Course Code 005636 (Fall 2017)

Multimedia

Lossy Compression Algorithms

Prof. S. M. Riazul Islam, Dept. of Computer Engineering, Sejong University, Korea

E-mail: riaz@sejong.ac.kr

Outline

- Concept of Distortion
- Quantitative Analysis of Quantization
  - Uniform Scaler Quantization
  - Non-Uniform Scaler Quantization
  - Vector Quantization
- Discrete Cosine Transform (DCT)

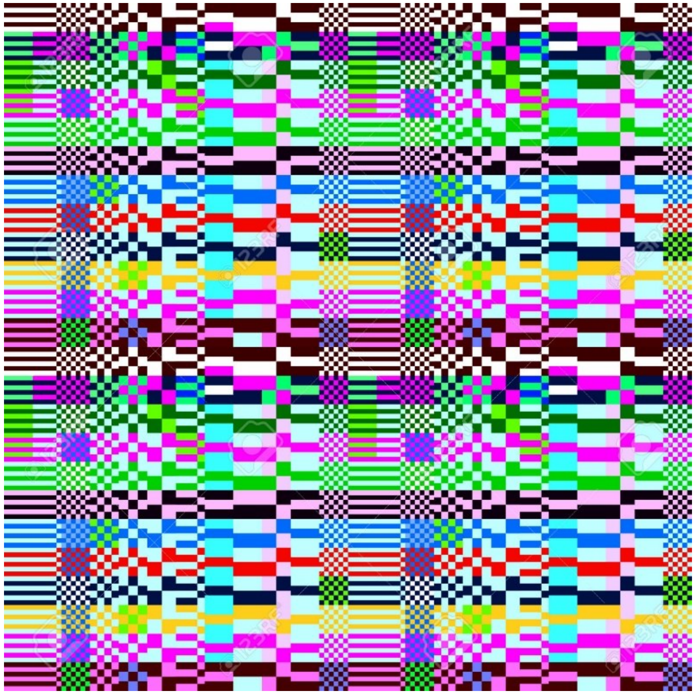
Introduction

- Lossless compression algorithms do not deliver compression ratios that are high enough. Hence, most multimedia compression algorithms are lossy.

- What is lossy compression?
  - The compressed data is not the same as the original data, but a close approximation of it.
  - Yields a much higher compression ratio than that of lossless compression.

Distortion

- A distortion measure is a mathematical quantity that specifies how close an approximation is to its original, using some distortion criteria.



- *Quantitative* vs. *Perceptual Distortion*

Distortion Measures

The three most commonly used distortion measures in image compression are:

- *mean square error* (MSE) σ^2,

$$\sigma^2 = \frac{1}{N}\sum_{n=1}^{N}(x_n - y_n)^2$$

where x_n, y_n, and N are the input data sequence, reconstructed data sequence, and length of the data sequence respectively.

- *signal to noise ratio* (SNR), in decibel units (dB),

$$SNR = 10\log_{10}\frac{\sigma_x^2}{\sigma_d^2}$$

where σ_x^2 is the average square value of the original data sequence and σ_d^2 is the MSE.

- *peak signal to noise ratio* (PSNR),

$$PSNR = 10\log_{10}\frac{x_{peak}^2}{\sigma_d^2}$$

Example

- Suppose, {4, 5, 6, 6, 2} is the reconstructed data sequence of the input data sequence {5, 4, 8, 7, 1} in a lossy compression/decompression system.

  - Calculate the signal-to-noise ratio as measure of the associated distortion.

Do It Now

- Hint

Calculate the signal power

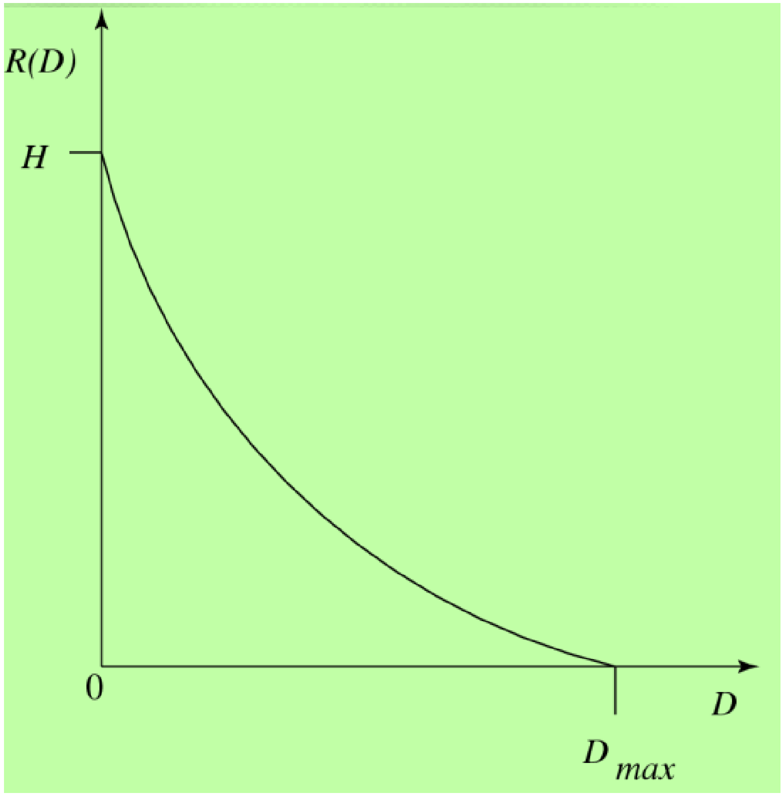
Calculate the MSE

Find the SNR in dB

?? 12.87 dB

The Rate-Distortion Theory

- Provides a framework for the study of tradeoffs between Rate and Distortion. A Typical Rate Distortion Function.



Quantizer

- Reduce the number of distinct output values to a much smaller set.

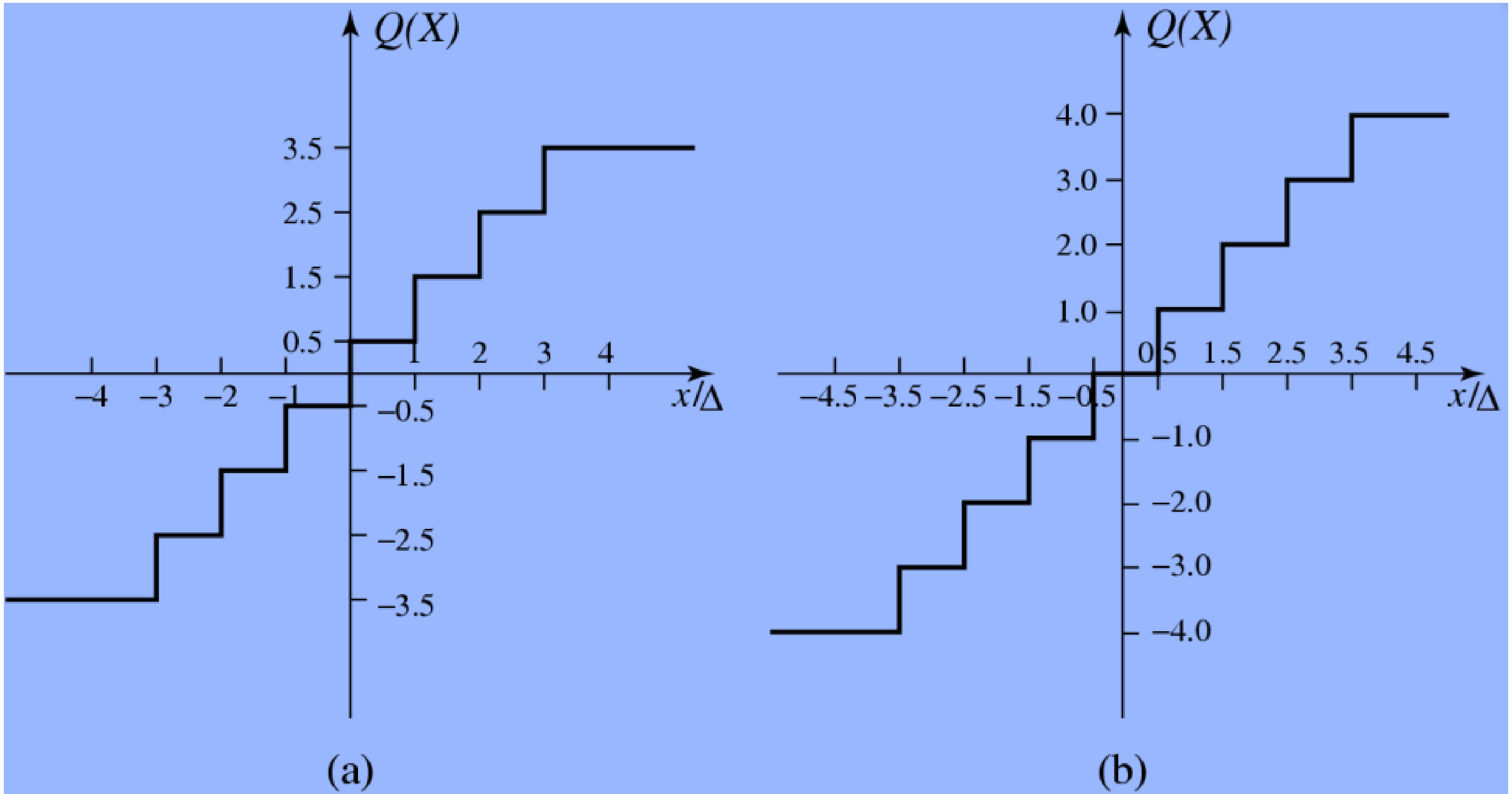
- Main source of the ‘loss’ in lossy compression.

- Three different forms of quantization.
  - Uniform: midrise and midtread quantizers.
  - Nonuniform: companded quantizer.
  - Vector

Quantizer

- A uniform scalar quantizer partitions the domain of input values into equally spaced intervals, except possibly at the two outer intervals.
  - The output or reconstruction value corresponding to each interval is taken to be the midpoint of the interval.
  - The length of each interval is referred to as the step size, denoted by the symbol Δ.
- Two types of uniform scalar quantizers:
  - Midrise quantizers have even number of output levels.
  - Midtread quantizers have odd number of output levels, including zero as one of them

Quantizer



Uniform Scalar Quantizers: (a) Midrise, (b) Midtread.

Uniform Scaler Quantizer

- Design a successful uniform quantizer to minimize the distortion for a given source input with a desired number of output values.

- This can be done by adjusting the step size.

Performance of an M level quantizer. Let $B = \{b_0, b_1, \ldots, b_M\}$ be the set of decision boundaries and $Y = \{y_1, y_2, \ldots, y_M\}$ be the set of reconstruction or output values.

Suppose the input is uniformly distributed in the interval $[-X_{max}, X_{max}]$. The rate of the quantizer is:

$$R = \lceil \log_2 M \rceil$$

Bounded vs. Unbounded Input

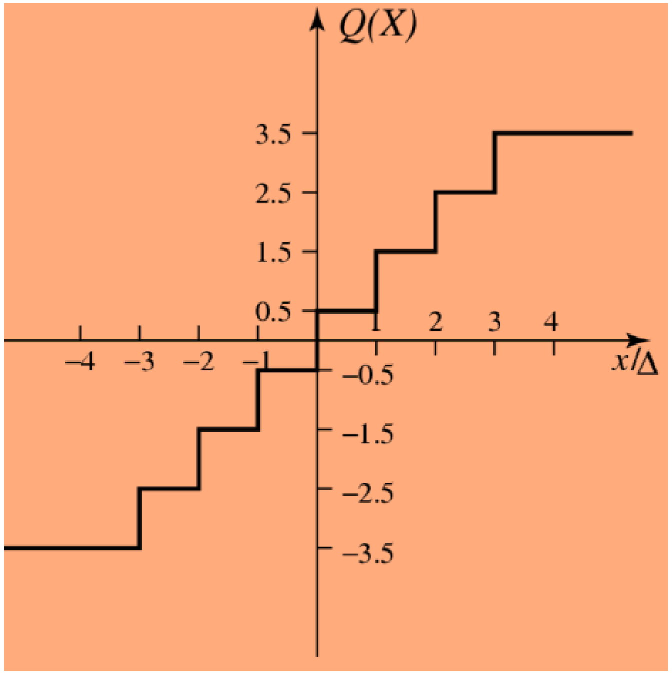
- **Granular Distortion:** the quantization error for bounded input.

