Compressing a 1D Discrete Signal

- Divide the signal into 1×8 blocks.
- Subtract the sample mean from each value.
- Compute the 8×8 covariance matrix for the blocks.
- Compute the eigenvectors of the covariance matrix.
- Compute the KL transform of each block.
- Threshold each transformed block to create long runs of zeros.
- Compute differences of KL transform coefficients in raster scan order.
- Run length encode differences.
- Huffman-code run-lengths.
- Huffman-code labels.

Average Over all Initial Positions

Given a 1D discrete signal $\mathbf{f}$ of length N, construct an 8×8 covariance matrix for the 8 consecutive values starting at all initial positions n:

$$\mathbf{C} = \frac{1}{N} \sum_{n=0}^{N-1} \left[f_{n+0} \cdots \; f_{n+7} \right]^{\mathrm{T}} \left[f_{n+0} \; \cdots \; f_{n+7} \right]$$

$$= \begin{bmatrix} \langle f_{n+0} f_{n+0} \rangle & \langle f_{n+0} f_{n+1} \rangle & \cdots & \langle f_{n+0} f_{n+7} \rangle \\ \langle f_{n+1} f_{n+0} \rangle & \langle f_{n+1} f_{n+1} \rangle & \cdots & \langle f_{n+1} f_{n+7} \rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle f_{n+7} f_{n+0} \rangle & \langle f_{n+7} f_{n+1} \rangle & \cdots & \langle f_{n+7} f_{n+7} \rangle \end{bmatrix}.$$

Because the covariance matrix is an average over all initial positions, there can be no dependence on n:

$$\langle f_{n+i} f_{n+j} \rangle = \langle f_{m+i} f_{m+j} \rangle .$$

It follows that $\mathbf{C}$ is *circulant* and the eigenvectors of $\mathbf{C}$ are sampled harmonic signals!

Compressing a 1D Discrete Signal (revised)

- Divide the signal into 1×8 blocks.
- Subtract the sample mean from each value.
- Compute DFT of each block.
- Threshold each transformed block to create long runs of zeros.
- Compute differences of thresholded DFT coefficients in raster scan order.
- Run length encode differences.
- Huffman-code run-lengths.
- Huffman-code labels.

Symmetric Extension

Given a vector **f** of length N construct a vector **g** of length $2N$ such that

$$g(n) = \begin{cases} f(n) & \text{if } 0 \leq n \leq N-1 \\ f(2N-n-1) & \text{if } N \leq n \leq 2N-1. \end{cases}$$

The DFT of **g** shifted by one-half is:

$$G(m) = \frac{1}{2N} \sum_{n=0}^{2N-1} g(n) \exp\left(\frac{-j\pi(n+\frac{1}{2})m}{N}\right)$$

for $0 \leq m \leq 2N-1$. Recall that

$$\exp\left(\frac{-j\pi(n+\frac{1}{2})m}{N}\right) = \cos\left(\frac{\pi(n+\frac{1}{2})m}{N}\right) + j\sin\left(\frac{\pi(n+\frac{1}{2})m}{N}\right).$$

Now, because **g** is symmetric:

$$\frac{1}{2N} \sum_{n=0}^{2N-1} g(n)\, j\sin\left(\frac{\pi(n+\frac{1}{2})m}{N}\right) = 0.$$

Symmetric Extension (contd.)

Consequently

$$G(m) = \frac{1}{2N} \sum_{n=0}^{2N-1} g(n) \cos\left(\frac{\pi(n+\frac{1}{2})m}{N}\right).$$

Dividing the sum into two and substituting $f(n)$ for $g(n)$ in the first sum and $f(2N-n-1)$ for $g(n)$ in the second yields

$$G(m) = \frac{1}{2N} \sum_{n=0}^{N-1} f(n) \cos\left(\frac{\pi(n+\frac{1}{2})m}{N}\right) +$$

$$\frac{1}{2N} \sum_{n=N}^{2N-1} f(2N-n-1) \cos\left(\frac{\pi(n+\frac{1}{2})m}{N}\right).$$

Symmetric Extension (contd.)

