Module
6

STILL IMAGE COMPRESSION STANDARDS

Lesson 17

JPEG-2000 – Architecture and Features

Instructional Objectives

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

1. State the shortcomings of JPEG standard.
2. State the scope and objectives of JPEG-2000.
3. Name the main application areas of JPEG-2000.
4. List the requirements of its major applications.
5. Explain the major building blocks of JPEG-2000.
6. Define tiling and its significance.
7. Distinguish between reversible and irreversible wavelet transforms.
8. Perform periodic extension of signals.
9. Distinguish between convolution based filtering and lifting based filtering.
10. Show the relations for reversible and irreversible component transformations.
11. Define precincts and packets for JPEG-2000 bit stream.

17.0 Introduction

In lesson-16, we had discussed in details about the JPEG standard. This was in fact the first ever international standard on continuous tone still image compression and coding. Despite its very impressive performance and the massive popularity it gained in storage of still images and transmission through internet, some shortcomings were felt as the technology further matured. First, JPEG encoded images show severe blocking artifacts at very low bit rate. This problem basically originates from the use of block-based DCTs, which we had discussed earlier. Moreover, JPEG provides insufficient support for spatial and SNR (quality) scalability. JPEG lacks object-based and region-based representation and has no support for error resiliency. Subsequent to the acceptance of JPEG as a standard, the technology pertaining to image compression and coding matured further, especially in the field of wavelet transforms and subband coding, which we had discussed in lessons-10 to 15. The shortcomings of JPEG, maturity of wavelet transforms and subband coding and proliferation of new application areas prompted the international community to work on a more advanced still image coding standard, known as JPEG-2000, whose work was first initiated in 1996.

In this lesson, we are going to present the major application areas of JPEG-2000 and their requirements, the architecture of the coding engine and the encoding concepts. JPEG-2000 encoding is based on EBCOT wavelet coding technique, which we have already discussed in lesson-15. Hence, the focus of this lesson will not be on the coding algorithm, but more on its architectural aspects. Because of several added features not found in earlier coding standards, improved capabilities and vast future potentials, we shall devote two more lessons, viz. lesson-18 and lesson-19, in addition to the present one to discuss various aspects of JPEG-2000 coding.

17.1 Scope and objectives of JPEG-2000

The JPEG-2000 standards were intended for use in different types of still images, such as bi-level, gray-scale, color (or multi-component) with different characteristics, such as natural images, scientific, medical, remote sensing, text, rendered graphics etc and on different imaging models, such as client/server, real-time transmission, image library archival, limited buffer and bandwidth resources etc – all within a unified framework. The objective was to make it 30% more efficient than the baseline JPEG. The standardization committee developed the Part-I of JPEG-2000 standard to make it royalty free, so as to promote its wide usage.

17.2 Applications of JPEG-2000 and their requirements

The JPEG-2000 standards have several applications and more are emerging with time. Some of the major application areas are:

- Internet
- Color facsimile
- Printing
- Scanning
- Digital photography
- Remote Sensing
- Mobile
- Medical imagery
- Digital libraries and archives
- E-commerce

Each application area has some requirements which the standard should fulfill. The requirements from different application areas may be summarized as follows:

- **Improved low bit-rate performance:** It should give acceptable quality below 0.25 bpp. Networked image delivery and remote sensing applications have this requirements.

- **Lossless and lossy compression:** Lossless compression should be included in progressive decoding. It should support embedded bit-stream that allow progressive lossy to lossless build-up.

- **Progressive transmission:** The standard should allow progressive transmission that allows images to be reconstructed with increasing pixel accuracy and resolution.

- **Region of Interest Coding:** It should preferentially allocate more bits to the regions of interest (ROIs) as compared to the non-ROI ones.

- **Random code stream access:** The user defined ROIs in an image should be randomly accessible.

- **Error resilience:** The standard should provide error resiliency, especially for wireless communication channels.

- **Open architecture:** It should support open architecture to optimize the system for different image types and applications. The decoder should have only the core tool set and a parser that understands the core stream. If necessary, the unknown tools could be requested by the decoder.

- **Content based description:** Finding the desired image from a large archive of images is a challenging task. This has applications in medical images, forensic, digital libraries etc. These issues are being addressed by MPEG-7.

- **Image Security:** This is another major requirement to protect the intellectual property rights of the images. Digital images can be protected using watermarking, labeling, stamping, encryption etc.

