2026 年度秋学期　画像情報処理　第９回

第２部・画像情報圧縮／ 離散フーリエ変換と離散コサイン変換

前々回に説明した KL 変換は，画像の「重要でない成分をごまかす」ことをもっとも効率的に行い，画像データを圧縮する方法を示しましたが，実行するには画像の統計的性質がわからなければならないという問題点がありました。また，前回は，画像を行列で表し，縦方向と横方向それぞれに作用する直交変換・ユニタリー変換を行う枠組みを示して，基底画像の考えを導きました。これにより，「基底画像にそれぞれ倍率をかけて足し合わせることで，元の画像が表現される」という仕組みが得られました。この方法では，基底画像は行列の形で表されて，目に見えますから，どのような基底画像を作って，どこをごまかせば効率よくデータ圧縮ができるかを，目で見て考えることができます。

基底画像の考えは，フーリエ変換のときに出てきた「いろいろな周波数の三角関数（指数関数）にそれぞれフーリエ係数をかけて足し合わせることで，元の関数が表現される」という考えと同じものです。そこで，今回は，変換としてフーリエ変換を用いることにします。このときの「ごまかし」の方法は，高い周波数に対応する成分は，各画像間で違いが少ないので，省略してもかまわない，というものです。画像情報圧縮の規格として広く用いられている JPEG 方式には，フーリエ変換から派生したコサイン変換が用いられています。

今回は，前回説明した直交変換の枠組みでフーリエ変換を表し，フーリエ変換・コサイン変換による画像情報圧縮の方法を説明します。

2 次元離散フーリエ変換

2 次元フーリエ変換と 2 次元離散フーリエ変換

この講義の第 1 部で，2 次元の関数 $f(x,y)$ のフーリエ変換を

$$F(\nu_x,\nu_y)=\iint_{-\infty}^{\infty} f(x,y)\exp\{-i2\pi(\nu_x x+\nu_y y)\}dxdy \tag{1}$$

と定義しました。この式は

$$\begin{aligned} F(\nu_x,\nu_y) &= \iint_{-\infty}^{\infty} f(x,y)\exp(-i2\pi\nu_x x)\exp(-i2\pi\nu_y y)dxdy \\ &= \int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} f(x,y)\exp(-i2\pi\nu_x x)dx\right]\exp(-i2\pi\nu_y y)dy \end{aligned} \tag{2}$$

と書き直すことができるので，x,y それぞれの方向について 1 次元のフーリエ変換を行なっていることに相当します。また，1 次元の N 項の数列 $u(n)$ の離散フーリエ変換は

$$U(k)=\sum_{n=0}^{N-1} u(n)\exp(-i2\pi\frac{k}{N}n)\quad(k=0,1,\ldots,N-1) \tag{3}$$

となることも，第 5 回で説明しました。これらのことから，m 方向に M 項，n 方向に N 項の 2 次元数列 $u(m,n)$ の 2 次元離散フーリエ変換は

$$U(k,l) = \sum_{n=0}^{N-1} \left[\sum_{m=0}^{M-1} u(m,n) \exp(-i2\pi \frac{k}{M} m) \right] \exp(-i2\pi \frac{l}{N} n) \tag{4}$$

$(k = 0, 1, \ldots, M-1, l = 0, 1, \ldots, N-1)$ と定義することができます。とくに $M = N$ の場合，つまり正方形のディジタル画像のフーリエ変換を想定すると，(4) 式は

$$U(k,l) = \sum_{n=0}^{N-1} \left[\sum_{m=0}^{N-1} u(m,n) \exp(-i2\pi \frac{k}{N} m) \right] \exp(-i2\pi \frac{l}{N} n) \quad (k,l = 0, 1, \ldots, N-1) \tag{5}$$