Let's look at the range in the second sum:

$$\frac{1}{2N}\sum_{n=N}^{2N-1} f(2N-n-1)\cos\left(\frac{\pi(n+\frac{1}{2})m}{N}\right).$$

When $n = N$:

$$\begin{aligned} f(2N-n-1)\cos\left(\frac{\pi(n+\frac{1}{2})m}{N}\right) &= f(N-1)\cos\left(\frac{\pi(N+\frac{1}{2})m}{N}\right) \\ &= f(N-1)\cos\left(\frac{\pi(N-\frac{1}{2})m}{N}\right) \\ &= f(N-1)\cos\left(\frac{\pi((N-1)+\frac{1}{2})m}{N}\right). \end{aligned}$$

When $n = 2N-1$:

$$\begin{aligned} f(2N-n-1)\cos\left(\frac{\pi(n+\frac{1}{2})m}{N}\right) &= f(0)\cos\left(\frac{\pi((2N-1)+\frac{1}{2})m}{N}\right) \\ &= f(0)\cos\left(\frac{\pi(0-\frac{1}{2})m}{N}\right) \\ &= f(0)\cos\left(\frac{\pi(0+\frac{1}{2})m}{N}\right). \end{aligned}$$

Symmetric Extension (contd.)

It follows that the two sums are equal:

$$\frac{1}{2N}\sum_{n=N}^{2N-1} f(2N-n-1)\cos\left(\frac{\pi(n+\frac{1}{2})m}{N}\right) =$$

$$\frac{1}{2N}\sum_{n=0}^{N-1} f(n)\cos\left(\frac{\pi(n+\frac{1}{2})m}{N}\right).$$

Consequently

$$\begin{aligned} G(m) &= \frac{2}{2N}\sum_{n=0}^{N-1} f(n)\cos\left(\frac{\pi(n+\frac{1}{2})m}{N}\right) \\ &= \frac{1}{N}\sum_{n=0}^{N-1} f(n)\cos\left(\frac{\pi(n+\frac{1}{2})m}{N}\right). \end{aligned}$$

Discrete Cosine Transform

The *Discrete Cosine Transform* is defined as follows:

$$F(m) = \frac{1}{N}\sum_{n=0}^{N-1} f(n)\cos\left(\frac{\pi(n+\frac{1}{2})m}{N}\right).$$

- Because the DCT is a real valued transform there is no need for complex arithmetic.
- Unlike **f** which had a discontinuity at $f(0)$, **g** is continuous at both $g(0)$ and $g(N)$!
- For this reason, the energy in the DCT representation of a function is localized in fewer non-zero values.

Periodicity of DFT and DCT

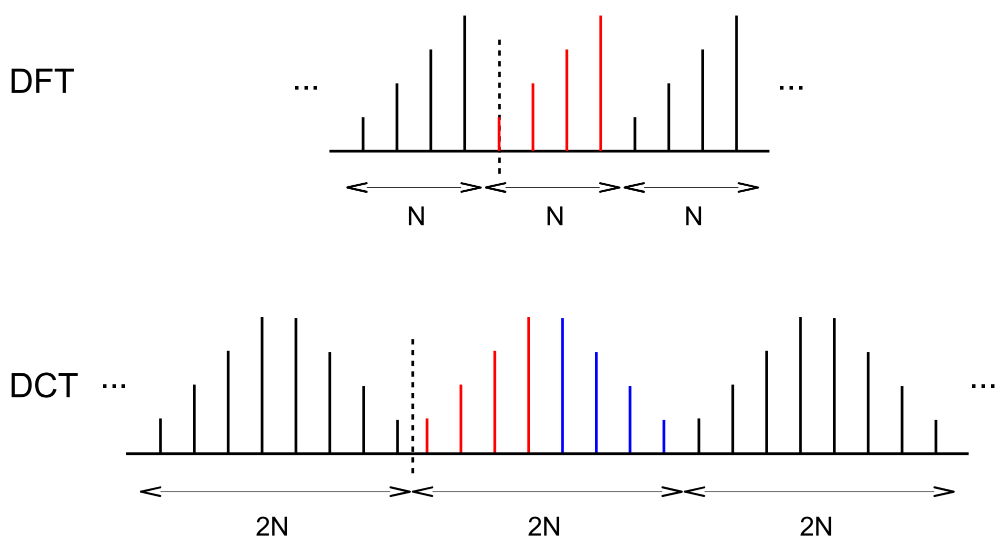

Figure 1: The DCT is the unique half of the real part of the DFT of a length $2N$ symmetric extension of a length N signal shifted by one-half.

