierarchical encoding:** The hierarchical encoding is also known as the pyramidal encoding in which the image to be encoded is organized in a pyramidal structure of multiple resolutions, with the original, that is, the finest resolution image on the lowermost layer and reduced resolution images on the successive upper layers. Each layer decreases its resolution with respect to its adjacent lower layer by a factor of two in either the horizontal or the vertical direction or both. Hierarchical encoding may be regarded as a special case of progressive encoding with increasing spatial resolution between the progressive stages.

The steps involved in hierarchical encoding may be summarized below:

- Obtain the reduced resolution images starting with the original and for each, reduce the resolution by a factor of two, as described above.

- Encode the reduced resolution image from the topmost layer of the pyramid (that is, the coarsest form of the image) using baseline (sequential) encoding (Section-16.3.1), progressive encoding (Section-16.3.2) or lossless encoding (Section-16.3.4).

- Decode the above reduced resolution image. Interpolate and up-sample it by a factor of two horizontally and/or vertically, using the identical interpolation filter which the decoder must use. Use this interpolated and up-sampled image as a predicted image for encoding the next lower layer (finer resolution) of the pyramid.

- Encode the difference between the image in the next lower layer and the predicted image using baseline, progressive or lossless encoding.
- Repeat the steps of encoding and decoding until the lowermost layer (finest resolution) of the pyramid is encoded.

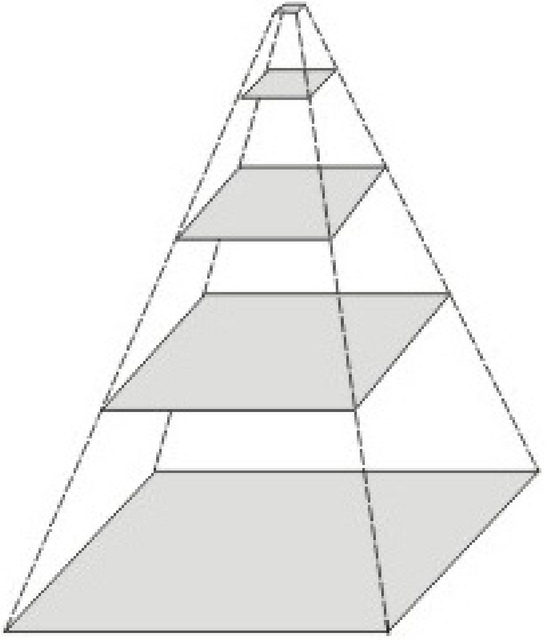

***Fig. 16.5 Hierarchical encoding (Pyramid structure)***

Fig.16.5 illustrates the hierarchical encoding process. In hierarchical encoding, the image quality at low bit rates surpass the other JPEG encoding methods, but at the cost of increased number of bits at the full resolution. Hierarchical encoding is used for applications in which a high-resolution image should be accessed by a low resolution display device. For example, the image may be printed by a high-resolution printer, while it is being displayed on a low resolution monitor.

**16.3.4 Lossless encoding:** The lossless mode of encoding in JPEG follows a simple predictive coding mechanism, rather than having FDCT + Entropy coder for encoding and Entropy decoder + IDCT for decoding. Theoretically, it should have been possible to achieve lossless encoding by eliminating the quantization block, but because of finite precision representation of the cosine kernels, IDCT can not exactly recover what the image was before IDCT. This led to a modified and simpler mechanism of predictive coding, which we discussed in section-5.3 of lesson-5.

In lossless encoding, the 8x8 block structure is not used and each pixel is predicted based on three adjacent pixels, as illustrated in fig.16.6 using one of the eight possible predictor modes, listed in table-16.1.

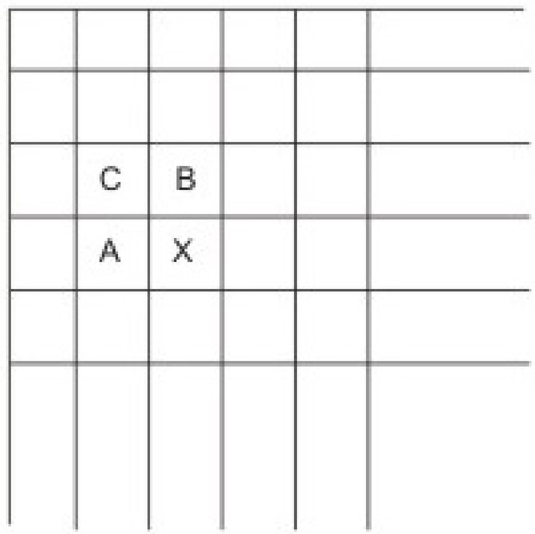

***Fig. 16.6 Predictive coding for lossless JPEG***

An entropy encoder is then used to encode the predicted pixel obtained from the lossless encoder. Lossless codecs typically produce around 2:1 compression for color images with moderately complex scenes. Lossless JPEG encoding finds applications in transmission and storage of medical images.