となります。

分離可能なユニタリー変換の形であらわす

さて，前回の講義で，行列 X の縦横に分離可能 (separable) なユニタリー変換は，ユニタリー行列 R を使って

$$Z = RXR' \tag{6}$$

と表せることを示しました。この形式で (5) 式の変換を表すことを考えてみましょう。(6) 式の行・列と (5) 式の各変数を対応させると

$$^{l\downarrow}(Z = \overset{k\rightarrow}{U}(k,l)) = {}^{l\downarrow}\overset{n\rightarrow}{(R)} \cdot {}^{n\downarrow}(X = \overset{m\rightarrow}{u}(m,n)) \cdot {}^{m\downarrow}\overset{k\rightarrow}{(R')} \tag{7}$$

という形になります。そこで，(5) 式の $\sum$ の計算と (7) 式の行列の演算を対応させると，

$$R' = {}^{m\downarrow}\begin{pmatrix} e^{-i2\pi\frac{0}{N}0} & \cdots & \overset{k\rightarrow}{e^{-i2\pi\frac{k}{N}0}} & \cdots & e^{-i2\pi\frac{N-1}{N}0} \\ \vdots & \ddots & & & \\ e^{-i2\pi\frac{0}{N}m} & & e^{-i2\pi\frac{k}{N}m} & & \\ \vdots & & & \ddots & \\ e^{-i2\pi\frac{0}{N}(N-1)} & & & & e^{-i2\pi\frac{N-1}{N}(N-1)} \end{pmatrix}$$

となり，また

$$R = {}^{l\downarrow}\begin{pmatrix} e^{-i2\pi\frac{0}{N}0} & \cdots & \overset{n\rightarrow}{e^{-i2\pi\frac{0}{N}n}} & \cdots & e^{-i2\pi\frac{0}{N}(N-1)} \\ \vdots & \ddots & & & \\ e^{-i2\pi\frac{l}{N}0} & & e^{-i2\pi\frac{l}{N}n} & & \\ \vdots & & & \ddots & \\ e^{-i2\pi\frac{N-1}{N}0} & & & & e^{-i2\pi\frac{N-1}{N}(N-1)} \end{pmatrix} \tag{8}$$

となります。これらは対称行列ですから，

$$W_N = \exp(-\frac{i2\pi}{N}) \tag{9}$$

とおいて

$$R = \begin{pmatrix} W_N^{0\cdot 0} & \cdots & W_N^{0\cdot n} & \cdots & W_N^{0\cdot(N-1)} \\ \vdots & \ddots & & & \\ W_N^{l\cdot 0} & & W_N^{ln} & & \\ \vdots & & & \ddots & \\ W_N^{(N-1)\cdot 0} & & & & W_N^{(N-1)(N-1)} \end{pmatrix} \tag{10}$$

とすると

$$Z = RXR \tag{11}$$

と表すことができます。

実はユニタリーではないので

ところで，行列 R と行列 R'^* の積は $RR'^* = NI$ となります（付録を参照してください）。R がユニタリー行列ならば $RR'^* = I$ になるはずで，R はユニタリー行列ではありません。

そこで，RR'^* は単位行列の N 倍になっているわけですから，R をユニタリー行列とするには，W_N をすこし変えて，

$$W_N = \frac{1}{\sqrt{N}} \exp(-\frac{i2\pi}{N}) \tag{12}$$

と定義すればよいことになります。このとき，逆変換は

$$X = R^* Z R^* \tag{13}$$

となります。このように定義した離散フーリエ変換を，とくに**ユニタリー離散フーリエ変換** (unitary DFT) ということもあります。

フーリエ変換は「座標の回転」

このように，離散フーリエ変換がユニタリー変換の一種であるということは，離散フーリエ変換も「座標の回転」の一種であり，いわば「画像を表す空間の見方を変えただけ」ということができます。もともとは「画素値の並び」だったのが，「『波』を表す基底画像の組み合わせ」に変わったわけです。

2 次元離散フーリエ変換の要素と周波数

さて，第 1 部で説明したとおり，もとの関数をサンプリングすると，周波数空間ではもとの関数の周波数分布が周期的に繰り返されることになります。離散フーリエ変換は，もとの周波数分布の周波数 0 のところから 1 周期分を取り出して N 等分にサンプリングしたものに相当しています。したがって，第

