

# 画像工学

Rev. 2017.07.27

# 講義ホームページ

---

[cvwww.ee.ous.ac.jp/lect/ip/](http://cvwww.ee.ous.ac.jp/lect/ip/)



# 講義内容

---

## ❖ デジタル画像処理

画像変換、特徴量抽出、パターン認識、  
画像符号化、動画画像処理

## ❖ 画像処理プログラミング

Processingを使ったプログラム作成

→ レポート課題

## ❖ コンピュータグラフィックスの基礎

# 今日のテーマ

---

✦ 知覚とは？

✦ 画像処理の応用例は？

✦ コンピュータで画像を扱うには？

# 知覚と画像処理

---

人は見たものをどのようにして、認識・理解しているのか？

光 → 眼球 → 視神経 → 脳(情報処理)

コンピュータはどのような処理を行えば、「もの」を画像から認識・理解できるのか？

# 視細胞

---

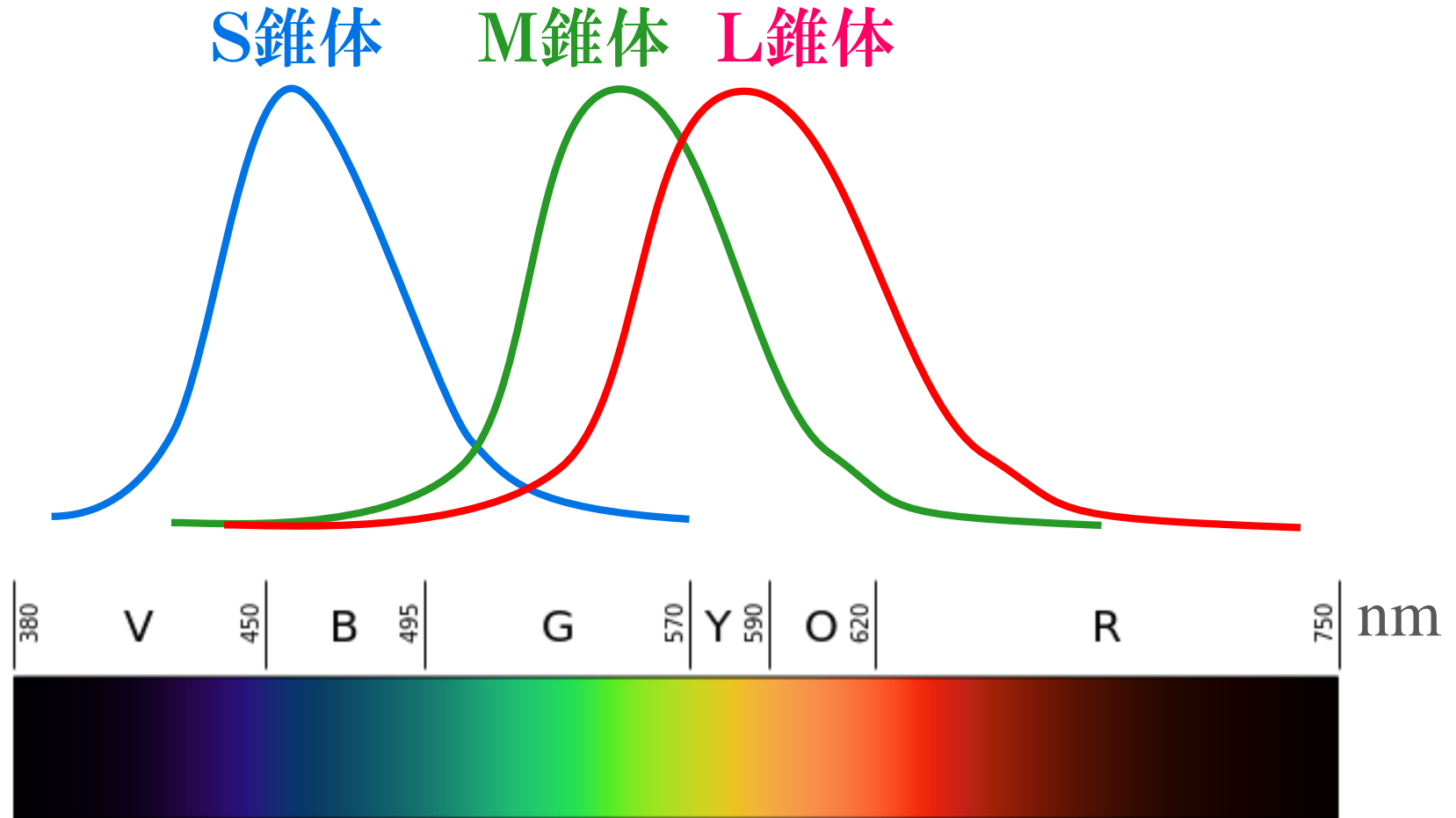
## 錐体細胞

明るい場所で反応する。3種類あり、色を区別できる。

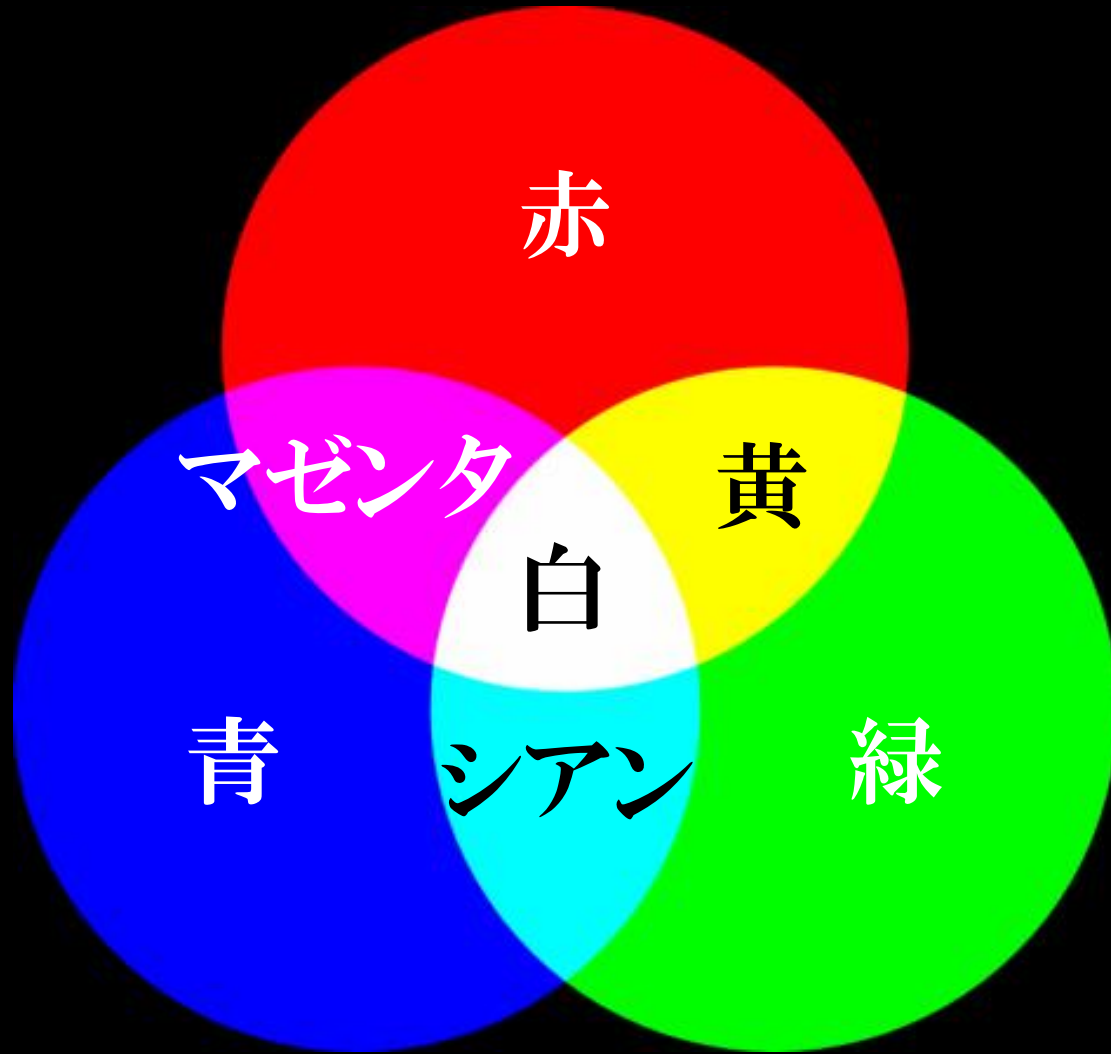
## 桿体細胞

暗い場所で反応する。色を区別できない。

# 可視光波長と錐体細胞の反応



# 光の3原色





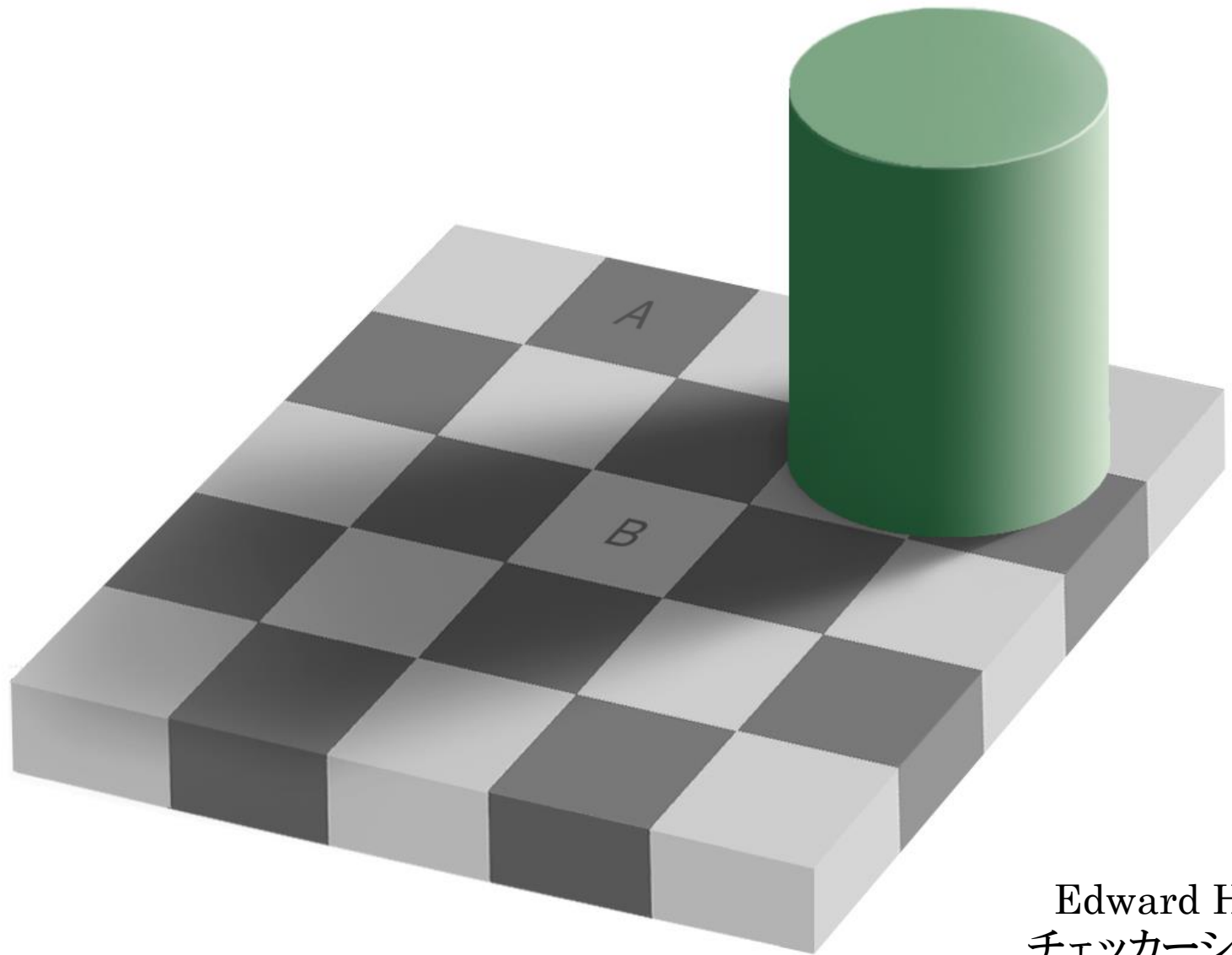
# 人間の知覚

---

人間は、正確にものが  
見えているのか？

# AとBはどちらが明るい？

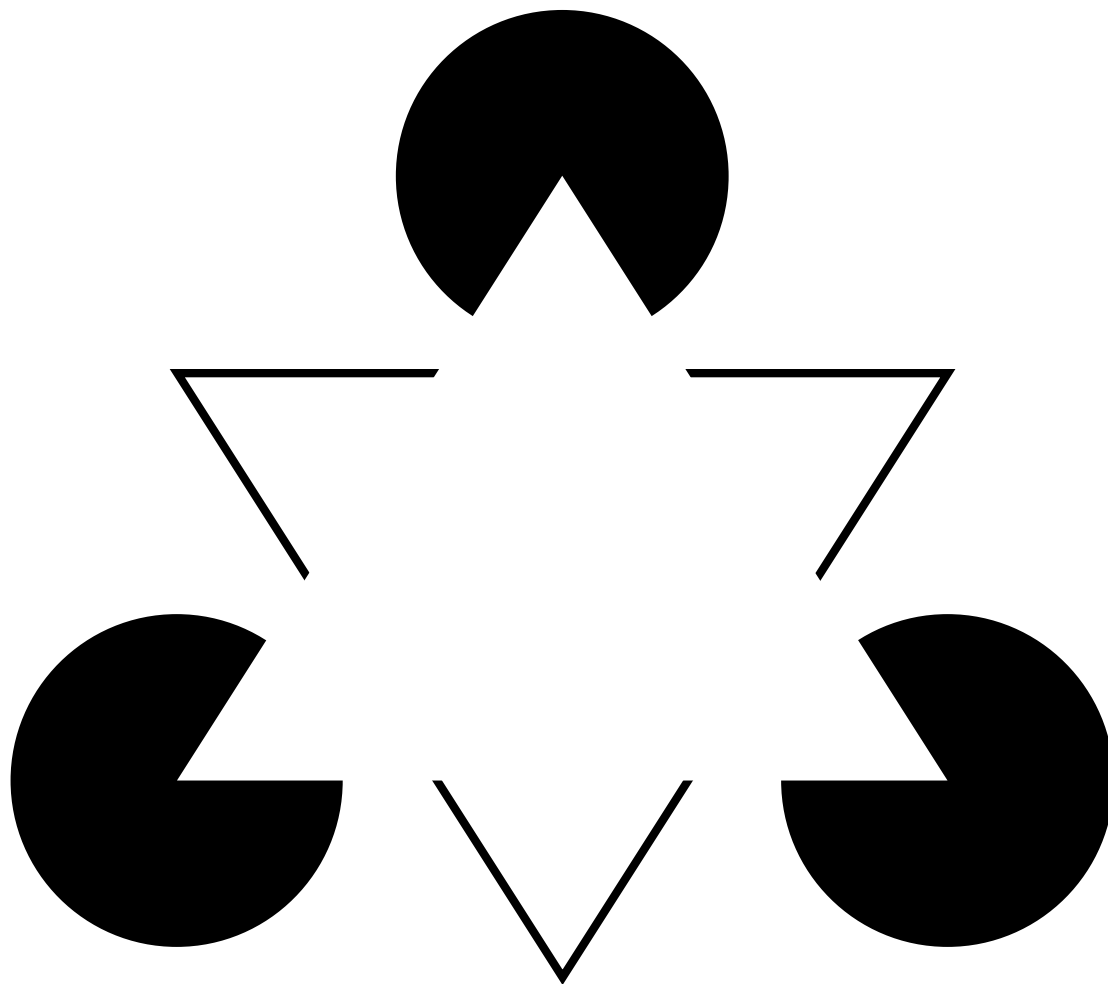
---



Edward H. Adelson  
チェッカーシャドウ錯視

# 主觀的輪廓

---



# 顔細胞(顔ニューロン)

---

