

# Cosine Transform – Continuous and Discrete

Benjamin W. Ong

## 1 Introduction

Suppose we have an even function,  $f(x)$ , in the interval from  $[-L, L]$ . We would like to approximate this function as a combination of cosine functions. Specifically,

$$f(x) \approx \frac{a_0}{2} + \sum_{k=1}^N a_k \cos\left(\frac{k\pi x}{L}\right). \quad (1)$$

The coefficient  $a_k$  is sometimes referred to as the *amplitude* corresponding to a wave with *wave number* =  $k$ . Figure 1 shows that the square wave function (red curve) can be approximated by the the “cosine expansion”,  $L(x) = \frac{4}{\pi} \cos(2\pi x) - \frac{4}{3\pi} \cos(6\pi x) + \frac{4}{5\pi} \cos(10\pi x)$ .

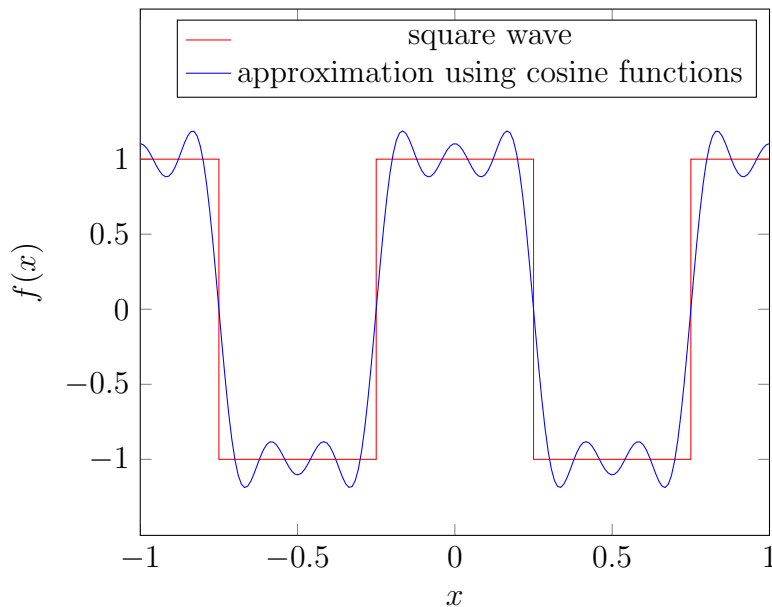


Figure 1: Approximating a square wave function (red curve) using combinations of cosine functions (blue curve).

Why approximate functions using combinations of cosine functions? It turns out that for many digital media sources (such as audio waves and digital pictures), the cosine representation results in many small coefficients for large wave numbers. The idea behind image or audio compression is to approximate the digital signal using only coefficients that are sufficiently large. Thus, one is often able to use only 10% of the coefficients to approximate the original signal.

To obtain the cosine expansion for a more general function  $f(x), x \in [0, L]$  that is not even,

we first form the even-extension of the function, e.g.

$$f_e(x) = \begin{cases} f(x), & x \in [0, L], \\ f(-x), & x \in [-L, 0), \end{cases}$$

before attempting to obtain a cosine expansion.

## 2 Continuous Cosine Transform

Given an even (continuous) function,  $f(x)$ , in the interval  $[-L, L]$ , the coefficients for the cosine expansion in equation (1) satisfy the following formulas:

$$a_0 = \frac{1}{L} \int_{-L}^L f(x) dx, \quad a_k = \frac{1}{L} \int_{-L}^L f(x) \cos\left(\frac{k\pi x}{L}\right) dx. \quad (2)$$

These formulas can be derived by considering the orthogonality properties of the cosine function, which we won't discuss here.

