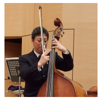
2016年度秋学期　画像情報処理　第８回

行列の直交変換と基底画像

浅野　晃
関西大学総合情報学部

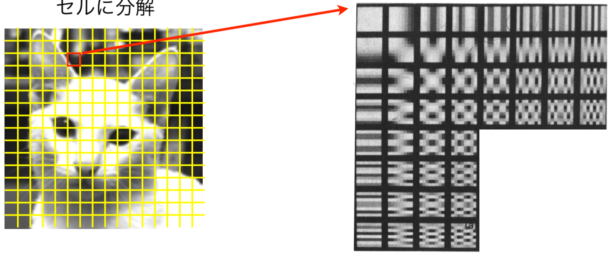


JPEG方式による画像圧縮

画像を波の重ね合わせで表わし，
一部を省略して，データ量を減らす

8×8ピクセルずつの
セルに分解

ひとつのセルを，
これらの波の重ね合わせで表す

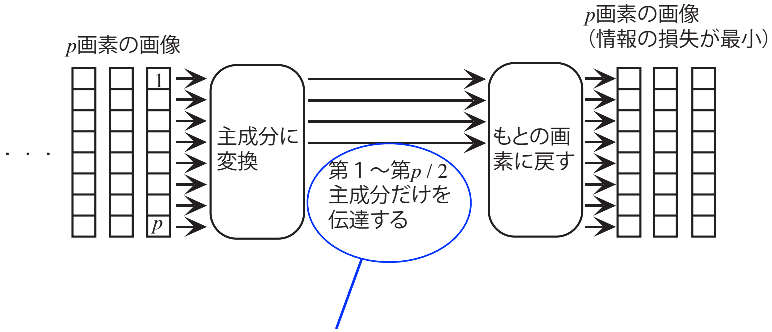


細かい部分は，どの画像でも大してかわらないから，省略しても気づかない

省略すると，データ量が減る

Karhunen-Loève変換（KL変換）

画像を主成分に変換してから伝送する



データ量が半分でも
情報の損失は最小

KL変換の大問題

主成分を求めるには，
分散共分散行列が必要

分散共分散行列を求めるには，
「いまから取り扱うすべての画像」が
事前にわかっていないといけない

そんなことは不可能。

A. Asano, Kansai Univ.

じゃあ，主成分を求めるのはあきらめて，
どういう直交変換をするか「直観的」に

A. Asano, Kansai Univ.

画像をベクトルにしてしまったら，
直観がはたらかない…

A. Asano, Kansai Univ.

行列の直交変換

A. Asano, Kansai Univ.

画像を行列であらわす

素直に表せばいいのですが。

前回はベクトルで考えていたので，

$$\boldsymbol{z} = P'\boldsymbol{x}$$

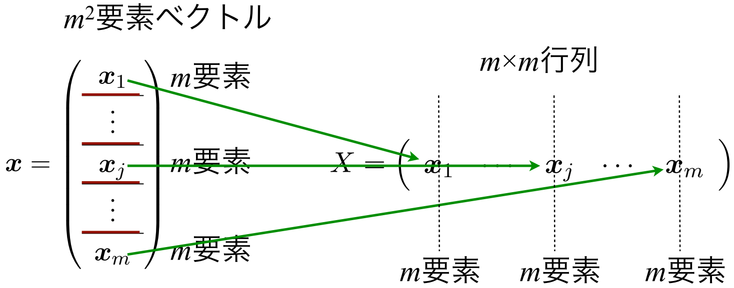
変換後の画像を表すベクトル（m^2要素）

原画像を表すベクトル（m^2要素）

直交変換を表す行列（$m^2 \times m^2$）

ベクトルから行列に書き換える（戻す）ことを考える

ベクトルを行列に書き換える



zも同じ

直交変換行列P'は？

P'がこういう形になっているのなら

$$P' = \begin{pmatrix} r_{11}c_{11} & \cdots & r_{11}c_{1m} & & r_{1m}c_{11} & \cdots & r_{1m}c_{1m} \\ \vdots & \ddots & \vdots & \cdots & \vdots & \ddots & \vdots \\ r_{11}c_{m1} & \cdots & r_{11}c_{mm} & & r_{1m}c_{m1} & \cdots & r_{1m}c_{mm} \\ & \vdots & & \ddots & & \vdots & \\ r_{m1}c_{11} & \cdots & r_{m1}c_{1m} & & r_{mm}c_{11} & \cdots & r_{mm}c_{1m} \\ \vdots & \ddots & \vdots & \cdots & \vdots & \ddots & \vdots \\ r_{m1}c_{m1} & \cdots & r_{m1}c_{mm} & & r_{mm}c_{m1} & \cdots & r_{mm}c_{mm} \end{pmatrix}$$

行列のKronecker積