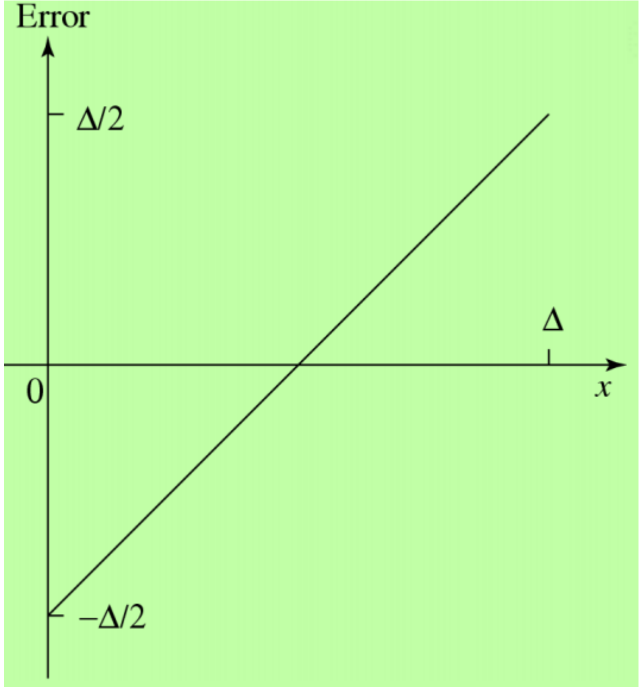
- **Overload Distortion:** If the quantizer replaces a whole range of values, from a maximum value to infinity and similarly for negative values (That part of the distortion)

To get an overall figure for granular distortion, notice that decision boundaries b_i for a midrise quantizer are $[(i-1)\Delta, i\Delta]$, $i = 1..M/2$, covering positive data X (and another half for negative X values).

Output values y_i are the midpoints $i\Delta - \Delta/2$, $i = 1..M/2$, again just considering the positive data. The total distortion is twice the sum over the positive data, or

$$D_{gran} = 2\sum_{i=1}^{\frac{M}{2}} \int_{(i-1)\Delta}^{i\Delta} \left(x - \frac{2i-1}{2}\Delta \right)^2 \frac{1}{2X_{max}}\, dx$$





The error value at x is $e(x) = x - \Delta/2$,

The variance of the error is

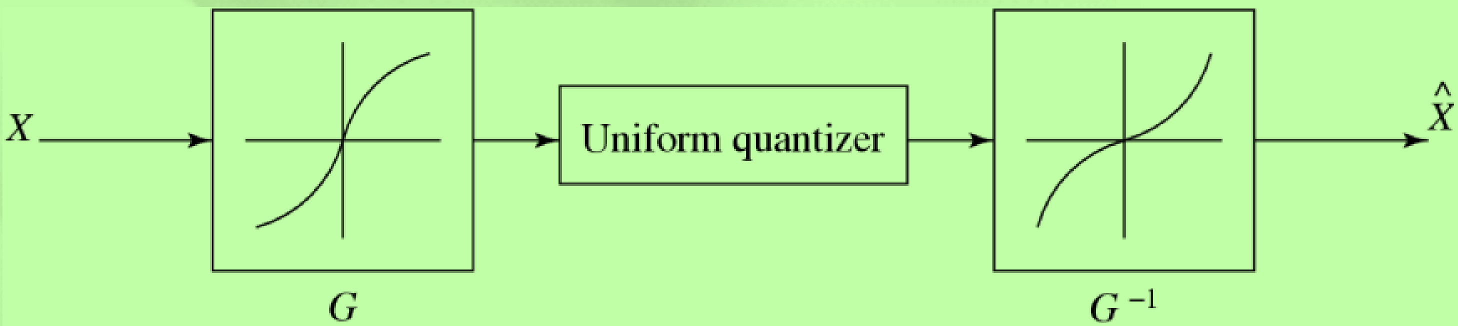
$$\sigma_d^2 = \frac{1}{\Delta}\int_0^{\Delta}\left(x - \frac{\Delta}{2}\right)^2 dx = \frac{\Delta^2}{12}$$

the *signal* variance is $\sigma_x^2 = (2X_{max})^2/12$

$$\begin{aligned}
SQNR &= 10\log_{10}\left(\frac{\sigma_x^2}{\sigma_d^2}\right) \\
&= 10\log_{10}\left(\frac{(2X_{max})^2}{12}\cdot\frac{12}{\Delta^2}\right) \\
&= 10\log_{10}\left(\frac{(2X_{max})^2}{12}\cdot\frac{12}{\left(\frac{2X_{max}}{M}\right)^2}\right) \\
&= 10\log_{10} M^2 = 20\,n\,\log_{10} 2 \\
&= 6.02\,n\ (dB)
\end{aligned}$$

Non-Uniform Scaler Quantizer

Companded quantization is **nonlinear**.



As shown above, a *compander* consists of a *compressor function* G, a uniform quantizer, and an *expander function* G^{-1}.

The two commonly used companders are the μ-law and A-law companders.

$\mu - law$

For a given input x, the equation for μ-law encoding is

$$F(x) = \operatorname{sgn}(x)\frac{\ln(1+\mu|x|)}{\ln(1+\mu)} \quad -1 \leq x \leq 1$$

where $\mu = 255$ (8 bits) in the North American and Japanese standards.

μ-law expansion is then given by the inverse equation:

$$F^{-1}(y) = \operatorname{sgn}(y)(1/\mu)((1+\mu)^{|y|} - 1) \quad -1 \leq y \leq 1$$

A-law

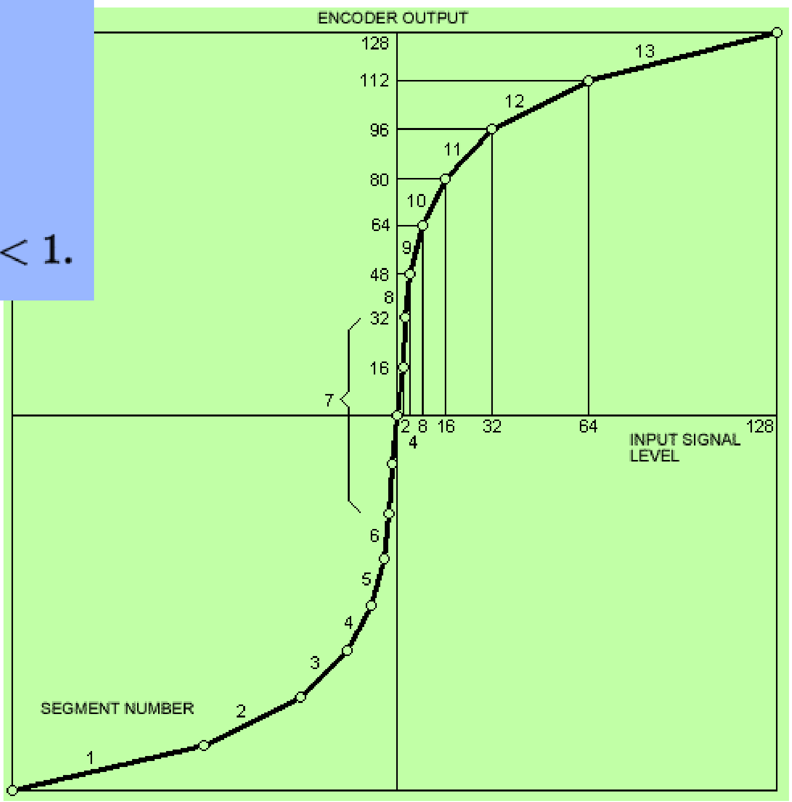
For a given input *x*, the equation for A-law encoding is as follows,

$$F(x) = \operatorname{sgn}(x) \begin{cases} \dfrac{A|x|}{1+\log(A)}, & |x| < \dfrac{1}{A} \\ \dfrac{1+\log(A|x|)}{1+\log(A)}, & \dfrac{1}{A} \leq |x| \leq 1, \end{cases}$$

where *A* is the compression parameter. In Europe, $A = 87.6$.

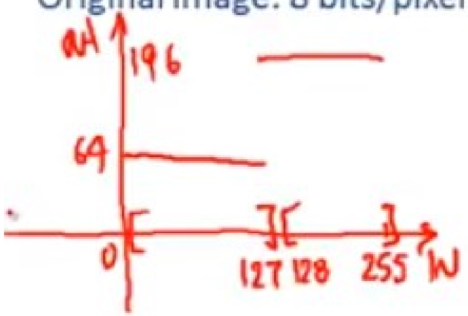
A-law expansion is given by the inverse function,

$$F^{-1}(y) = \operatorname{sgn}(y) \begin{cases} \dfrac{|y|(1+\ln(A))}{A}, & |y| < \dfrac{1}{1+\ln(A)} \\ \dfrac{\exp(|y|(1+\ln(A))-1)}{A}, & \dfrac{1}{1+\ln(A)} \leq |y| < 1. \end{cases}$$



Example: Uniform Quantized Image

Original image: 8 bits/pixel



1 bit/pixel

3 bits/pixel

2 bits/pixel

4 bits/pixel

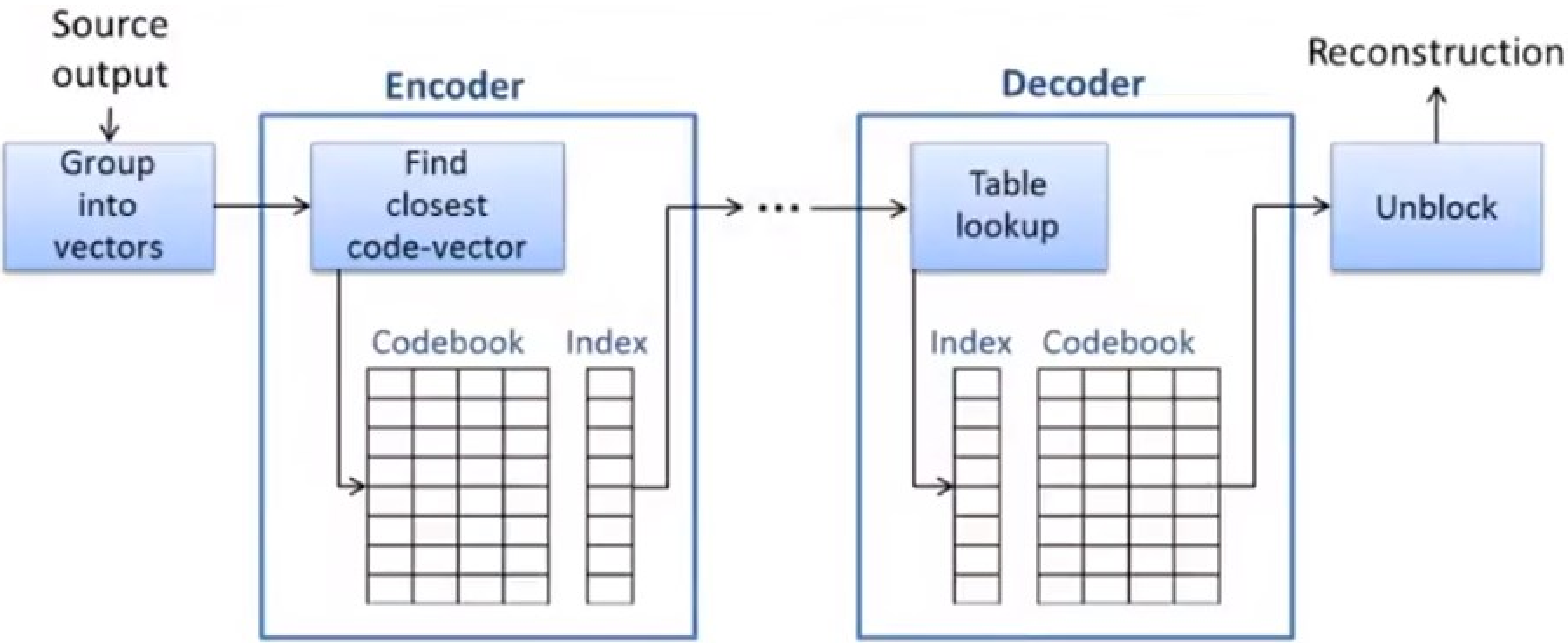
Vector Quantization

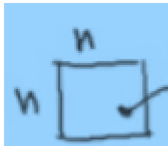
According to Shannon's original work on information theory, any compression system performs better if it operates on vectors or groups of samples rather than individual symbols or samples.

Form vectors of input samples by simply concatenating a number of consecutive samples into a single vector.

Instead of single reconstruction values as in scalar quantization, in VQ code *vectors* with n components are used. A collection of these code vectors form the *codebook*.