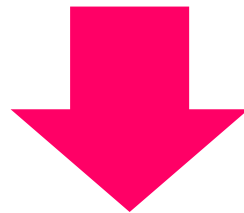
顔の認識に特化した神経細胞

# コンピュータによる知覚

---

コンピュータは、点の色や座標を正確に検出できる。(ミクロの視点)

しかし、顔認識などのように、画像を全体的に捉えることは簡単ではない。  
(マクロの視点)



研究の対象となる

# 画像処理技術の応用

---

- ❖ 見やすい画像への変換・補正
- ❖ 文字認識、指紋照合、顔識別、年齢推定
- ❖ 製品検査、交通量計測
- ❖ ジェスチャー認識、スポーツ映像解析
- ❖ 高画質での映像記録・映像伝送

# 画像処理の事例

# 画像処理の種類

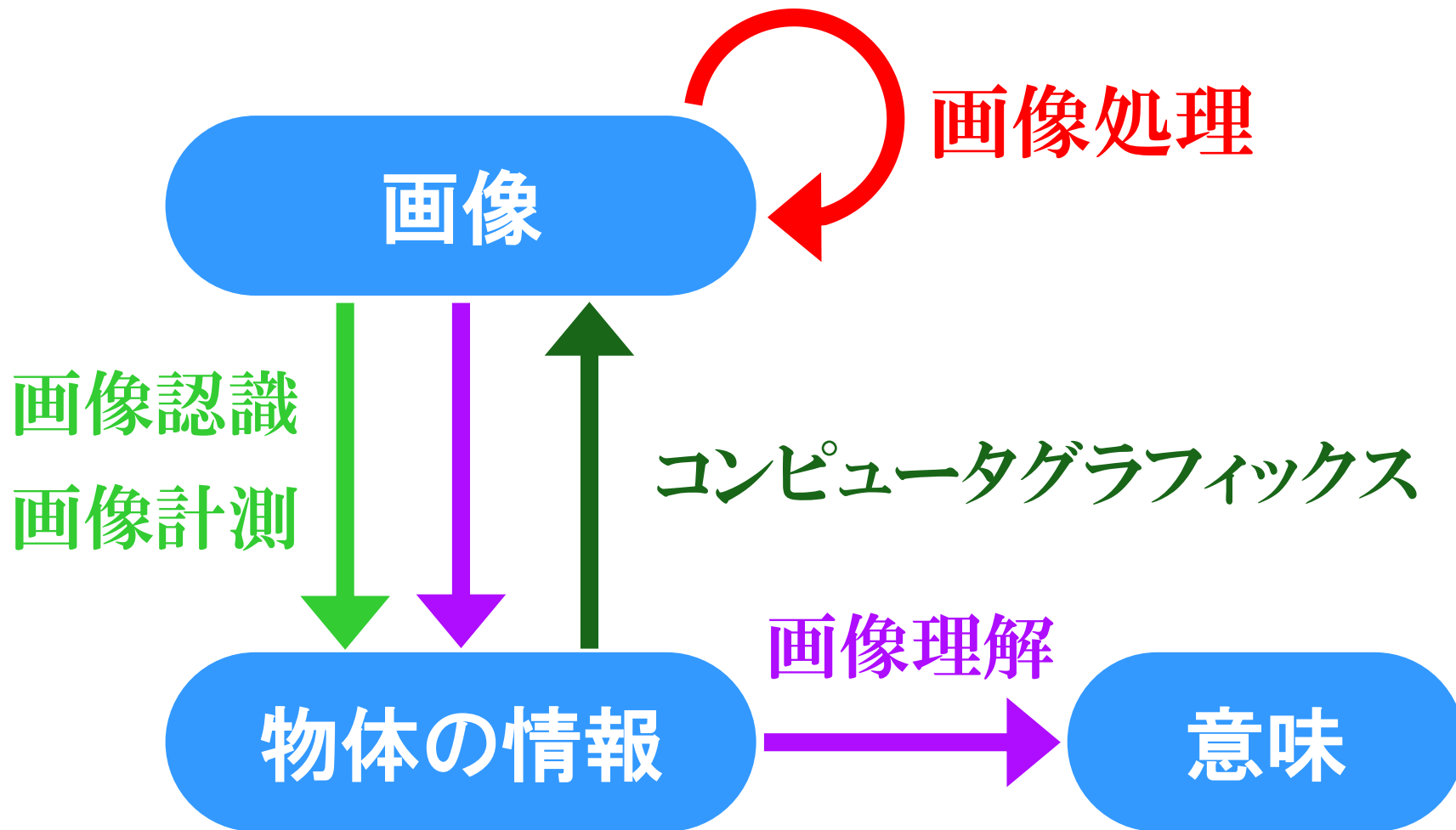
---

- ❖ **画像処理** (image processing)  
画像を加工して新たな画像を作る。
- ❖ **画像認識** (image recognition)  
画像中の対象物が何であるか同定する。
- ❖ **画像計測** (image measurement)  
画像から対象物の形状や個数、動きの情報を得る。
- ❖ **画像理解** (image understanding)  
画像から得られる情報を統合し、画像がもつ意味を構造的に理解する。

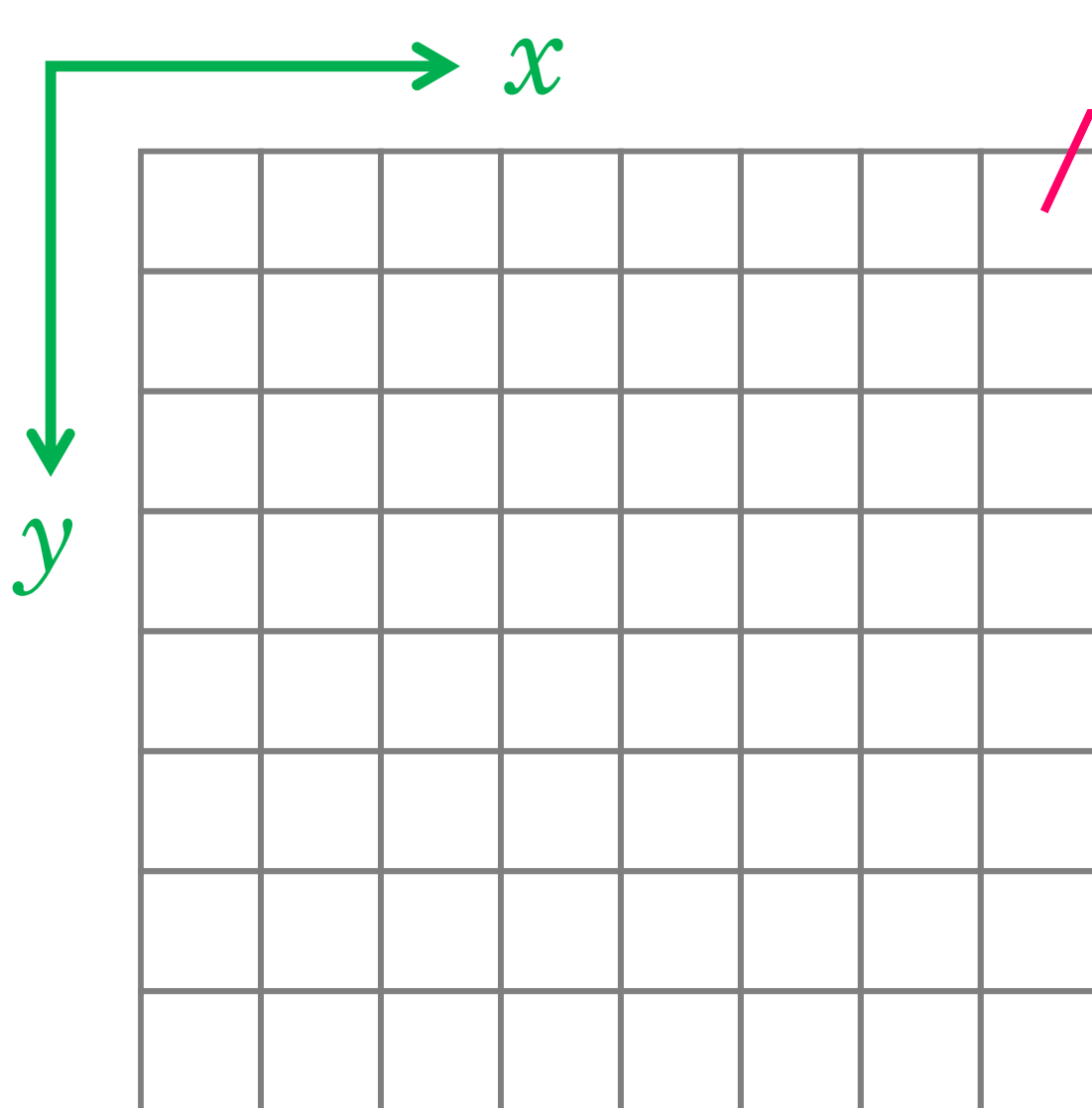


# 画像処理とCG

---



# デジタル画像



画素(pixel)

画素値  
輝度値  
濃度値  
濃淡値  
階調値

画素の明るさ  
(0~255)

# デジタル画像の取得

---

## ✦ AD変換

重要

### ① 標本化 (sampling)

空間のデジタル化

標本化間隔 (sampling interval)

解像度 (resolution)