## 3 Discrete Cosine Transform

Suppose we now have discrete data,  $\vec{f} = [f_1, f_2, \dots, f_N]$  corresponding to function values at  $x_i = \frac{(2i-1)}{2} \Delta x$ , where  $\Delta x = \frac{L}{N}$ . (Note: the discrete data for  $x \in [-L, 0]$  can be obtained by taking an even extension). How do we find the coefficients  $a_k$ , so that

$$L(x) = \frac{a_0}{2} + \sum_{k=1}^{N-1} a_k \cos\left(\frac{k\pi x}{L}\right) \quad (3)$$

interpolates the data, providing an approximation to the original signal? If  $L(x)$  interpolates the data, then  $L(x_i) = f_i$ , giving  $N$  equations for the  $N$  unknown coefficients. One can assemble these equations in matrix form, and solve the linear system for the unknown coefficients. First, observe

$$\begin{aligned} L(x_i) &= \frac{a_0}{2} + \sum_{k=1}^{N-1} a_k \cos\left(\frac{k\pi x_i}{L}\right) \\ &= \frac{a_0}{2} + \sum_{k=1}^{N-1} a_k \cos\left(\frac{k\pi (2i-1) L}{L \cdot 2 \cdot N}\right) \\ &= \frac{a_0}{2} + \sum_{k=1}^{N-1} a_k \cos\left(\frac{k\pi(2i-1)}{2N}\right) \end{aligned}$$

Hence, the linear system takes the form

$$\begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \cos\left(\frac{\pi}{2N}\right) & \cos\left(\frac{2\pi}{2N}\right) & \dots & \cos\left(\frac{(N-1)\pi}{2N}\right) \\ \frac{1}{2} & \cos\left(\frac{3\pi}{2N}\right) & \cos\left(\frac{2\cdot 3\pi}{2N}\right) & \dots & \cos\left(\frac{(N-1)\cdot 3\pi}{2N}\right) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{1}{2} & \cos\left(\frac{(2N-1)\pi}{2N}\right) & \cos\left(\frac{2\cdot(2N-1)\pi}{2N}\right) & \dots & \cos\left(\frac{(N-1)\cdot(2N-1)\pi}{2N}\right) \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_{N-1} \end{bmatrix},$$

and can be solved for the unknown coefficients  $\{a_0, a_1, \dots, a_{N-1}\}$ .

Interestingly, it turns out that the inverse of this matrix has a very simple form due to it's structure. Specifically,

$$= \frac{N}{2} \begin{bmatrix} \frac{1}{2} & \cos\left(\frac{\pi}{2N}\right) & \cos\left(\frac{2\pi}{2N}\right) & \dots & \cos\left(\frac{(N-1)\pi}{2N}\right) \\ \frac{1}{2} & \cos\left(\frac{3\pi}{2N}\right) & \cos\left(\frac{2\cdot 3\pi}{2N}\right) & \dots & \cos\left(\frac{(N-1)\cdot 3\pi}{2N}\right) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{1}{2} & \cos\left(\frac{(2N-1)\pi}{2N}\right) & \cos\left(\frac{2\cdot(2N-1)\pi}{2N}\right) & \dots & \cos\left(\frac{(N-1)\cdot(2N-1)\pi}{2N}\right) \end{bmatrix}^{-1} \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ \cos\left(\frac{\pi}{2N}\right) & \cos\left(\frac{3\pi}{2N}\right) & \cos\left(\frac{5\pi}{2N}\right) & \dots & \cos\left(\frac{(2N-1)\pi}{2N}\right) \\ \cos\left(\frac{2\cdot\pi}{2N}\right) & \cos\left(\frac{2\cdot 3\pi}{2N}\right) & \cos\left(\frac{2\cdot 5\pi}{2N}\right) & \dots & \cos\left(\frac{2\cdot(2N-1)\pi}{2N}\right) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \cos\left(\frac{(N-1)\cdot\pi}{2N}\right) & \cos\left(\frac{(N-1)\cdot 3\pi}{2N}\right) & \cos\left(\frac{(N-1)\cdot 5\pi}{2N}\right) & \dots & \cos\left(\frac{(N-1)\cdot(2N-1)\pi}{2N}\right) \end{bmatrix}$$

This gives rise to the following DCT formulas that are typically found (up to a scaling factor). For example, the MATLAB DCT<sup>1</sup> transformation uses the formula

$$a_k = w_k \sum_{n=1}^N f_n \cos\left(\frac{(2n-1)k\pi}{2N}\right), \quad k = 0, 2, \dots, (N-1)$$

where  $w_k$  is a scaling factor which satisfies

$$w(k) = \begin{cases} \frac{1}{\sqrt{N}}, & k = 0, \\ \sqrt{\frac{2}{N}}, & 1 \leq k \leq (N-1). \end{cases}$$

---

<sup>1</sup><http://www.mathworks.com/help/signal/ref/dct.html>