

情報デザイン専攻

画像情報処理論及び演習II

- 周波数分解 -

フーリエ変換、DCTと周波数操作

第2回講義
水曜日 1 限
教室 6218

吉澤 信
shin@riken.jp, 非常勤講師
大妻女子大学 社会情報学部

独立行政法人
理化学研究所

Shin Yoshizawa: shin@riken.jp

今日の授業内容

www.riken.jp/briect/Yoshizawa/Lectures/index.html
www.riken.jp/briect/Yoshizawa/Lectures/Lec14.pdf

1. フーリエ変換と周波数操作.
2. 演習: Discrete Cosine Transform (DCT, 離散コサイン変換)によるフィルタ処理.

今日の演習は最初のレポートで出すので、
みなさん頑張ってくださいねーp(^)q

Shin Yoshizawa: shin@riken.jp

周波数分解

入力画像 → フーリエ変換、Wavelet変換、KL展開等の関数展開 → 周波数・係数列

空間領域 $f(x, y)$ → フーリエ変換 → 周波数領域 $F(u, v)$
フーリエ逆変換

COG-ARTS協会

Shin Yoshizawa: shin@riken.jp

周波数操作

入力画像 → 変換 → 周波数 → 処理 → 処理後の周波数 → 逆変換 → 出力画像

入力画像 $f(x, y)$ → フーリエ変換 → 周波数 $F(u, v)$ → COG-ARTS協会 → 周波数 $H(u, v)$ → フーリエ逆変換 → 出力画像 $g(x, y)$

Shin Yoshizawa: shin@riken.jp

周波数(Frequency)

- ✓ 周波数・振動数: 波動・振動周期の逆数(1/周期).
- ✓ 周期(Period): 1循環するまでの時間.
- ✓ 振幅(Amplitude): 振動の大きさ.

低周波: ゆるやか = 大きな特徴

高周波: こまやか = シャープな特徴

振幅 周期 時間

Shin Yoshizawa: shin@riken.jp

関数展開

- ✓ 関数を基底と係数の1次(線形)結合で表す事.
- ✓ 例えば三角関数を基底とすると...

$$f(x) = \frac{a_0}{2} + a_1 \cos \frac{\pi x}{L} + a_2 \cos \frac{2\pi x}{L} + a_3 \cos \frac{3\pi x}{L} + \dots$$

$$+ b_1 \sin \frac{\pi x}{L} + b_2 \sin \frac{2\pi x}{L} + b_3 \sin \frac{3\pi x}{L} + \dots$$

低周波 ←→ 高周波

係数 = 周波数成分: a, b 基底: sin, cos

関数 = 係数 × 基底

Shin Yoshizawa: shin@riken.jp

関数展開2

✓ 1次元では…(音声・信号処理等)

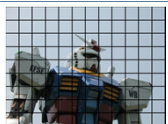
関数 $f(x)$ = 係数 $\begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix}$ × 基底 $\begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix}$

低周波
高周波

✓ 2次元では…(画像処理等)

関数 $f(u, v)$ = 係数 $\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}$ × 基底 $\begin{pmatrix} v_{11} & v_{12} & \dots & v_{1n} \\ v_{21} & v_{22} & \dots & v_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ v_{n1} & v_{n2} & \dots & v_{nn} \end{pmatrix}$

低周波
高周波



Shin Yoshizawa: shin@riken.jp

フーリエ級数

✓ フーリエ級数: $[-L, L]$ のパターンを繰り返す周期関数を、 $\sin(x)$ と $\cos(x)$ の和で表す。

$$f(x) = \frac{a_0}{2} + a_1 \cos \frac{\pi x}{L} + a_2 \cos \frac{2\pi x}{L} + a_3 \cos \frac{3\pi x}{L} + \dots$$

