13.5 Quantization and Coding of Transform Coefficients

If the amount of information conveyed by each coefficient is different, it makes sense to assign differing numbers of bits to the different coefficients. There are two approaches to assigning bits. One approach relies on the average properties of the transform coefficients, while the other approach assigns bits as needed by individual transform coefficients.

In the first approach, we first obtain an estimate of the variances of the transform coefficients. These estimates can be used by one of two algorithms to assign the number of bits used to quantize each of the coefficients. We assume that the relative variance of the coefficients corresponds to the amount of information contained in each coefficient. Thus, coefficients with higher variance are assigned more bits than coefficients with smaller variance.

Let us find an expression for the distortion, then find the bit allocation that minimizes the distortion. To perform the minimization we will use the method of Lagrange [189]. If the average number of bits per sample to be used by the transform coding system is R, and the average number of bits per sample used by the kth coefficient is R_k, then

$$R = \frac{1}{M}\sum_{k=1}^{M} R_k \tag{13.51}$$

where M is the number of transform coefficients. The reconstruction error variance for the kth quantizer $\sigma_{r_k}^2$ is related to the kth quantizer input variance $\sigma_{\theta_k}^2$ by the following:

$$\sigma_{r_k}^2 = \alpha_k 2^{-2R_k}\sigma_{\theta_k}^2 \tag{13.52}$$

where α_k is a factor that depends on the input distribution and the quantizer.

The total reconstruction error is given by

$$\sigma_r^2 = \sum_{k=1}^{M} \alpha_k 2^{-2R_k}\sigma_{\theta_k}^2. \tag{13.53}$$

The objective of the bit allocation procedure is to find R_k to minimize (13.53) subject to the constraint of (13.51). If we assume that α_k is a constant α for all k, we can set up the minimization problem in terms of Lagrange multipliers as

$$J = \alpha\sum_{k=1}^{M} 2^{-2R_k}\sigma_{\theta_k}^2 - \lambda\left(R - \frac{1}{M}\sum_{k=1}^{M} R_k\right). \tag{13.54}$$

Taking the derivative of J with respect to R_k and setting it equal to zero, we can obtain this expression for R_k:

$$R_k = \frac{1}{2}\log_2\left(2\alpha\ln 2\sigma_{\theta_k}^2\right) - \frac{1}{2}\log_2\lambda. \tag{13.55}$$

Substituting this expression for R_k in (13.51), we get a value for λ:

$$\lambda = \prod_{k=1}^{M}\left(2\alpha\ln 2\sigma_{\theta_k}^2\right)^{\frac{1}{M}} 2^{-2R}. \tag{13.56}$$

Substituting this expression for λ in (13.55), we finally obtain the individual bit allocations:

$$R_k = R + \frac{1}{2}\log_2 \frac{\sigma^2_{\theta_k}}{\prod_{k=1}^{M}(\sigma^2_{\theta_k})^{\frac{1}{M}}}. \tag{13.57}$$

Although these values of R_k will minimize (13.53), they are not guaranteed to be integers, or even positive. The standard approach at this point is to set the negative R_ks to zero. This will increase the average bit rate above the constraint. Therefore, the nonzero R_ks are uniformly reduced until the average rate is equal to R.

The second algorithm that uses estimates of the variance is a recursive algorithm and functions as follows:

1. Compute $\sigma^2_{\theta_k}$ for each coefficient.
2. Set $R_k = 0$ for all k and set $R_b = MR$, where R_b is the total number of bits available for distribution.
3. Sort the variances $\{\sigma^2_{\theta_k}\}$. Suppose $\sigma^2_{\theta_l}$ is the maximum.
4. Increment R_l by 1, and divide $\sigma^2_{\theta_l}$ by 2.
5. Decrement R_b by 1. If $R_b = 0$, then stop; otherwise, go to 3.

If we follow this procedure, we end up allocating more bits to the coefficients with higher variance.

This form of bit allocation is called *zonal sampling*. The reason for this name can be seen from the example of a bit allocation map for the 8 × 8 DCT of an image shown in Table 13.4. Notice that there is a zone of coefficients that roughly comprises the right lower diagonal of the bit map that has been assigned zero bits. In other words, these coefficients are to be discarded. The advantage to this approach is its simplicity. Once the bit allocation has been obtained, every coefficient at a particular location is always quantized using the same number of bits. The disadvantage is that, because the bit allocations are performed based on average value, variations that occur on the local level are not reconstructed properly. For example, consider an image of an object with sharp edges in front of a relatively plain background. The number of pixels that occur on edges is quite small compared to the total number of pixels. Therefore, if we allocate bits based on average variances, the coefficients that are important for representing edges (the high-frequency coefficients) will get few or

TABLE 13.4 Bit allocation map for an 8 × 8 transform.

8	7	5	3	1	1	0	0
7	5	3	2	1	0	0	0
4	3	2	1	1	0	0	0
3	3	2	1	1	0	0	0
2	1	1	1	0	0	0	0
1	1	0	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

no bits assigned to them. This means that the reconstructed image will not contain a very good representation of the edges.