Periodicity of 2D DFT and DCT

Figure 2: Periodicity of the 2D Discrete Fourier Transform (DFT).

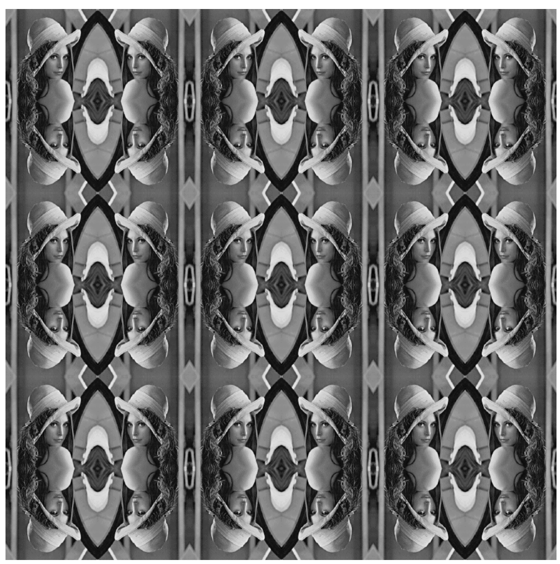

Figure 3: Periodicity of 2D Discrete Cosine Transform (DCT).

JPEG[1]

- Divide image into 8×8 blocks.
- Center gray values (i.e., subtract 128).
- Compute the DCT of each block.
- Quantize DCT coefficients using quantization table.
- Compute differences of quantized DCT coefficients along zigzag path.
- Run-length encode differences.
- Huffman-code run-lengths.
- Huffman-code labels.

[1] Joint Photographic Experts Group

Example Block

$$\begin{bmatrix} 52 & 55 & 61 & 66 & 70 & 61 & 64 & 73 \\ 63 & 59 & 55 & 90 & 109 & 85 & 69 & 72 \\ 62 & 59 & 68 & 113 & 144 & 104 & 66 & 73 \\ 63 & 58 & 71 & 122 & 154 & 106 & 70 & 69 \\ 67 & 61 & 68 & 104 & 126 & 88 & 68 & 70 \\ 79 & 65 & 60 & 70 & 77 & 68 & 58 & 75 \\ 85 & 71 & 64 & 59 & 55 & 61 & 65 & 83 \\ 87 & 79 & 69 & 68 & 65 & 76 & 78 & 94 \end{bmatrix}$$

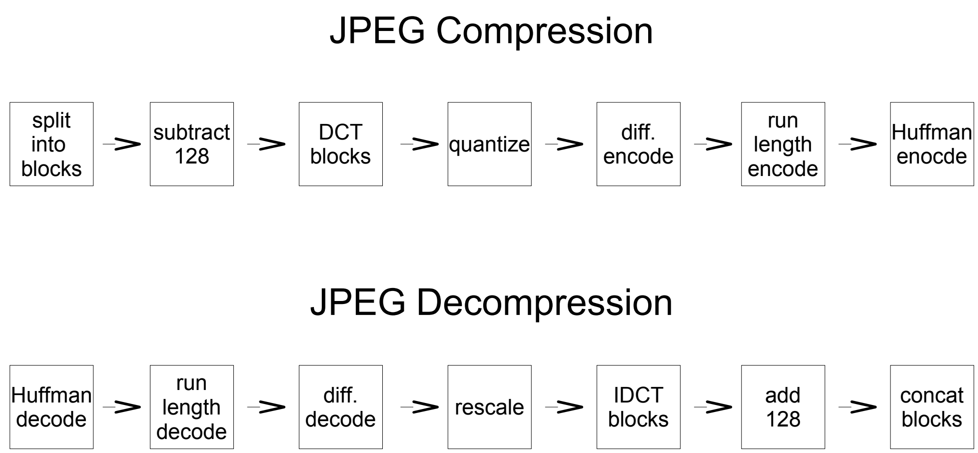

Figure 4: JPEG compression (top) and decompression (bottom) algorithms.

