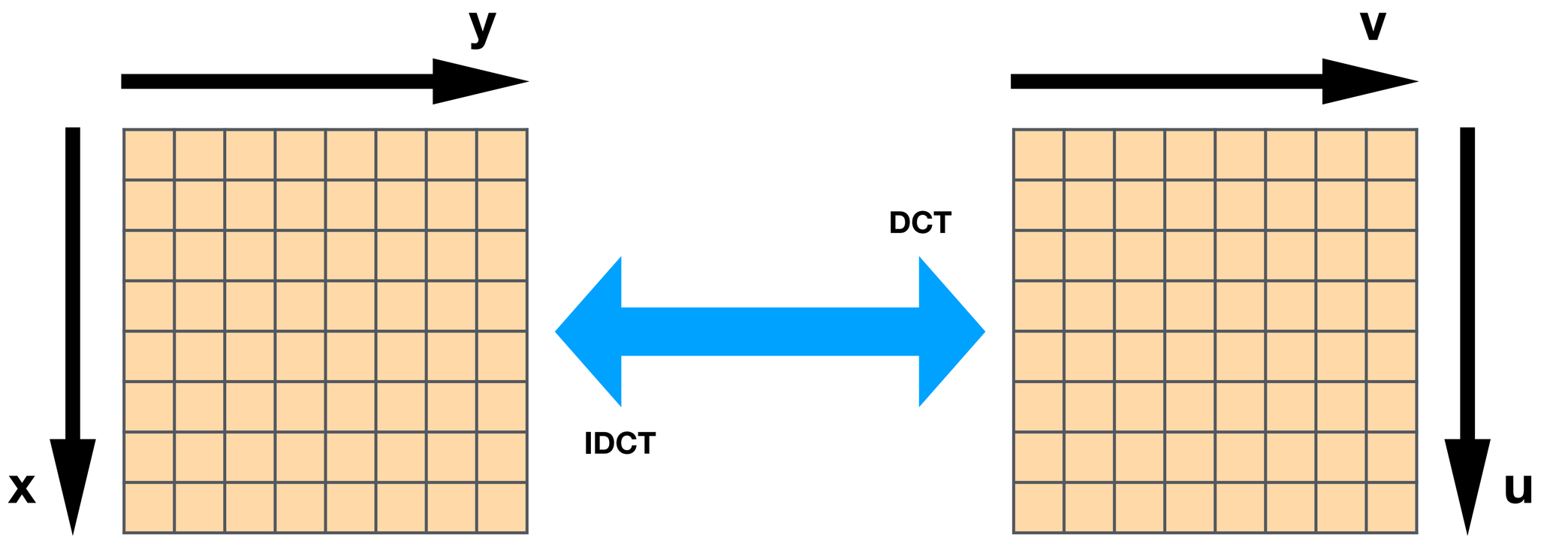
Algoritmo de compressão de imagens: DCT, quantização e RLE

DCT : $$D_{u,v} = \frac{1}{4}\,\alpha(u)\,\alpha(v)\sum_{x=0}^{7}\sum_{y=0}^{7} p_{x,y}\,cos\left[\frac{(2x+1)\,u\pi}{16}\right]cos\left[\frac{(2y+1)\,v\pi}{16}\right]$$

$$\begin{cases}\alpha(u=0) = \frac{1}{\sqrt{2}} \\ \alpha(u\neq 0) = 1\end{cases}$$

IDCT : $$p_{x,y} = \frac{1}{4}\sum_{u=0}^{7}\sum_{v=0}^{7}\alpha(u)\,\alpha(v)\,D_{u,v}\,cos\left[\frac{(2x+1)\,u\pi}{16}\right]cos\left[\frac{(2y+1)\,v\pi}{16}\right]$$



$p_{x,y}$: pixels (de -128 a +127)