Basic VQ Procedure

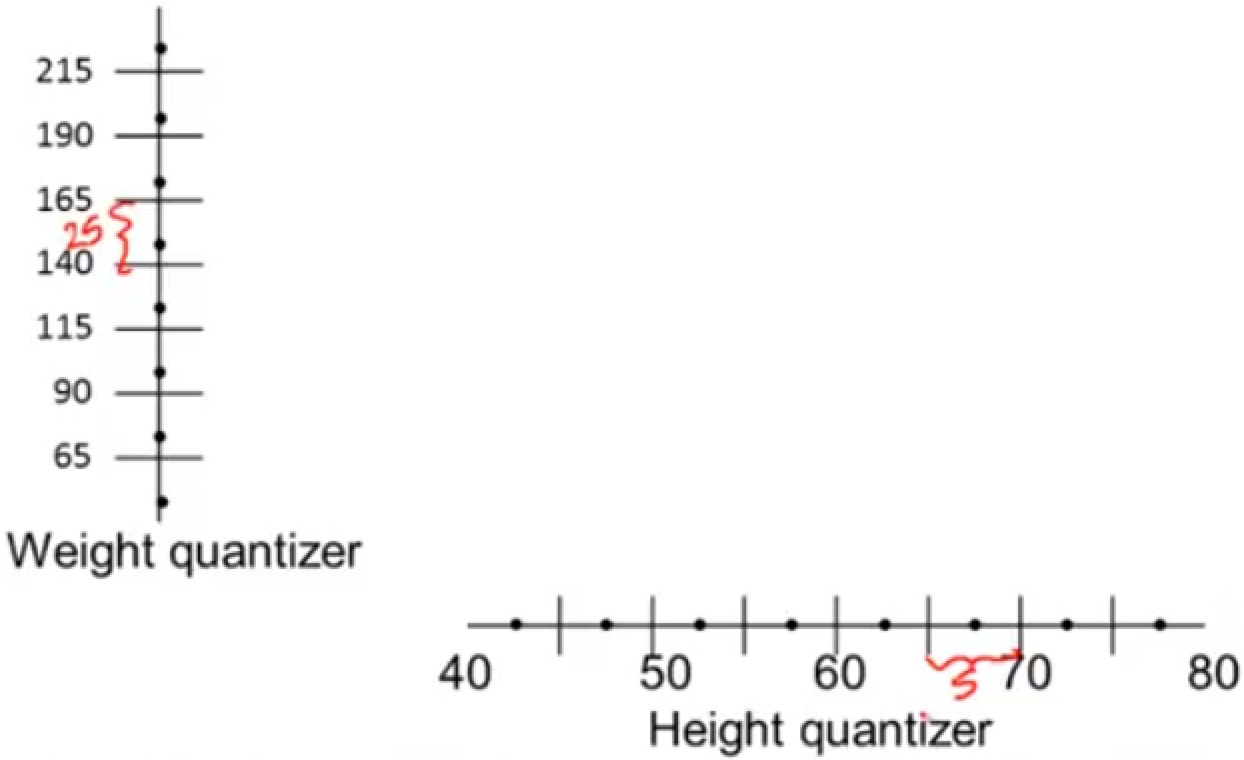




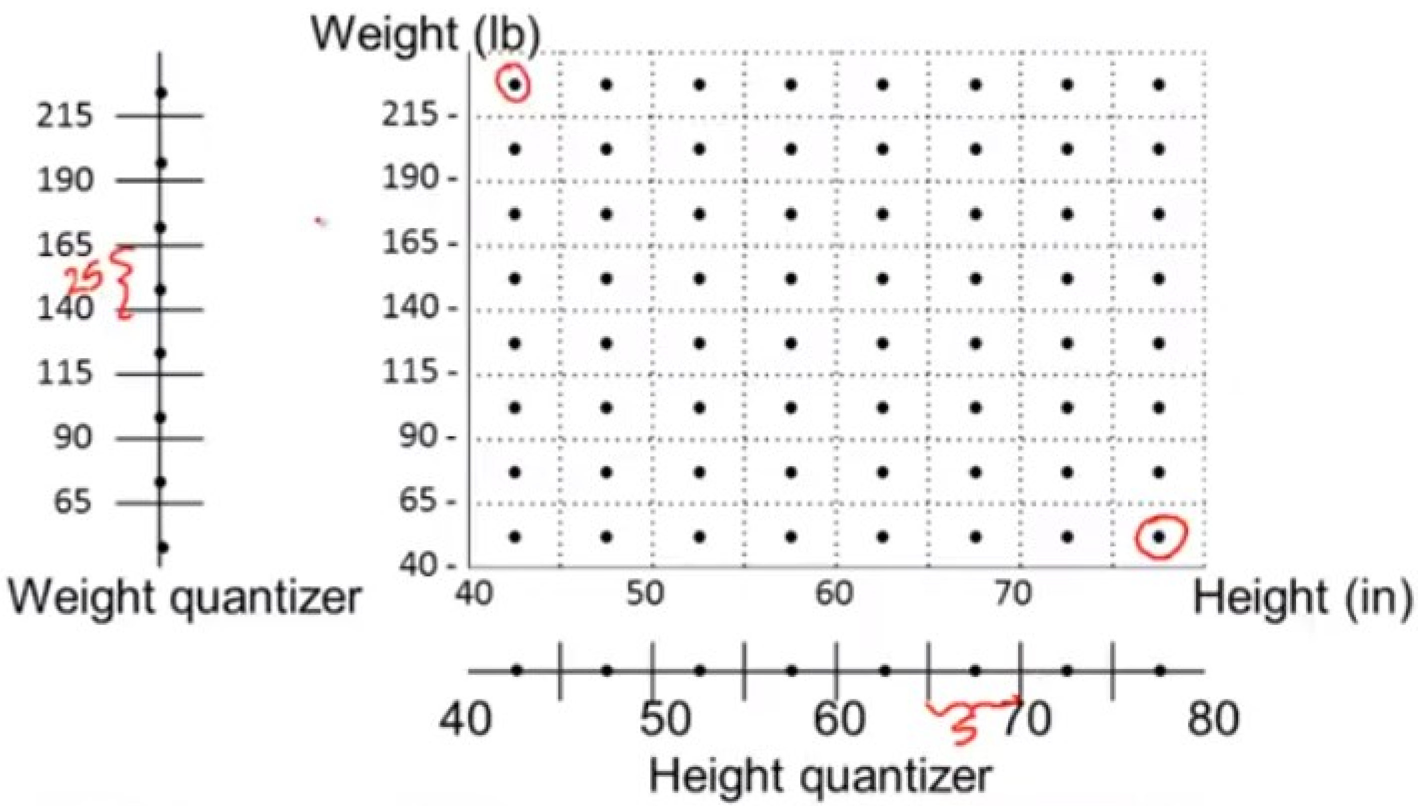
$L = 2^{l} \leftarrow$ bits

$\frac{n \times n \times b}{l} = \text{rate}$

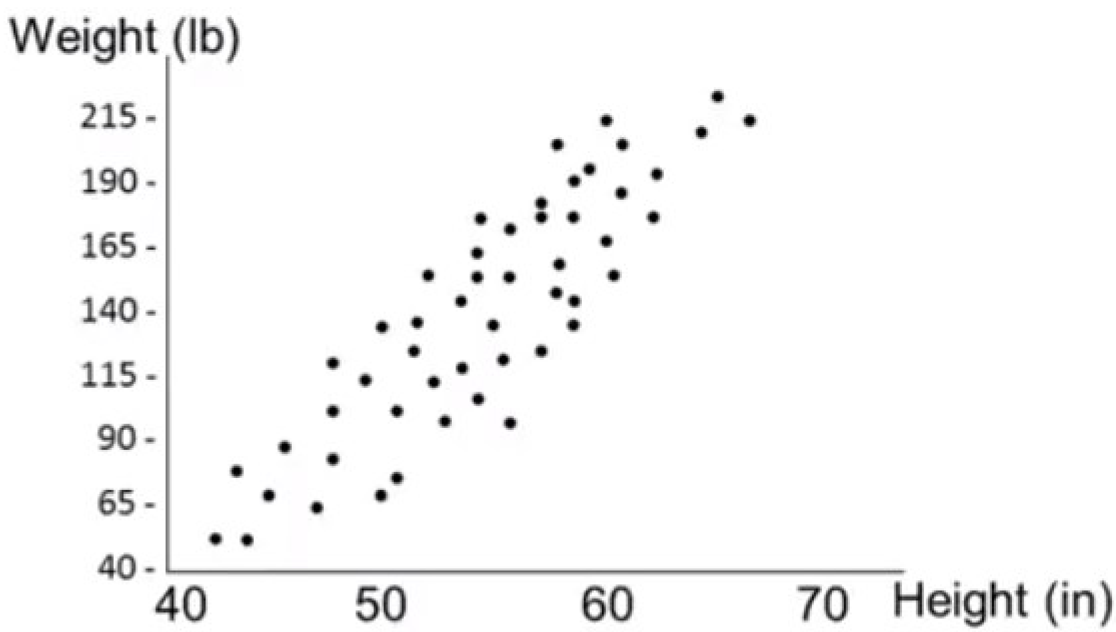
Scaler vs. Vector Quantization



Scaler vs. Vector Quantization



Scaler vs. Vector Quantization



Example: VQ

Codebook:

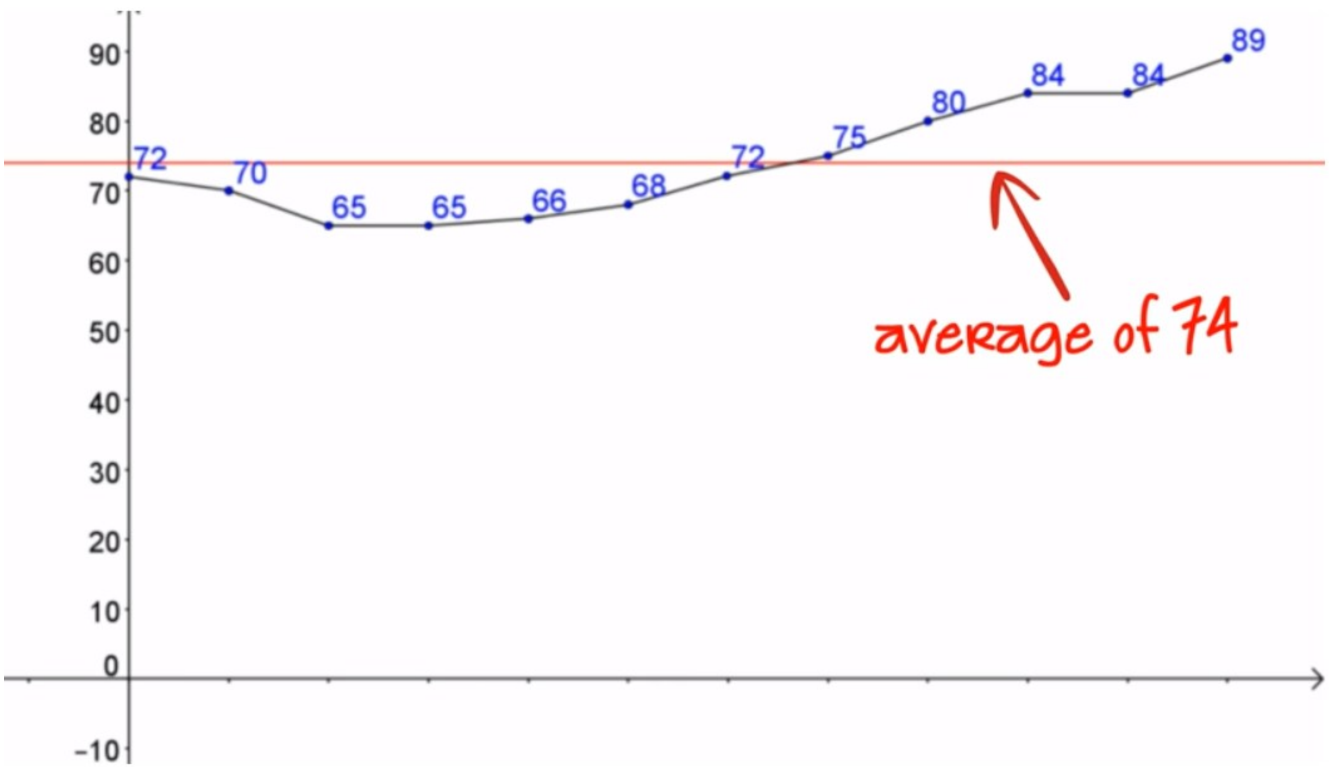
- i=1 : $y_1 = (0, 0)$ $(0-0)^2+(0-1)^2 = 1$
- i=2 : $y_2 = (2, 1)$
- i=3 : $y_3 = (1, 3)$
- i=4 : $y_4 = (1, 4)$ $(1-0)^2+(4-1)^2 = 10$