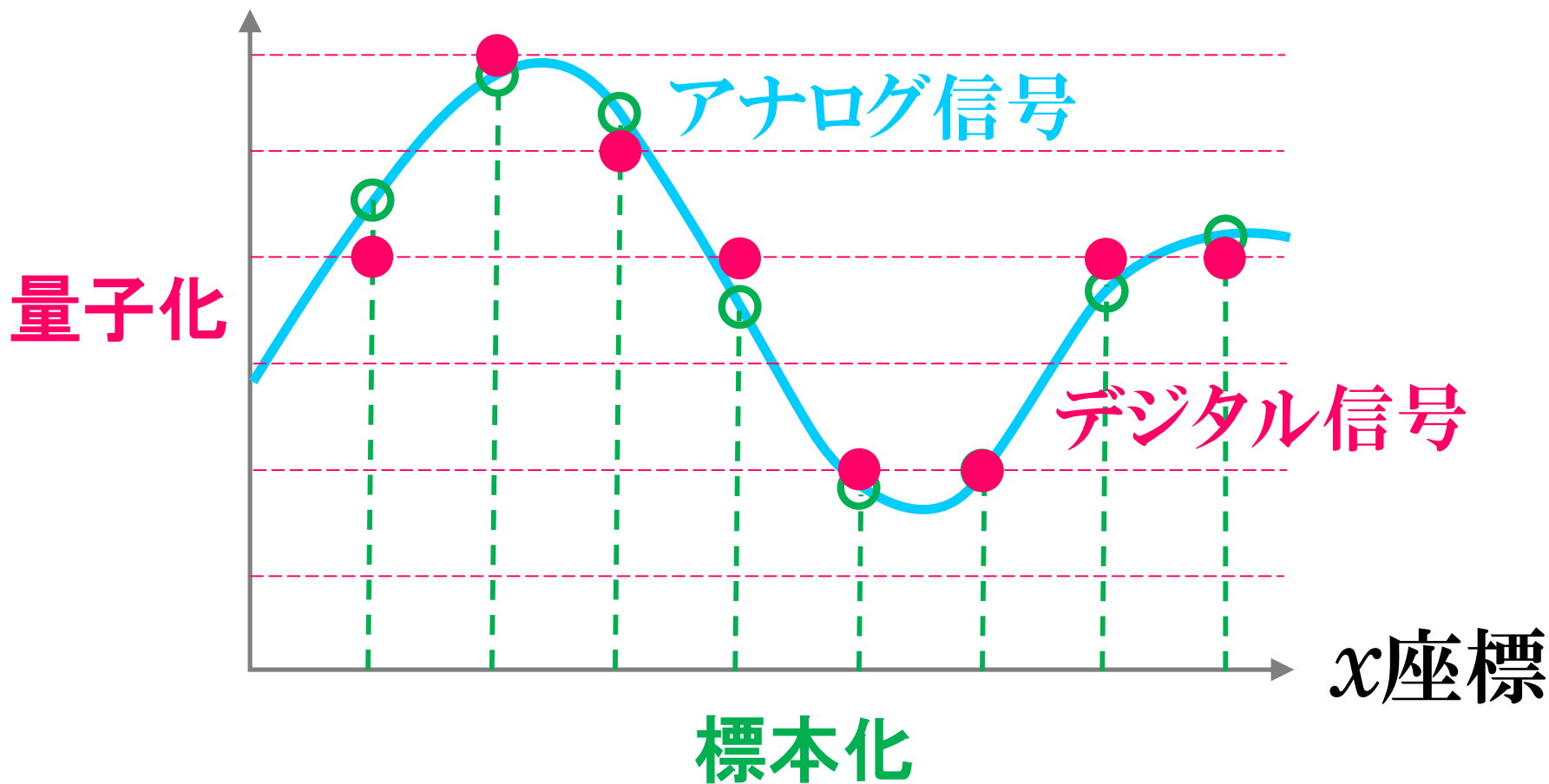
### ② 量子化 (quantization)

色のデジタル化

階調 (gradation)

# AD変換

画素値 (明るさ)



# 階調数による画像の種類

---

❖ **2値画像** (binary image)

2階調(黒・白)

❖ **白黒濃淡画像** (grayscale image)

256階調

❖ **カラー画像** (color image)

RGB(赤・緑・青) 各256階調

# 画像処理

---

## 画像を加工して新たな画像を作る

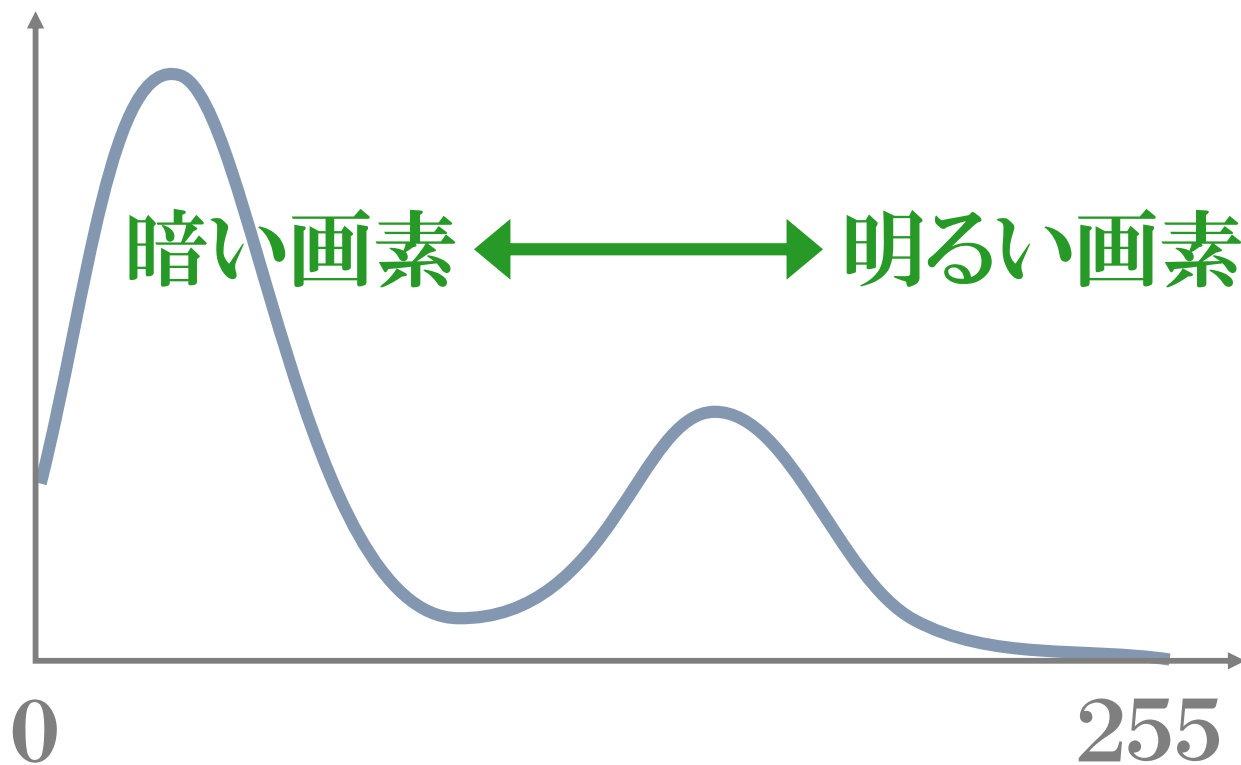
- ✦ 画素ベースの画素値変換
- ✦ 領域ベースの画素値変換
- ✦ 幾何学的変換
- ✦ 画像間演算

# ヒストグラム (histogram)

画素値の頻度分布を表したグラフ

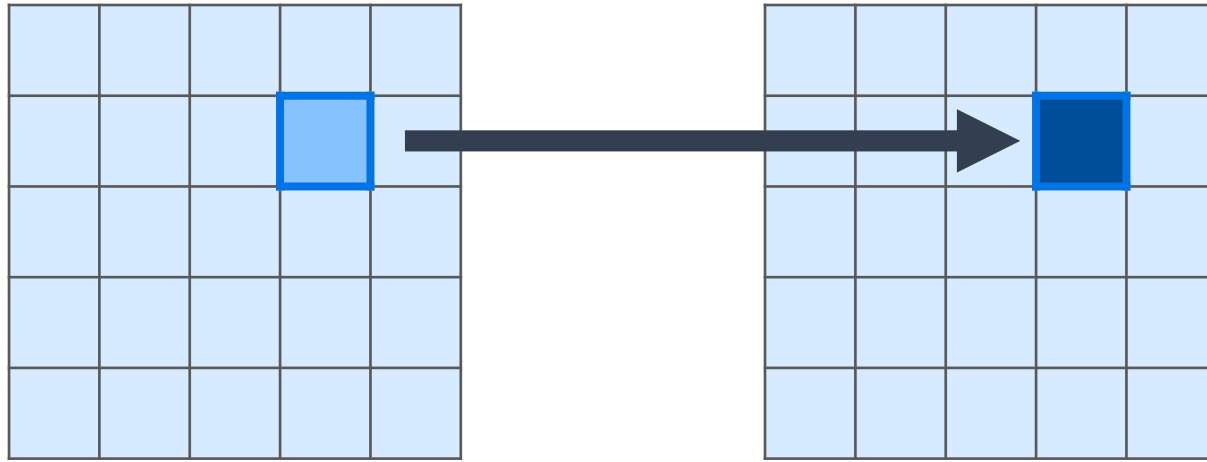
重要

画素数



# 画素ベースの画素値変換

---



原画像

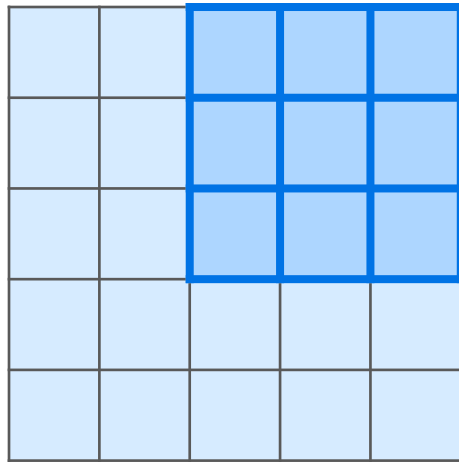
変換後画像

- 階調変換 (gray-level transformation)
- コントラスト強調 (contrast enhancement)



# 領域ベースの画素値変換

重要

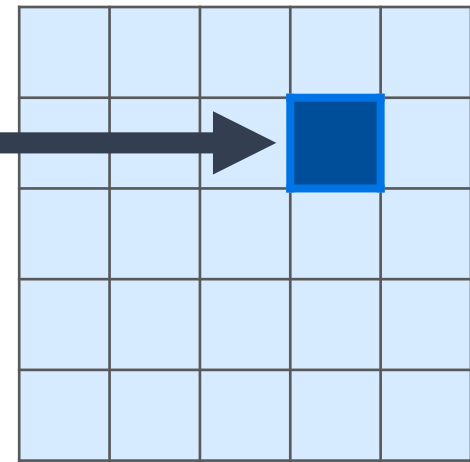
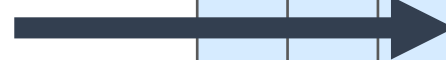


原画像

⊗

$a_0$	$a_1$	$a_2$
$a_3$	$a_4$	$a_5$
$a_6$	$a_7$	$a_8$

空間フィルタ  
(spatial filter)



変換後画像

- 平滑化 (smoothing)
- 輪郭抽出 (edge extraction)
- 鮮鋭化 (sharpening)

# 空間フィルタリング

---

## 畳み込み演算

$$g(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b f(x + s, y + t) \cdot w(s, t)$$

原画像

空間フィルタ

# 平滑化

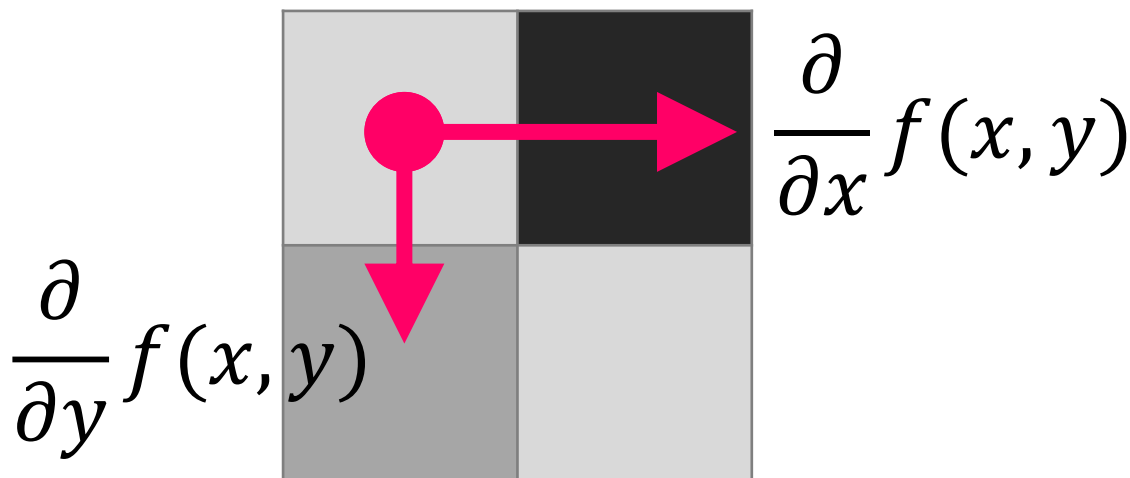
---

- ✿ 平均値フィルタ (average filter)
- ✿ 重み付き平均値フィルタ  
(weighted average filter)
- ✿ ガウシアンフィルタ (Gaussian filter)
- ✿ メディアンフィルタ (median filter)