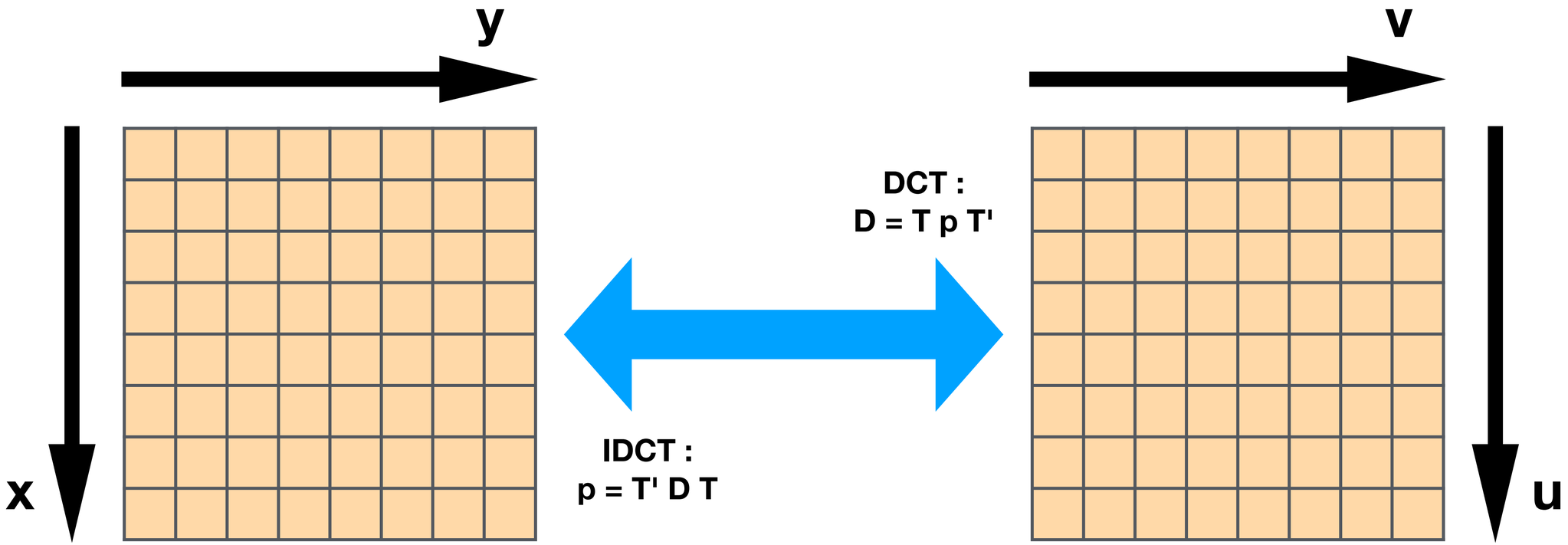
$D_{u,v}$: coeficientes DCT (de -1024 a +1023)

Matriz de transformação T ortogonal :

