

Discrete Cosine Transform — E186 Handout

Definition of DCT

The DFT transforms a complex signal into its complex spectrum. However, if the signal is real (as in most of the applications), half of the data is redundant. (The imaginary part of the signal is all zero and both the real and imaginary parts of the spectrum are symmetry.) As a real transform, Discrete cosine transform (DCT) transforms real data into real spectrum and therefore avoids the problem of redundancy. Also as DCT is derived from DFT, all the desirable properties of DFT are reserved. The DCT of a real sequence $\{x(m), \quad (m = 0, \dots, N - 1)\}$ can be derived as below.

First, we construct a new sequence of $2N$ points

$$x'(m) \triangleq \begin{cases} x(m) & (0 \leq m \leq N - 1) \\ x(-m - 1) & (-N \leq m \leq -1) \end{cases}$$

which repeats its self outside the range $-N \leq n \leq N - 1$.

In other words, this new sequence $x'(m)$ is a periodic (period $2N$) and even symmetric with respect to the point $m = -1/2$:

$$x'(m) = x'(-m - 1) = x'(2N - m - 1)$$

If we shift the signals $x'(m)$ to the right by $1/2$, or, equivalently, shift m to the left by $1/2$ by defining another index $m' = m + 1/2$, then $x'(m) = x'(m' - 1/2)$ is even symmetric with respect to $m' = 0$. In the following we simply represent this new function by $x(m)$.

Next find the DFT of this $2N$ -point even symmetric sequence:

$$X(n) = \frac{1}{\sqrt{2N}} \sum_{m'=-N+1/2}^{N-1/2} x(m' - \frac{1}{2}) e^{-j2\pi m' n/2N} \quad (n = 0, \dots, 2N-1)$$

Replacing m' by $m + 1/2$, we get

$$\begin{aligned} X(n) &= \frac{1}{\sqrt{2N}} \sum_{m=-N}^{N-1} x(m) e^{-j2\pi(m-1/2)n/2N} = \frac{1}{\sqrt{2N}} \sum_{m=-N}^{N-1} x(m) e^{-j\pi(2m-1)n/2N} \\ &= \sqrt{\frac{2}{N}} \sum_{m=0}^{N-1} x(m) \cos\left(\frac{(2m+1)n\pi}{2N}\right) \quad (n = 0, \dots, N-1) \end{aligned}$$

The last equal sign is due to the fact that $x(m)$ is real and even symmetric. Also, for the same reason, only the first half of the data points in the spectrum, $X(n)$ ($n = 0, \dots, N-1$), is kept, as the second half is symmetric and therefore redundant. (Is $X(n)$ even or odd symmetric?)

The above equation defines the forward DCT. The element in the m th row and n th column of the transform matrix is

$$C(n, m) \triangleq \sqrt{\frac{2}{N}} \cos\left(\frac{(2m+1)n\pi}{2N}\right)$$

All row vectors are orthogonal and, except the first one ($n = 0$), normal:

$$\sqrt{\sum_{m=0}^{N-1} C^2(n, m)} = \sqrt{\sum_{m=0}^{N-1} \frac{2}{N} \cos^2\left(\frac{(2m+1)n\pi}{2N}\right)} = \begin{cases} \sqrt{2} & \text{for } n = 0 \\ 1 & \text{for } n = 1, 2, \dots, N-1 \end{cases}$$

A normalization coefficient defined as

$$c(n) = \begin{cases} \sqrt{1/N} & \text{for } n = 0 \\ \sqrt{2/N} & \text{for } n = 1, 2, \dots, N-1 \end{cases}$$

is therefore included so that the orthonormal DCT can now be written as

$$X(n) = c(n) \sum_{m=0}^{N-1} x(m) \cos\left(\frac{(2m+1)n\pi}{2N}\right) \quad (n = 0, \dots, N-1)$$

and the inverse DCT is

$$x(m) = \sum_{n=0}^{N-1} c(n) X(n) \cos\left(\frac{(2m+1)n\pi}{2N}\right) \quad (m = 0, \dots, N-1)$$

Fast DCT algorithm

Forward DCT

The DCT of a sequence $\{x(m), \quad (m = 0, \dots, N-1)\}$ can be implemented by FFT. First we define a new sequence $\{y(m), \quad (m = 0, \dots, N-1)\}$:

$$\begin{cases} y(m) \triangleq x(2m) \\ y(N-1-m) \triangleq x(2m+1) \quad (i = 0, \dots, N/2-1) \end{cases}$$

Then the DCT of $x(n)$ can be written as the following (the coefficient $c(n)$ is dropped for now for simplicity):

$$\begin{aligned} X(n) &= \sum_{m=0}^{N-1} x(m) \cos\left(\frac{(2m+1)n\pi}{2N}\right) \\ &= \sum_{m=0}^{N/2-1} x(2m) \cos\left(\frac{(4m+1)n\pi}{2N}\right) + \sum_{m=0}^{N/2-1} x(2m+1) \cos\left(\frac{(4m+3)n\pi}{2N}\right) \\ &= \sum_{m=0}^{N/2-1} y(m) \cos\left(\frac{(4m+1)n\pi}{2N}\right) + \sum_{m=0}^{N/2-1} y(N-1-m) \cos\left(\frac{(4m+3)n\pi}{2N}\right) \end{aligned}$$

where the first summation is for all even terms and second all odd terms. We define for the second summation $m' \triangleq N-1-m$, then the limits of the summation 0 and $N/2-1$ for m becomes $N-1$ and $N/2$ for m' , and the second summation can be written as

$$\sum_{m'=N/2}^{N-1} y(m') \cos\left(2n\pi - \frac{(4m'+1)n\pi}{2N}\right) = \sum_{m'=N/2}^{N-1} y(m') \cos\left(\frac{(4m'+1)n\pi}{2N}\right)$$

where the equal sign is due to

$$\cos(\alpha - \beta) = \cos \alpha \cos \beta + \sin \alpha \sin \beta$$

Now the two summations in the expression of $X(n)$ can be combined

$$X(n) = \sum_{m=0}^{N-1} y(m) \cos\left(\frac{(4m+1)n\pi}{2N}\right)$$

Next, consider the DFT of $y(m)$:

$$Y(n) = \sum_{m=0}^{N-1} y(m) e^{-j2\pi mn/N} = \sum_{m=0}^{N-1} y(m) [\cos(\frac{2\pi mn}{N}) - j \sin(\frac{2\pi mn}{N})]$$

