UNIT IV WAVELETS AND IMAGE COMPRESSION

PART A

   In terms of storage, the capacity of a storage device can be effectively increased with methods that compress a body of data on its way to a storage device and decompresses it when it is retrieved. In terms of communications, the bandwidth of a digital communication link can be effectively increased by compressing data at the sending end and decompressing data at the receiving end. At any given time, the ability of the Internet to transfer data is fixed. Thus, if data can effectively be compressed wherever possible, significant improvements of data throughput can be achieved. Many files can be combined into one compressed document making sending easier.

2. Define compression ratio. (Nov./Dec. 2009)
   Compression Ratio is defined as original size / compressed size: 1

3. What are the coding systems in JPEG? (Nov./Dec. 2012)
   1. A lossy baseline coding system, which is based on the DCT and is adequate for most compression application.
   2. An extended coding system for greater compression, higher precision or progressive reconstruction applications.
   3. A lossless independent coding system for reversible compression.

   The acronym is expanded as "Joint Photographic Expert Group". It is an international standard in 1992. It perfectly Works with color and grayscale images, Many applications e.g., satellite, medical,...

5. What are the basic steps in JPEG? (Nov./Dec. 2009)
   The Major Steps in JPEG Coding involve:
   _ DCT (Discrete Cosine Transformation)
   _ Quantization
   _ Zigzag Scan _ DPCM on DC component
   _ RLE on AC Components
   _ Entropy Coding

   The acronym is expanded as "Moving Picture Expert Group". It is an international standard in 1992. It perfectly Works with video and also used in teleconferencing
PART B


The acronym is expanded as "Joint Photographic Expert Group". It is an international standard in 1992. It perfectly works with color and grayscale images. Many applications e.g., satellite, medical.

The Major Steps in JPEG Coding involve:

DCT (Discrete Cosine Transformation)
Quantization
Zigzag Scan
DPCM on DC component
RLE on AC Components
Entropy Coding

ENCODER

DECODER

JPEG is a transform coding approach using DCT. Consider 8*8 block of the image as shown in table

Table: an 8*8 block of an image

<p>| | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>124</td>
<td>125</td>
<td>122</td>
<td>120</td>
<td>122</td>
<td>119</td>
<td>117</td>
<td>118</td>
</tr>
<tr>
<td>121</td>
<td>121</td>
<td>120</td>
<td>119</td>
<td>119</td>
<td>120</td>
<td>120</td>
<td>118</td>
</tr>
<tr>
<td>125</td>
<td>124</td>
<td>123</td>
<td>122</td>
<td>121</td>
<td>121</td>
<td>120</td>
<td>120</td>
</tr>
<tr>
<td>124</td>
<td>124</td>
<td>125</td>
<td>125</td>
<td>126</td>
<td>125</td>
<td>124</td>
<td>124</td>
</tr>
<tr>
<td>127</td>
<td>127</td>
<td>128</td>
<td>129</td>
<td>130</td>
<td>128</td>
<td>127</td>
<td>125</td>
</tr>
<tr>
<td>143</td>
<td>142</td>
<td>143</td>
<td>142</td>
<td>140</td>
<td>139</td>
<td>139</td>
<td>139</td>
</tr>
</tbody>
</table>
The Transform

The transform used in the JPEG scheme is the DCT. The input image is first "level shifted by 2^(p-1)" i.e., subtract 2^(p-1) from each pixel value. Then the image is divided into blocks of size 8*8, which are transformed using an 8*8 forward DCT. The table shows the DCT coefficient.