# 画像の微分

輝度勾配 (intensity gradient)

重要



ラプラシアン (Laplacian)

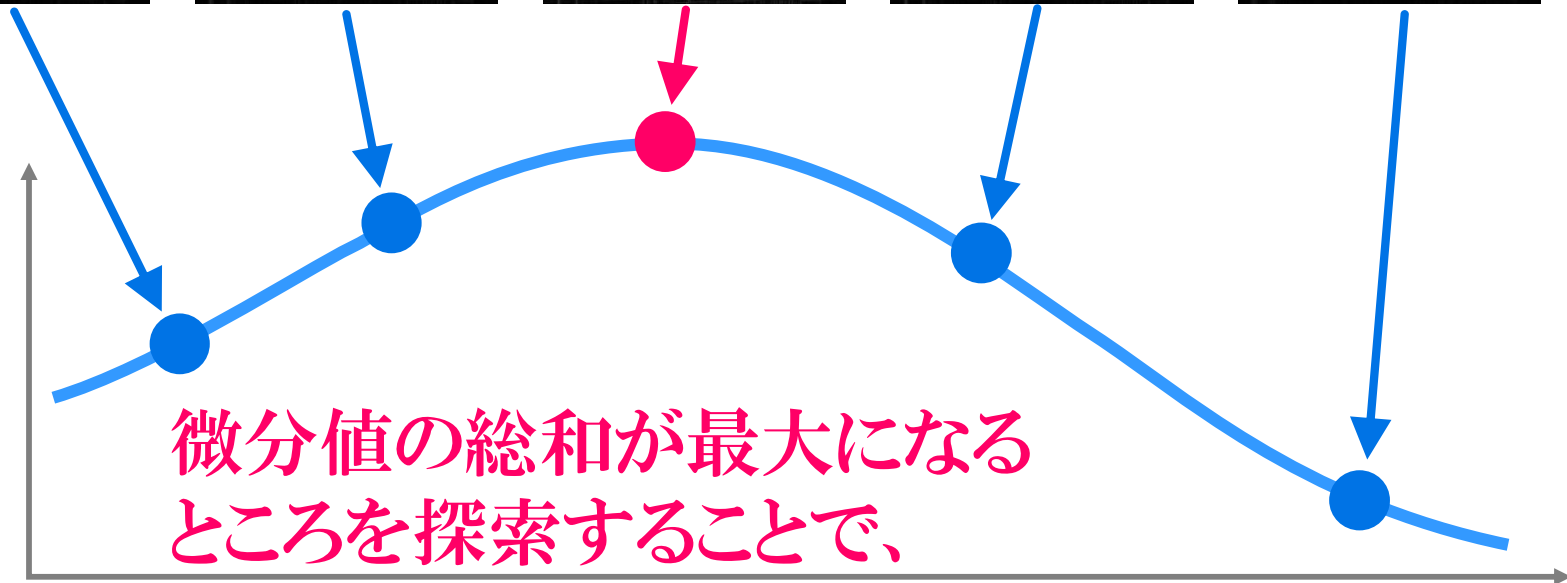
$$\nabla^2 f(x, y) = \frac{\partial^2}{\partial x^2} f(x, y) + \frac{\partial^2}{\partial y^2} f(x, y)$$

# オートフォーカス

応用



全面素の  
微分値の総和



微分値の総和が最大になるところを探索することで、フォーカスを自動調整できる。

フォーカス値

# 鮮鋭化フィルタ

---

原画像の濃淡を残したまま、エッジを強調する。

$$g(x, y) = f(x, y) - \nabla^2 f(x, y)$$

0	0	0
0	1	0
0	0	0

 $-$ 

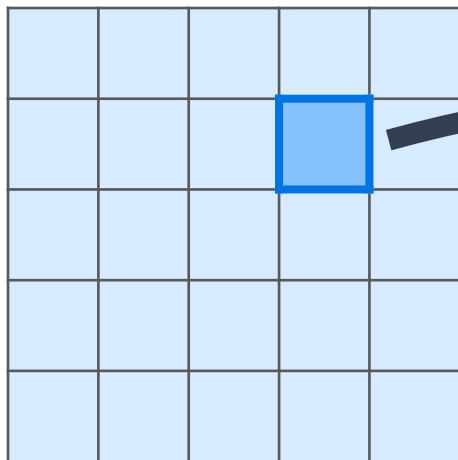
0	1	0
1	-4	1
0	1	0

 $=$ 

0	-1	0
-1	5	-1
0	-1	0

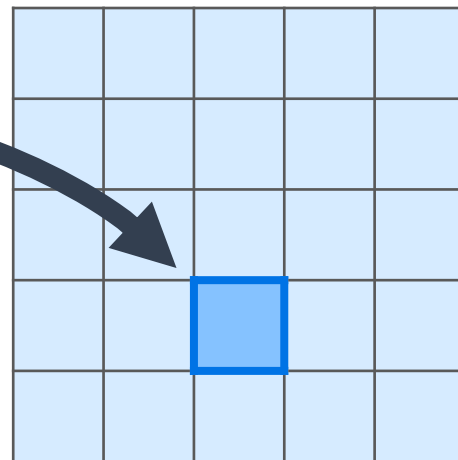
# 幾何学的変換

重要



原画像

座標変換

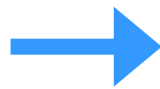


変換後画像

- 線形変換 (linear transformation)
- アフィン変換 (affine transformation)
- 射影変換 (projective transformation)

# アフィン変換

$$\begin{cases} x = aX + bY + c \\ y = dX + eY + f \end{cases}$$



$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}$$

## 拡大縮小

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}$$

## 平行移動

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}$$

## 回転

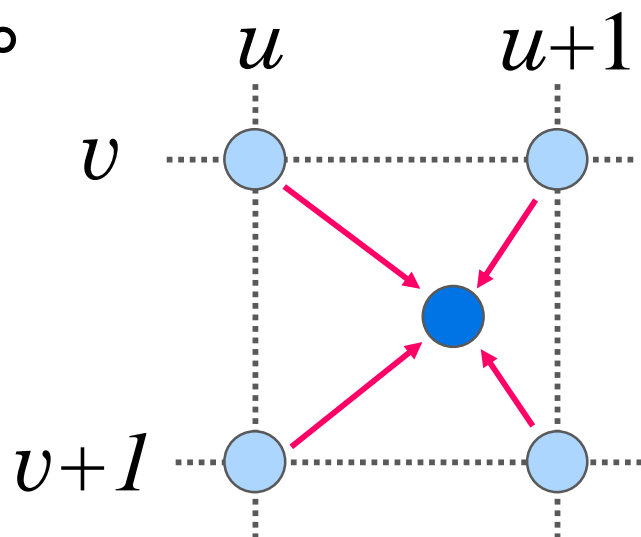
$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}$$



# 再標本化 (resampling)

重要

標本化された画像から、実数座標の画素値を求める。

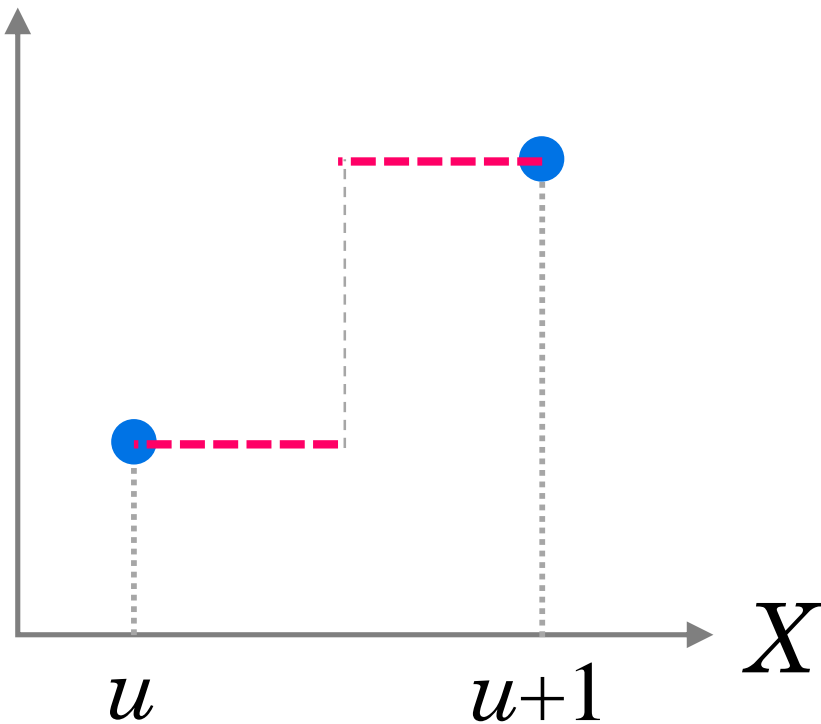


$u, v$  : 整数値

- 最近隣内挿 (nearest neighbor interpolation)
- 共一次内挿 (bi-linear interpolation)

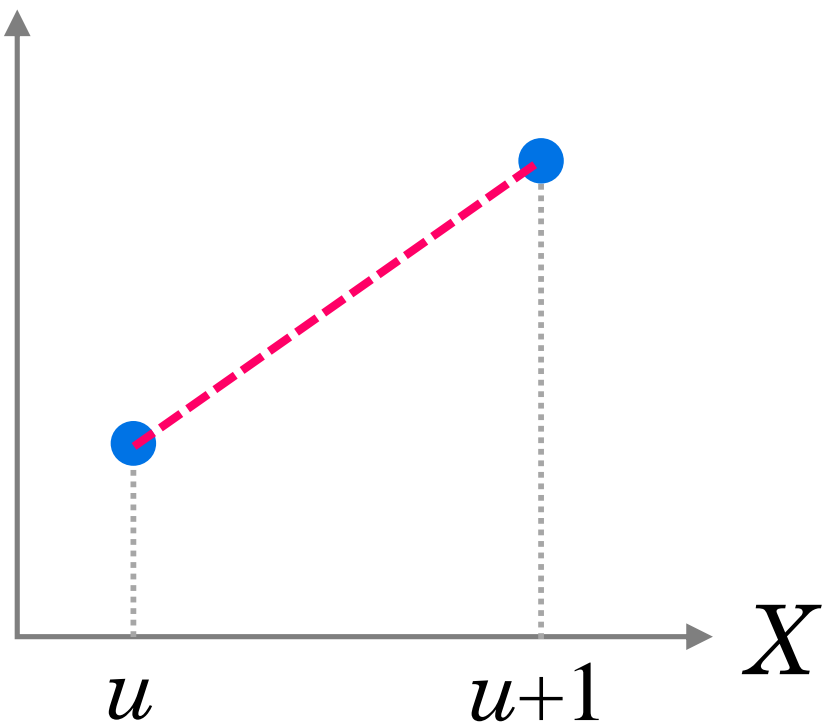
# 内插

画素值



最近隣内插

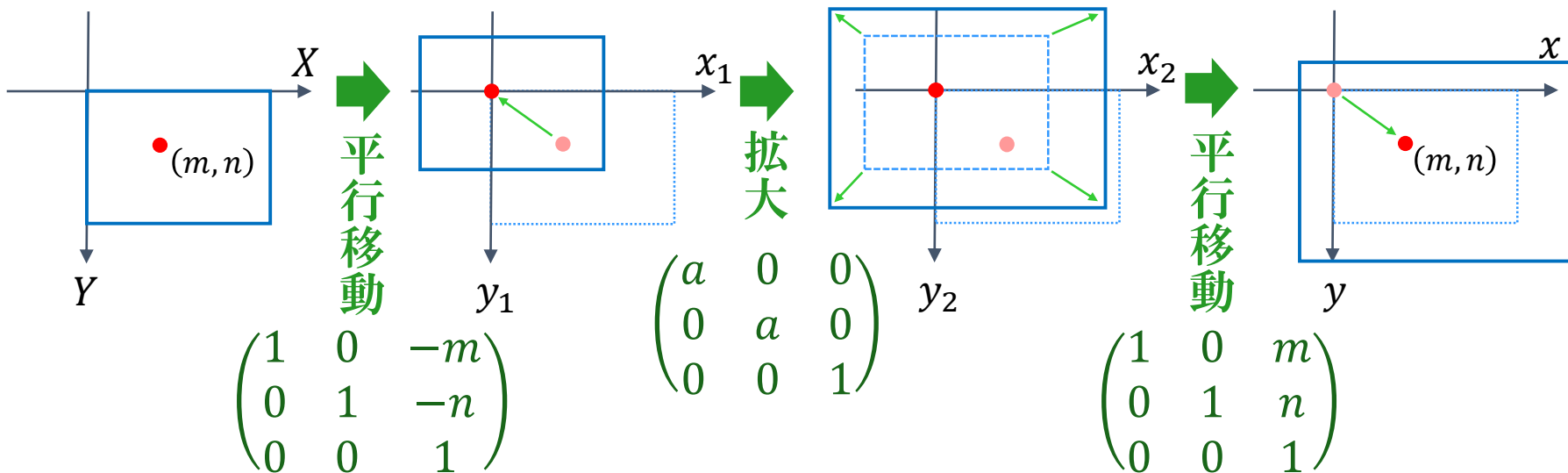
画素值



共一次内插

# 合成変換

例) 画像の座標 $(m, n)$ を中心にして $a$ 倍に拡大



順変換 
$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & m \\ 0 & 1 & n \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} a & 0 & 0 \\ 0 & a & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & -m \\ 0 & 1 & -n \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}$$