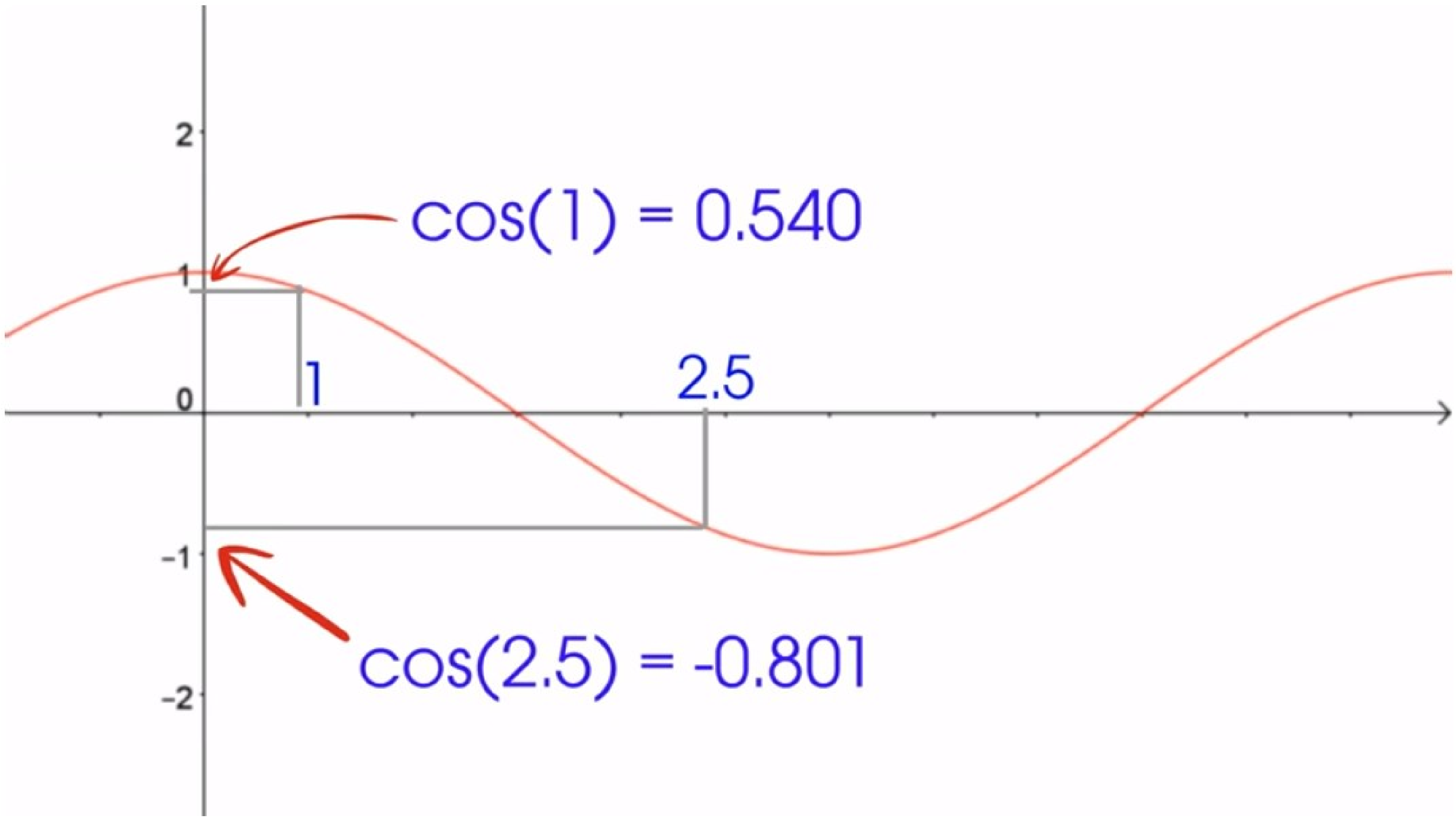
Signal :	0	1	2	3	2	0
Transmit to decoder :	1		3		2	
Decoded signal :	0	0	1	3	2	1
Quantization error :	0	-1	-1	0	0	1