Table: The DCT coefficient

<table>
<thead>
<tr>
<th>39.88</th>
<th>6.56</th>
<th>-2.24</th>
<th>1.22</th>
<th>-0.37</th>
<th>-1.08</th>
<th>0.79</th>
<th>1.13</th>
</tr>
</thead>
<tbody>
<tr>
<td>-102.43</td>
<td>4.56</td>
<td>2.26</td>
<td>1.12</td>
<td>0.35</td>
<td>-0.63</td>
<td>-1.05</td>
<td>-0.48</td>
</tr>
<tr>
<td>37.77</td>
<td>1.31</td>
<td>1.77</td>
<td>0.25</td>
<td>-1.50</td>
<td>-2.21</td>
<td>-0.10</td>
<td>0.23</td>
</tr>
<tr>
<td>-5.67</td>
<td>2.24</td>
<td>-1.32</td>
<td>-0.81</td>
<td>1.41</td>
<td>0.22</td>
<td>-0.13</td>
<td>0.17</td>
</tr>
<tr>
<td>-3.37</td>
<td>-0.74</td>
<td>-1.75</td>
<td>0.77</td>
<td>-0.62</td>
<td>-2.65</td>
<td>-1.30</td>
<td>0.76</td>
</tr>
<tr>
<td>5.98</td>
<td>-0.13</td>
<td>-0.45</td>
<td>-0.77</td>
<td>1.99</td>
<td>-0.26</td>
<td>1.46</td>
<td>0.00</td>
</tr>
<tr>
<td>3.97</td>
<td>5.52</td>
<td>2.39</td>
<td>-0.55</td>
<td>-0.051</td>
<td>-0.84</td>
<td>-0.52</td>
<td>-0.13</td>
</tr>
<tr>
<td>-3.43</td>
<td>0.51</td>
<td>-1.07</td>
<td>0.87</td>
<td>0.96</td>
<td>0.09</td>
<td>0.33</td>
<td>0.01</td>
</tr>
</tbody>
</table>

Quantization

The JPEG algorithm uses uniform midthread quantization to quantize the various coefficients. The quantizer step sizes are organized in a table called the quantization table as shown in the table.

Table: Sample Quantization table

<table>
<thead>
<tr>
<th>16</th>
<th>11</th>
<th>10</th>
<th>16</th>
<th>24</th>
<th>40</th>
<th>51</th>
<th>61</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>12</td>
<td>14</td>
<td>19</td>
<td>26</td>
<td>58</td>
<td>60</td>
<td>55</td>
</tr>
<tr>
<td>14</td>
<td>13</td>
<td>16</td>
<td>24</td>
<td>40</td>
<td>57</td>
<td>69</td>
<td>56</td>
</tr>
<tr>
<td>14</td>
<td>17</td>
<td>22</td>
<td>29</td>
<td>51</td>
<td>87</td>
<td>80</td>
<td>62</td>
</tr>
<tr>
<td>18</td>
<td>22</td>
<td>37</td>
<td>56</td>
<td>68</td>
<td>109</td>
<td>103</td>
<td>77</td>
</tr>
<tr>
<td>24</td>
<td>35</td>
<td>55</td>
<td>64</td>
<td>81</td>
<td>104</td>
<td>113</td>
<td>92</td>
</tr>
<tr>
<td>49</td>
<td>64</td>
<td>78</td>
<td>87</td>
<td>103</td>
<td>121</td>
<td>120</td>
<td>101</td>
</tr>
</tbody>
</table>
The label corresponding to the quantized value of the transform coefficient $\theta_{ij}$ is obtained as

$$L_{ij} = \theta_{ij}/Q_{ij} + 0.5$$

Where $Q_{ij}$ is the $(i,j)$th element of the quantization table. The reconstructed value is obtained by multiplying the label with corresponding entry in the quantization table.

Table: The quantizer label

<table>
<thead>
<tr>
<th>L_{ij}</th>
<th>2</th>
<th>1</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>-9</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Coding

In this approach the label for the DC and AC coefficient are coded differently using Huffman codes. The DC coefficient values partitioned into categories. The categories are then Huffman coded. The AC coefficient is generated in slightly different manner. There are two special codes: End-of-block(EOF) and ZRL.

Table: Coding of the differences of the DC labels

<table>
<thead>
<tr>
<th>Difference</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>-1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>-3</td>
<td>-2</td>
<td>2</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>-4</td>
<td>4</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table: sample table for obtaining the Huffman code for a given label value and run length
To encode the AC coefficient First using Zigzag scan. We obtain

-9 3 0 0 0 0 ....... 0

The first value belong to category 1. transmit the code corresponding to 0/1 follow by a single bit 1 to indicate that the value being transmitted is 1 and not -1. Similarly other AC coefficient code are transmitted.