- **Continuous-tone and bi-level image compression:** Previously, we had different standards for bi-level images (JBIG standards) and continuous-tome images (JPEG standard). A requirement was felt that a single standard should address both these domains, including the color images and encompass applications involving compound images with overlaying texts.

- **Sequential one-pass decoding:** The decoding should be done in a single sequential pass and support interleaved and non-interleaved mode.

- **Side-channel support:** The standard should support alpha planes and transparency planes.

All these requirements, originating from different application domains were addressed and incorporated in the JPEG-2000 standard.
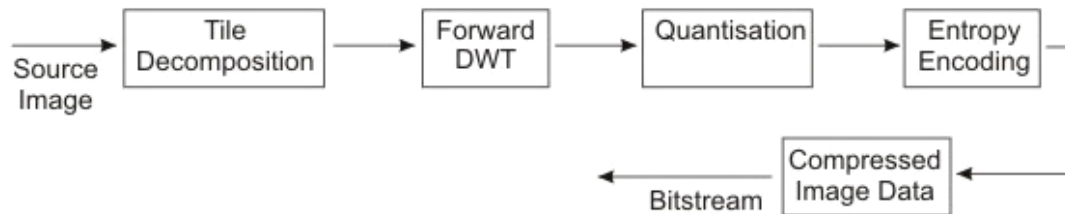
17.3 Architecture of JPEG-2000

Fig. 17.1: *Block Diagram of JPEG-2000 encoder.*

Fig.17.1 shows the block-diagram of JPEG-2000 encoder and fig.17.2 shows its corresponding decoder.
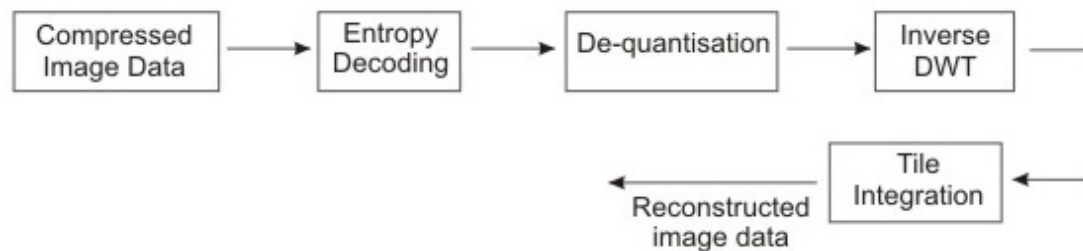

Fig. 17.2: *Block Diagram of JPEG-2000 decoder*

The block diagrams are not significantly different from those of JPEG, except for the fact that the transform used is Discrete Wavelet Transform (DWT), rather than the DCT. Before applying the DWT, the source image is divided into components (colors) and each component is divided into tiles (to be described in Section-17.4) which are compressed independently, as though they are independent images and all the samples in the tile are DC level shifted. The DWT coefficients in different subbands are quantized and then composed into an embedded bit-stream following the EBCOT algorithm, presented in lesson-15. The embedded bit-stream is composed of quality layers so as to offer both resolution and SNR scalability. The entropy coding is based on a context-adaptive arithmetic coder, known as MQ coder, which has error correction capability. The encoding process may be summarized as follows:

- The source image is decomposed into components, such as RGB or YUV.

- The image and its components are divided into rectangular tiles, which form the basic unit to be encoded.

- DWT at different resolution levels is applied on each tile to compose the subbands. The DWT may be reversible or irreversible, to be described in Section-17.5.

- The subband coefficients are quantized and collected into rectangular arrays of code blocks (described in lesson-15, Section-15.3).

- The bit-planes of the code-blocks are arithmetic coded in multiple passes with fractional bit-plane concept of EBCOT algorithm (refer lesson-15).

- The encoding is done in such a way that the Regions of Interest (ROI) get higher quality, as compared to the other areas (to be discussed in lesson-18).

- Markers are introduced in the bit-stream for error resilience (to be discussed in lesson-19).

17.4 Tiling and its significance

Before applying the DWT, the image and its components are divided into smaller non-overlapping blocks, known as tiles, which can be coded independently, as if each tile is an independent image. All operations, such as component mixing, DWT, quantization and entropy coding are therefore done independently for each tile.

Tiling has the advantage of reducing memory requirements for DWT and its processing and is amenable to parallelization. Moreover, tiles may be independently accessed and used for decoding specific parts of the image, rather than the complete one.