逆変換 
$$\begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & -m \\ 0 & 1 & -n \\ 0 & 0 & 1 \end{pmatrix}^{-1} \begin{pmatrix} a & 0 & 0 \\ 0 & a & 0 \\ 0 & 0 & 1 \end{pmatrix}^{-1} \begin{pmatrix} 1 & 0 & m \\ 0 & 1 & n \\ 0 & 0 & 1 \end{pmatrix}^{-1} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

# スマホの写真表示の操作

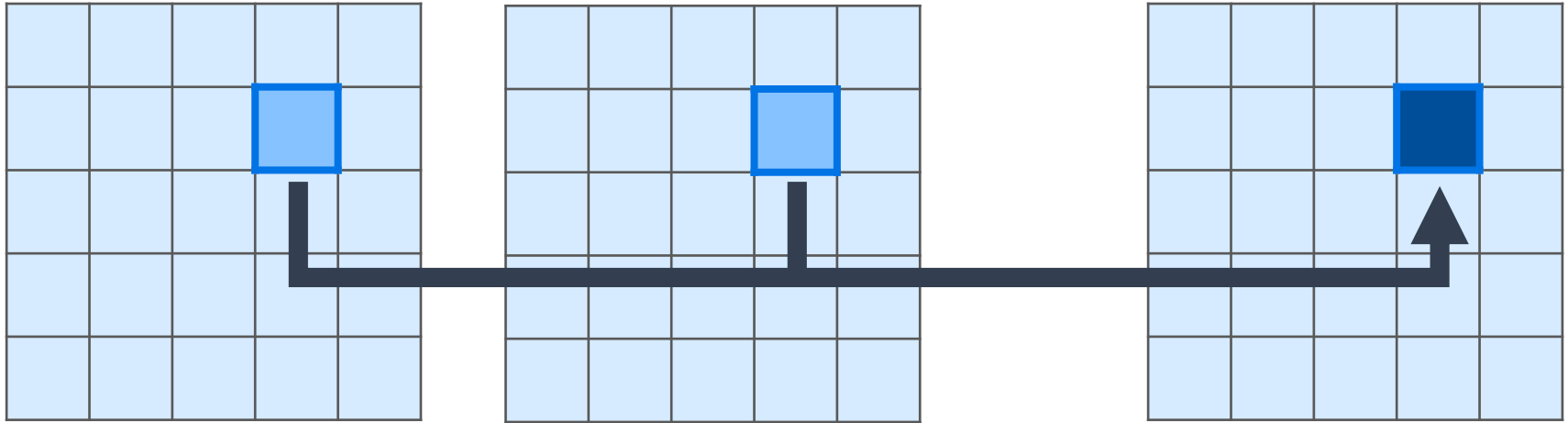
応用



- ① 2本の指の移動前と移動後の座標を検出する。
- ② 幾何学変換式に座標値を入れ、式を解いてパラメータ値(平行移動量、拡大縮小率、回転角度)を求める。
- ③ 得られたパラメータ値を基にして画像全体を幾何学変換する。

# 画像間演算

---



原画像A

原画像B

変換後画像

- ブレンディング (blending)
- 差分画像 (subtraction image)

# 2値画像処理

---

画像から対象物を抽出して、その形状特徴などから対象物の認識を行う。

- ① 各画素を**対象物**または**背景**に分類する。
- ② 対象物の画素の連結を調べ、対象物の領域を抽出する。
- ③ 対象物の固有の特徴を検出する。

# 2値化 (binarization)

---

画素を対象物と背景とに分ける。

ある画素値をしきい値にして、白と黒の2階調に画像を変換する。

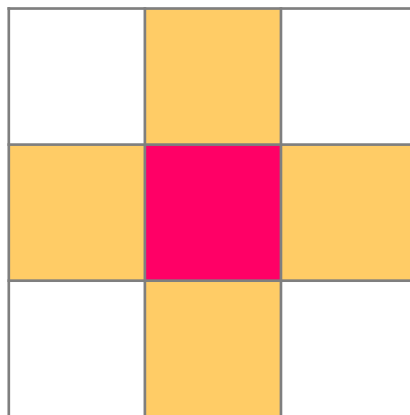
## しきい値の決定手法

- モード法 (mode method)
- P-タイル法 (p-tile method)
- 判別分析法 (discriminant analysis method)

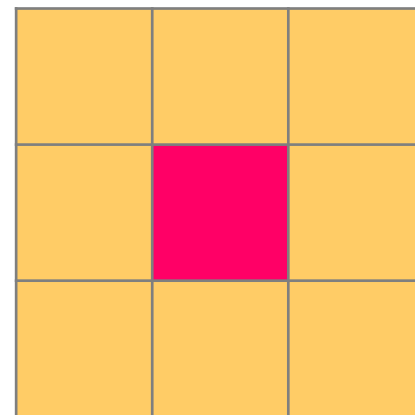
# 連結性 (connectivity)

隣り合う画素同士  
のつながり方

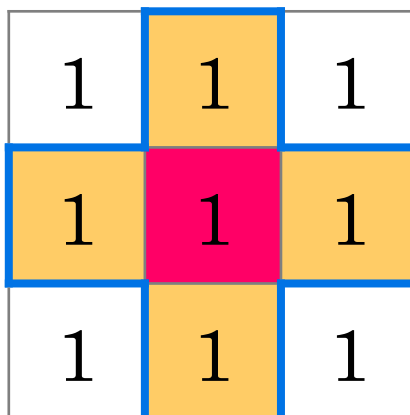
画素同士のつながりを調べることで、領域が検出できる。



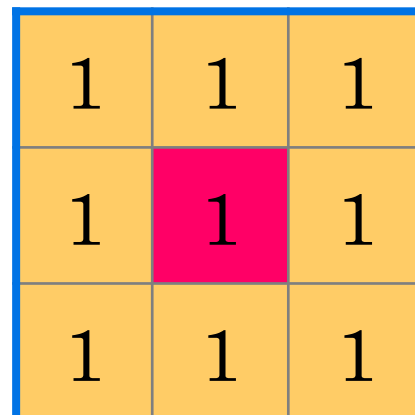
4近傍



8近傍



4連結



8連結





# QRコードの読み取り

応用



2値化



連結成分検出  
マーカー検出

画像抽出  
幾何学変換



# 特徴抽出

---

連結成分ごとに特徴を抽出する。

❖ **形状特徴** (geometric feature)

❖ **モルフォロジー演算**  
(morphological operation)

❖ **骨格化** (skeletonization)

❖ **細線化** (thinning)

# 形状特徴

---

- ✿ 重心 (center of gravity)
- ✿ 外接長方形 (bounding box)
- ✿ 面積 (area)
- ✿ 周囲長 (perimeter)
- ✿ 円形度 (roundness)
- ✿ オイラー数 (Euler number)
- ✿ 主軸 (principal axis)

# モルフォロジー演算

---

✿ 膨張 (expansion)

✿ 収縮 (contraction)

✿ オープニング (opening)

収縮n回 → 膨張n回

✿ クロージング (closing)

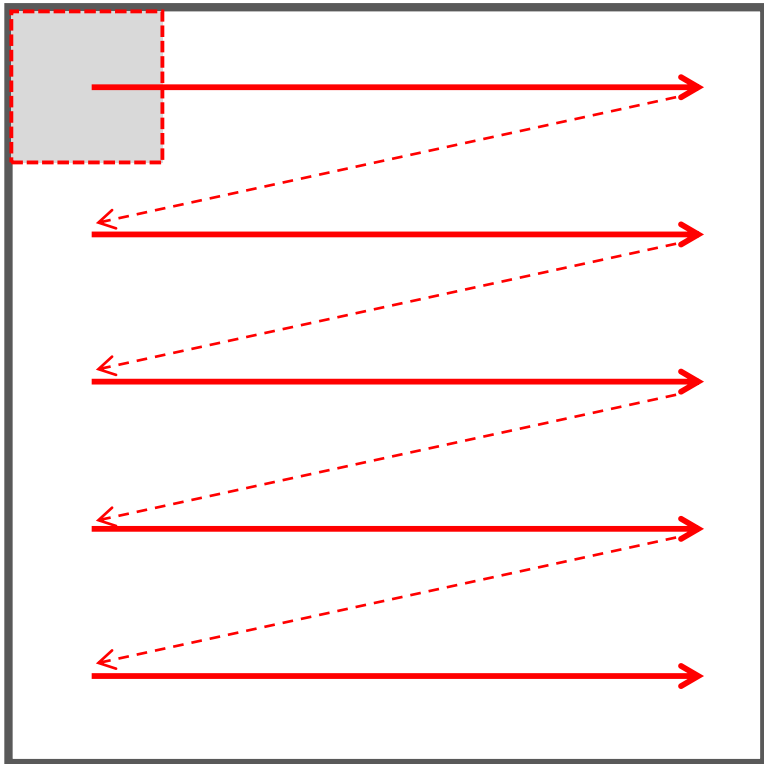
膨張n回 → 収縮n回

2値画像のノイズ除去に有効

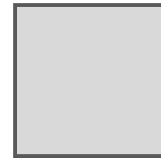
# テンプレートマッチング

1画素ずつずらしながら類似度を計算

**重要**



対象画像

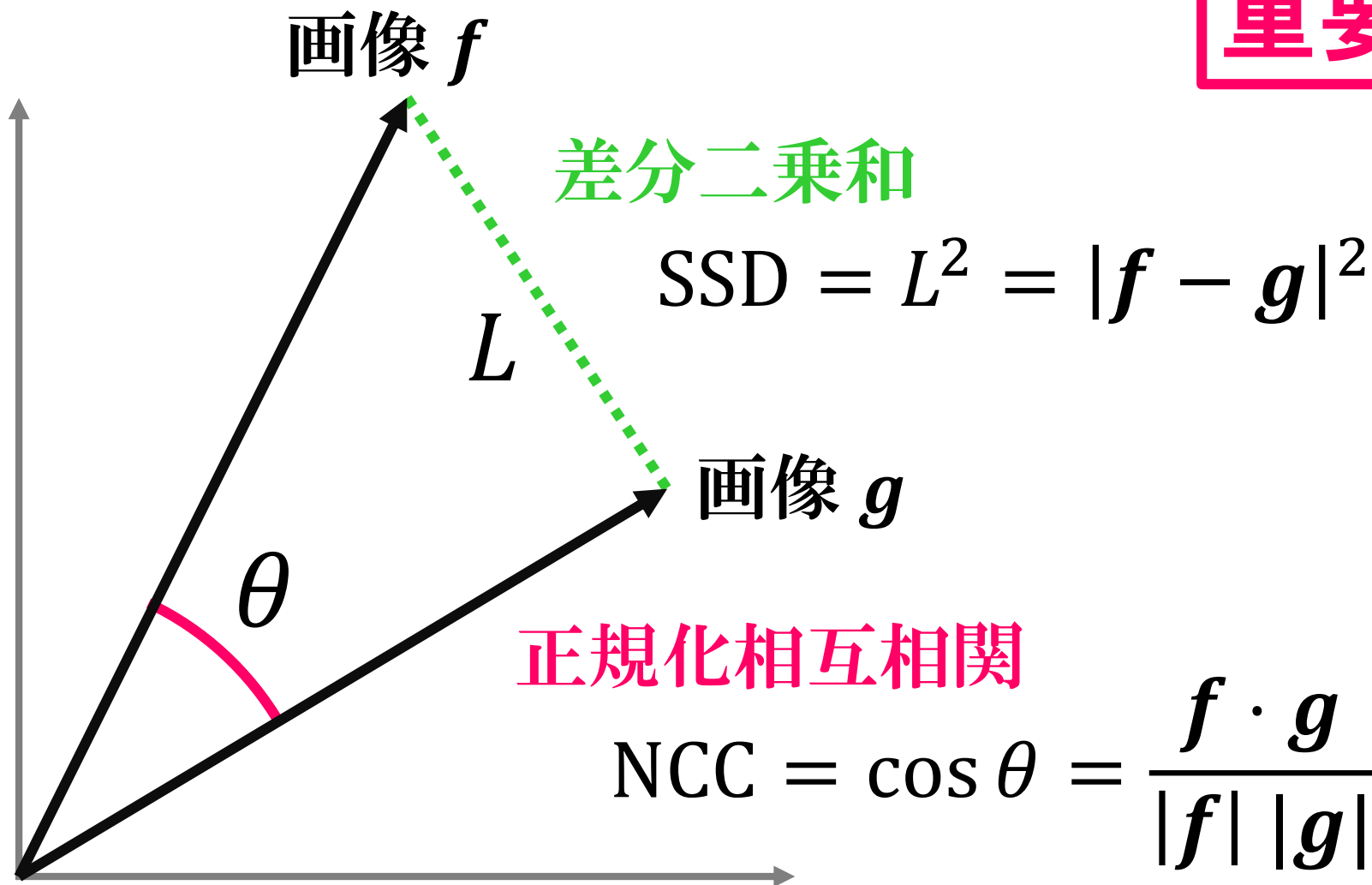


テンプレート画像  
(探索物の画像)