To obtain the reconstruction of the original block Dequantization is performed and taking inverse transform of the coefficient we get the reconstructed block.

2. Describe run length encoding with an example. (April/May 2015)

Run Length Encoding

Consider a matrix A with 15 elements,
A= [10 10 9 9 9 4 0 0 0 0 10 10 10]
In the given example,
10 has occurred 2 times, 9 has occurred 4 times, 4 has occurred once, 0 has occurred 5 times and 10 has occurred 3 times.
After Run length encoding, we obtain the matrix without any repetition in the adjacent elements,
[10 9 4 0 10]. And the occurrences of each element [2 4 1 5 3]
Thus the matrix is reduced to 10 elements from 15 elements.

Reduction method.
Consider the above matrix A,
1. Find the difference between adjacent elements. Use the function ‘diff(A)’ to find the difference.
   [0 -1 0 0 0 -5 -4 0 0 0 0 10 0 0]
2. Convert it to logical format. The elements without repetition are denoted with one and the repeated elements with zero.
   [0 1 0 0 0 1 1 0 0 0 0 0 0 1]
3. Find the position of the elements that has the value one in the above step.
   [2 6 7 12 15].
4. Find the unique element values using the positions obtained from the above step. In the matrix A, the element at the position 2 is 10, the element at the position 6 is 9, the element at the position 7 is 4, the element at the position 12 is 0 and the element at the position 15 is 10.
   [10 9 4 0 10]
5. The first element in the matrix is 10, it has occurred 2 times. We obtained the occurrence of the first element alone from the matrix in the step 3. For the remaining elements, find the difference of the matrix in the step 3.
i.e. \( \text{diff}([2 \ 6 \ 7 \ 12 \ 15]) \); The result after concatenating the first element of the matrix obtained in step 3 with difference for the matrix in the step 3 is \([2 \ 4 \ 1 \ 5 \ 3]\).

6. Thus in the step 4 we obtain the elements without repetition, \([10 \ 9 \ 4 \ 0 \ 10]\) and the occurrences in step 5, \([2 \ 4 \ 1 \ 5 \ 3]\).


In terms of storage, the capacity of a storage device can be effectively increased with methods that compress a body of data on its way to a storage device and decompresses it when it is retrieved. In terms of communications, the bandwidth of a digital communication link can be effectively increased by compressing data at the sending end and decompressing data at the receiving end.

**Compression**: It is the process of reducing the size of the given data or an image. It will help us to reduce the storage space required to store an image or File.

**Image Compression Model**:

There are two Structural model and they are broadly Classified as follows

An Encoder
A Decoder.

An Input image \( f(x,y) \) is fed in to encoder and create a set of symbols and after transmission over the channel ,the encoded representation is fed in to the decoder.

**A General Compression system model**:

The General system model consist of the following components,They are broadly classified as

Source Encoder
Channel Encoder
Channel
Channel Decoder
Source Decoder
The Source Encoder Will removes the input redundancies. The channel encoder will increase the noise immunity of the source encoder’s output. If the channel between encoder and decoder is noise free then the channel encoder and decoder can be omitted.

**MAPPER:**

It transforms the input data into a format designed to reduce the interpixel redundancy in the input image.

**QUANTIZER:**

It reduces the accuracy of the mapper’s output.

**SYMBOL ENCODER:**

It creates a fixed or variable length code to represent the quantizer’s output and maps the output in accordance with the code.

**SYMBOL DECODER:**

The inverse operation of the source encoder’s symbol will be performed and maps the blocks.
4. Draw and explain the block diagram of MPEG ENCODER. (NOV/DEC 2012).

MPEG: Moving Pictures Experts Group, established in 1988 for the development of digital video. • It is appropriately recognized that proprietary interests need to be maintained within the family of MPEG standards: – Accomplished by defining only a compressed bitstream that implicitly defines the decoder. – The compression algorithms, and thus the encoders, are completely up to the manufacturers.

MPEG-1 adopts the CCIR601 digital TV format also known as SIF (Source Input Format).

