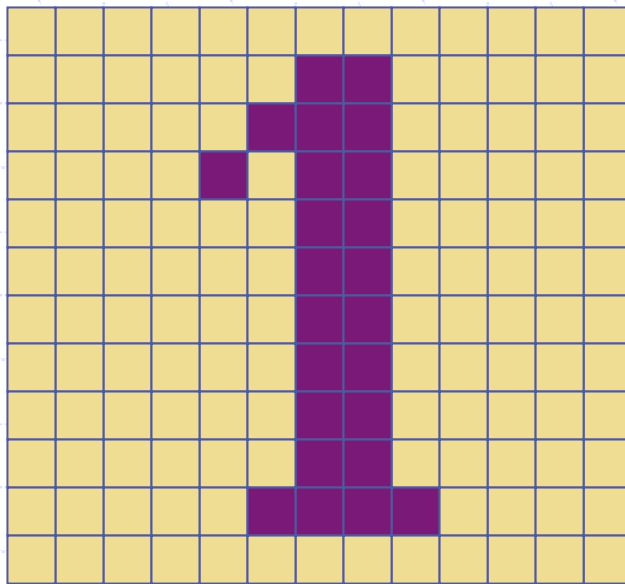


# 二次元画像処理

日浦慎作

# パターンとシンボル

## パターン



- 均質な要素の配列
- 各要素値の並びが重要

## シンボル

0	青
1	赤
2	緑
3	黄

- 不均質均質な要素の配列
- 各要素が独立に意味を持つ

# 画像の処理と認識・理解

## ◆ 画像処理・画像変換 (パターン→パターン)

- 画質改善
- 画像符号化・圧縮
- メディア変換 (不可視情報の可視化)

狭義の画像処理

## ◆ 画像認識・画像理解 (パターン→シンボル)

- 2次元パターン認識
- 3次元画像計測・認識

## ◆ 画像生成 (シンボル→パターン)

- コンピュータグラフィックス

# 二次元画像処理

## ◆ フィルタ演算

- 平滑化・エッジ抽出・ラプラシアン
- メディアンフィルタ等の非線形フィルタ

## ◆ 明度変換

- コントラスト強調・二値化・多値化

## ◆ 二値画像処理

- ラベリング・細線化・領域特徴量

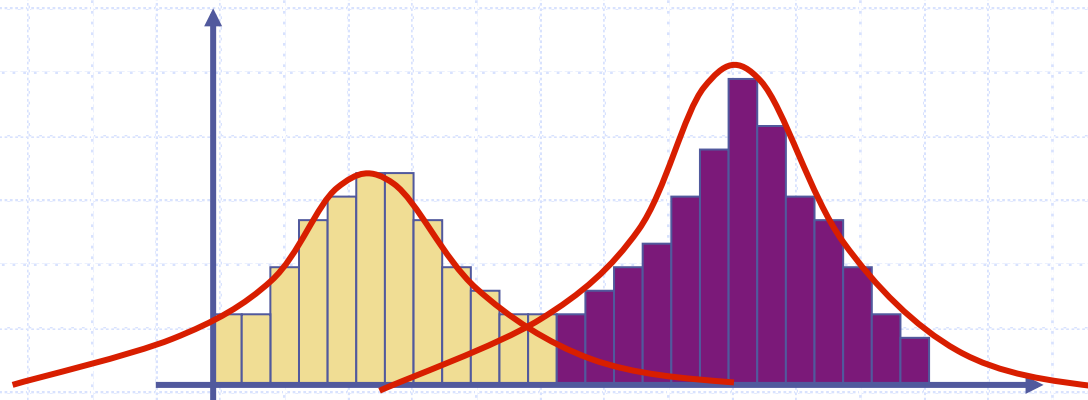
## ◆ 画像圧縮

- 可逆圧縮・非可逆圧縮

# 二値化と閾値決定

- ◆ 図と地の割合が予測できる場合（文書等）
  - P-タイル法  
ヒストグラムを一方から加算した結果がちょうど  $p$  になる値を利用
- ◆ ヒストグラムがはっきりとした双峰性
  - ピーク間の最小値
- ◆ その他の場合
  - 判別分析法