$$P' = \begin{pmatrix} \boxed{\begin{matrix} r_{11}c_{11} & \cdots & r_{11}c_{1m} \\ \vdots & r_{11}\times C & \vdots \\ r_{11}c_{m1} & \cdots & r_{11}c_{mm} \end{matrix}} & \cdots & \boxed{\begin{matrix} r_{1m}c_{11} & \cdots & r_{1m}c_{1m} \\ \vdots & r_{1m}\times C & \vdots \\ r_{1m}c_{m1} & \cdots & r_{1m}c_{mm} \end{matrix}} \\ \vdots & \ddots & \vdots \\ \boxed{\begin{matrix} r_{m1}c_{11} & \cdots & r_{m1}c_{1m} \\ \vdots & r_{m1}\times C & \vdots \\ r_{m1}c_{m1} & \cdots & r_{m1}c_{mm} \end{matrix}} & \cdots & \boxed{\begin{matrix} r_{mm}c_{11} & \cdots & r_{mm}c_{1m} \\ \vdots & r_{mm}\times C & \vdots \\ r_{mm}c_{m1} & \cdots & r_{mm}c_{mm} \end{matrix}} \end{pmatrix}$$

$$C = \begin{pmatrix} c_{11} & \cdots & c_{1m} \\ \vdots & \ddots & \vdots \\ c_{m1} & \cdots & c_{mm} \end{pmatrix}, \quad R = \begin{pmatrix} r_{11} & \cdots & r_{1m} \\ \vdots & \ddots & \vdots \\ r_{m1} & \cdots & r_{mm} \end{pmatrix}$$

Rの各要素に
Cを貼付けたもの

$$P' = R \otimes C$$ Kronecker積

こうなっているのなら

行列の変換に書き換える

$$\boldsymbol{z} = P'\boldsymbol{x}$$

ベクトル$\boldsymbol{x}$から
ベクトル$\boldsymbol{z}$への
行列P'による変換

$$Z = CXR'$$

行列Xから
行列Zへの
行列CとR'による変換

証明は…ひたすら計算（(8),(9)式）

P'が正規直交行列であるためには

正規直交… 異なる列の内積は０，同じ列同士の内積は１

$$
P' = \begin{pmatrix}
r_{11}c_{11} & \cdots & r_{11}c_{1m} & & r_{1m}c_{11} & \cdots & r_{1m}c_{1m} \\
\vdots & \ddots & \vdots & \cdots & \vdots & \ddots & \vdots \\
r_{11}c_{m1} & \cdots & r_{11}c_{mm} & & r_{1m}c_{m1} & \cdots & r_{1m}c_{mm} \\
 & \vdots & & \ddots & & \vdots & \\
r_{m1}c_{11} & \cdots & r_{m1}c_{1m} & & r_{mm}c_{11} & \cdots & r_{mm}c_{1m} \\
\vdots & \ddots & \vdots & \cdots & \vdots & \ddots & \vdots \\
r_{m1}c_{m1} & \cdots & r_{m1}c_{mm} & & r_{mm}c_{m1} & \cdots & r_{mm}c_{mm}
\end{pmatrix}
$$