Para i = 0 : $T_{i,j} = \frac{1}{\sqrt{8}}$

Para i ≠ 0 : $T_{i,j} = \frac{1}{2} \cos\left[\frac{(2j+1)\, i\pi}{16}\right]$

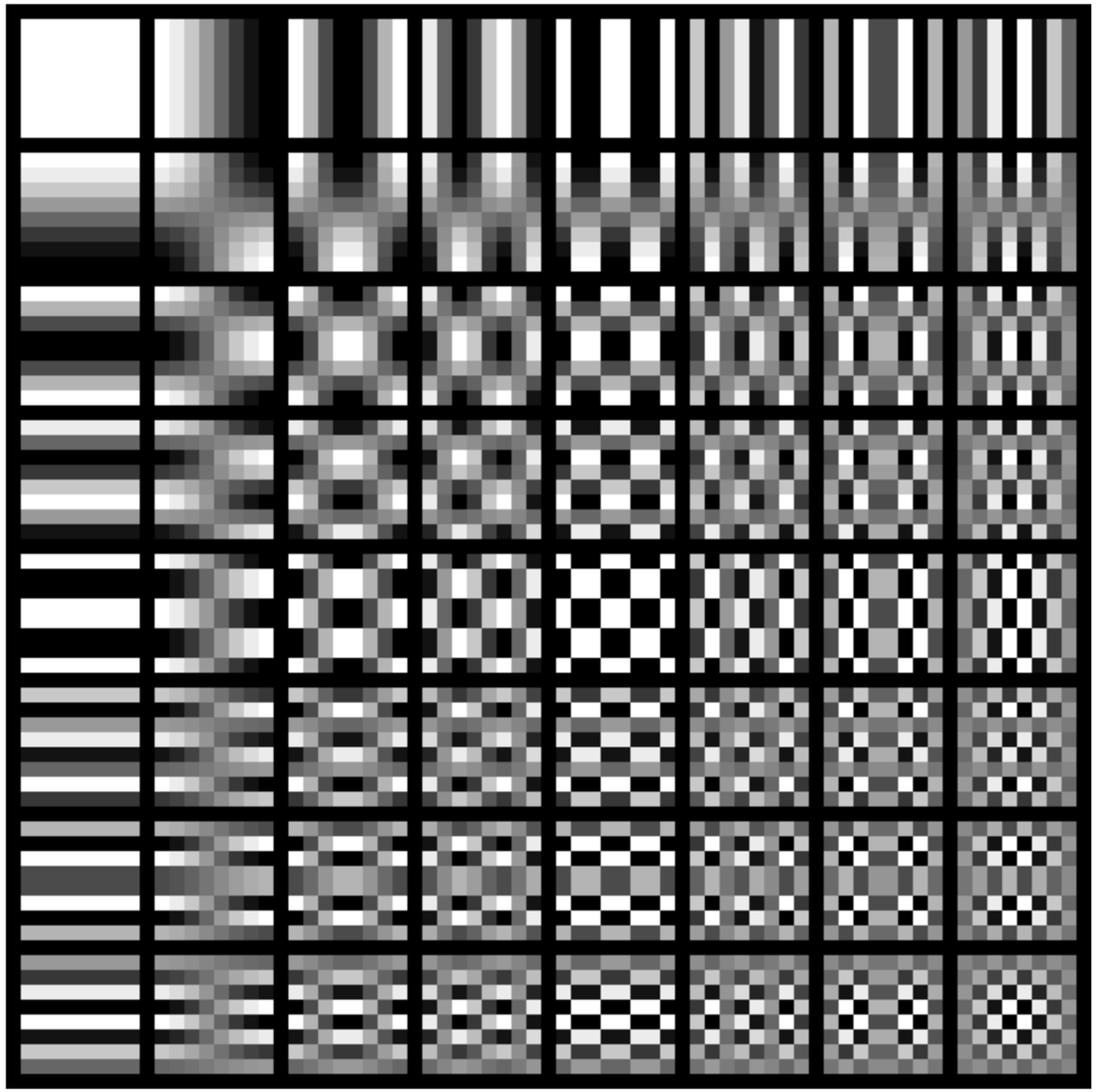
$$T = \begin{bmatrix}
.3536 & .3536 & .3536 & .3536 & .3536 & .3536 & .3536 & .3536 \\
.4904 & .4157 & .2778 & .0975 & -.0975 & -.2778 & -.4157 & -.4904 \\
.4619 & .1913 & -.1913 & -.4619 & -.4619 & -.1913 & .1913 & .4619 \\
.4157 & -.0975 & -.4904 & -.2778 & .2778 & .4904 & .0975 & -.4157 \\
.3536 & -.3536 & -.3536 & .3536 & .3536 & -.3536 & -.3536 & .3536 \\
.2778 & -.4904 & .0975 & .4157 & -.4157 & -.0975 & .4904 & -.2778 \\
.1913 & -.4619 & .4619 & -.1913 & -.1913 & .4619 & -.4619 & .1913 \\
.0975 & -.2778 & .4157 & -.4904 & .4904 & -.4157 & .2778 & -.0975
\end{bmatrix}$$



$p_{x,y}$: pixels (de -128 a +127)

$D_{u,v}$: coeficientes DCT (de -1024 a +1023)

Explicação gráfica dos coeficientes DCT :



64 imagens-base normalizadas para a faixa de 0 a 255

O algoritmo DCT transforma um bloco 8x8 de valores de entrada (pixels) em uma combinação linear das imagens-base (os 64 padrões à esquerda divididos pelos fatores de amplificação).