# 判別分析法



## ◆ 特徴量

- 全画素の明度値の平均  $\mu$ , 分散  $\sigma^2$
- 閾値以上・以下の分布をクラス1,2に分類
- 各クラス  $x$  の割合  $w_x$ , 平均  $\mu_x$ , 分散  $\sigma_x^2$

クラス内分散  $\sigma_w^2 = w_1\sigma_1^2 + w_2\sigma_2^2$

クラス間分散  $\sigma_B^2 = w_1(\mu_1 - \mu)^2 + w_2(\mu_2 - \mu)^2 = w_1w_2(\mu_1 - \mu_2)^2$

$$\sigma^2 = \sigma_w^2 + \sigma_B^2$$

## ◆ 閾値の決定

- クラス間の分離度  $\sigma_B^2 / \sigma^2$  を最大にする閾値

# ラベリング

1	1	0	2	0
1	1	0	2	0
1	1	1	●	●
·	·	·	●	●
·	·	·	·	●

重複リスト

1	2
·	·
·	·
·	·

2
1 ●

[1パス]

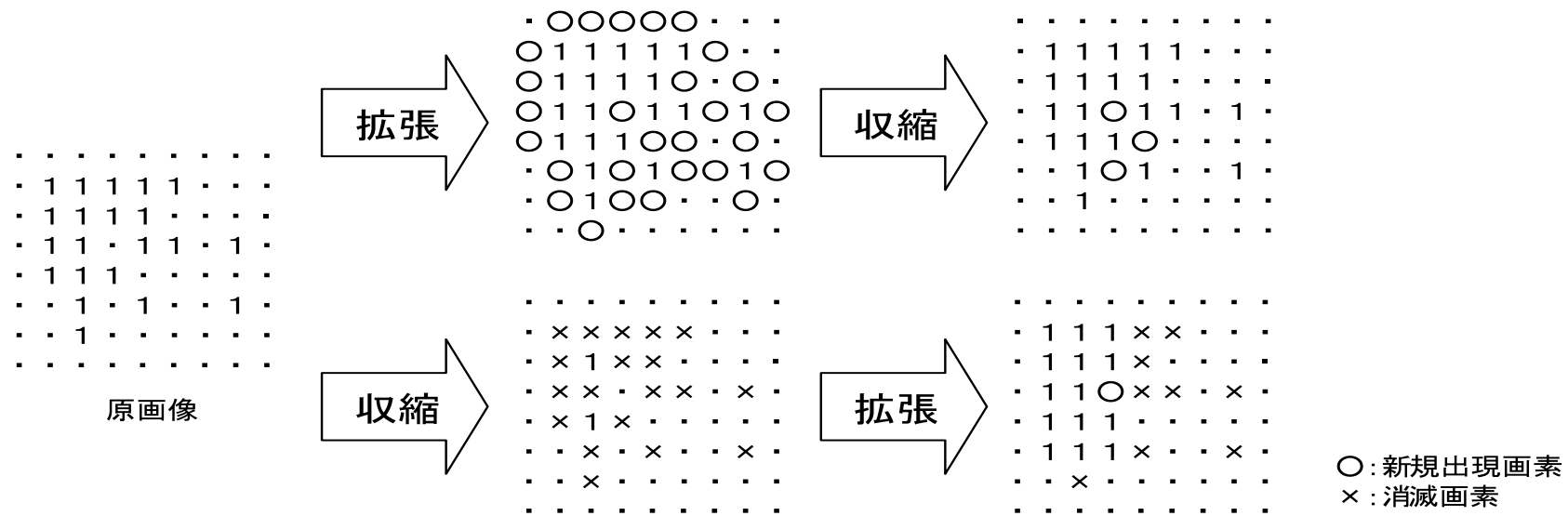
上または左の画素と同じラベルを付与

- 左と上の画素が異なるラベルを持つ場合
  - 重複リストに追加
- 上も左も、0画素である場合
  - 新しいラベル番号を付与

[2パス]