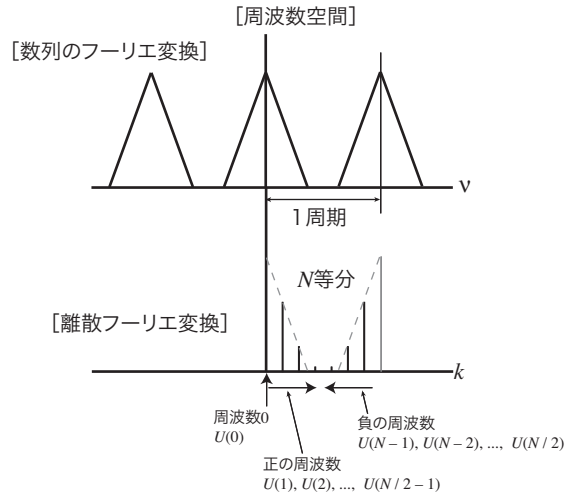


図 1: 離散フーリエ変換における要素と周波数

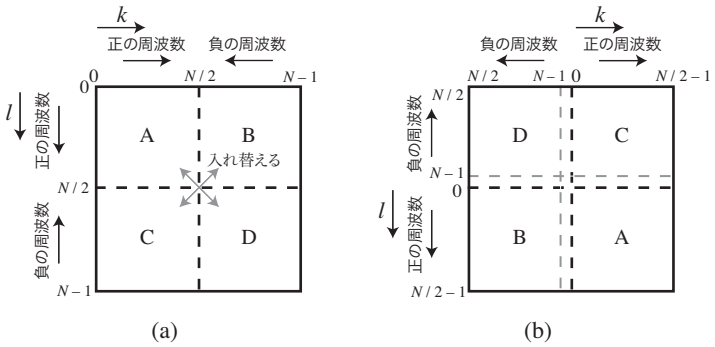


図 2: 2次元離散フーリエ変換における要素と周波数

5回の付録2で説明したとおり，1次元の離散フーリエ変換においては，$U(0)$ が元の周波数分布の周波数0に，$U(1), U(2), \ldots, U(N/2-1)$ までが正の周波数に，$U(N-1), U(N-2), \ldots, U(N/2)$ 個は逆順に負の周波数に相当します（図1)。また，実関数のフーリエ変換は振幅が偶関数，位相が奇関数となることが知られています[1]。このことは，これもで第5回の付録2で説明したとおり，したとおり，(3) 式で定義される，実数列の1次元の離散フーリエ変換について，

$$U^*(N-k) = U(k) \tag{14}$$

がなりたつことに対応しています。

2次元離散フーリエ変換においては，縦横両方向に同じ原理が成り立ちますから，2次元離散フーリエ変換の要素は図2(a) のように周波数に対応します。これは，図2(b) のように領域を入れ替えるとわかりやすくなります。

また，実行列の2次元離散フーリエ変換においては，(14) 式と同様に

[1]関数 $f(x)$ が，任意の x について $f(x) = f(-x)$ をみたすとき，$f(x)$ は偶関数であるといいます。また，$f(-x) = -f(x)$ であるとき，$f(x)$ は奇関数であるといいます。

$$U(k,l) = U^*(M-k, N-l) \tag{15}$$

の関係がなりたつので，2 次元離散フーリエ変換においても全体の 1/4 の領域の値が決まれば他の値は全て決まることになります。

離散コサイン変換と画像圧縮

複素数を使わずに変換したい

画像では一般に，低い周波数成分は画像中の物体のおおまかな形を表し，高い周波数成分は物体の周辺の細かな様子を表しています。したがって，高い周波数成分はその値も小さく，省略しても画像が伝える情報はそれほど損なわれません。したがって第 7 回の KL 変換のところで示した考え方にしたがうと，高い周波数成分を省略することで画像圧縮ができます。