対象画像全体の中で、  
類似度が大きい座標を  
検出する。

# 2つの画像の類似度

重要



# パターン認識

重要

画像の特徴を用いて、あらかじめ定めたクラスに画像を分類する処理

## ❖ 学習 (learning)

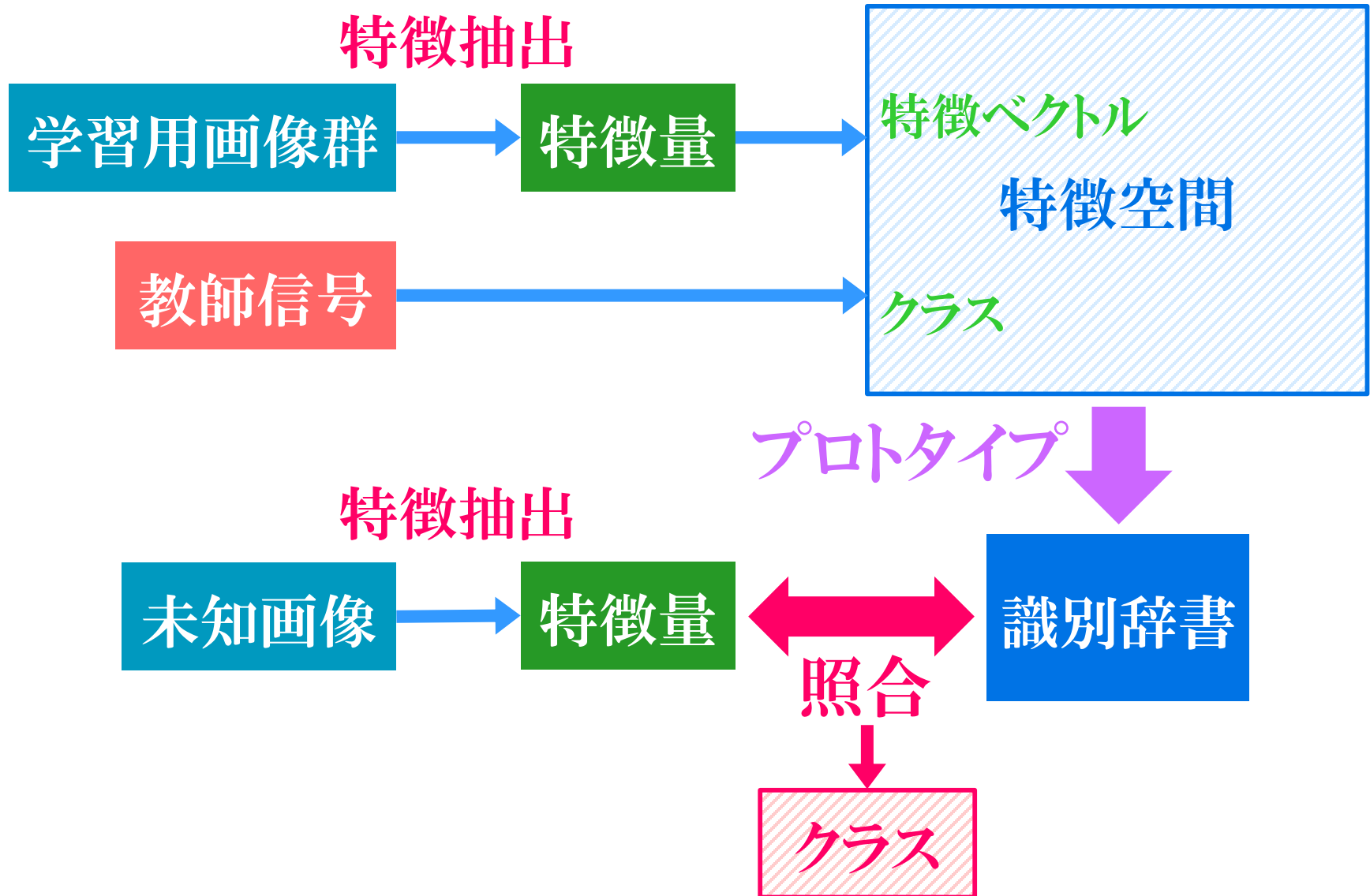
学習用画像群の特徴抽出を行い、各クラスのプロトタイプを定めて識別辞書を作る。

## ❖ 識別 (discrimination)

未知画像の特徴抽出を行い、識別辞書と比較して未知画像のクラスを決定する。

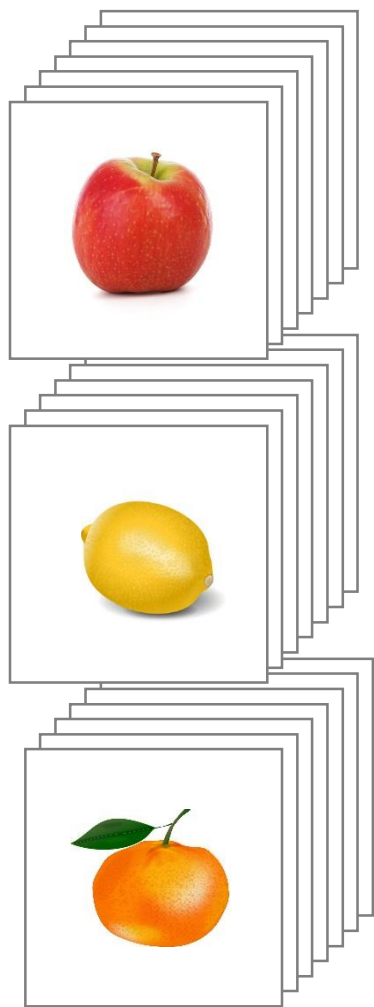


# パターン認識の流れ

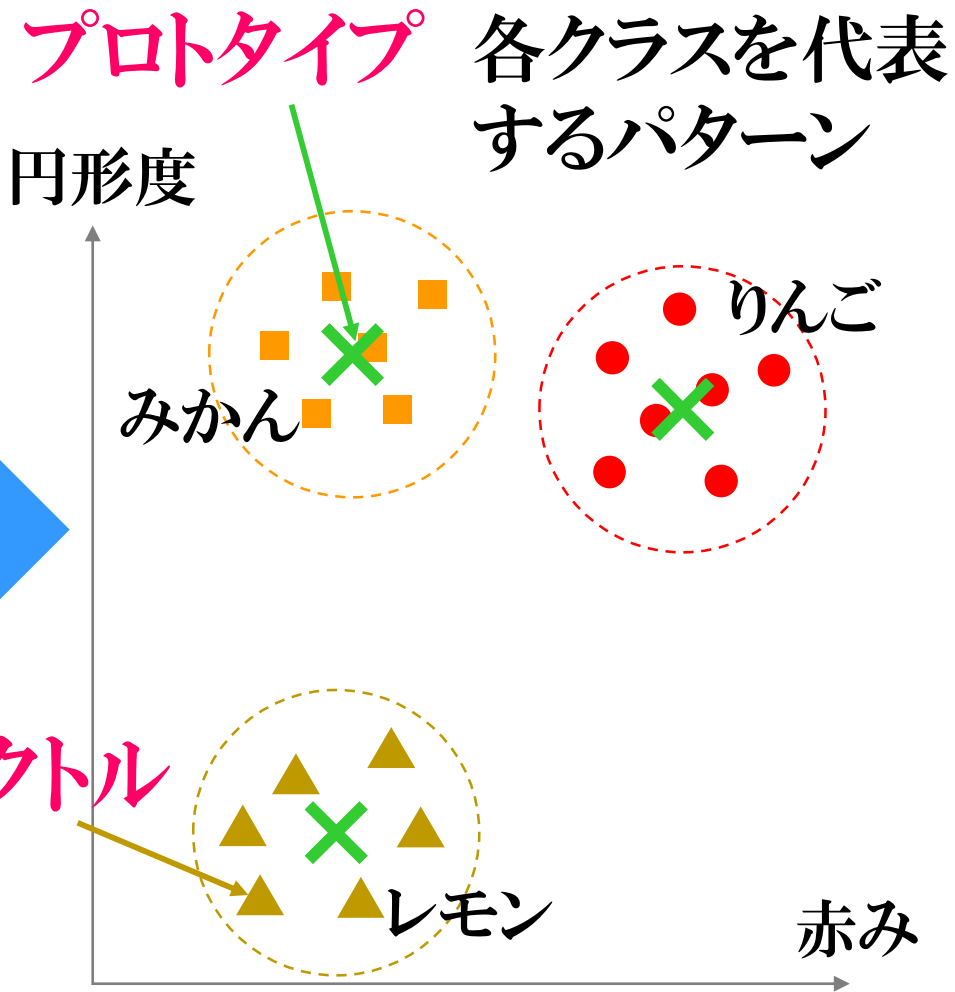


# 学習

重要



学習用画像



特徴空間

# 識別

重要



未知画像

特徴抽出

円形度

プロトタイプ

りんご

みかん

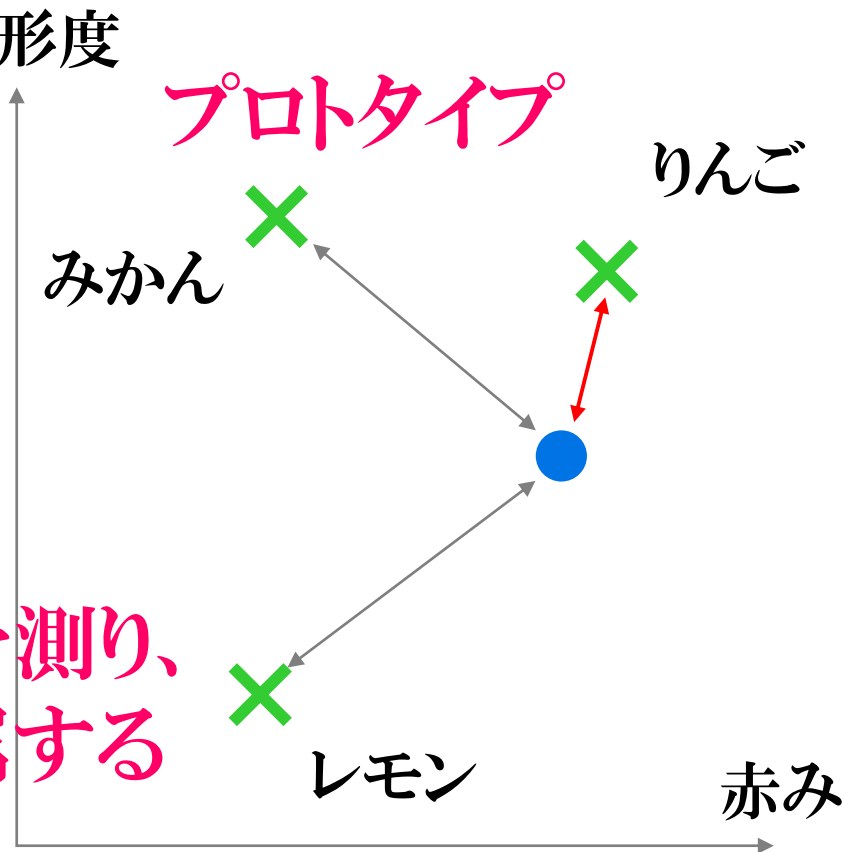
レモン

レモン

赤み

各プロトタイプとの距離を測り、最も近いプロトタイプが属するクラスに識別する。

特徴空間



# 画像認識の手法

---

## ❖ テンプレートマッチング

テンプレート画像に類似している部分を抽出する。  
特徴空間の軸は、各画素の画素値

## ❖ プロトタイプによるパターン認識

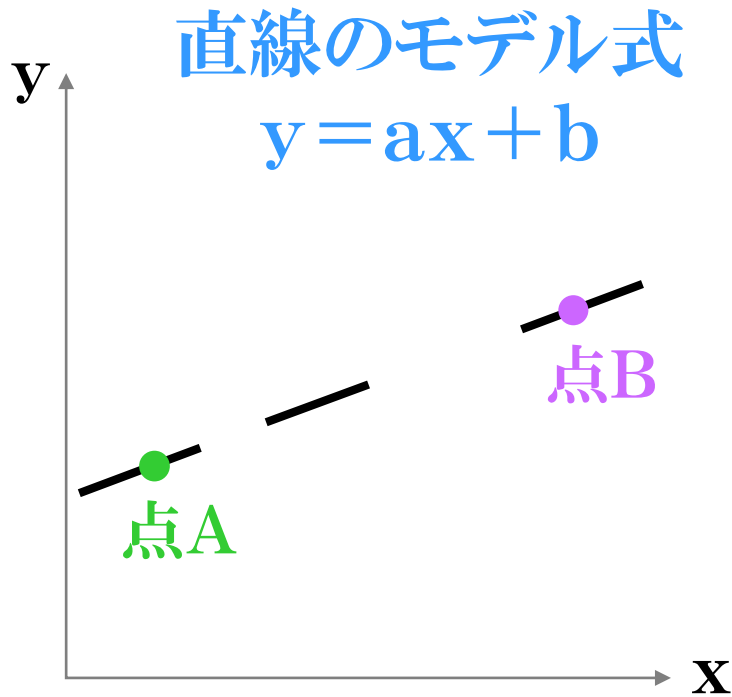
プロトタイプとの距離を測り、未知画像を識別する。  
特徴空間の軸は、形状特徴量など