| Selection Value | Prediction |
|---|---|
| 0 | None |
| 1 | A |
| 2 | B |
| 3 | C |
| 4 | A+B-C |
| 5 | A+(B-C)/2 |
| 6 | B+(A-C)/2 |
| 7 | (A+B)/2 |

Table-16.1 Predictors in lossless JPEG mode

## 16.4 Color image formats and interleaving

The most commonly used color image representation format is RGB, the encoding of which may be regarded as three independent gray scale image encoding. However, from efficient encoding considerations, RGB is not the best format. Color spaces such as YUV, CIELUV, CIELAB and others represent the chromatic (color) information in two components and the luminance (intensity) information in one component. These formats are more efficient from image compression considerations, since our eyes are relatively insensitive to the high frequency information from the chrominance channels and thus the chrominance components can be represented at a reduced resolution as compared to the luminance components for which full resolution representation is necessary.

It is possible to convert an RGB image into YUV, using the following relations:

$$Y = 0.3R + 0.6G + 0.1B \qquad (16.1)$$

$$U = \frac{B - Y}{2} + 0.5 \qquad (16.2)$$

$$V = \frac{R - Y}{1.6} + 0.5 \qquad (16.3)$$

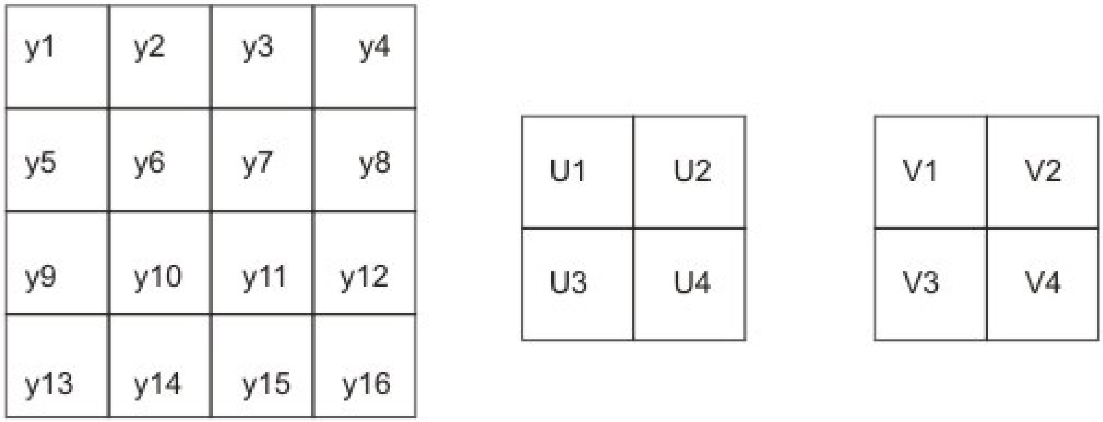

*Fig. 16.7 YUV representation of an example 4x4 image*

Fig.16.7 illustrates the YUV representation by considering an example of a 4x4 image. The Y components are shown as Y1, Y2, ……, Y16. The U and the V components are sub-sampled by a factor of two in both horizontal and vertical directions and are therefore of 2x2 size. The three components may be transmitted in either a non-interleaved manner or an interleaved manner.

The non-interleaved ordering can be shown as

Scan-1: Y1,Y2,Y3,……,Y15,Y16.

Scan-2: U1,U2,U3,U4.

Scan-3: V1,V2,V3,V4.

The interleaved ordering encodes in a single scan and proceeds like

Y1, Y2, Y3, Y4, U1, V1, Y5, Y6, Y7, Y8, U2, V2, ………

Interleaving requires minimum of buffering to decode the image at the decoder.

## 16.5 JPEG Performance

Considering color images having 8-bits/sample luminance components and 8-bits/sample for each of the two chrominance components U and V, each pixel requires 16-bits for representation, if both U and V are sub-sampled by a factor of two in either of the directions. Using JPEG compression on a wide variety of such color images, the following image qualities were measured subjectively:

| Bits/pixel | Quality | Compression Ratio |
|---|---|---|
| $\geq 2$ | Indistinguishable | 8:1 |
| 1.5 | Excellent | 10.7:1 |
| 0.75 | Very good | 21.4:1 |
| 0.5 | Good | 32:1 |
| 0.25 | Fair | 64:1 |

A more advanced still image compression standard JPEG-2000 has evolved in recent times. This will be our topic in the next lesson.