Subtracting 128

$$f = \begin{bmatrix} -76 & -73 & -67 & -62 & -58 & -67 & -64 & -55 \\ -65 & -69 & -73 & -38 & -19 & -43 & -59 & -56 \\ -66 & -69 & -60 & -15 & 16 & -24 & -62 & -55 \\ -65 & -70 & -57 & -6 & 26 & -22 & -58 & -59 \\ -61 & -67 & -60 & -24 & -2 & -40 & -60 & -58 \\ -49 & -63 & -68 & -58 & -51 & -60 & -70 & -53 \\ -43 & -57 & -64 & -69 & -73 & -67 & -63 & -45 \\ -41 & -49 & -59 & -60 & -63 & -52 & -50 & -34 \end{bmatrix}$$

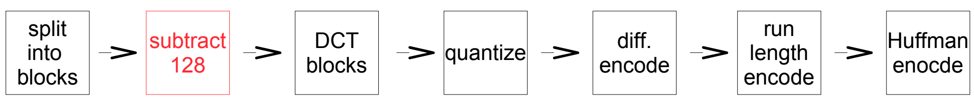

Figure 5: Subtracting 128 from blocks.

DCT of 8×8 Block

$$F(u,v) = \frac{1}{4}\sum_{x=0}^{7}\sum_{y=0}^{7} f(x,y)\cos\left(\frac{\pi(2x+1)u}{16}\right)\cos\left(\frac{\pi(2y+1)v}{16}\right)$$

Figure 6: Discrete cosine transform of blocks.

$$F = \begin{bmatrix} -415.38 & -30.19 & -61.20 & 27.24 & 56.12 & -20.10 & -2.39 & 0.46 \\ 4.47 & -21.86 & -60.76 & 10.25 & 13.15 & -7.09 & -8.54 & 4.88 \\ -46.83 & 7.37 & 77.13 & -24.56 & -28.91 & 9.93 & 5.42 & -5.65 \\ -48.53 & 12.07 & 34.10 & -14.76 & -10.24 & 6.30 & 1.83 & 1.95 \\ 12.12 & -6.55 & -13.20 & -3.95 & -1.87 & 1.75 & -2.79 & 3.14 \\ -7.73 & 2.91 & 2.38 & -5.94 & -2.38 & 0.94 & 4.30 & 1.85 \\ -1.03 & 0.18 & 0.42 & -2.42 & -0.88 & -3.02 & 4.12 & -0.66 \\ -0.17 & 0.14 & -1.07 & -4.19 & -1.17 & -0.10 & 0.50 & 1.68 \end{bmatrix}$$