$P' = R \otimes C$ なら

$$
C = \begin{pmatrix} c_{11} & \cdots & c_{1m} \\ \vdots & \ddots & \vdots \\ c_{m1} & \cdots & c_{mm} \end{pmatrix}, \quad R = \begin{pmatrix} r_{11} & \cdots & r_{1m} \\ \vdots & \ddots & \vdots \\ r_{m1} & \cdots & r_{mm} \end{pmatrix}
$$

C, Rそれぞれが正規直交なら，P'は正規直交

A. Asano, Kansai Univ.

分離可能性

$$
CXR' =
\begin{pmatrix} c_{11} & \cdots & c_{1m} \\ \vdots & \ddots & \vdots \\ c_{m1} & \cdots & c_{mm} \end{pmatrix}
\begin{pmatrix} x_{11} & \cdots & x_{1m} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mm} \end{pmatrix}
\begin{pmatrix} r_{11} & \cdots & r_{m1} \\ \vdots & \ddots & \vdots \\ r_{1m} & \cdots & r_{mm} \end{pmatrix}
$$

CはXの列に作用

RはXの行に作用

縦方向と横方向の作用を分離できる
ことを，分離可能(separable)という

A. Asano, Kansai Univ.

行列の直交変換とユニタリー変換

縦横の作用を区別する必要はない場合，
C=Rとする

$$Z = RXR' \qquad X = R'ZR$$

ただし　$RR'=I$

行列Xの行列Rによる直交変換

要素が複素数の場合は，R'のかわりに R'^*を用いる

行列Xの行列Rによる
ユニタリー変換

A. Asano, Kansai Univ.

ちょっと余談ですが

A. Asano, Kansai Univ.

縦横の作用を区別する必要はないのか？

画像処理としてはその仮定はおかしくないが,

現実世界においては,
重力があるので，左右と上下は異なる

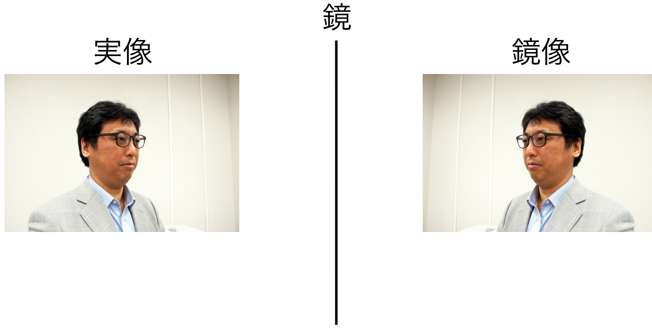
上下反転のほうが
違和感が大きい

だから

鏡ではなぜ左右だけ逆になるのか？

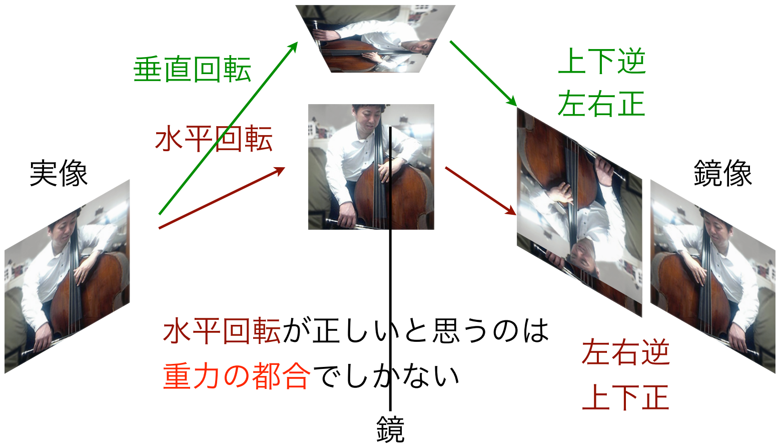
鏡で逆になっているのは，左右でも上下でもなく

前後。



鏡ではなぜ左右だけ逆になるのか？

「鏡で逆になる」というなら，「正解」はなにか？



水平回転が正しいと思うのは
重力の都合でしかない

基底画像

基底画像

$$Z = RXR'$$

どういうRを用いれば,
最適に画像データを圧縮できるか？

それは，依然わからない

しかし，画像をベクトルでなく行列で表したことで,
直交変換の効果がヴィジュアルにわかる

基底画像

変換後の画像Zのm^2個の要素を,
それぞれ行列に分ける

$$Z = \begin{pmatrix} z_{11} & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix} + \begin{pmatrix} 0 & z_{12} & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix} + \cdots + \begin{pmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & z_{mm} \end{pmatrix}$$

$X = R'ZR$ を，上の各行列で行う。たとえば