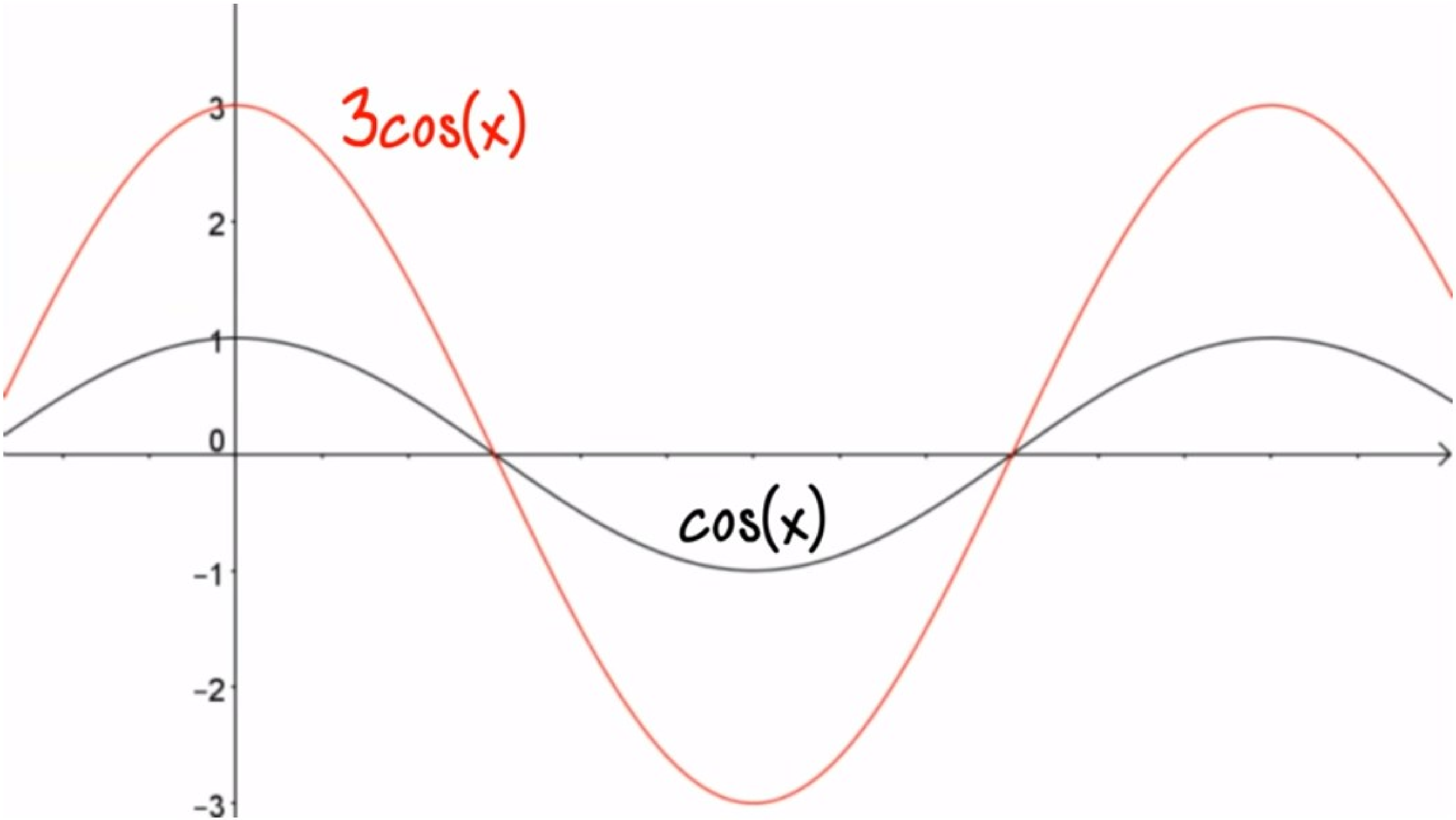
Transform Coding and DCT

Towards DCT

72 70 65 65 66 68 72 75 80 84 84 89




cos(1) = 0.540
cos(2.5) = -0.801
2
1
0
-1
-2
1
2.5



3cos(x)
cos(x)
3
2
1
0
−1
−2
−3

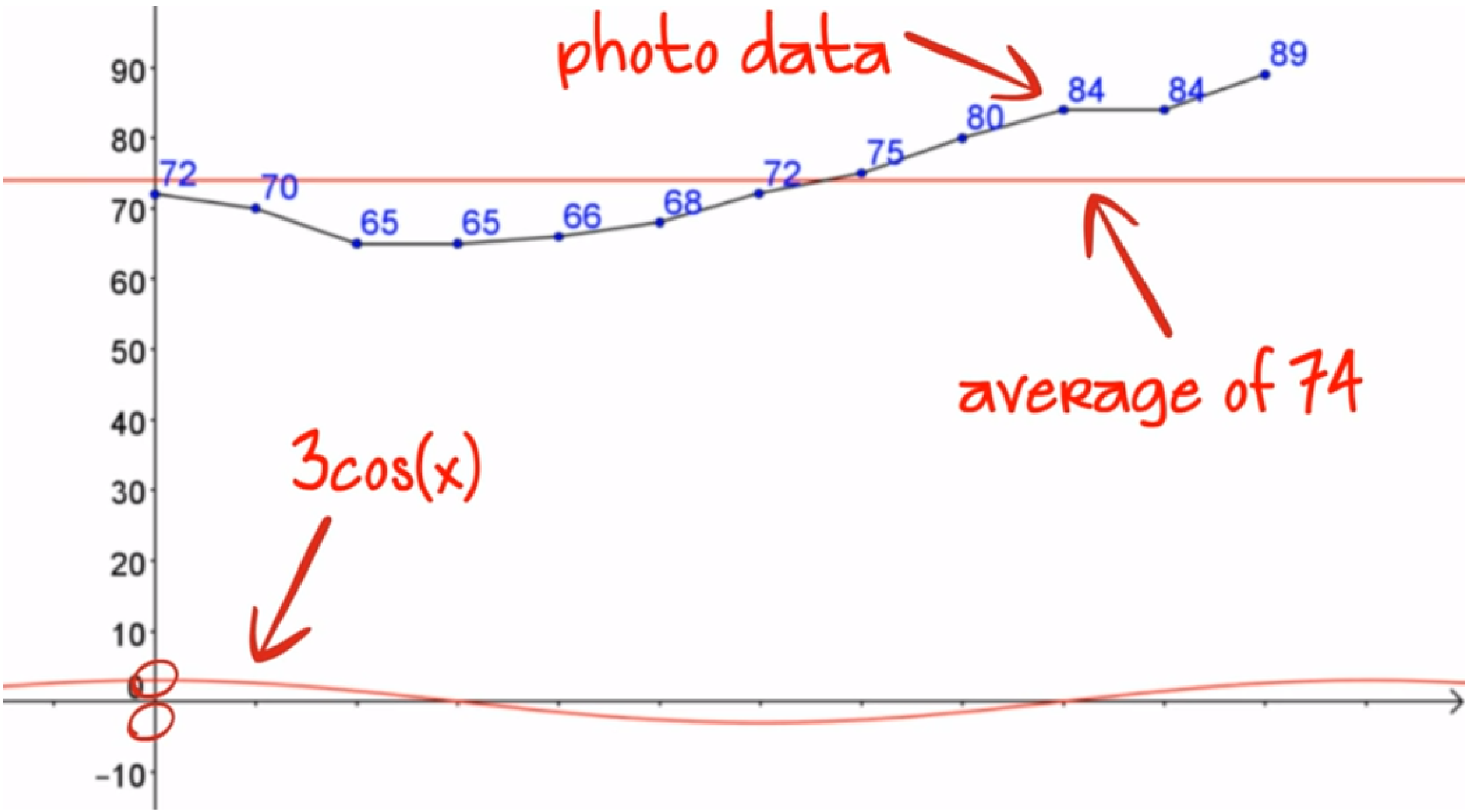
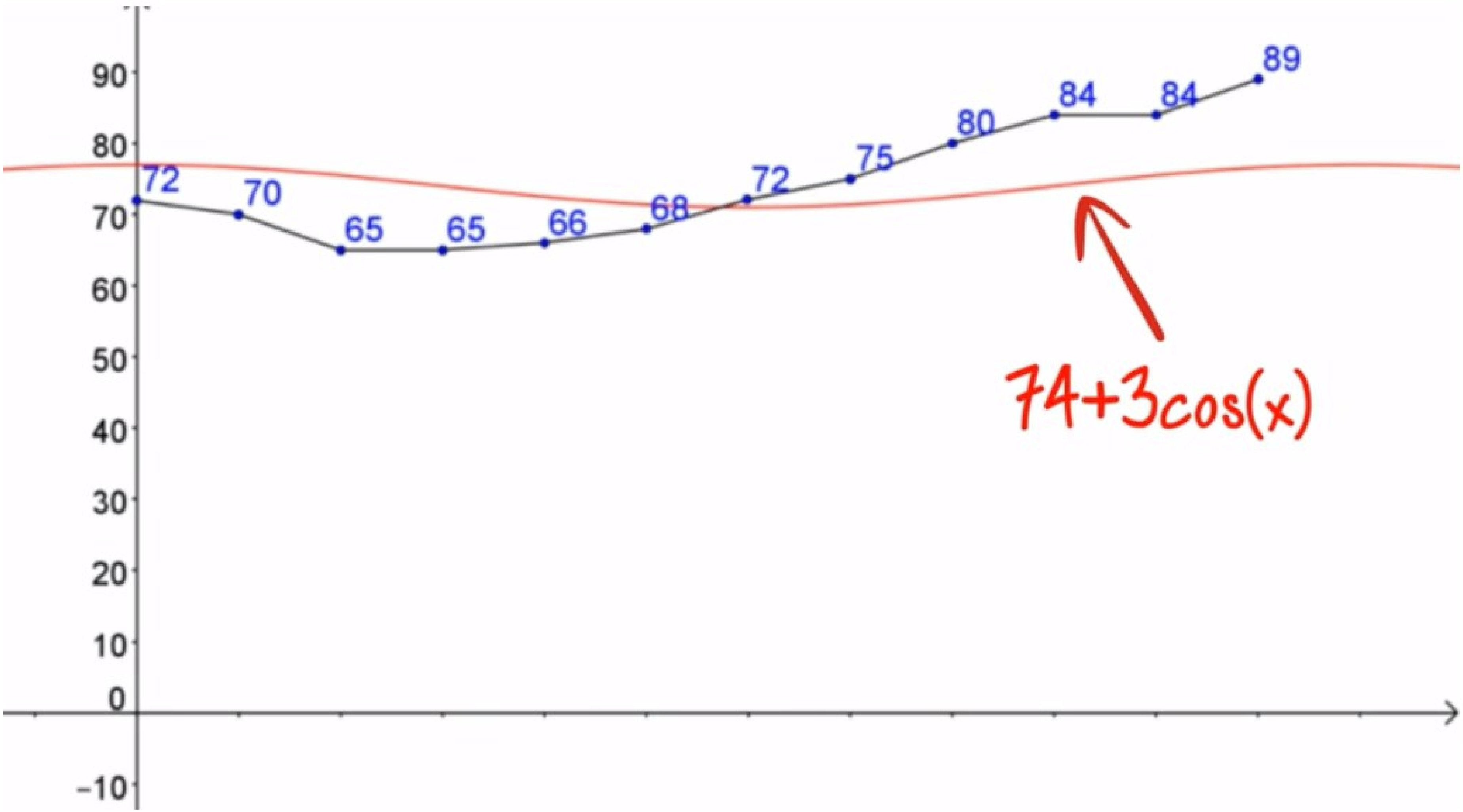
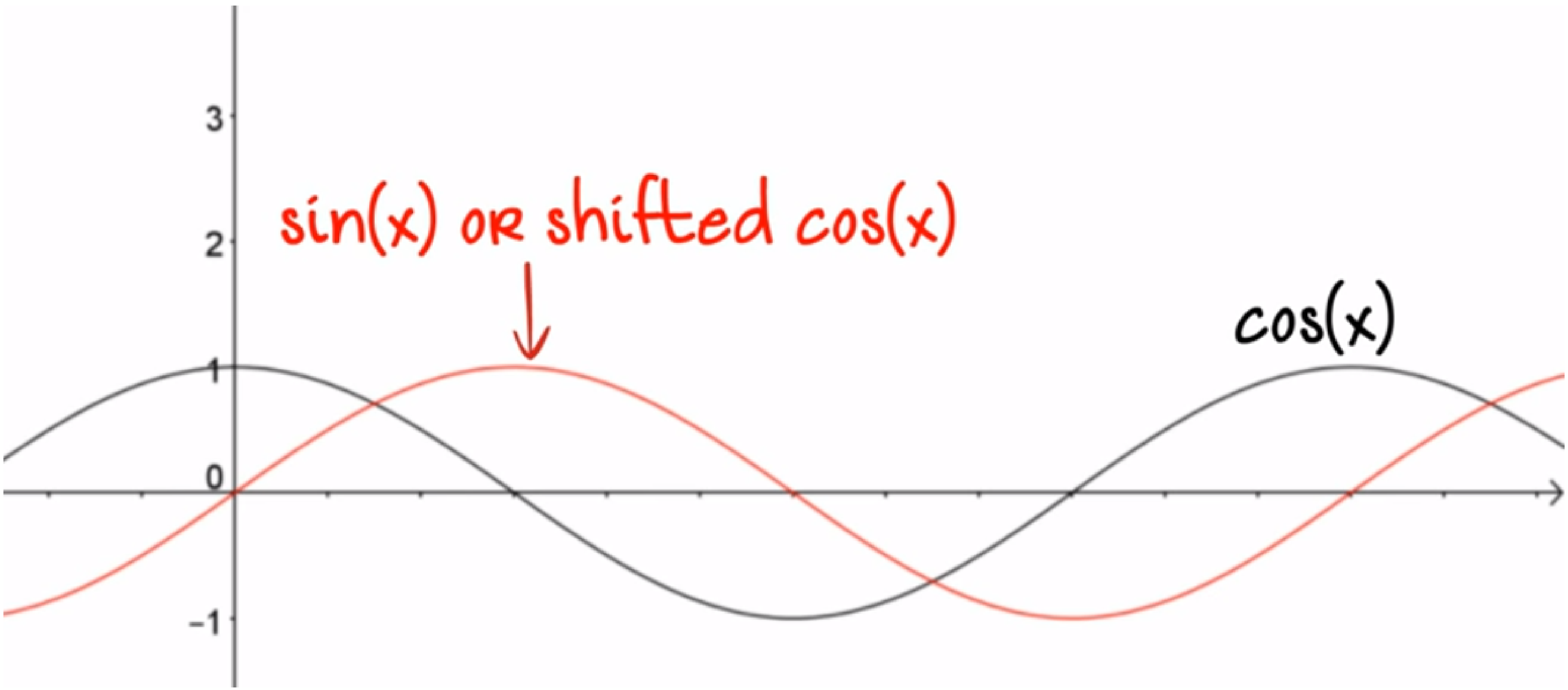
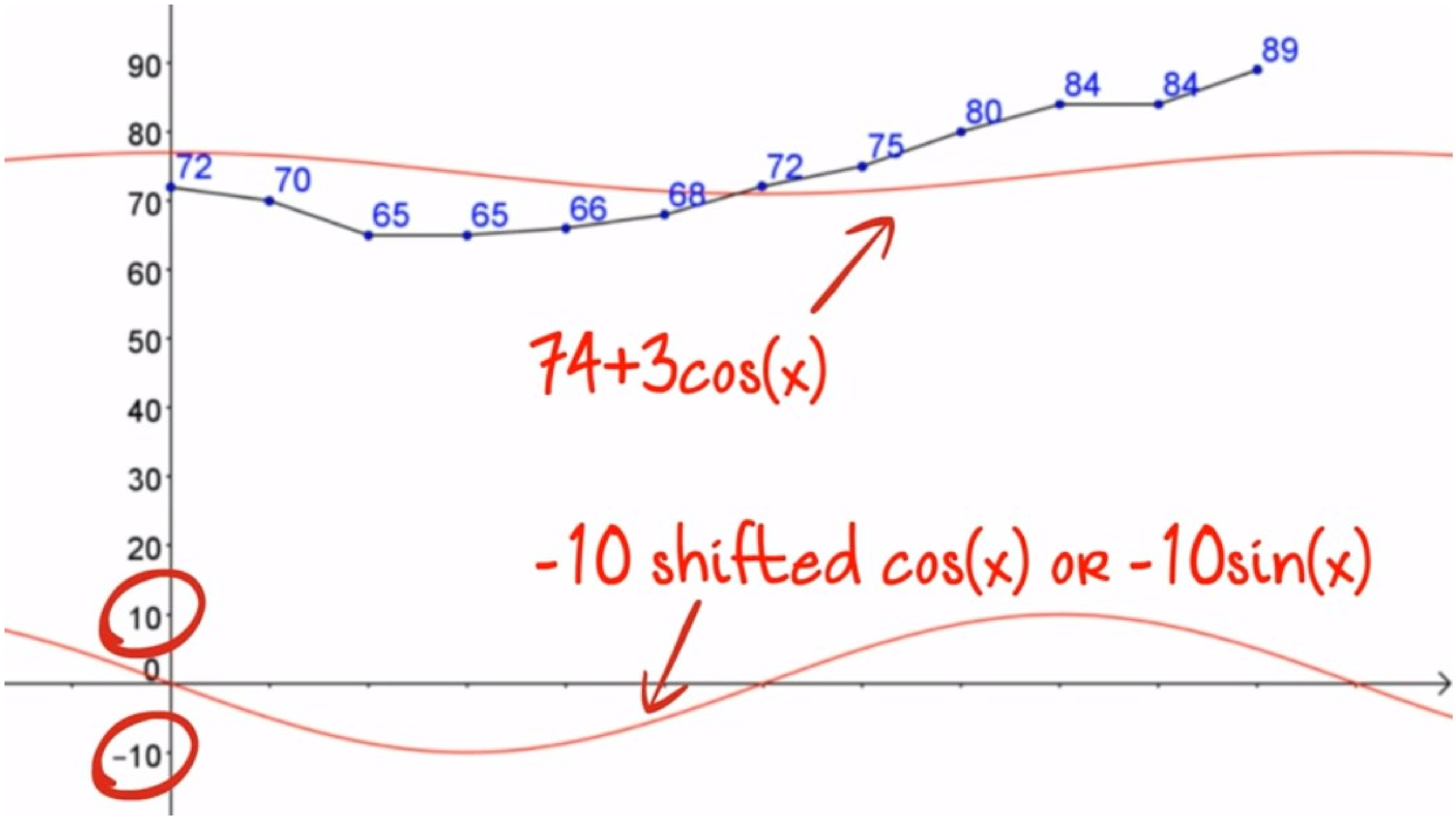


photo data
average of 74
3cos(x)
90
80
70
60
50
40
30
20
10
0
-10
72
70
65
65
66
68
72
75
80
84
84
89


90
80
70
60
50
40
30
20
10
0
−10
72
70
65
65
66
68
72
75
80
84
84
89
74+3cos(x)

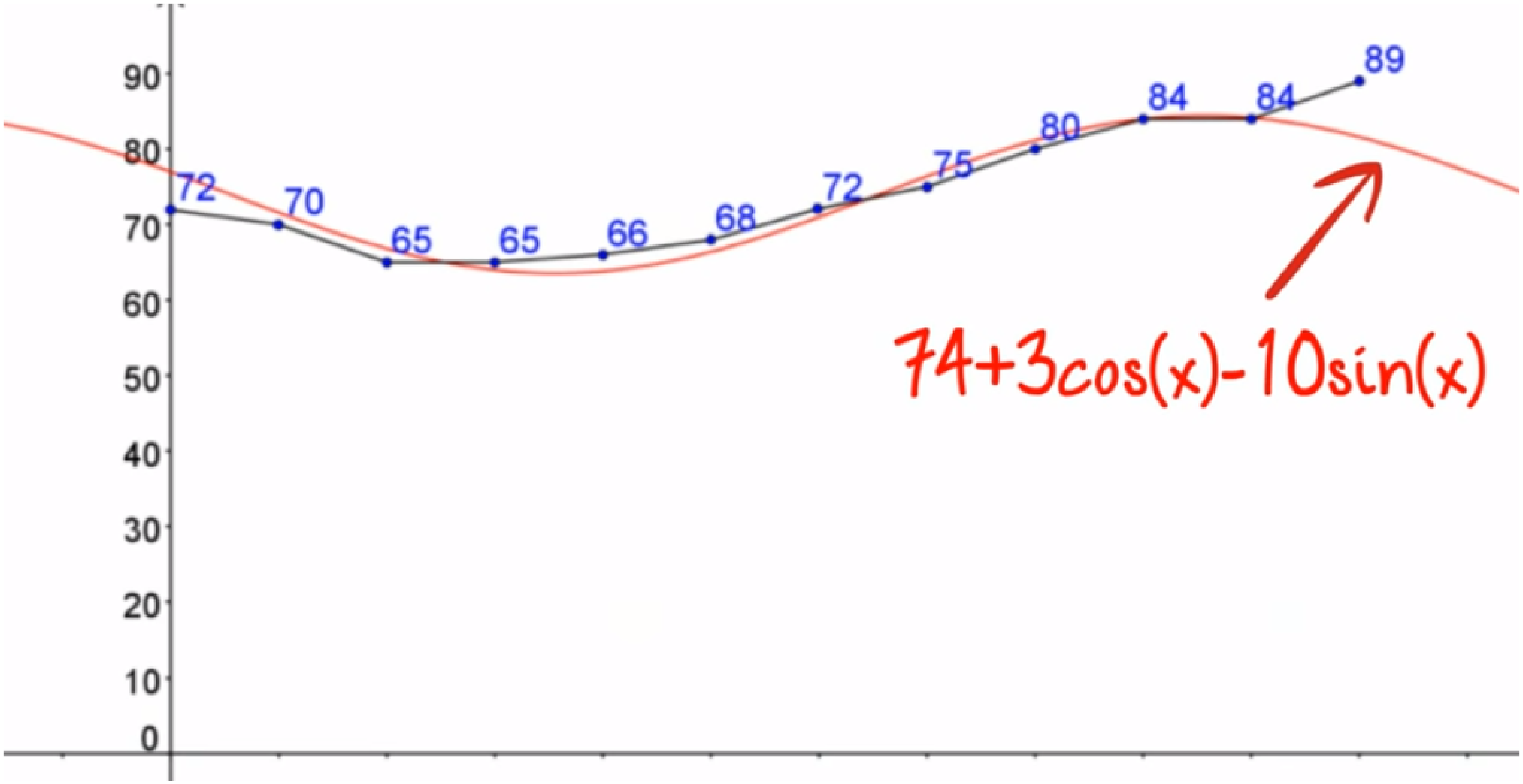


sin(x) or shifted cos(x)
cos(x)
3
2
1
0
−1



90
80
70
60
50
40
30
20
10
0
-10
72
70
65
65
66
68
72
75
80
84
84
89
74+3cos(x)
-10 shifted cos(x) or -10sin(x)