離散コサイン変換

しかし，フーリエ変換は複素数を扱う必要があるため，計算機での取り扱いがいくぶん面倒です。そこで，JPEG 方式をはじめ，実際の画像圧縮でよく用いられているのが，**離散コサイン変換** (discrete cosine transformation, DCT) です。離散コサイン変換では，行列 R は (10) 式のかわりに

$$\begin{aligned} R &= \begin{array}{c} \\ l\downarrow \end{array} \begin{array}{c} \overset{n\rightarrow}{} \\ \begin{pmatrix} \ddots & & \\ & r(n,l) & \\ & & \ddots \end{pmatrix} \end{array}, \\ r(n,l) &= \begin{cases} \frac{1}{\sqrt{N}} & l = 0 \\ \frac{2}{\sqrt{N}} \cos \frac{(2n+1)l\pi}{2N} & l \neq 0 \end{cases} \end{aligned} \tag{16}$$

と定義されます。離散コサイン変換では，R は実行列になっています。

偶関数のフーリエ変換

ところで，偶関数のフーリエ変換は実関数になります[2]。このことを，1 次元の場合に見てみましょう。関数 $f(x)$ が偶関数であるとは，任意の x について $f(x) = f(-x)$ であることをいいます。このとき，$f(x)$ のフーリエ変換 $F(\nu)$ は

$$\begin{aligned} F(\nu) &= \int_{-\infty}^{\infty} f(x) \exp\left(-i2\pi(\nu x)\right) dx \\ &= \int_{-\infty}^{0} f(x) \exp\left(-i2\pi(\nu x)\right) dx + \int_{0}^{\infty} f(x) \exp\left(-i2\pi(\nu x)\right) dx \end{aligned} \tag{17}$$

となります。ここで，第 1 項の x を $-x$ に変数変換すると

[2] フーリエコサイン変換といいます。

$$
\begin{aligned}
F(\nu) &= \int_{\infty}^{0} f(-x)\exp\left(-i2\pi(\nu(-x))\right)(-dx) + \int_{0}^{\infty} f(x)\exp\left(-i2\pi(\nu x)\right)dx \\
&= \int_{\infty}^{0} f(-x)\exp\left(i2\pi(\nu x)\right)(-dx) + \int_{0}^{\infty} f(x)\exp\left(-i2\pi(\nu x)\right)dx
\end{aligned}
\tag{18}
$$

であり，$f(x) = f(-x)$ を使って整理すると

$$
\begin{aligned}
F(\nu) &= \int_{\infty}^{0} f(x)\exp\left(i2\pi(\nu x)\right)(-dx) + \int_{0}^{\infty} f(x)\exp\left(-i2\pi(\nu x)\right)dx \\
&= \int_{0}^{\infty} f(x)\exp\left(i2\pi(\nu x)\right)dx + \int_{0}^{\infty} f(x)\exp\left(-i2\pi(\nu x)\right)dx \\
&= \int_{0}^{\infty} f(x)\left[\exp\left(i2\pi(\nu x)\right)dx + \exp\left(-i2\pi(\nu x)\right)\right]dx
\end{aligned}
\tag{19}
$$

となります。よって，指数関数と三角関数の関係から

$$
F(\nu) = 2\int_{0}^{\infty} f(x)\cos 2\pi(\nu x)dx \tag{20}
$$

となるので，実数の計算として扱えることがわかります。

離散コサイン変換と偶関数のフーリエ変換

離散コサイン変換は，元の画像を縦横とも座標軸に対称に折り返し，偶関数にしたものの離散フーリエ変換に相当します。偶関数のフーリエ変換なので，実関数として計算できるわけです。1 次元の離散コサイン変換の場合にこれを見てみましょう。元の 1 次元信号を N 要素の数列 $u(0), u(1), \ldots, u(N-1)$ とするとき，これを折り返した数列は $2N$ 要素の $u(N-1), u(N-2), \ldots, u(1), u(0), u(0), u(1), \ldots, u(N-1)$ となります。N 要素の 1 次元の離散コサイン変換は (16) 式から

$$
U(k) = \begin{cases} \frac{1}{\sqrt{N}}\sum_{n=0}^{N-1} u(n) & k = 0 \\ \frac{2}{\sqrt{N}} u(n)\cos\frac{(2n+1)k\pi}{2N} & k \neq 0 \end{cases} \tag{21}
$$