$$\begin{pmatrix} r_{11} & \cdots & r_{m1} \\ \vdots & \ddots & \vdots \\ r_{1m} & \cdots & r_{mm} \end{pmatrix} \begin{pmatrix} z_{11} & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} r_{11} & \cdots & r_{1m} \\ \vdots & \ddots & \vdots \\ r_{m1} & \cdots & r_{mm} \end{pmatrix}$$

基底画像

$$\begin{pmatrix} r_{11} & \cdots & r_{m1} \\ \vdots & \ddots & \vdots \\ r_{1m} & \cdots & r_{mm} \end{pmatrix} \begin{pmatrix} z_{11} & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} r_{11} & \cdots & r_{1m} \\ \vdots & \ddots & \vdots \\ r_{m1} & \cdots & r_{mm} \end{pmatrix}$$

$$\begin{pmatrix} r_{11} & \cdots & r_{m1} \\ \vdots & \ddots & \vdots \\ r_{1m} & \cdots & r_{mm} \end{pmatrix} \begin{pmatrix} z_{11} & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} r_{11} & \cdots & r_{1m} \\ \vdots & \ddots & \vdots \\ r_{m1} & \cdots & r_{mm} \end{pmatrix}$$

$$z_{11} \begin{pmatrix} r_{11} \\ \vdots \\ r_{1m} \end{pmatrix} \begin{pmatrix} r_{11} & \cdots & r_{1m} \end{pmatrix} = z_{11} \begin{pmatrix} r_{11}r_{11} & r_{11}r_{12} & \cdots & r_{11}r_{1m} \\ r_{12}r_{11} & r_{12}r_{12} & \cdots & r_{12}r_{1m} \\ \vdots & \vdots & \ddots & \vdots \\ r_{1m}r_{11} & r_{1m}r_{12} & \cdots & r_{1m}r_{1m} \end{pmatrix}$$

ベクトルの直積　　　行列すなわち画像

基底画像

つまり

$$X = z_{11}\boldsymbol{r}_1\boldsymbol{r}_1' + z_{12}\boldsymbol{r}_1\boldsymbol{r}_2' + \cdots + z_{mm}\boldsymbol{r}_m\boldsymbol{r}_m'$$

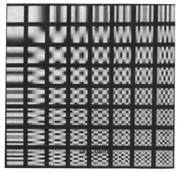
基底画像

原画像Xは，m^2個の基底画像に
それぞれZの各要素をかけて足し合わせた
ものになっている

つづきは

原画像Xは，m^2個の基底画像に
それぞれZの各要素をかけて足し合わせた
ものになっている

つまり，今日の最初にでてきたこれ（の8×8の１つ１つ）が基底画像です



元の関数は，いろいろな周波数の波に，
各々対応するフーリエ係数をかけて足し合わせた
ものになっている…

第１部のこれと同じ？

つづきは

原画像Xは，m^2個の基底画像に
それぞれZの各要素をかけて足し合わせた
ものになっている

元の関数は，いろいろな周波数の波に，
各々対応するフーリエ係数をかけて足し合わせた
ものになっている…

逆フーリエ変換？

フーリエ変換も，ユニタリー変換の一種
フーリエ変換を基本に，
画像圧縮に適した基底画像（一部を省略しても
影響が少ない基底画像）を選ぶ