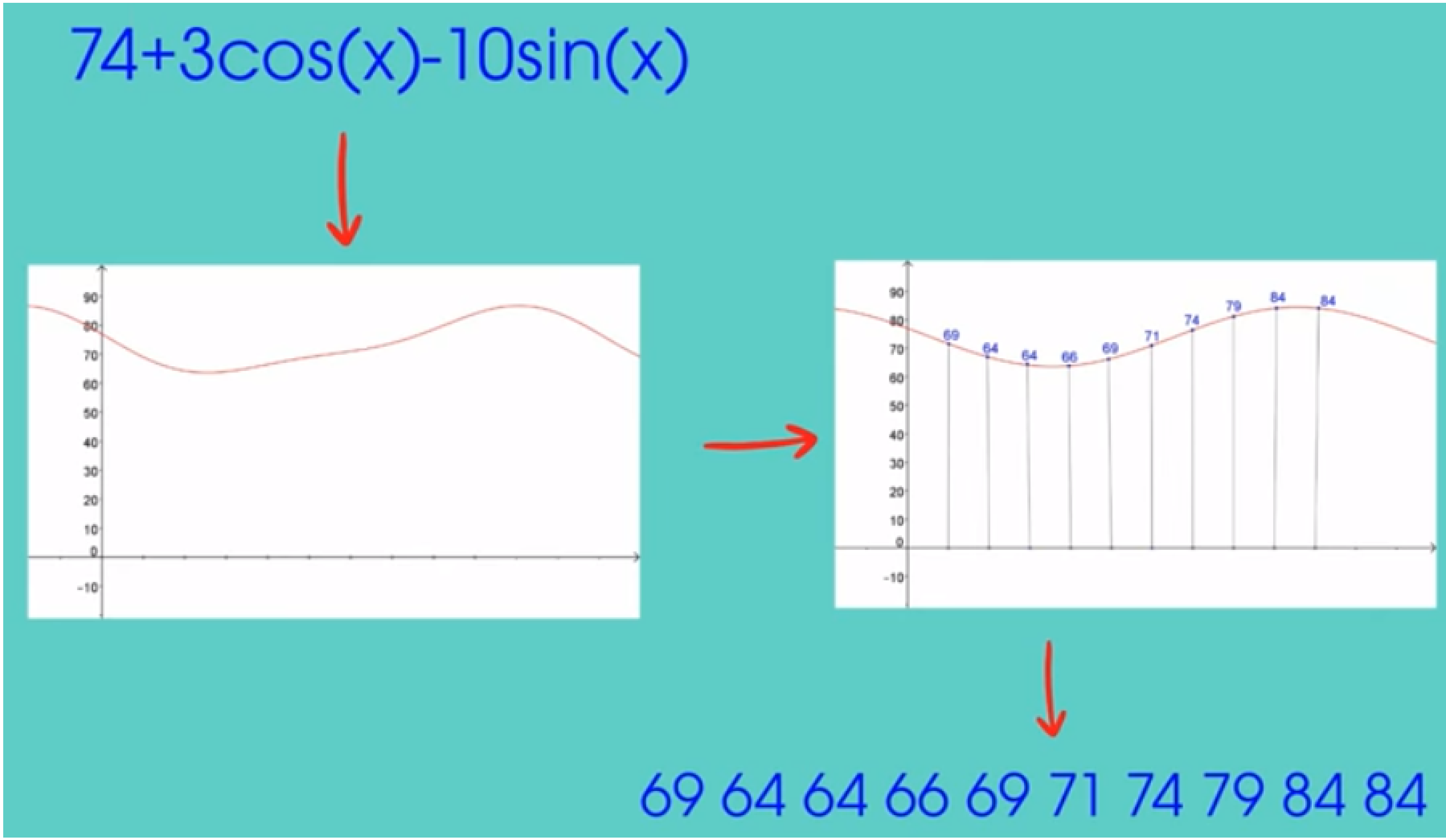
Towards DCT



72 70 65 65 66 68 72 75 80 84 84 89 → 74 3 -10

Towards DCT

72 70 65 65 66 68 72 75 80 84 84 89 → 74 3 -10



Discrete-Cosine Transform (DCT)

1D Discrete Cosine Transform (1D DCT).

$$F(u) = \frac{C(u)}{2} \sum_{i=0}^{7} \cos \frac{(2i+1)u\pi}{16} f(i),$$

where $i = 0, 1, \ldots, 7$, $u = 0, 1, \ldots, 7$.

1D Inverse Discrete Cosine Transform (1D-IDCT).

$$\tilde{f}(i) = \sum_{u=0}^{7} \frac{C(u)}{2} \cos \frac{(2i+1)u\pi}{16} F(u),$$

where $i = 0, 1, \ldots, 7$, $u = 0, 1, \ldots, 7$.

Discrete-Cosine Transform (DCT)

Definition of DCT. Let's start with the two-dimensional DCT. Given a function $f(i, j)$ over two integer variables i and j (a piece of an image), the 2D DCT transforms it into a new function $F(u, v)$, with integer u and v running over the same range as i and j. The general definition of the transform is

$$F(u, v) = \frac{2\,C(u)\,C(v)}{\sqrt{MN}} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \cos\frac{(2i+1)u\pi}{2M} \cos\frac{(2j+1)v\pi}{2N} f(i, j)$$

where $i, u = 0, 1, \ldots, M-1$, $j, v = 0, 1, \ldots, N-1$, and the constants $C(u)$ and $C(v)$ are determined by

$$C(\xi) = \begin{cases} \frac{\sqrt{2}}{2} & \text{if } \xi = 0, \\ 1 & \text{otherwise.} \end{cases}$$

Discrete-Cosine Transform (DCT)

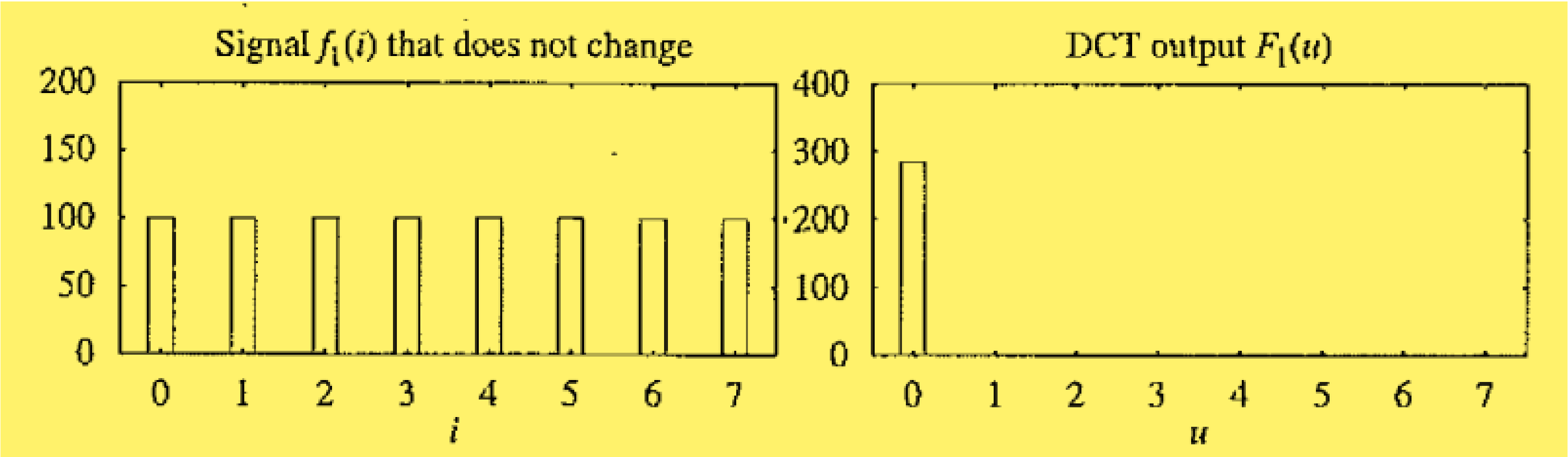
- In JPEG M=N=8

2D Discrete Cosine Transform (2D DCT).

$$F(u,v) = \frac{C(u)\,C(v)}{4} \sum_{i=0}^{7} \sum_{j=0}^{7} \cos\frac{(2i+1)u\pi}{16} \cos\frac{(2j+1)v\pi}{16}\, f(i,j),$$

$$\tilde{f}(i,j) = \sum_{u=0}^{7} \sum_{v=0}^{7} \frac{C(u)\,C(v)}{4} \cos\frac{(2i+1)u\pi}{16} \cos\frac{(2j+1)v\pi}{16}\, F(u,v)$$

Example



$$\begin{aligned} F_1(0) &= \frac{\sqrt{2}}{2\cdot 2}\cdot(1\cdot 100+1\cdot 100+1\cdot 100+1\cdot 100 \\ &\quad +1\cdot 100+1\cdot 100+1\cdot 100+1\cdot 100) \\ &\approx 283 \end{aligned}$$

When $u = 1$, $F_1(u)$ is as below. Because $\cos\frac{\pi}{16} = -\cos\frac{15\pi}{16}$, $\cos\frac{3\pi}{16} = -\cos\frac{13\pi}{16}$, etc. and $C(1) = 1$, we have