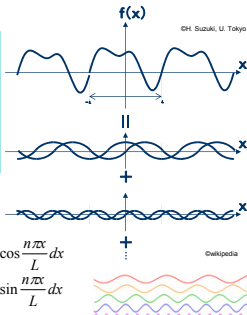
$$+ b_1 \sin \frac{\pi x}{L} + b_2 \sin \frac{2\pi x}{L} + b_3 \sin \frac{3\pi x}{L} + \dots$$

$$= \frac{a_0}{2} + \sum_{n=1}^{\infty} \left(a_n \cos \frac{n\pi x}{L} + b_n \sin \frac{n\pi x}{L} \right)$$

$$f(x) = \sum_{n=-\infty}^{\infty} c_n e^{i \frac{n\pi}{L} x} \quad e^{i\theta} = \cos \theta + i \sin \theta$$

周波数成分: 基底の係数 a_n と b_n の決め方は、 $f(x)$ に \cos, \sin をかけて積分。

$$a_n = \frac{1}{L} \int_{-L}^L f(x) \cos \frac{n\pi x}{L} dx$$

$$b_n = \frac{1}{L} \int_{-L}^L f(x) \sin \frac{n\pi x}{L} dx$$


Shin Yoshizawa: shin@riken.jp

フーリエ変換

✓ もとの関数 $f(x)$ から、別の関数 $F(k)$ への変換:

順変換 $F(k) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(x) e^{-ikx} dx$

逆変換 $f(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} F(k) e^{ikx} dk$

$e^{i\theta} = \cos \theta + i \sin \theta$

フーリエ変換の性質

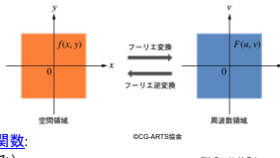
$f(x) + g(x) \Rightarrow F(k) + G(k)$, $af(x) \Rightarrow aF(k)$

- ある関数 $f(x)$ を $F(k)$ の積分(〜和)。
- $F(k)$ は $f(x)$ から積分によって計算。
- $f(x)$ と $F(k)$ の式は対称。
- $f(x)$ が実数の関数でも、 $F(k)$ は一般に複素関数。
 $f(x)$ が偶関数の場合には $F(k)$ は実関数 (cosのみ)

空間領域 $f(x, y)$ → フーリエ変換 → 周波数領域 $F(u, v)$

逆変換: 周波数領域 $F(u, v)$ → フーリエ逆変換 → 空間領域 $f(x, y)$

は規格化係数なので、あまり気にしないで良い



Shin Yoshizawa: shin@riken.jp

離散フーリエ変換

✓ デジタル画像:

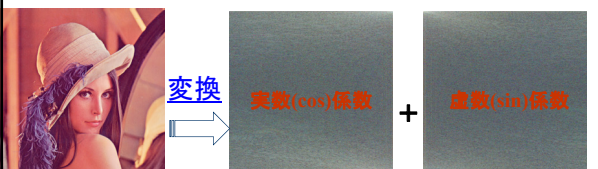
- フーリエ変換の式は連続関数に対するもの。
- デジタル画像はサンプリングされて、飛び飛び(離散データ)。

✓ 離散フーリエ変換・逆変換:

- 離散データに対するフーリエ変換・逆変換。
- 変換の結果の周波数列も離散的に求まる。

1024x1024の画像 → 2x1024x1024の係数列: ($k = 0, 1, \dots, N-1$)

変換: 実数(cos)係数 + 虚数(sin)係数



Shin Yoshizawa: shin@riken.jp

離散フーリエ変換2

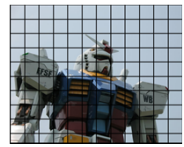
✓ 画像では2次元なので…

$$F(u, v) = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} f(i, j) \exp(\dots)$$

- 1画素の離散フーリエ変換を計算するのに全ての画素の重み付和が必要! → 全ての画素の変換を計算するには入力画素数の2乗に比例する計算量が必要!