重複リストを元に、ラベルを更新

# 膨張・収縮

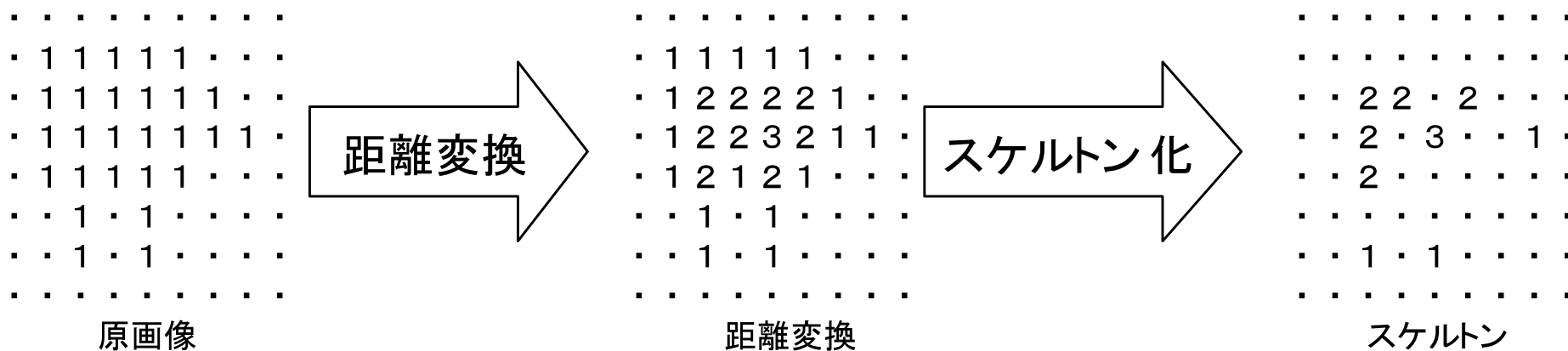


◆ 膨張・収縮 - 穴を埋める効果

◆ 収縮・膨張 - 孤立点を除去する効果

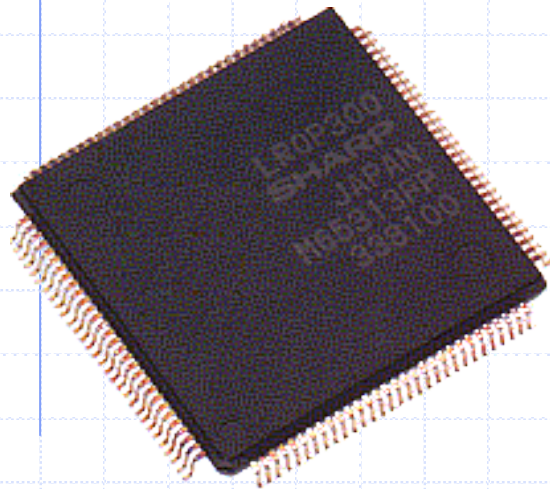


# 距離変換・スケルトン



- ◆ 距離変換 – 何度目の収縮処理で0画素になるか
- ◆ スケルトン – 距離変換画像の極大点  
(近傍画素値が中央画素の値以下)
- ◆ 元の画像を復元可能