If we multiply both sides by

$$e^{-jn\pi/2N} = \cos(\frac{n\pi}{2N}) - j \sin(\frac{n\pi}{2N})$$

and take the real part of the result (and recall both $x(m)$ and $y(m)$ are real), we get:

$$\begin{aligned} \text{Re}[e^{-jn\pi/2N} Y(n)] &= \sum_{m=0}^{N-1} y(m) [\cos(\frac{2\pi mn}{N}) \cos(\frac{n\pi}{2N}) - \sin(\frac{2\pi mn}{N}) \sin(\frac{n\pi}{2N})] \\ &= \sum_{m=0}^{N-1} y(m) \cos(\frac{(4m+1)n\pi}{2N}) \end{aligned}$$

Note the last equal sign is due to the same trigonometric equation

$$\cos(\alpha + \beta) = \cos \alpha \cos \beta - \sin \alpha \sin \beta$$

Comparing this with the expression of $X(n)$ obtained above, we see that DCT can be computed as

$$X(n) = \text{Re}[e^{-jn\pi/2N} Y(n)]$$

where $Y(n)$ is the DFT of $y(m)$ (defined from $x(m)$) which can be computed using FFT algorithm in $N \log_2 N$ time complexity.

In summary, fast forward DCT can be implemented in 3 steps:

- **Step 1** Generate a sequence $y(m)$ from the given sequence $x(m)$:

$$\begin{cases} y(m) = x(2m) \\ y(N-1-m) = x(2m+1) \end{cases} \quad (i = 0, \dots, N/2 - 1)$$

- **step 2** Obtain DFT $Y(n)$ of $y(m)$ using FFT in $N \log_2 N$ complexity. (As $y(m)$ is real, $Y(n)$ is symmetric and only half of the data points need be computed.)

$$Y(n) = \mathcal{F}[y(m)]$$

- **step 3** Obtain DCT $X(n)$ from $Y(n)$ by

$$X(n) = \text{Re}[e^{-jn\pi/2N} Y(n)]$$

Inverse DCT

The most obvious way to do inverse DCT is to reverse the order and the mathematical operations of the three steps for the forward DCT:

- **step 1** Obtain $Y(n)$ from $X(n)$. In step 3 above there are N equations but $2N$ variables (both real and imaginary parts of $Y(n)$). However, note that as $y(m)$'s are *real*, the real part of its spectrum $Y(n)$ is even symmetric ($N+1$ independent variables) and imaginary part odd symmetric ($N-1$ independent variables). So there are only N variables which can be obtained by solving the N equations.

$$X(n) = \text{Re}[e^{-jn\pi/2N} Y(n)] \quad (n = 0, \dots, N-1)$$

- **step 2** Obtain $y(m)$ from $Y(n)$ by inverse DFT also using FFT in $N \log_2 N$ complexity.

$$y(m) = \mathcal{F}^{-1}[Y(n)]$$

- **Step 3** Obtain $x(m)$ from $y(m)$ by

$$\begin{cases} x(2m) = y(m) \\ x(2m+1) = y(N-1-m) \end{cases} \quad (i = 0, \dots, N/2-1)$$

However, there is a more efficient way to do the inverse DCT. Consider first the real part of the inverse DFT of the sequence $X(n)e^{jn\pi/2N}$:

$$\begin{aligned} \text{Re} \sum_{n=0}^{N-1} [X(n)e^{jn\pi/2N}] e^{j2\pi mn/N} &= \text{Re} \sum_{n=0}^{N-1} X(n)e^{j(4m+1)n\pi/2N} \\ &= \sum_{n=0}^{N-1} X(n) \cos\left(\frac{(4m+1)n\pi}{2N}\right) = x(2m) \quad (m = 0, \dots, N-1) \end{aligned}$$

This equation gives the inverse DCT of all $N/2$ even data points $x(2m)$ ($m = 0, \dots, N/2-1$). To obtain the odd data points, recall that $x(m) = x(2N-m-1)$, and all odd data points

$$x(2m+1) = x(2N-2(m+1)-1) = x(2(N-m-1)) \quad (m = 0, \dots, N/2-1)$$

can be obtained from the second half of the previous equation in reverse order ($m = N-1, N-2, \dots, N/2$).

In summary, we have these steps to compute IDCT:

- **step 1** Generate a sequence $Y(n)$ from the given DCT sequence $X(n)$:

$$Y(n) = X(n)e^{jn\pi/2N} \quad (n = 0, \dots, N-1)$$

- **step 2** Obtain $y(m)$ from $Y(n)$ by inverse DFT also using FFT in $N \log_2 N$ complexity. (Only the real part need be computed.)

$$y(m) = \text{Re}[\mathcal{F}^{-1}[Y(n)]]$$

- **Step 3** Obtain $x(m)$'s from $y(m)$'s by

$$\begin{cases} x(2m) = y(m) \\ x(2m+1) = y(N-1-m) \end{cases} \quad (i = 0, \dots, N/2-1)$$

These three steps are mathematically equivalent to the steps of the first method.