✓ 高速フーリエ変換(FFT): 次週やります。

- サンプリング数が2のべき乗(例: 512や1,024)の時に高速に計算する方法(Nのべき乗の方法もある)。



Shin Yoshizawa: shin@riken.jp

フーリエ変換2

✓ パワースペクトル: 周波数の強度。

$$|F(u, v)|^2 = \text{Re}\{F(u, v)\}^2 + \text{Im}\{F(u, v)\}^2$$

実数(cos)係数
虚数(sin)係数

画像処理ではよく象限を1→3, 2→4, 3→1, 4→2と入れ替えた画像を用いる。(周期性を用いた離散変換を行うため)



Shin Yoshizawa: shin@riken.jp

フーリエ変換3

2次元の基本波

$f(x, y) = A \cos(ux + vy)$

縦方向に太い縞 (低い周波数成分) 横方向に細い縞 (高い周波数成分)

CCG-ARTS協会

(a) 画像 (正弦波) (b) (a)のフーリエスペクトル (c) (b)の輪郭図 (ピーク位置)

(d) 画像 (方形波) (e) (d)のフーリエスペクトル (f) (e)における (c)の断面

Shin Yoshizawa: shin@riken.jp

フーリエ変換4

(a) 画像 (b) 画像 (c) 画像

(d) (a)のフーリエスペクトル (e) (b)のフーリエスペクトル (f) (c)のフーリエスペクトル

Shin Yoshizawa: shin@riken.jp

離散コサイン変換(DCT)

✓ 余弦関数列(cos)のみを基底に用いた変換:

- 入力を y 軸で折り返して偶関数化して離散フーリエ変換する事と同義: 離散フーリエ変換は実数に対して複素数を返すのに対して、DCTは常に実数を返す。
- 低周波成分に集中度が上がるため圧縮やフィルタ処理でよく用いられている。

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

逆変換 $f(x, y) = \frac{4}{MN} \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} C(i)C(j) \cos\left(\frac{2(j+1)x\pi}{2M}\right) \cos\left(\frac{2(i+1)y\pi}{2N}\right) F(i, j)$

$C(x) = \begin{cases} 1 & (x=0) \\ \sqrt{2} & (x \neq 0) \end{cases}$

偶関数の例 奇関数の例

Shin Yoshizawa: shin@riken.jp

離散コサイン変換(DCT)2

低周波 高周波

DCT

低周波成分のみで逆変換 高周波成分も使って逆変換

Shin Yoshizawa: shin@riken.jp

離散コサイン変換(DCT)3

✓ 非常に処理が重いので、FFTを使わない簡単な実装は画像を部分画像(ブロック)に分割してブロック毎に変換する:

ブロック

DCT

32x32のブロック毎のDCT例:

Shin Yoshizawa: shin@riken.jp

周波数フィルタリング

✓ 画像のフーリエ変換:

- 空間領域から周波数領域へ。
- フーリエ逆変換すれば、画像になる。

入力画像 $f(x, y)$ 出力画像 $g(x, y)$

フーリエ変換 フーリエ逆変換

Hで周波数特性を操作.

$G(u, v) = F(u, v)H(u, v)$

CCG-ARTS協会

©H. Suzuki, U. Tokyo

✓ フーリエ変換して、画像を周波数領域に変換してしまえば、フィルタリングは、二つの関数を単純に掛け算するだけ。

Shin Yoshizawa: shin@riken.jp

周波数操作

入力画像 $\xrightarrow{\text{変換}}$ 周波数 $\xrightarrow{\text{処理}}$ 処理後の周波数 $\xrightarrow{\text{逆変換}}$ 出力画像