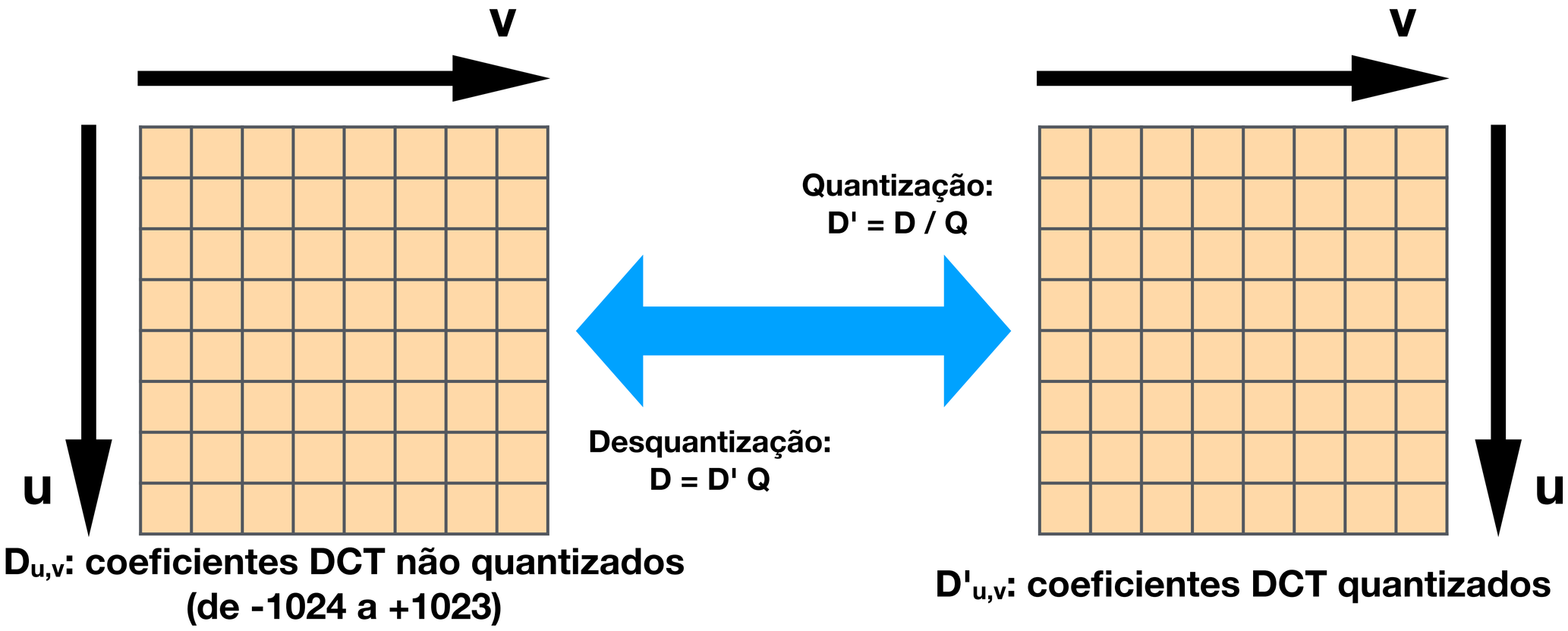
= fatores de amplificação x imagens-base

```
fatores_amplificacao =
[[2040  735  781  735 1020  735  781  735]
 [ 735  530  563  530  735  530  563  530]
 [ 781  563  598  563  781  563  598  563]
 [ 735  530  563  530  735  530  563  530]
 [1020  735  781  735 1020  735  781  735]
 [ 735  530  563  530  735  530  563  530]
 [ 781  563  598  563  781  563  598  563]
 [ 735  530  563  530  735  530  563  530]]
```