This problem can be avoided by using a different approach to bit allocation known as *threshold coding* [190, 93, 191]. In this approach, which coefficient to keep and which to discard is not decided a priori. In the simplest form of threshold coding, we specify a threshold value. Coefficients with magnitude below this threshold are discarded, while the other coefficients are quantized and transmitted. The information about which coefficients have been retained is sent to the receiver as side information. A simple approach described by Pratt [93] is to code the first coefficient on each line regardless of the magnitude. After this, when we encounter a coefficient with a magnitude above the threshold value, we send two codewords: one for the quantized value of the coefficient, and one for the count of the number of coefficients since the last coefficient with magnitude greater than the threshold. For the two-dimensional case, the block size is usually small, and each "line" of the transform is very short. Thus, this approach would be quite expensive. Chen and Pratt [191] suggest scanning the block of transformed coefficients in a zigzag fashion, as shown in Figure 13.7. If we scan an 8 × 8 block of quantized transform coefficients in this manner, we will find that in general a large section of the tail end of the scan will consist of zeros. This is because

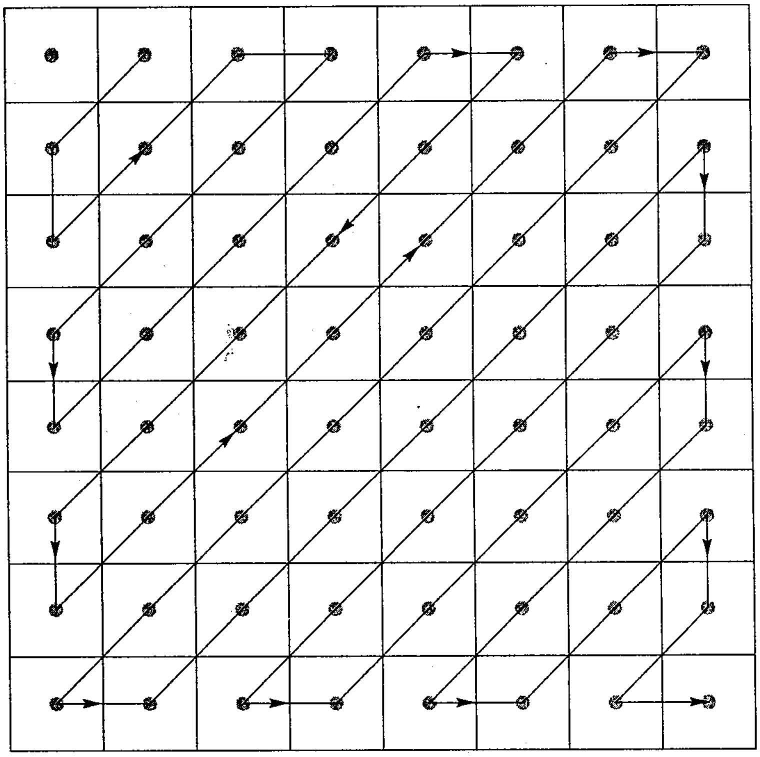

FIGURE 13. 7 The zigzag scanning pattern for an 8 × 8 transform.

generally the higher-order coefficients have smaller amplitude. This is reflected in the bit allocation table shown in Table 13.4. As we shall see later, if we use midtread quantizers (quantizers with a zero output level), combined with the fact that the step sizes for the higher-order coefficients are generally chosen to be quite large, this means that many of these coefficients will be quantized to zero. Therefore, there is a high probability that after a few coefficients along the zigzag scan, all coefficients will be zero. In this situation, Chen and Pratt suggest the transmission of a special *end-of-block* (EOB) symbol. Upon reception of the EOB signal, the receiver would automatically set all remaining coefficients along the zigzag scan to zero.

The algorithm developed by the Joint Photographic Experts Group (JPEG), described in the next section, uses a rather clever variation of this approach.

13.6 Application to Image Compression—JPEG

The JPEG standard is one of the most widely known standards for lossy image compression. It is a result of the collaboration of the International Standards Organization (ISO), which is a private organization, and what was the CCITT (now ITU-T), a part of the United Nations. The approach recommended by JPEG is a transform coding approach using the DCT. The approach is a modification of the scheme proposed by Chen and Pratt [191]. In this section we will briefly describe the baseline JPEG algorithm. In order to illustrate the various components of the algorithm, we will use an 8 × 8 block of the Sena image, shown in Table 13.5. For more details, see [10].

13.6.1 The Transform

The transform used in the JPEG scheme is the DCT transform described earlier. The input image is first "level shifted" by 2^{P-1}; that is, we subtract 2^{P-1} from each pixel value, where P is the number of bits used to represent each pixel. Thus, if we are dealing with 8-bit images whose pixels take on values between 0 and 255, we would subtract 128 from each pixel so that the value of the pixel varies between -128 and 127. The image is divided into blocks of size 8 × 8, which are then transformed using an 8 × 8 forward DCT. If any dimension of the image is not a multiple of eight, the encoder replicates the last column or row until the

TABLE 13.5 An 8 × 8 block from the Sena image.

124	125	122	120	122	119	117	118
121	121	120	119	119	120	120	118
126	124	123	122	121	121	120	120
124	124	125	125	126	125	124	124
127	127	128	129	130	128	127	125
143	142	143	142	140	139	139	139
150	148	152	152	152	152	150	151
156	159	158	155	158	158	157	156