JPEG Basis Functions

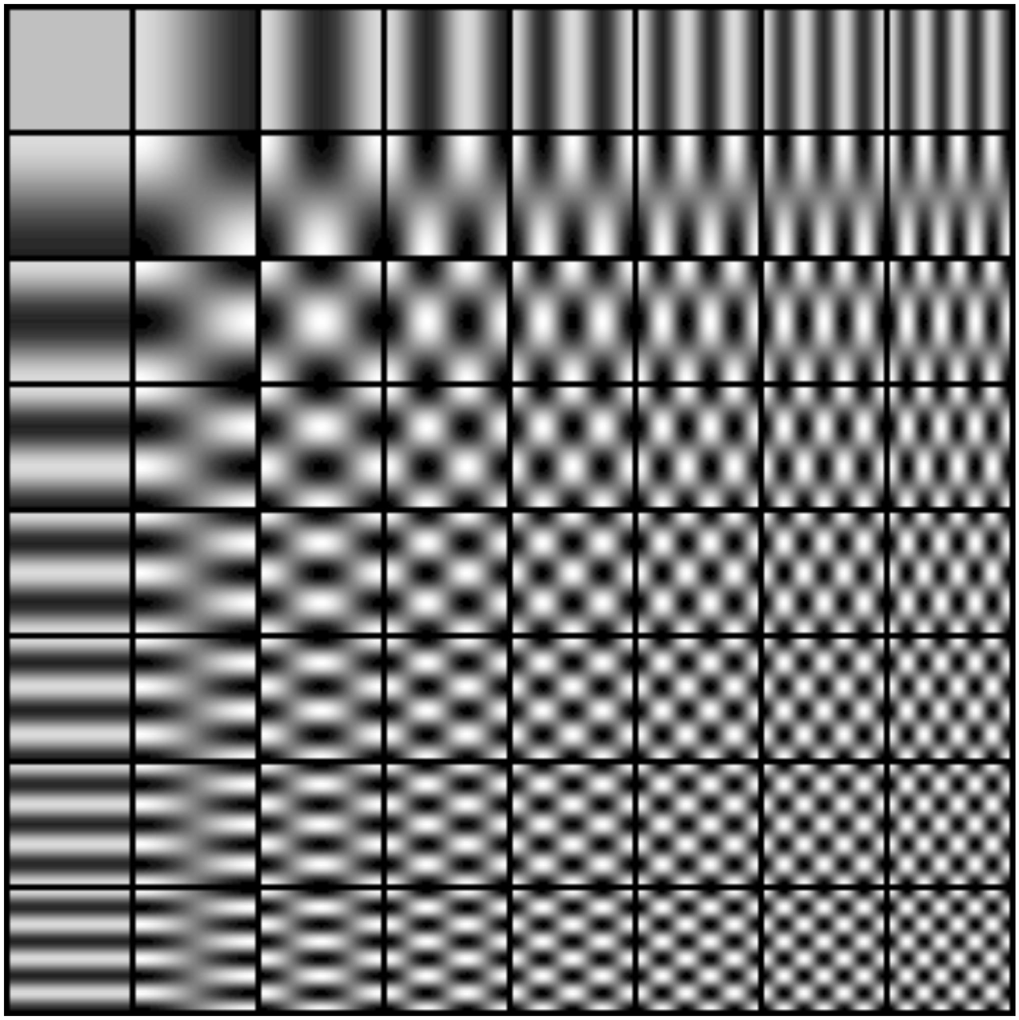

Figure 7: JPEG basis functions.

Block Quantization

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

$$G(u,v) = \left\lfloor \frac{F(u,v)}{Q(u,v)} + 0.5 \right\rfloor$$

Figure 8: Quantize blocks.

$$G = \begin{bmatrix} -26 & -3 & -6 & 2 & 2 & -1 & 0 & 0 \\ 0 & -2 & -4 & 1 & 1 & 0 & 0 & 0 \\ -3 & 1 & 5 & -1 & -1 & 0 & 0 & 0 \\ -3 & 1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Difference Encode along Zigzag Path

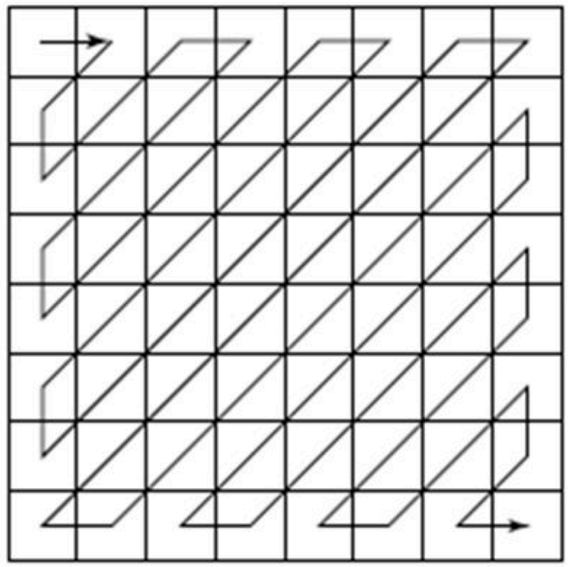

Figure 9: Zigzag scan order.

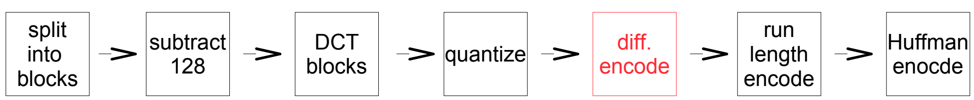

Figure 10: Difference encode blocks.