The tile may be as large as the entire image size (that is, single tile) or of smaller partitions, such as 256x256, 128x128 etc. In terms of PSNR, tiling degrades the performance, as compared to no tiling and smaller tile sizes lead to tiling artifacts.

17.5 Wavelet Filters

The JPEG-2000 standard supports lossy as well as lossless encoding. Two types of wavelet filters are included in Part-I of the standard –

(a) *Irreversible*, where exact reconstruction will not be possible at the decoder and is used for lossy encoding. This is implemented using a 9/7 Daubechies filter, whose analysis and synthesis filter coefficients are shown in Table-17.1.

(b) *Reversible,* where exact reconstruction at the decoder is possible and is therefore included for lossless JPEG-2000. This is implemented using a 5/3 filter, whose analysis and synthesis filter coefficients are shown in **Table-17.2.**

Analysis Filter Coefficients		
i	Low-pass filter $h_l(i)$	High-pass filter $h_u(i)$
0	0.6029490182363579	1.115087052456994
± 1	0.2668641184428723	-0.5912717631142470
± 2	-0.07822326652898785	-0.05754352622849957
± 3	-0.01686411844287495	0.09127176311424948
± 4	0.02674875741080976	

Synthesis Filter Coefficients		
i	Low-pass filter $h_l(i)$	High-pass filter $h_u(i)$
0	1.115087052456994	0.6029490182363579
± 1	0.5912717631142470	-0.2668641184428723
± 2	-0.05754352622849957	-0.07822326652898785
± 3	-0.09127176311424948	0.01686411844287495
± 4		0.02674875741080976

Table-17.1 Daubechies 9/7 analysis and synthesis filter coefficients

	Analysis Filter Coefficients		Synthesis Filter Coefficients	
i	Low-pass filter $h_l(i)$	High-pass filter $h_u(i)$	Low-pass filter $g_l(i)$	High-pass filter $g_u(i)$
0	6/8	1	1	6/8
± 1	2/8	-1/2	1/2	-2/8
± 2	-1/8			-1/8

Table-17.2   5/3 Analysis and synthesis filter coefficients

17.5.1 Periodic Extension of Signals: Two filtering approaches, namely convolution and lifting based (to be discussed in Section-17.5.1) filters are used in JPEG-2000 standard. Both these approaches fail at the signal boundaries, since corresponding signal samples are not available for all the filter coefficients. One can consider the non-existent samples as zeros, but this leads to unsatisfactory results because of signal discontinuities. A better approach is to extend the signal at both the boundaries by including the mirror images of the samples about the boundaries.
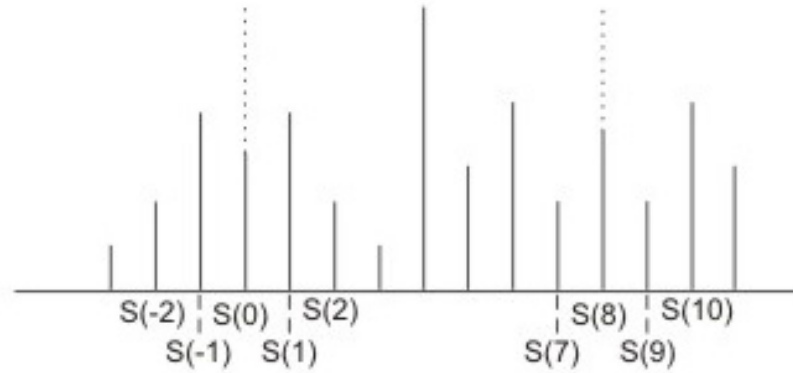

Fig. 17.3: *Periodic symmetric extension of 1-d signal S(0),S(1)……S(8).*

Fig. 17.3 illustrates the signal extension for a one-dimensional example of nine samples: $s(0), s(1), s(2), \cdots\cdots, s(8)$, where $s(0)$ and $s(8)$ serve as boundary samples. The samples are extended to the left of $s(0)$ and to the right of $s(8)$ as

$$s(i) = s(-i), \quad i = -1, -2, \cdots\cdots$$
$$s(8+i) = s(8-i) \quad i = 1, 2, 3, \cdots\cdots$$

The extent to which the samples should be extended depends on the filter length.