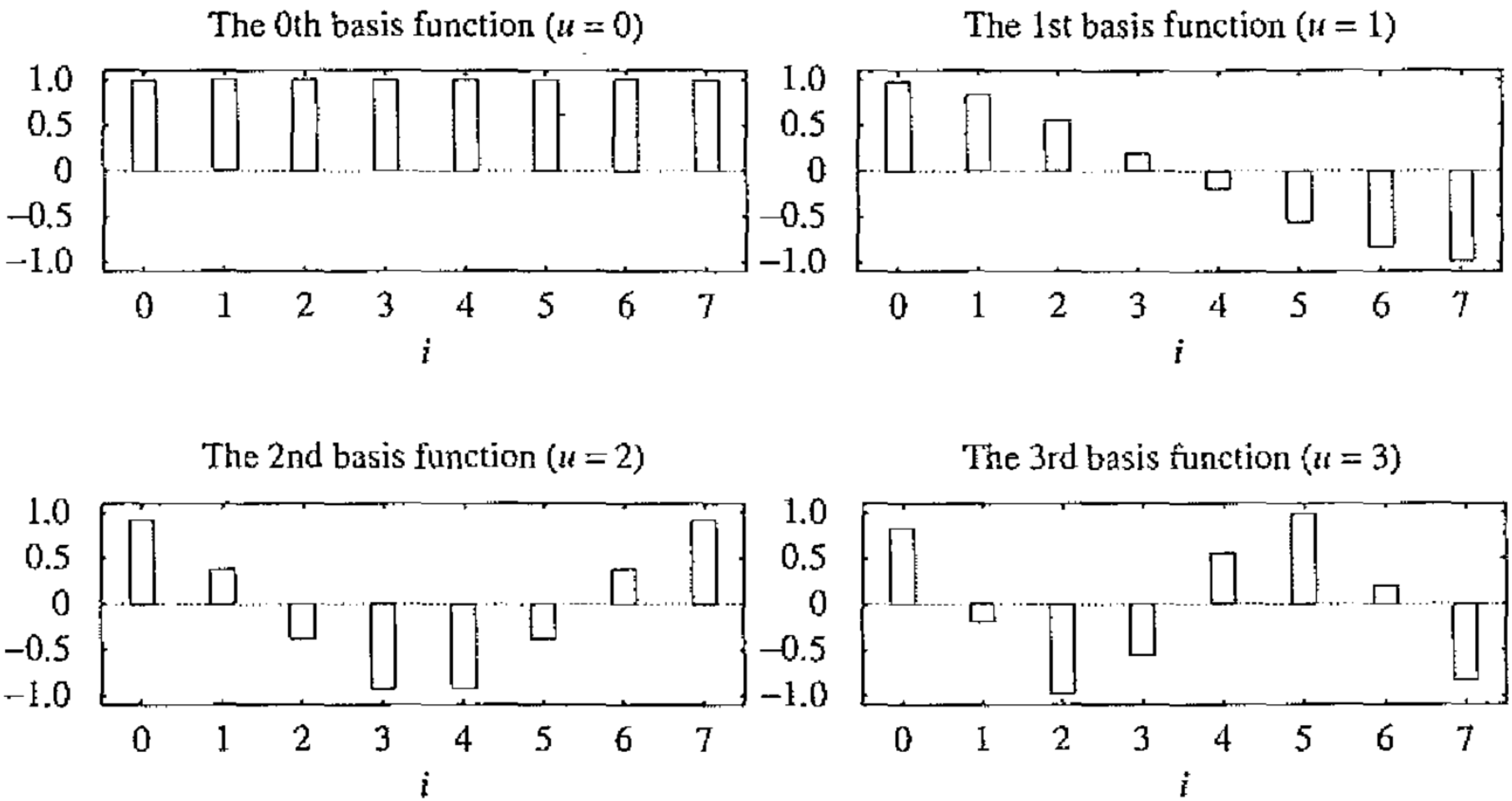
$$\begin{aligned} F_1(1) &= \frac{1}{2}\cdot(\cos\frac{\pi}{16}\cdot 100+\cos\frac{3\pi}{16}\cdot 100+\cos\frac{5\pi}{16}\cdot 100+\cos\frac{7\pi}{16}\cdot 100 \\ &\quad +\cos\frac{9\pi}{16}\cdot 100+\cos\frac{11\pi}{16}\cdot 100+\cos\frac{13\pi}{16}\cdot 100+\cos\frac{15\pi}{16}\cdot 100) \\ &= 0 \end{aligned}$$

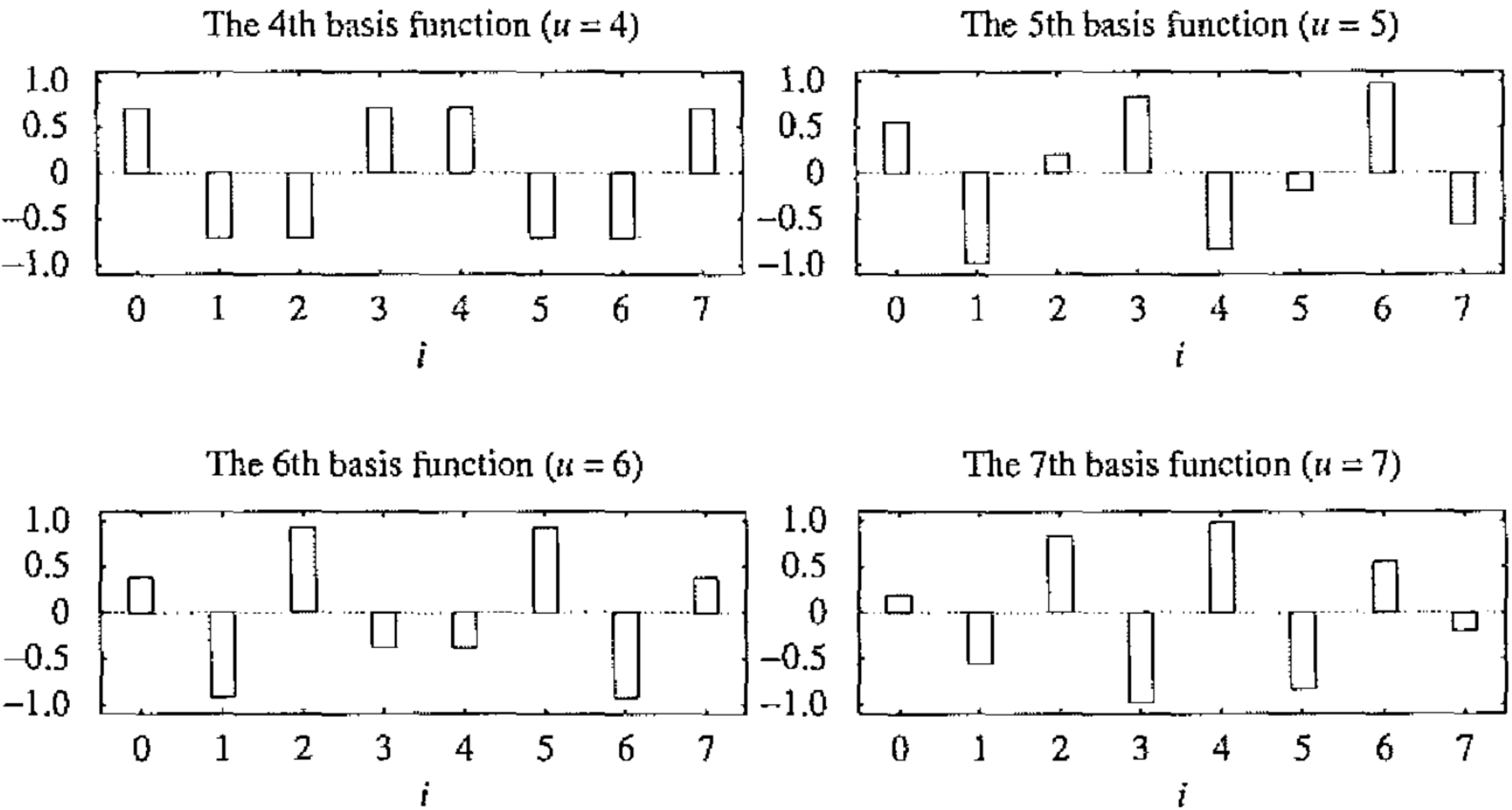
Basis Function

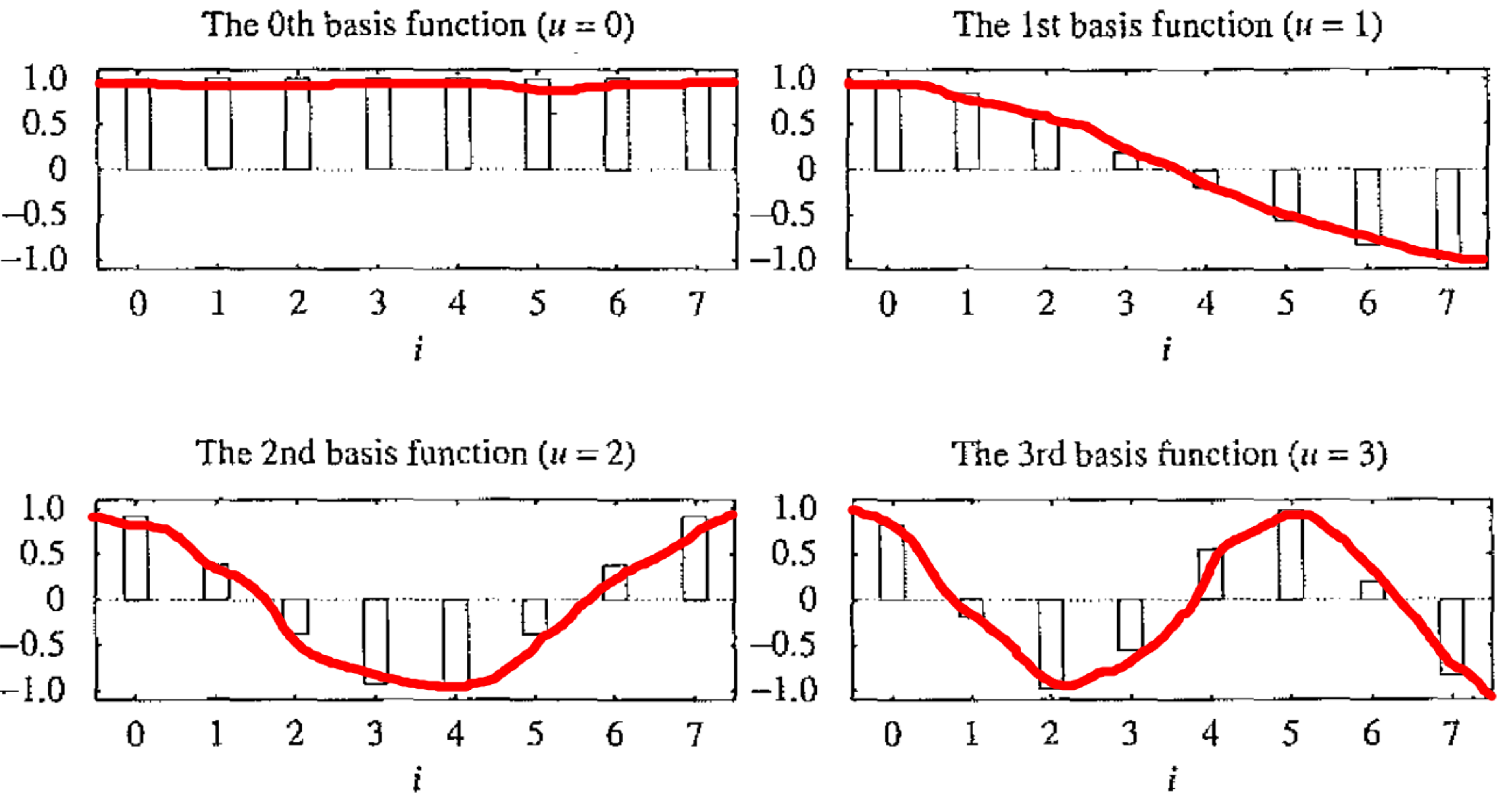
- A basis function is an element of a particular basis for a function space.
- Every continuous function in the function space can be represented as a linear combination of basis functions, just as every vector in a vector space can be represented as a linear combination of basis vectors.
- The collection of quadratic polynomials with real coefficients has

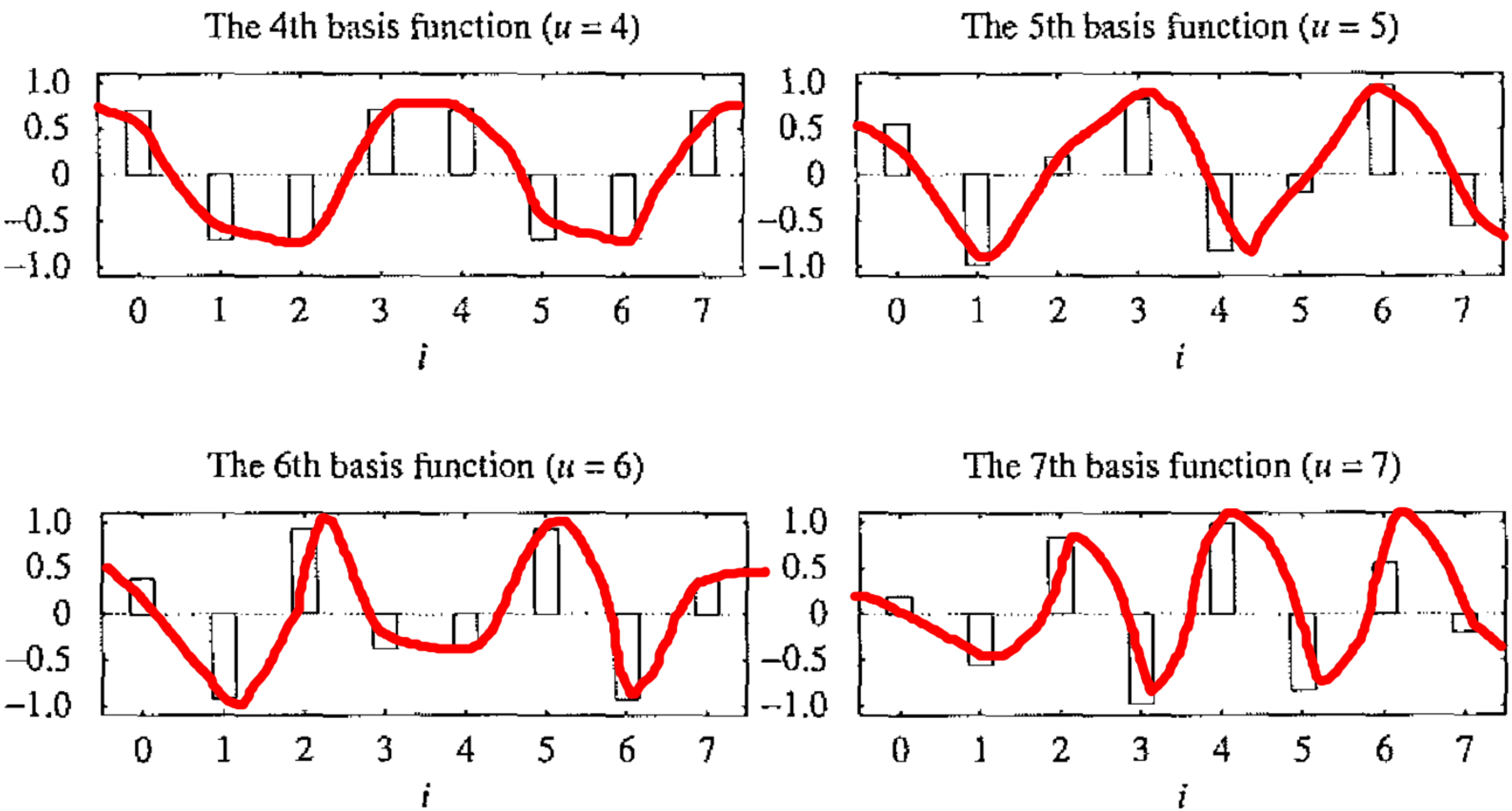
  $\{1, t, t^2\}$ as a basis.