# 汎用画像処理LSI

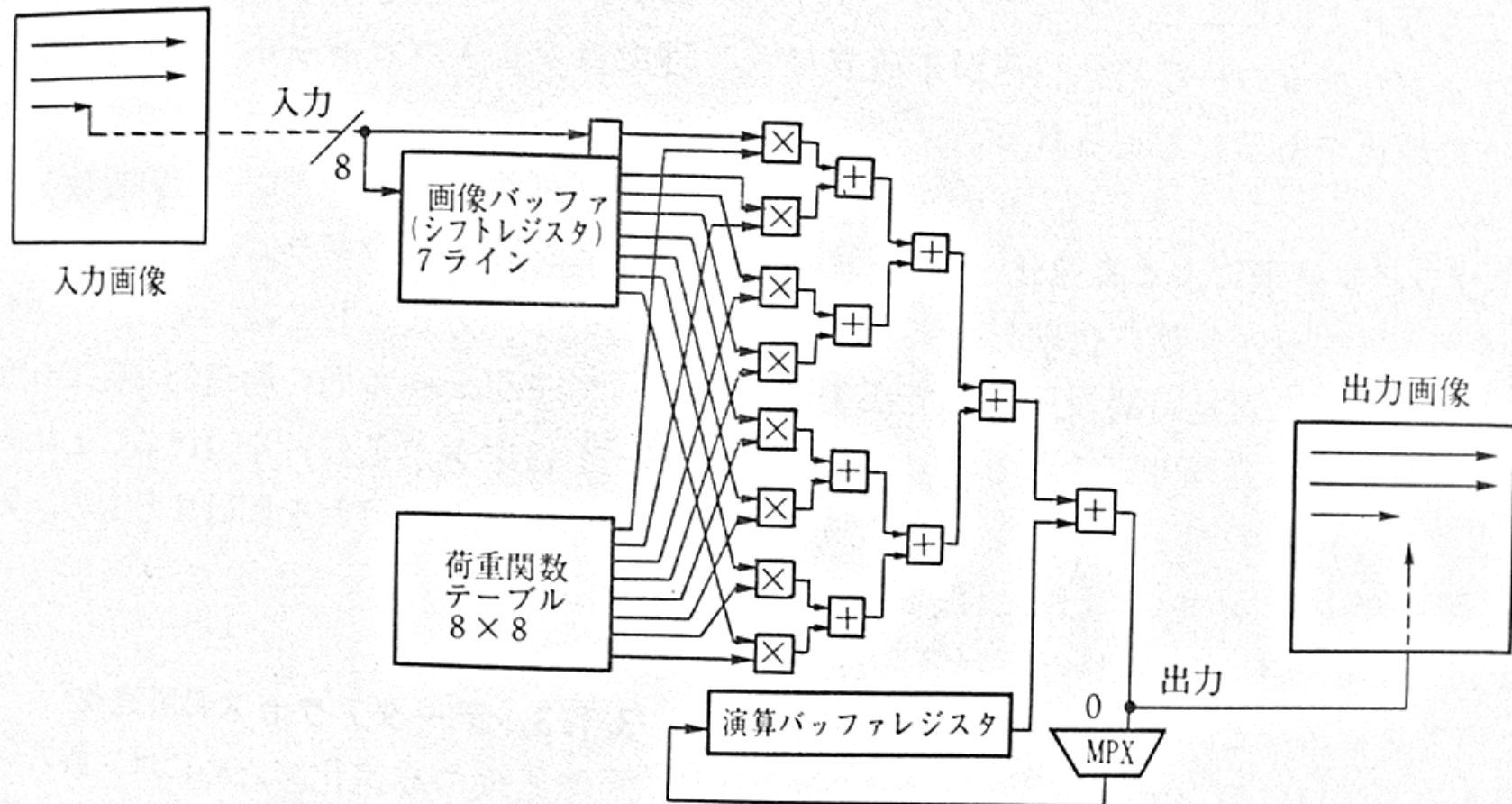


製品例(2000年頃)

- ◆ 3x3 近傍演算に特化
- ◆ 1画素 40nsec (25MHz)
- ◆ 線形・非線形演算が可能
  - フィルタ処理
  - ヒストグラム等の濃淡処理
  - 二値画像処理
  - ラベリング 等