## ❖ モデル式マッチング

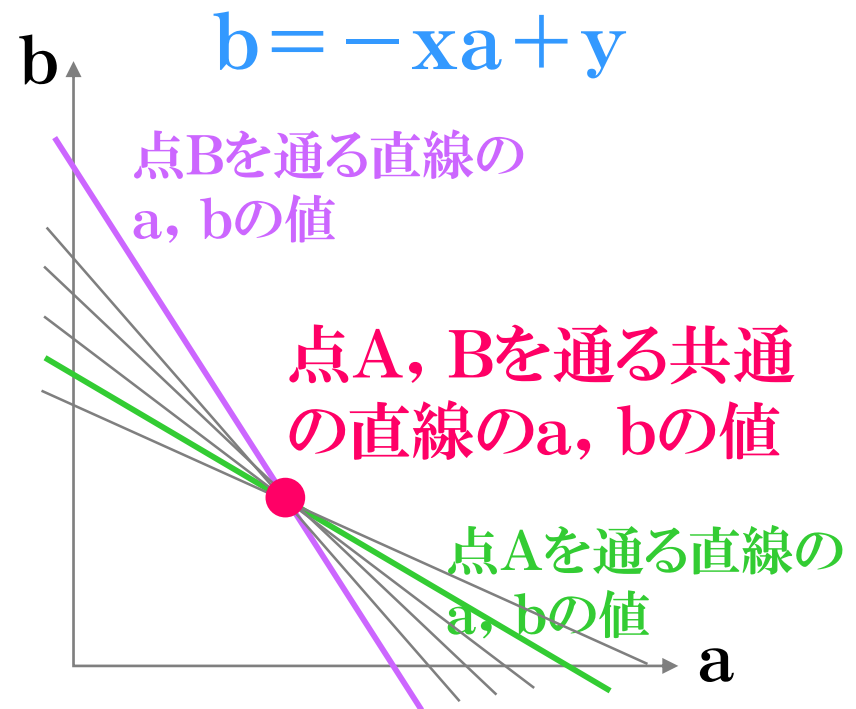
対象物の数式(モデル式)で表し、モデル式のパラメータを画像から決めることで、対象物を検出する。  
特徴空間の軸は、モデル式のパラメータ

# ハフ変換 (Hough transform)

不連続な輪郭線やノイズが含まれる画像から、モデル式で表された図形を抽出する。



x-y 画像平面



パラメータ空間

# 人物姿勢(ポーズ)の推定

発展



# 動画画像処理

---

オプティカルフロー (optical flow)

時間連続した2フレーム間の見かけの動きの分布

❖ ブロックマッチング法

❖ 勾配法 (gradient method)

仮定:  $f(x, y, t) = f(x + \Delta x, y + \Delta y, t + \Delta t)$

$$\frac{\partial f(x, y, t)}{\partial x} \frac{dx}{dt} + \frac{\partial f(x, y, t)}{\partial y} \frac{dy}{dt} + \frac{\partial f(x, y, t)}{\partial t} = 0$$

を解いて、 $(\frac{dx}{dt}, \frac{dy}{dt})$ を求める。

# 移動物体検出

---

## ❖ 背景差分法

移動物体が存在していない背景だけの画像との差分により、移動物体を検出する。

## ❖ フレーム間差分法

3枚のフレーム画像の差分の論理積から、移動物体を検出する。



# 3次元画像処理

---

## 投影変換

- ❖ 平行投影 (orthogonal projection)
- ❖ 透視投影 (perspective projection)

3次元空間座標  $(X, Y, Z)$

2次元平面座標  $(x, y)$

焦点距離  $f$

$$x = f \cdot \frac{X}{Z}, \quad y = f \cdot \frac{Y}{Z}$$

# 3次元再構成

---

## ❖ ステレオ視 (stereo vision)

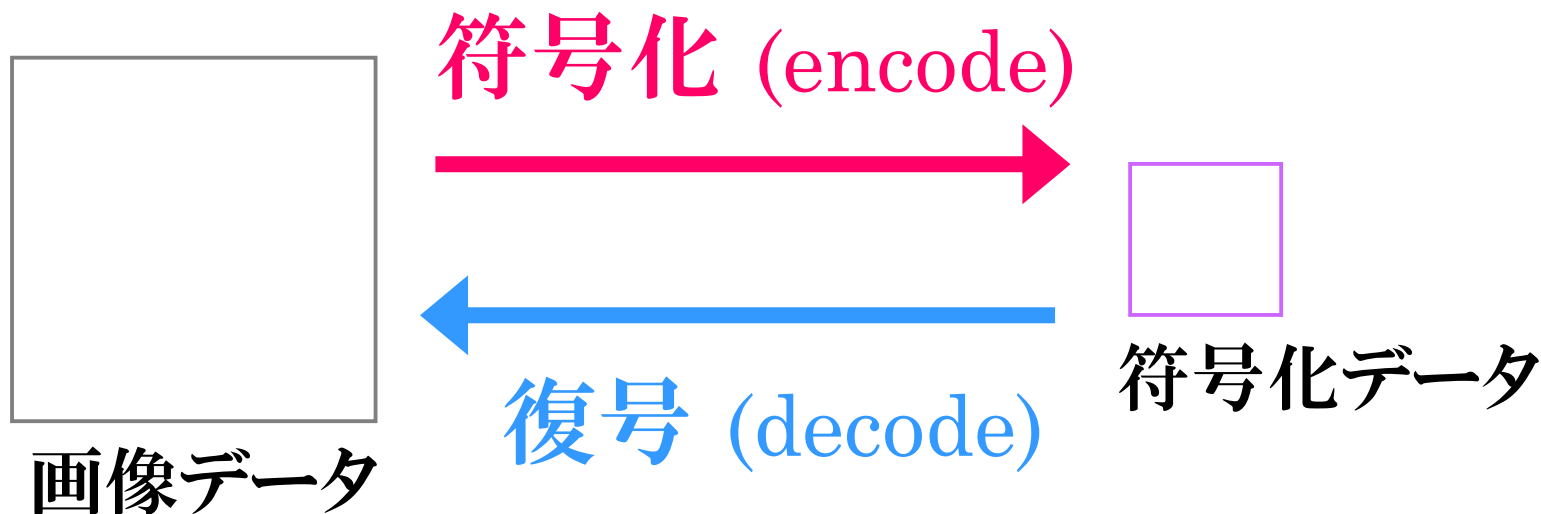
視点の異なる2枚の画像の対応点を見つけ (ステレオマッチング)、三角測量を用いて3次元座標を計測する方法。

## ❖ 能動的ステレオ視 (active stereo vision)

ステレオ視において、片方のカメラをレーザ光源などに変えて、ステレオマッチングの精度を向上させた方法。

# 画像符号化

重要



- **可逆符号化 (lossless coding)**  
復号すると、元のデータと同じになる。
- **非可逆符号化 (lossy coding)**  
復号すると、元のデータと厳密には同じにならない。

# 可變長符号

重要

画素値	固定長符号		可變長符号		
	符号語	符号長	出現確率	符号語	符号長
0	00	2 bit	1/2	0	1 bit
1	01	2 bit	1/4	10	2 bit
2	10	2 bit	1/8	110	3 bit
3	11	2 bit	1/8	111	3 bit

平均符号長

2 bit

1.75 bit

# 可逆符号化

---

## ❖ エントロピー符号化 (entropy coding)

出現確率が高い画素ほど、短い符号語を与える。

### ● ハフマン符号 (Huffman coding)

## ❖ 予測符号化 (predictive coding)

隣接する画素値の差分をとることで、データのエントロピーを小さくしてからエントロピー符号化をする。

## ❖ ランレングス符号化 (run-length coding)

2値画像において、同じ画素値が連続する長さを符号化する。

# フーリエ変換

---

## ❖ フーリエ変換

$$F(u) = \int_{-\infty}^{\infty} f(x) \exp(-j2\pi ux) dx$$

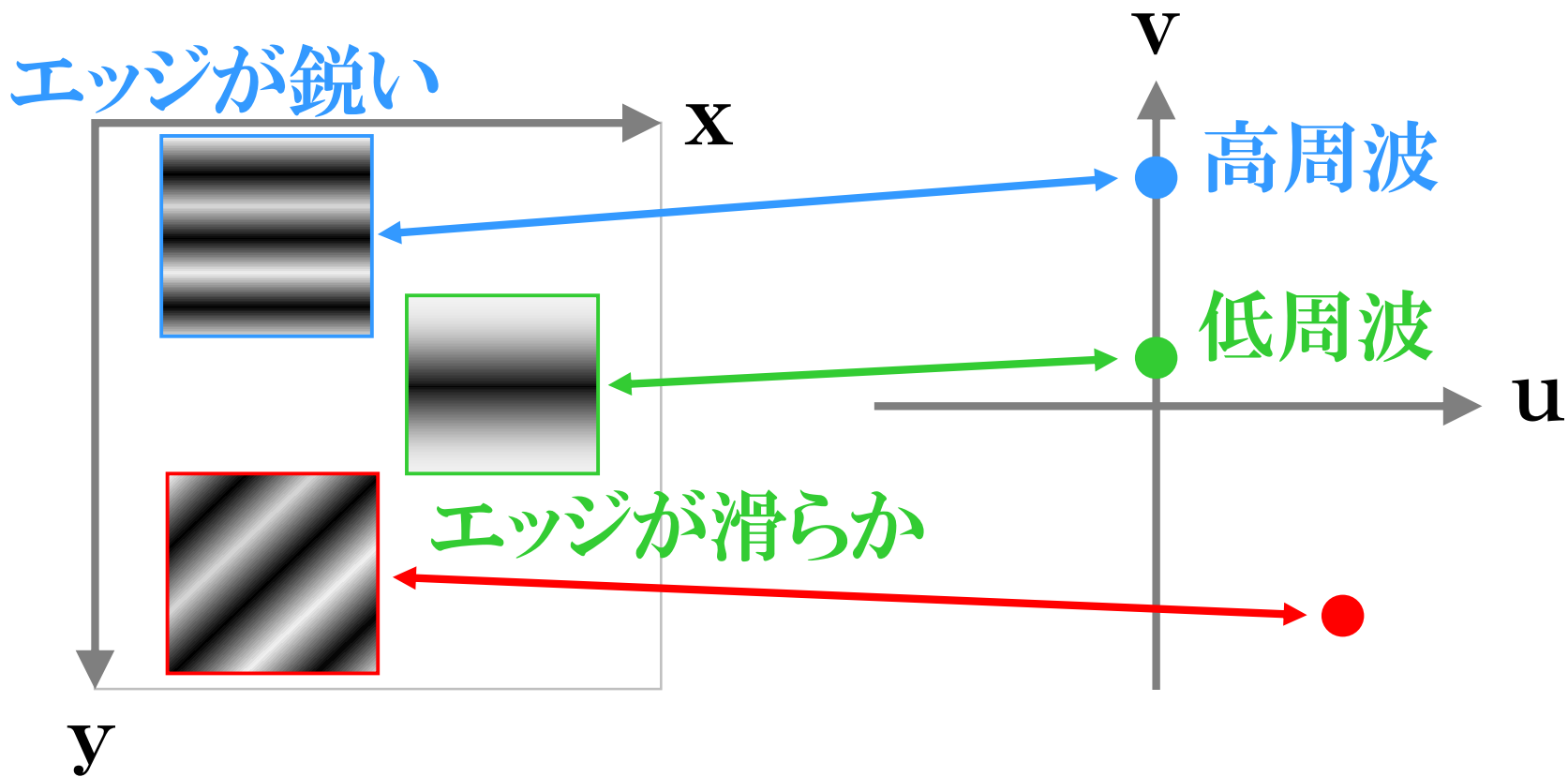
## ❖ 2次元フーリエ変換

$$F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \exp(-j2\pi(ux + vy)) dx dy$$