1D DCT Basis Functions

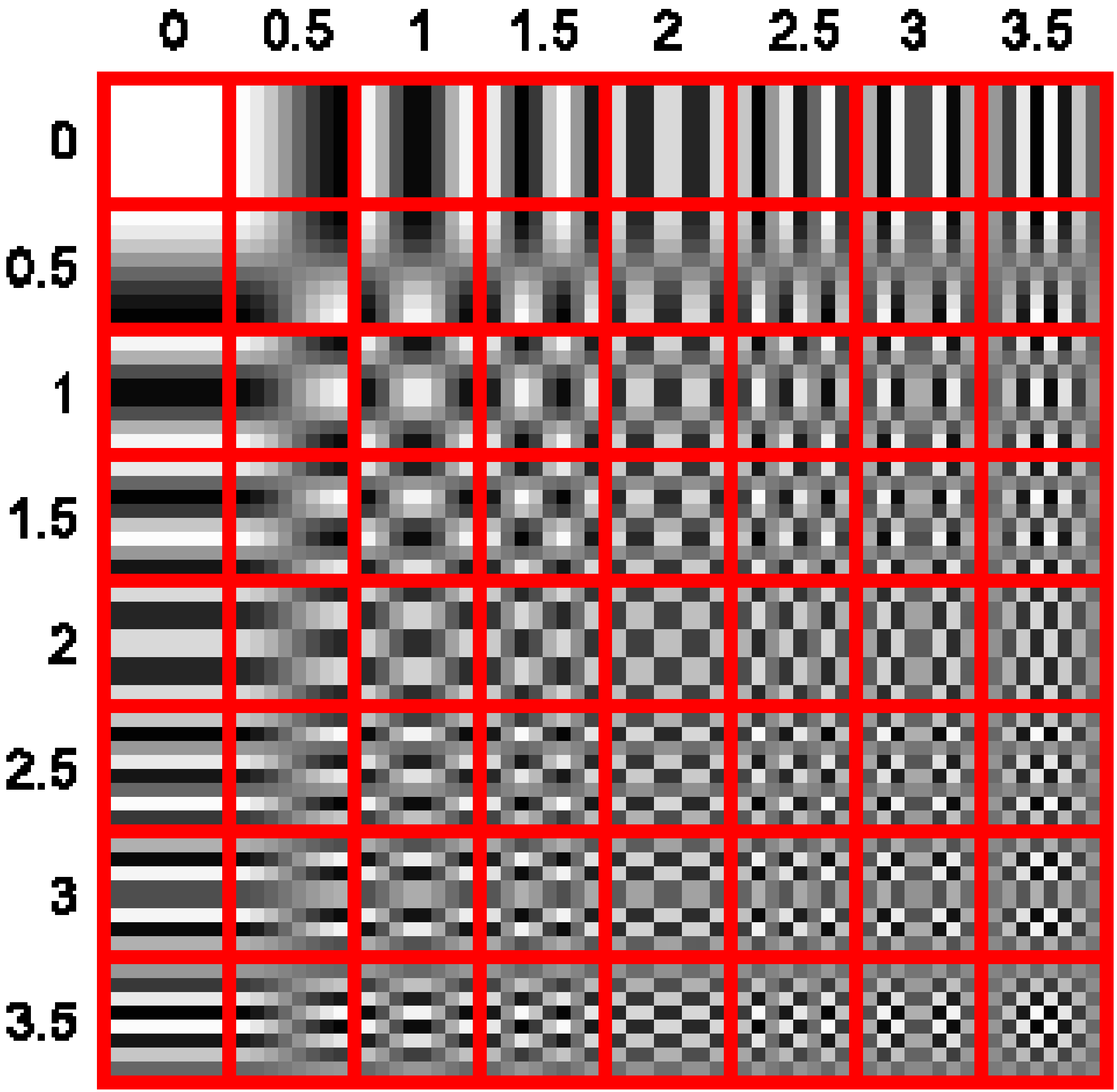



The 4th basis function ($u = 4$)
The 5th basis function ($u = 5$)
The 6th basis function ($u = 6$)
The 7th basis function ($u = 7$)
1.0
0.5
0
−0.5
−1.0
0 1 2 3 4 5 6 7
i

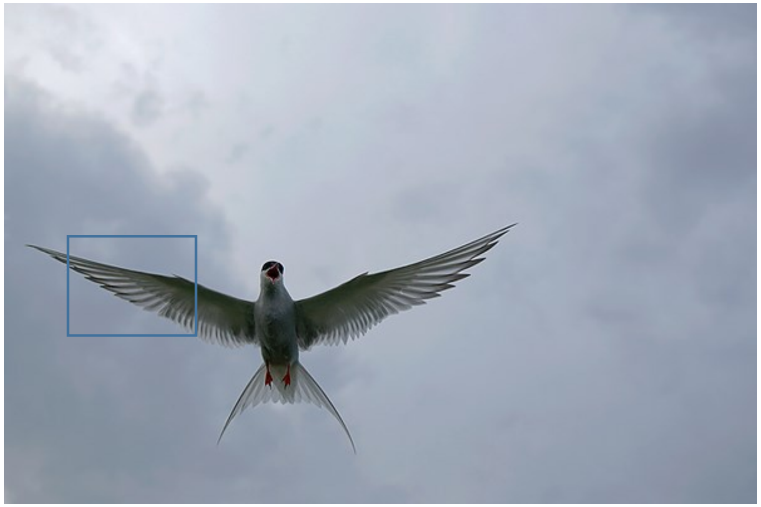

The 0th basis function ($u = 0$)
The 1st basis function ($u = 1$)
The 2nd basis function ($u = 2$)
The 3rd basis function ($u = 3$)
1.0
0.5
0
−0.5
−1.0
0 1 2 3 4 5 6 7
i


The 4th basis function (u = 4)
The 5th basis function (u = 5)
The 6th basis function (u = 6)
The 7th basis function (u = 7)
1.0
0.5
0
−0.5
−1.0
0 1 2 3 4 5 6 7
i

JPEG DCT Basis Functions



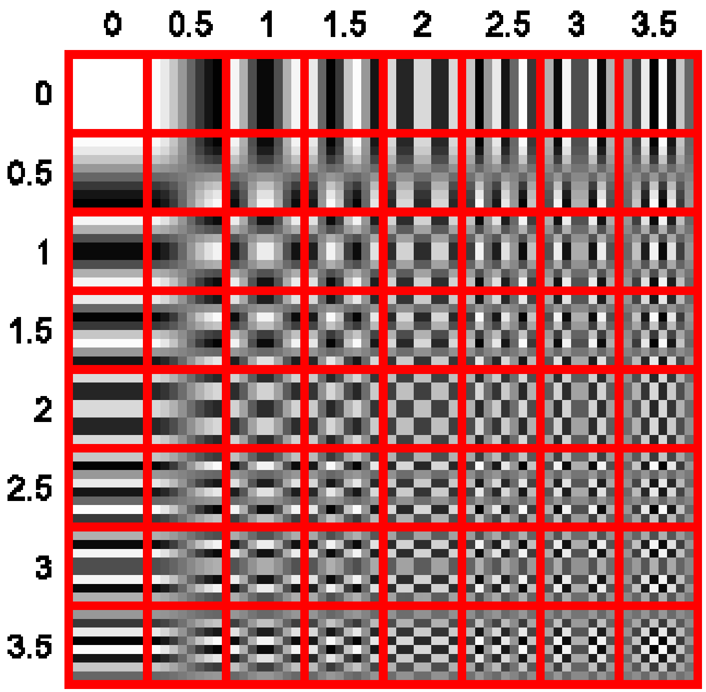
DCT Coefficients



139	144	149	153	155	155	155	155
144	151	153	156	159	156	156	156
150	155	160	163	158	156	156	156
159	161	162	160	160	159	159	159
159	160	161	162	162	155	155	155
161	161	161	161	160	157	157	157
162	162	161	163	162	157	157	157
162	162	161	161	163	158	158	158

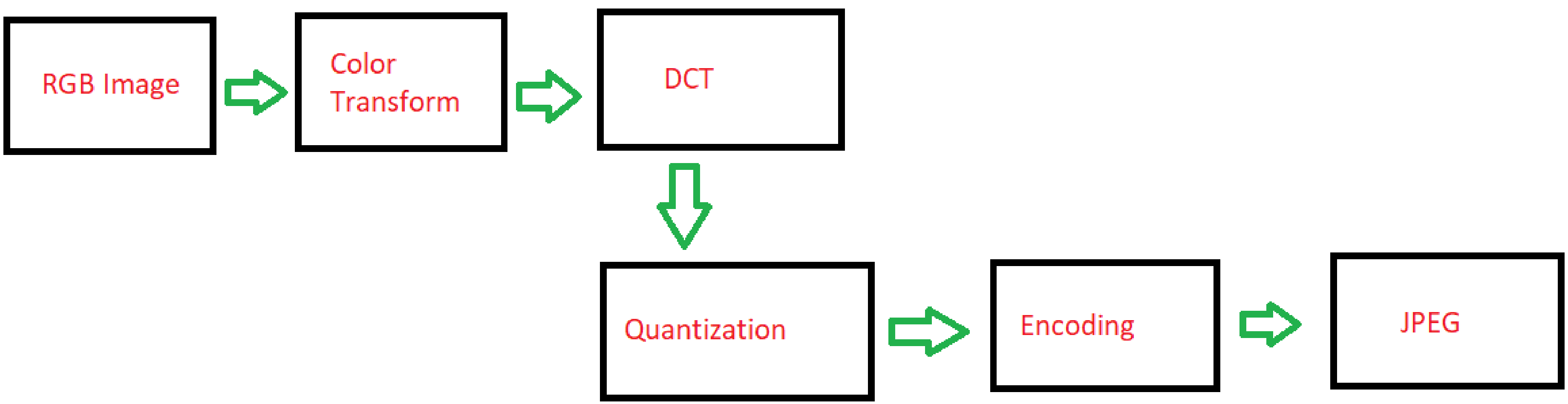
235.6	-1.0	-12.1	-5.2	2.1	-1.7	-2.7	1.3
-22.6	-17.5	-6.2	-3.2	-2.9	-0.1	0.4	-1.2
-10.9	-9.3	-1.6	1.5	0.2	-0.9	-0.6	-0.1
-7.1	-1.9	0.2	1.5	0.9	-0.1	0.0	0.3
-0.6	-0.8	1.5	1.6	-0.1	-0.7	0.6	1.3
1.8	-0.2	1.6	-0.3	-0.8	1.5	1.0	-1.0
-1.3	-0.4	-0.3	-1.5	-0.5	1.7	1.1	-0.8
-2.6	1.6	-3.8	-1.8	1.9	1.2	-0.6	-0.4

DCT Coefficients



235.6	-1.0	-12.1	-5.2	2.1	-1.7	-2.7	1.3
-22.6	-17.5	-6.2	-3.2	-2.9	-0.1	0.4	-1.2
-10.9	-9.3	-1.6	1.5	0.2	-0.9	-0.6	-0.1
-7.1	-1.9	0.2	1.5	0.9	-0.1	0.0	0.3
-0.6	-0.8	1.5	1.6	-0.1	-0.7	0.6	1.3
1.8	-0.2	1.6	-0.3	-0.8	1.5	1.0	-1.0
-1.3	-0.4	-0.3	-1.5	-0.5	1.7	1.1	-0.8
-2.6	1.6	-3.8	-1.8	1.9	1.2	-0.6	-0.4

JPEG



Q&A