# 近傍演算回路の概要



- ◆ 近傍画素メモリ(シフトレジスタ)と演算回路からなる

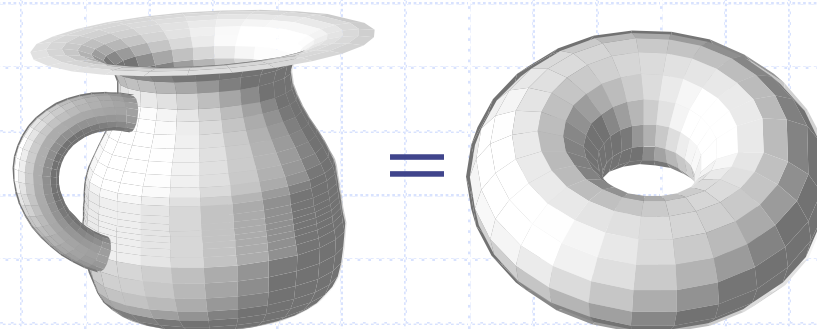
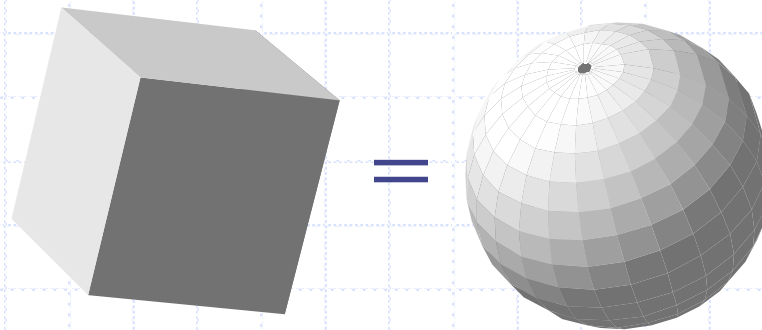
# 近傍演算で出来る処理

- ◆ 線形フィルタリング (1次微分, 2次微分)
- ◆ 非線形フィルタリング
  - メディアンフィルタ等
- ◆ 二値画像処理
  - ラベリング・細線化・膨張・収縮等

# 近傍演算による大局的情報： トポロジーの利用

## ◆トポロジーとは

- 変形しても変化しない図形の性質



# 二次元二値画像のトポロジー

## ◆ 連結成分の穴の数に対応

000000000000	000000000000
001110000000	001111100000
001111110000	001100110000
000011110000	000110010000
000110011000	000111111000
000000000000	000000000000

穴なし

穴1つ

# オイラー数

◆(連結成分の個数) - (穴の個数)

00000000000000	00000000000000	00000000000000
00111000000000	00111110000000	00111110000000
00111111000000	00110011000000	00110111000000
00001111000000	00011001000000	00011101000000
00011001100000	00011111100000	00011111100000
00000000000000	00000000000000	00000000000000

オイラー数: 1

オイラー数: 0

オイラー数: -1



# オイラー数の計算方法

## ◆ 4近傍の場合

```
00000000000000
00111110000000
00110011000000
00011001000000
00011111000000
00000000000000
```

1の個数

(“1”の数:18)  
の数:2) = 0

```
00000000000000
00(1111)000000
00(11)00(11)0000
000(11)00(1)0000
000(11111)0000
00000000000000
```

縦・横2接続

(縦・横2接続の数:8+12)

```
00000000000000
00(1111)000000
00(11)00(11)0000
000(11)00(1)0000
000(11111)0000
00000000000000
```

2x2接続

+ (2x2接続



# オイラー数の原理(1)

## ◆位相が不変な操作

	000000	000000	000000	000000	000000
	011100	011100	011100	011100	011110
	001000	001110	001100	001100	001110
	000000	000000	000000	001100	001100
画素数	+1	+1	+1	+1	+1
縦・横接続	+1	+1	+2	+3	+4
2x2接続	±0	±0	+1	+2	+3
オイラー数	±0	±0	±0	±0	±0