• MPEG-1 supports only non-interlaced video. Normally, its picture resolution is: – 352 × 240 for NTSC video at 30 fps – 352 × 288 for PAL video at 25 fps – It uses 4:2:0 chroma subsampling

• The MPEG-1 standard is also referred to as ISO/IEC 11172. It has five parts: 11172-1 Systems, 11172-2 Video, 11172-3 Audio, 11172-4 Conformance, and 11172-5 Software.

Motion Compensation (MC) based video encoding in H.261 works as follows: – In Motion Estimation (ME), each macroblock (MB) of the Target P-frame is assigned a best matching MB from the previously coded I or P frame - prediction. – prediction error: The difference between the MB and its matching MB, sent to DCT and its subsequent encoding steps. – The prediction is from a previous frame — forward prediction.

The MB containing part of a ball in the Target frame cannot find a good matching MB in the previous frame because half of the ball was occluded by another object. A match however can readily be obtained from the next frame.

MPEG introduces a third frame type — B-frames, and its accompanying bi-directional motion compensation. • The MC-based B-frame coding idea is illustrated in Fig. 11.2: – Each MB from a B-frame will have up to two motion vectors (MVs) (one from the forward and one from the backward prediction). – If matching in both directions is successful, then two MVs will be sent and the two corresponding matching MBs are averaged (indicated by ‘%’ in the figure) before comparing to the
Target MB for generating the prediction error. If an acceptable match can be found in only one of the reference frames, then only one MV and its corresponding MB will be used from either the forward or backward prediction.

Source formats supported: – H.261 only supports CIF (352 × 288) and QCIF (176 × 144) source formats, MPEG-1 supports SIF (352 × 240 for NTSC, 352 × 288 for PAL). – MPEG-1 also allows specification of other formats as long as the Constrained Parameter Set (CPS) as shown in Table 11.1 is satisfied: Table 11.1: The MPEG-1 Constrained Parameter Set Parameter Value Horizontal size of picture ≤ 768 Vertical size of picture ≤ 576 No. of MBs / picture ≤ 396 No. of MBs / second ≤ 9, 900 Frame rate ≤ 30 fps Bit-rate ≤ 1, 856 kbps

Source formats supported: – H.261 only supports CIF (352 × 288) and QCIF (176 × 144) source formats, MPEG-1 supports SIF (352 × 240 for NTSC, 352 × 288 for PAL). – MPEG-1 also allows specification of other formats as long as the Constrained Parameter Set (CPS) as shown in Table 11.1 is satisfied: Table 11.1: The MPEG-1 Constrained Parameter Set Parameter Value Horizontal size of picture ≤ 768 Vertical size of picture ≤ 576 No. of MBs / picture ≤ 396 No. of MBs / second ≤ 9, 900 Frame rate ≤ 30 fps Bit-rate ≤ 1, 856 kbps.

Quantization: – MPEG-1 quantization uses different quantization tables for its Intra and Inter coding

For DCT coefficients in Intra mode: QDCT[i, j] = round \( 8 \times DCT[i, j] \) step size[i, j] " = round \( 8 \times DCT[i, j] Q1[i, j] \times \text{scale}" (11.1)

DCT coefficients in Inter mode, QDCT[i, j] = \# \( 8 \times DCT[i, j] \) step size[i, j] $\# = \# \times DCT[i, j] Q2[i, j] \times \text{scale}$

MPEG-1 allows motion vectors to be of sub-pixel precision (1/2 pixel). The technique of “bilinear interpolation” for H.263 can be used to generate the needed values at halfpixel locations.

• Compared to the maximum range of ±15 pixels for motion vectors in H.261, MPEG-1 supports a range of [−512, 511.5] for half-pixel precision and [−1, 024, 1, 023] for full-pixel precision motion vectors

• The MPEG-1 bitstream allows random access — accomplished by GOP layer in which each GOP is time coded.

• The typical size of compressed P-frames is significantly smaller than that of I-frames — because temporal redundancy is exploited in inter-frame compression. • B-frames are even smaller than P-frames — because of (a) the advantage of bi-directional prediction and (b) the lowest priority given to B-frames.