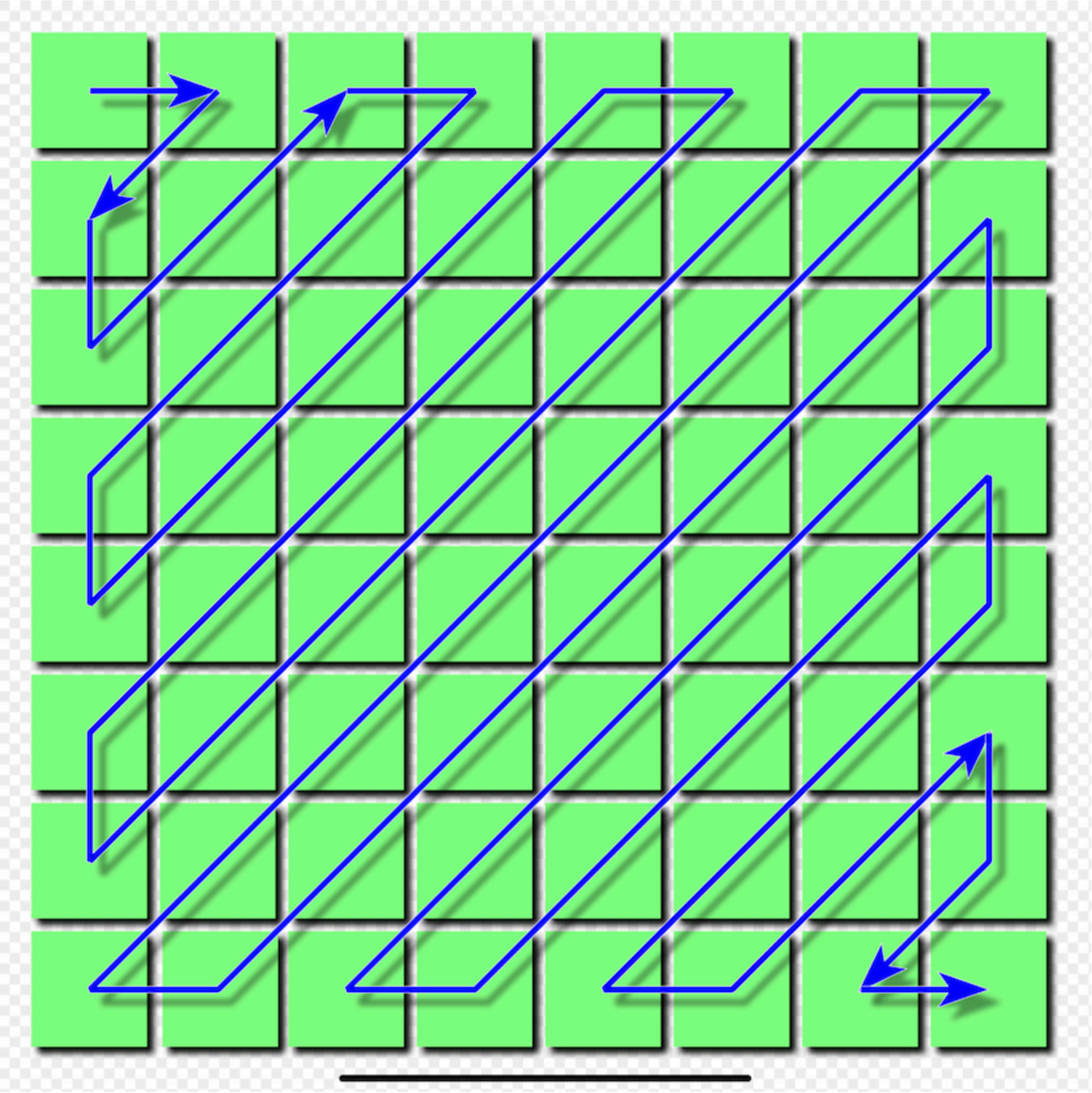
Matriz de quantização Q (p/ qualidade de 1 a 100) :

$$\begin{cases} Q_{(qual<50)} = \left(\frac{50}{qual}\right) Q50 \qquad Q_{(qual>50)} = \left(\frac{100 - qual}{50}\right) Q50 \\ \text{Após o cálculo acima, deve-se arredondar e clipar Q para que Q fique com valores inteiros de 1 a 255} \end{cases}$$

$$Q_{50} = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$



Codificação de entropia com Run-Length Encoding :



fonte: https://en.wikipedia.org/wiki/JPEG

O algoritmo RLE consiste em uma compressão *lossless* (sem perdas) aplicada à matriz de coeficientes DCT quantizados D'.

Percorre-se D' em zigue-zague, iniciando no canto superior esquerdo, até o último elemento não nulo.

Para cada elemento não nulo, usa-se:

- 4 bits denominados RUNLENGTH*
- 4 bits denominados SIZE
- SIZE + 1 bits com o valor do elemento

* O elemento superior esquerdo não usa RUNLENGTH.

Exemplo de DCT e RLE:

```
p2 =
[[-128  127 -128  127 -128  127 -128  127]
 [-128  127 -128  127 -128  127 -128  127]
 [-128  127 -128  127 -128  127 -128  127]
 [-128  127 -128  127 -128  127 -128  127]
 [-128  127 -128  127 -128  127 -128  127]
 [-128  127 -128  127 -128  127 -128  127]
 [-128  127 -128  127 -128  127 -128  127]
 [-128  127 -128  127 -128  127 -128  127]]

p2_dct =
[[  -4 -184    0 -217    0 -325    0 -924]
 [   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    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]]

Aplicação do algoritmo RLE na matriz de DCTs quantizados:

Posição do último elemento não nulo, de 0 a 63 (valor -1 indica que todos são nulos) = 28

(SIZE) (VALUE) = (2) (-4) ===> 7 bits
(RUNLENGTH, SIZE) (VALUE) = (0, 8) (-184) ===> 17 bits
(RUNLENGTH, SIZE) (VALUE) = (4, 8) (-217) ===> 17 bits
(RUNLENGTH, SIZE) (VALUE) = (8, 9) (-325) ===> 18 bits
(RUNLENGTH, SIZE) (VALUE) = (12, 10) (-924) ===> 19 bits
(RUNLENGTH, SIZE) = EOB = (0, 0) ===> 8 bits

Total de bits = 86 bits.
Total de bytes = 11 bytes (88 bits).
```