Table 1: AC Coefficient Encoding Scheme.

4 bits	4 bits	8 bits
zero run length	category	amplitude

Rescaling Block

$$\tilde{F}(u,v) = G(u,v) \cdot Q(u,v)$$

Figure 11: Rescale blocks.

$$\tilde{F} = \begin{bmatrix} -416 & -33 & -60 & 32 & 48 & -40 & 0 & 0 \\ 0 & -24 & -56 & 19 & 26 & 0 & 0 & 0 \\ -42 & 13 & 80 & -24 & -40 & 0 & 0 & 0 \\ -42 & 17 & 44 & -29 & 0 & 0 & 0 & 0 \\ 18 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Inverse DCT

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

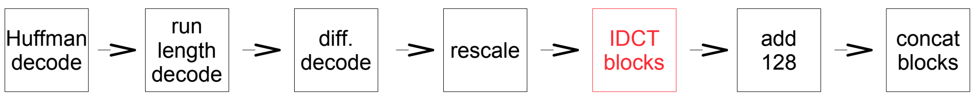

Figure 12: Inverse discrete cosine transform of blocks.

$$\tilde{f} = \begin{bmatrix} -66 & -63 & -71 & -68 & -56 & -65 & -68 & -46 \\ -71 & -73 & -72 & -46 & -20 & -41 & -66 & -57 \\ -70 & -78 & -68 & -17 & 20 & -14 & -61 & -63 \\ -63 & -73 & -62 & -8 & 27 & -14 & -60 & -58 \\ -58 & -65 & -61 & -27 & -6 & -40 & -68 & -50 \\ -57 & -57 & -64 & -58 & -48 & -66 & -72 & -47 \\ -53 & -46 & -61 & -74 & -65 & -63 & -62 & -45 \\ -47 & -34 & -53 & -74 & -60 & -47 & -47 & -41 \end{bmatrix}$$

Reconstruction Errors

$$f - \tilde{f} = \begin{bmatrix} -10 & -10 & 4 & 6 & -2 & -2 & 4 & -9 \\ 6 & 4 & -1 & 8 & 1 & -2 & 7 & 1 \\ 4 & 9 & 8 & 2 & -4 & -10 & -1 & 8 \\ -2 & 3 & 5 & 2 & -1 & -8 & 2 & -1 \\ -3 & -2 & 1 & 3 & 4 & 0 & 8 & -8 \\ 8 & -6 & -4 & -0 & -3 & 6 & 2 & -6 \\ 10 & -11 & -3 & 5 & -8 & -4 & -1 & -0 \\ 6 & -15 & -6 & 14 & -3 & -5 & -3 & 7 \end{bmatrix}$$

Adding 128

$$\begin{bmatrix} 62 & 65 & 57 & 60 & 72 & 63 & 60 & 82 \\ 57 & 55 & 56 & 82 & 108 & 87 & 62 & 71 \\ 58 & 50 & 60 & 111 & 148 & 114 & 67 & 65 \\ 65 & 55 & 66 & 120 & 155 & 114 & 68 & 70 \\ 70 & 63 & 67 & 101 & 122 & 88 & 60 & 78 \\ 71 & 71 & 64 & 70 & 80 & 62 & 56 & 81 \\ 75 & 82 & 67 & 54 & 63 & 65 & 66 & 83 \\ 81 & 94 & 75 & 54 & 68 & 81 & 81 & 87 \end{bmatrix}$$

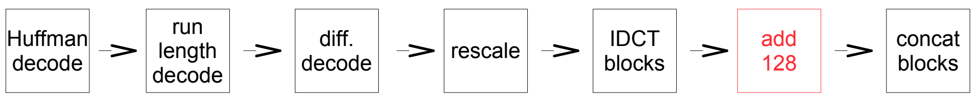

Figure 13: Add 128 to blocks.

Conclusion

- The DCT is really the KL transform of the block in disguise!
- The quantization table tells us which eigenvectors are most important perceptually.
- Compression ratios of 8 to 1 without perceptible differences are typical.