**MPEG-2**: For higher quality video at a bit-rate of more than 4 Mbps. • Defined seven profiles aimed at different applications: – Simple, Main, SNR scalable, Spatially scalable, High, 4:2:2, Multiview. – Within each profile, up to four levels are defined (Table 11.5). – The DVD video specification allows only four
Supporting Interlaced Video

- MPEG-2 must support interlaced video as well since this is one of the options for digital broadcast TV and HDTV.

In interlaced video each frame consists of two fields, referred to as the top-field and the bottom-field. — In a Frame-picture, all scanlines from both fields are interleaved to form a single frame, then divided into 16×16 macroblocks and coded using MC. — If each field is treated as a separate picture, then it is called Field-picture.

MPEG-2 defines Frame Prediction and Field Prediction as well as five prediction modes:

1. Frame Prediction for Frame-pictures: Identical to MPEG-1 MC-based prediction methods in both P-frames and B-frames.

2. Field Prediction for Field-pictures: A macroblock size of 16 × 16 from Field-pictures is used.

3. Field Prediction for Frame-pictures: The top-field and bottom-field of a Frame-picture are treated separately. Each 16 × 16 macroblock (MB) from the target Frame-picture is split into two .

4. 16 × 8 parts, each coming from one field. Field prediction is carried out for these 16 × 8 parts in a manner similar to that shown in Fig. 11.6(b). 4. 16 × 8 MC for Field-pictures: Each 16 × 16 macroblock (MB) from the target Field-picture is split into top and bottom 16 × 8 halves. Field prediction is performed on each half. This generates two motion vectors for each 16 × 16 MB in the P-Field-picture, and up to four motion vectors for each MB in the B-Field-picture. This mode is good for a finer MC when motion is rapid and irregular.

5. Dual-Prime for P-pictures: First, Field prediction from each previous field with the same parity (top or bottom) is made. Each motion vector mv is then used to derive a calculated motion vector cv in the field with the opposite parity taking into account the temporal scaling and vertical shift between lines in the top and bottom fields. For each MB the pair mv and cv yields two preliminary predictions. Their prediction errors are averaged and used as the final prediction error. This mode mimics B-picture
prediction for P-pictures without adopting backward prediction (and hence with less encoding delay). This is the only mode that can be used for either Framepictures or Field-pictures.

5. Discuss the method of constructing the masking function based on maximum variance and maximum magnitude. (NOV/DEC 2012)

A mask Processing Methods - A pixel’s value is computed from its old value and the values of pixels in its vicinity. - More costly operations than simple point processes, but more powerful.

A mask is a small matrix whose values are called weights.

Each mask has an origin, which is usually one of its positions.

The origins of symmetric masks are usually their center pixel position.

- For nonsymmetric masks, any pixel location may be chosen as the origin (depending on the intended use).

Applying Masks to Images (filtering) –

application of a mask to an input image produces an output image of the same size as the input.

Convolution

(1) For each pixel in the input image, the mask is conceptually placed on top of the image with its origin lying on that pixel.

(2) The values of each input image pixel under the mask are multiplied by the values of the corresponding mask weights.

(3) The results are summed together to yield a single output value that is placed in the output image at the location of the pixel being processed on the input.

Cross Correlation - Correlation translates the mask directly to the image without flipping it. - It is often used in applications where it is necessary to measure the similarity between images or parts of images. - If the mask is symmetric (i.e., the flipped mask is the same as the original one) then the results of convolution and correlation are the same.

Normalization of mask weights - The sum of weights in the convolution mask affect the overall intensity of the resulting image. - Many convolution masks have coefficients that sum to 1 (the convolved image will have the same average intensity as the original one)

Some masks have negative weights and sum to 0. - Pixels with negative values may be generated using masks with negative weights. - Negative values are mapped to the positive range through appropriate normalization.

Sharpening (or High-pass) - It is used to emphasize the fine details of an image (has the opposite effect of smoothing). - Points of high contrast can be detected by computing intensity differences in local image
regions. - The weights of the mask are both positive and negative. When the mask is over an area of constant or slowly varying gray level, the result of convolution will be close to zero. When gray level is varying rapidly within the neighborhood, the result of convolution will be a large number. Typically, such points form the border between different objects or scene parts (i.e., sharpening is a precursor step to edge detection).