入力画像 $f(x, y)$

↓ フーリエ変換

$F(u, v)$

↓ ECG-ART技法

$F(u, v)H(u, v)$

↑ フーリエ逆変換

出力画像 $g(x, y)$

Shin Yoshizawa: shin@riken.jp

ローパスフィルタ

✓ $-u_0$ から、 u_0 までの低周波数成分だけ残す。

周波数の高い横方向の波(縦線)を消す。

Shin Yoshizawa: shin@riken.jp

ローパスフィルタ2

$(u, v)=(0,0)$ のフィルタの値が1なので、 $(u, v)=(0,0)$ の成分が保存される
 → 画像の平均的な明るさが保持される。

Shin Yoshizawa: shin@riken.jp

ハイパスフィルタ

✓ $-u_0$ から、 u_0 までの高周波数成分だけ残す。

$H_{high}(u, v) = 1 - H_{low}(u, v)$

✓ 1からローパスを引く:

✓ **バンドパスフィルタ: 特定周波数成分の抽出。**

Shin Yoshizawa: shin@riken.jp

高域強調フィルタ

✓ ハイパスフィルタから作る事が出来る。

$H_{h-emph}(u, v) = 1 + kH_{high}(u, v)$

エッジ強調!

Shin Yoshizawa: shin@riken.jp

ギース現象・Overshooting

✓ ローパスフィルタ=高周波成分の切り捨てはデータにエッジがあった場合に不連続なデータを連続関数で近似するためエッジ周辺での誤差が非常に大きくなる事: 画像ではリングアーティファクトと呼ばれている: 圧縮や補間等でのカーネルの打ち切り誤差。

Shin Yoshizawa: shin@riken.jp

ギブス現象・Overshooting²

©MathWorld

Shin Yoshizawa: shin@riken.jp

その他の変換

✓ フーリエ変換以外にも、様々な基底を用いた関数展開が幅広く周波数解析に用いられている。

- KL(Karhunen-Loeve)展開: データの共分散行列の固有ベクトルを基底とする。最小二乗的に最もデータを近似出来る展開。

←主成分分析(PCA)の一般化

PCA: Principal Component Analysis

主成分分析: 与えられた点群データに対して最小二乗的に最も相関が強い方向と強度を計算する。

- 直線、平面、Hyperplane 等のデータへの当てはめ(最小二乗近似)
- Covariance matrix(共分散行列)=平均からの差の固有値。ベクトルは Best fit 楕円、ellipsoid等の近似。

データ点 固有ベクトル × 固有値

重心

- 球面調和関数: 超球面上の関数空間の正規直交基底(円や球への離散化で回転非依存にしやすい)。
- Zernike関数、固有関数展開、etc.

©MathWorld

Shin Yoshizawa: shin@riken.jp

その他の変換²

✓ Wavelet: 入力信号を小さな波形の拡大縮小と平行移動の重ね合わせで表現。

- フーリエ変換は時間軸上で常に一定のパターンを持ったデータ解析に有用だが、時刻によってパターンが変化するデータ解析には不向きである。ウェーブレット変換では局所的な波を平行移動と拡大縮小で波を表現するため、有限の区間内にあるデータの特性を解析するには三角関数より適している。
- 多重解像度解析(Multiresolution Analysis): パターンを周波数分解する作業を繰り返して特徴を解析。

Shin Yoshizawa: shin@riken.jp

周波数分解と操作

入力画像 $f(x, y)$ → 変換 → 周波数 $F(u, v)$ → 処理 → 処理後の周波数 $G(u, v)$ → 逆変換 → 出力画像 $g(x, y)$

フーリエ変換 \leftrightarrow FFT \leftrightarrow フーリエ逆変換

Shin Yoshizawa: shin@riken.jp

演習: DCT

www.riken.jp/brict/Yoshizawa/Lectures/index.html
www.riken.jp/brict/Yoshizawa/Lectures/Lec14.pdf

離散コサイン変換による周波数フィルタ

www.riken.jp/brict/Yoshizawa/Lectures/Ex08.zip

前回の演習(BMPとPPMの相互変換)が分からなかった or 出来ていない or 欠席した人は、前回の演習から始める事!