となります。そこで，$k \neq 0$ のとき

$$
\begin{aligned}
U(k) &= \frac{2}{\sqrt{N}}\sum_{n=0}^{N-1} u(n)\cos\frac{(2n+1)k\pi}{2N} \\
&= \frac{1}{\sqrt{N}}\sum_{n=0}^{N-1} u(n)\exp\frac{i(2n+1)k\pi}{2N} + \frac{1}{\sqrt{N}}\sum_{n=0}^{N-1} u(n)\exp\frac{-i(2n+1)k\pi}{2N} \\
&= \frac{1}{\sqrt{N}}\sum_{n=0}^{N-1} u(n)\exp\frac{-i((-n)-1/2)k\pi}{2N} + \frac{1}{\sqrt{N}}\sum_{n=0}^{N-1} u(n)\exp\frac{-i(n+1/2)k\pi}{2N}
\end{aligned}
\tag{22}
$$

となります。これは，$u(N-1), u(N-2), \ldots, u(1), u(0), u(0), u(1), \ldots, u(N-1)$ という数列に対して，折り返し点である 2 つの $u(0)$ の中間に仮に要素があると考えてこれを 0 番と想定し，それより後を $1/2$ 番，$3/2$ 番，…，前を $-1/2$ 番，$-3/2$ 番，…，と番号をつけて，離散フーリエ変換を行ったことに相当しています。$2N$ 項の数列のフーリエ変換をしているわけですから，$k=0$ が周波数 0 に相当し，以後 $k=1,\ldots,N-1$ と番号が大きいほど高い周波数に対応する成分となります。

したがって，離散コサイン変換では，$N \times N$ の実数要素の行列を変換すると $N \times N$ の実数要素の行列が得られ，$k,l=0,1,\ldots,N-1$ と番号が大きいほど高い周波数に相当するので，離散フーリエ変換のように周波数空間で折り返しをする必要はありません。したがって，k,l ともいくつかの番号までかを限定して残し他を省略することで，高周波成分を省略し情報量を圧縮することができます。画像圧縮でよく用いられている JPEG 方式では，画像を 8×8 画素の領域に分割し，各領域で離散コサイン変換を行って，それぞれの領域で高周波成分を省くことで情報圧縮を行っています。

付録：行列 R と行列 R'^* の積

行列 R と行列 R'^* の積を求めます。行列 R の，第 n 列と第 n' 列の複素共役の内積を求めると

$$\begin{aligned}\sum_{l=0}^{N-1} W_N^{ln}\cdot(W_N^{ln'})^* &= \sum_{l=0}^{N-1}\exp(-\frac{i2\pi ln}{N})\exp(\frac{i2\pi ln'}{N})\\ &= \sum_{l=0}^{N-1}\exp(-\frac{i\{(n-n')2\pi\}l}{N})\\ &= \sum_{l=0}^{N-1} W_N^{(n-n')l}\end{aligned} \tag{A1}$$

となります。この値は，$n \neq n'$ のとき

$$\begin{aligned}\sum_{l=0}^{N-1} W_N^{(n-n')l} &= \frac{1-W_N^{(n-n')N}}{1-W_N^{(n-n')}}\\ &= \frac{1-\left(W_N^N\right)^{(n-n')}}{1-W_N^{(n-n')}}\\ &= \frac{1-1^{(n-n')}}{1-W_N^{(n-n')}}=0\end{aligned} \tag{A2}$$

で，$n=n'$ のときは

$$\sum_{l=0}^{N-1} W_N^{(n-n')l} = \sum_{l=0}^{N-1} 1 = N \tag{A3}$$

となります。したがって $RR'^* = NI$ となります。