17.5.2 Convolution and Lifting Based filters: The convolution based filtering performs a dot product between the two filter masks (for low-pass and high-pass filters) and the extended 1-D signal. In lifting based filtering, the samples are split into odd and even groups. The odd samples are updated with a weighted sum of even sample values and the even samples are updated with a weighted sum of odd sample values. For the reversible filters (lossless case), the results are rounded to integer values. The lifting based filtering for the 5/3 analysis filter is achieved as follows:

$$y(2n+1) = x_{ext}(2n+1) - \left\lfloor \frac{x_{ext}(2n) + x_{ext}(2n+2)}{2} \right\rfloor$$

$$y(2n) = x_{ext}(2n) + \left\lfloor \frac{y(2n-1) + y(2n+1) + 2}{4} \right\rfloor$$

where x_{ext} is the extended input signal and *y* is the output signal.

17.6   Reversible and Irreversible Component Transformations

JPEG-2000 supports two different component transformations for multi-component images:

- *Irreversible Component Transformation (ICT)* used for lossy coding only. It may only be used with the 9/7 irreversible wavelet transform.

- *Reversible Component Transformation (RCT)* used for lossless as well as for lossy coding. It may only be used with the 5/3 reversible wavelet transform.

The forward (RGB to YUV) and the inverse ICT (YUV to RGB) transformations are given by

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.16875 & -0.33126 & 0.5 \\ 0.5 & -0.41869 & -0.08131 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.0 & 0 & 1.402 \\ 1.0 & -0.34413 & -0.71414 \\ 1.0 & 1.772 & 0 \end{bmatrix} \begin{bmatrix} Y \\ U \\ V \end{bmatrix}$$

The forward and the inverse RCT transformations are given by

$$\begin{bmatrix} Y_r \\ U_r \\ V_r \end{bmatrix} = \begin{bmatrix} \left\lfloor \dfrac{R + 2G + B}{4} \right\rfloor \\ R - G \\ B - G \end{bmatrix}$$

$$\begin{bmatrix} G \\ R \\ B \end{bmatrix} = \begin{bmatrix} Y_r - \left\lfloor \dfrac{U_r + V_r}{4} \right\rfloor \\ U_r + G \\ V_r + G \end{bmatrix}$$

The performance comparisons between lossless compression (RCT + 5/3 reversible filter) and lossy compression (ICT + 9/7 irreversible filter) at comparable bit rate indicates significantly better performance for the latter, as compared to the former.

It may be noted that the sub-sampling of the chrominance components, as shown in Section-16.4 of lesson-16 is not recommended for JPEG-2000, since a 2:1 sub-sampling in horizontal and vertical directions may be obtained by simply discarding the HL, LH and HH subbands of a component's wavelet decomposition and retaining all other subbands.

17.7 Precincts and packets for JPEG-2000 bit-stream

The arithmetically coded quantized wavelet coefficients are arranged in *packets* and the packet partition locations are referred to as *precincts*. After quantization, each subband is divided into some non-overlapping rectangles. Three spatially consistent rectangles - one from each subband at each resolution level comprise a packet partition location or precinct, as illustrated in fig.17.4. Each precinct is further divided into code-blocks (discussed in Section-15.3 of lesson-15), which form the input to the arithmetic coder.
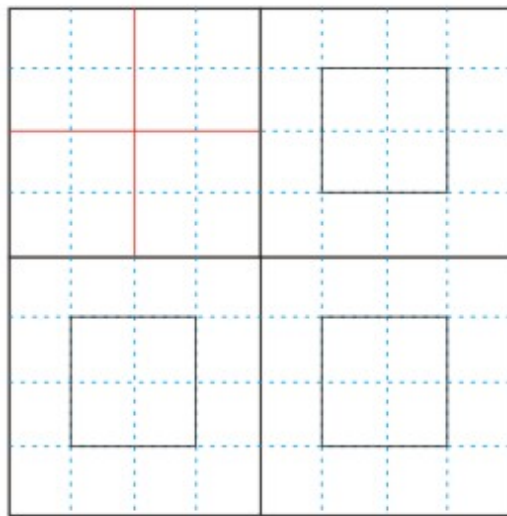

Fig. 17.4 Partitioning of a tile into subbands (red borders) , code blocks (blue borders) and precincts (black borders)

The data representing a specific tile, layer (refer to lesson-15), component, resolution and precinct appears in the code stream in a contiguous segment, called a *packet*. The data in a packet is ordered according to the subbands LL, HL, LH and HH and within each subband, the codeblocks are arranged in raster scanning order, confined within the bounds of the corresponding precinct.