Shin Yoshizawa: shin@riken.jp

演習: Ex08の説明

✓ DCT.h: 離散コサイン変換のブロック実装:

✓ 順変換: `Image *DCT(Image *in, int X, int Y)`: 入力画像inをX×YブロックでDCTを実行し周波数画像を戻り値で返す。

- 注意点: 周波数画像は入力画像サイズがブロックサイズで割り切れない場合は入力画像サイズより少し大きなサイズで作成される。

✓ 逆変換:

`void InverseDCT(Image *dct, Image *out, int X, int Y)`: DCT()にて変換された画像dctをX×Yブロックで逆変換し出力画像outへ保存。

Shin Yoshizawa: shin@riken.jp

演習: Ex08の説明2

- ✓ testDCT.cxx: 離散コサイン変換の例.
- ✓ makeでコンパイル.
- ✓ 引数4つ:
 1. 入力BMPファイル名.
 2. 出力ファイル名(ただし拡張子.bmpなし).
 3. ブロックサイズ(int).
 4. 周波数の閾値(int): 高周波を四角にカット.

/testDCT 入力BMP 出力ファイル名(.bmp抜き) ブロックサイズ(int) 周波数の閾値(int)

- ✓ 出力は3つのbmp画像ファイル:
 - 出力ファイル名_spectrum.bmp
 - 出力ファイル名_smooth.bmp
 - 出力ファイル名_smooth_spectrum.bmp

Shin Yoshizawa: shin@riken.jp

演習: 周波数フィルタ

- ✓ testDCT.cxxを編集して円状に高周波をゼロにするローパスフィルタを作ってみましょう!
- ✓ 16x16のブロックで半径1,2,3,4,8で実行してみてください.
- ✓ ヒント: testDCT.cxxは四角に低周波を残しているの、円状にするだけ.

Shin Yoshizawa: shin@riken.jp

演習の正解例

- ✓ 32x32のブロックで実行した場合:

Shin Yoshizawa: shin@riken.jp

演習の正解例2

- ✓ 32x32のブロックで実行した場合:

- ✓ Ex08中のSeikai.zip内にStrasborug2.bmpでの正解画像が入っています.

Shin Yoshizawa: shin@riken.jp

演習: 周波数フィルタ2

- ✓ 以下の周波数フィルタのプログラムを作ってみましょう!
- 1. ローパスフィルタ: 円状に高周波をゼロにする方法とガウス関数を使う方法両方.
- 2. ハイパスフィルタ: 円状に高周波をゼロにする方法.
- 3. バンドパスフィルタ: 円状に高周波をゼロにする方法.
- 4. エッジ強調フィルタ: 円状に高周波をゼロにする方法とガウス関数を使う方法両方.

- ヒント: ガウス関数の画像を作って正規化(画素の和で割る)し、DCT後に入力のDCT画像とかけて逆変換.

$$g(x, y) = \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

第1回レポートは↑を含むので頑張って-p(^.^)q

Shin Yoshizawa: shin@riken.jp

再来週の予定

- ✓ 今日の演習の続きとFFT、ピラミッド表現等.
- ✓ 10/5は体育祭で講義はありません.

内容(2-3): 周波数分解

- 1: フーリエ変換と周波数操作、多重解像度解析、ピラミッド等.
- 2: DCT、FFT、KL展開等.

1回	画像フォーマット
2回	周波数分解
3回	フィルタ処理・エッジ強調
4回	計算Photography・Artistic Stylization
5回	計算Photography・Artistic Stylization
6回	計算Photography・Artistic Stylization
7回	計算Photography・Artistic Stylization
8回	計算Photography・Artistic Stylization
9回	計算Photography・Artistic Stylization
10回	動画処理
11回	動画処理
12回	動画処理
13回	エッジ・形状・特徴抽出とパターン認識の基礎
14回	エッジ・形状・特徴抽出とパターン認識の基礎
15回	補講

ES. Yoshizawa, RIKEN