## ❖ 2次元離散フーリエ変換

$$F(u, v) = \sum_{y=0}^{H-1} \sum_{x=0}^{W-1} f(x, y) \exp\left(-j2\pi\left(\frac{xu}{W} + \frac{yv}{H}\right)\right)$$

# 画像の周波数領域

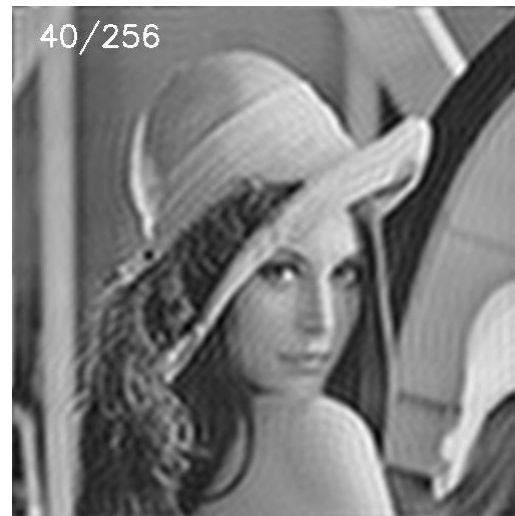
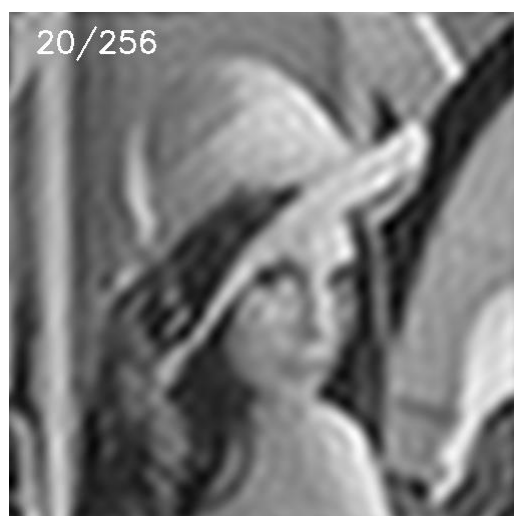
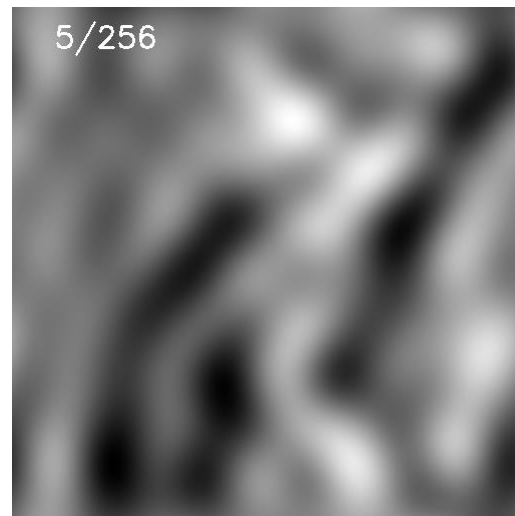
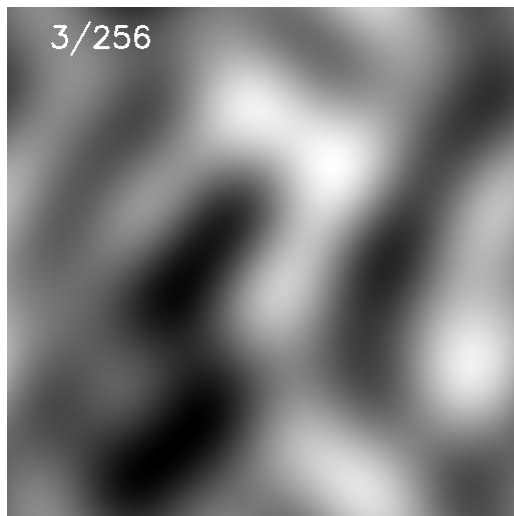
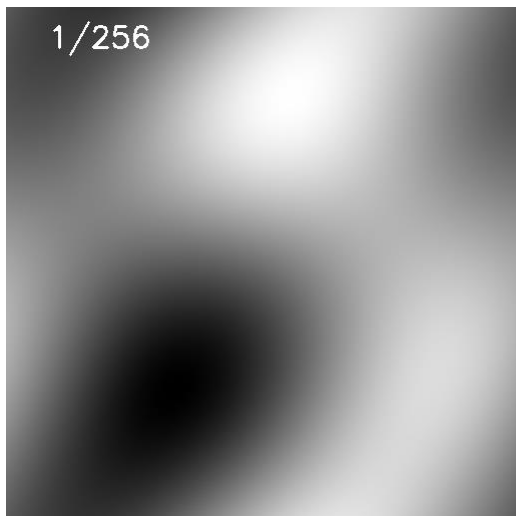


空間領域

周波数領域

# 波の合成による画像生成

---

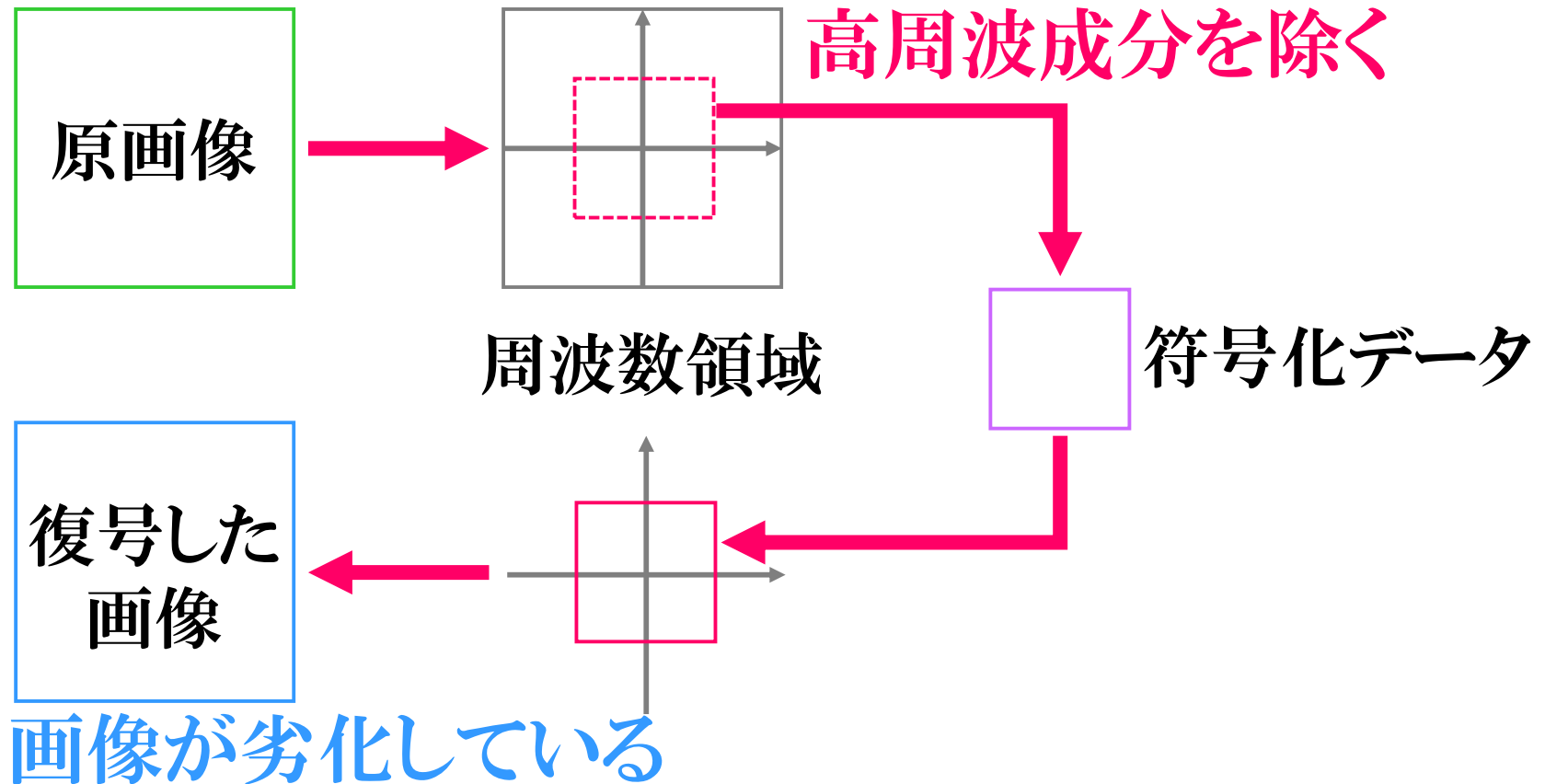




# 変換符号化

重要

画像の周波数成分から高周波成分を削除して、画像のデータ量を減らす。



# コンピュータグラフィックス

---

## ✦ モデリング

3次元空間における物体の形状や材質などを数値的に記述する。

## ✦ レンダリング

3次元物体のモデルを、光源や視点に基づいて、2次元画像に描画する。

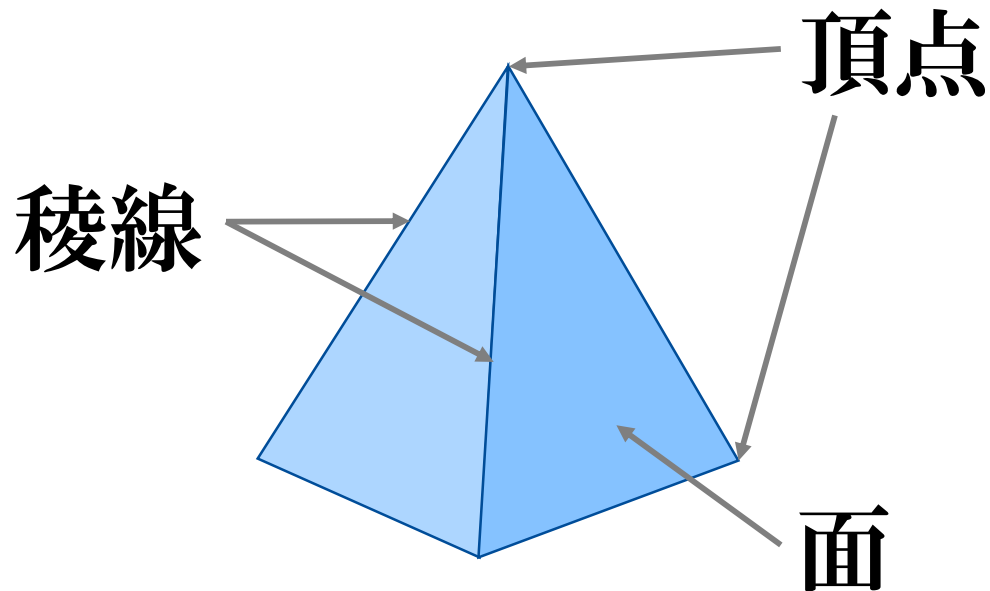
# 形状モデル

---

✦ ワイヤーフレームモデル

✦ サーフェスモデル

✦ ソリッドモデル



# レンダリング

---

- ① 透視投影
- ② 陰面消去
- ③ シェーディング
- ④ テクスチャマッピング

# 画像処理とCGの応用

---

- ❖ **拡張現実感** (AR: Augmented Reality)  
風景の画像上にCGを合成することで、情報を視覚的に補う技術
- ❖ **バーチャルリアリティ**  
コンピュータ上に作られた仮想的な空間をCGなどを用いて疑似体験させる技術
- ❖ **モーションキャプチャ**  
人の動作をリアルタイムに計測する技術

# 定期試験

---

試験日: 8月7日(月)

出題形式: 論述式 教科書・ノート持ち込み可

範囲:

- 画像変換 (AD変換、ヒストグラム、空間フィルタ、幾何学的変換、再標本化など)
- 2値画像処理 (2値化、連結性、特徴パラメータなど)
- 画像認識 (テンプレートマッチング、パターン認識、学習と識別など)
- 画像符号化 (ハフマン符号化、変換符号化など)