000000  
011110  
001010  
001100

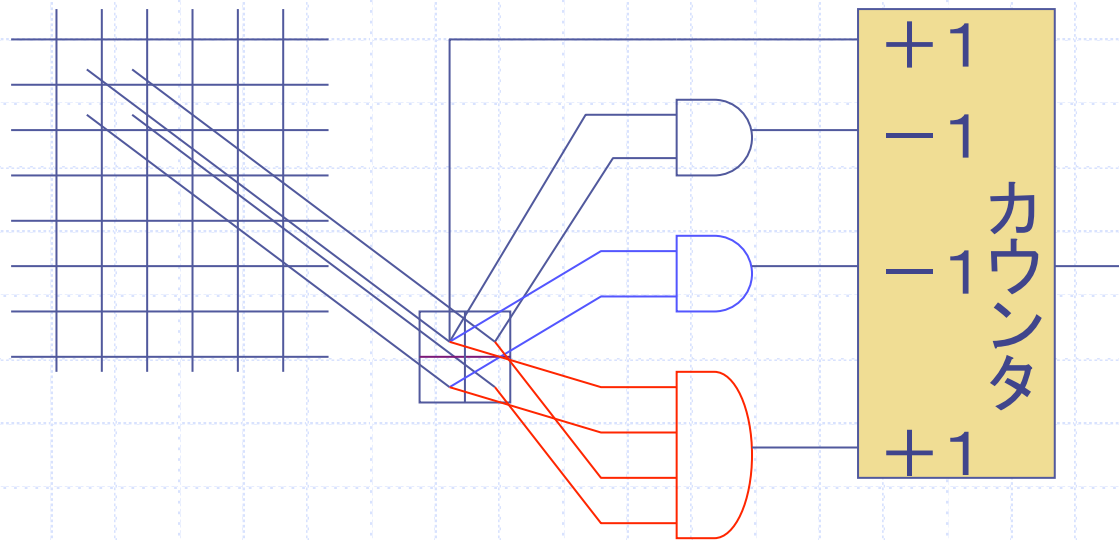
注意: 左の0は穴ではない  
 図("1")が4近傍の場合, 地("0")は8  
 近傍で考える

# オイラー数の原理(2)

## ◆位相が変化する操作

	000000	000000	000000	000000	000000
	000000	111010	000110	011110	011110
	001000	000010	001110	001110	001110
	000000	000010	001100	001110	000100
画素数	+1	+1	+1	+1	+1
縦・横接続	+0	+2	+4	+4	+4
2x2接続	±0	±0	+2	+4	+2
オイラー数	+1	-1	-1	+1	-1
操作	出現	接続	接続	穴埋め	接続


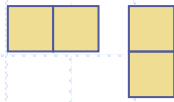
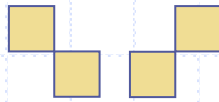
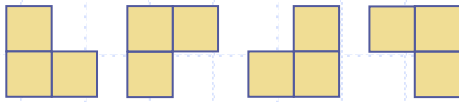
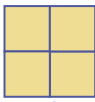
# ハードウェア構成



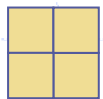
- ◆ 単純な回路で計数可能
- ◆ フィルタ演算 + 画素数数え上げでも可能

# 8近傍のオイラー数

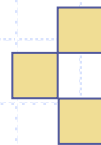
◆ ちょっと複雑

				
画素数	縦・横 2接続	斜め接続	2x2領域中、3画素	2x2接続
V	E	D	T	F

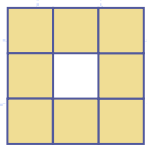
$$\text{オイラー数} = V - E - D + T - F$$



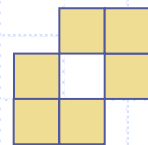
$$4 - 4 - 2 + 4 - 1 = 1$$



$$3 - 0 - 2 + 0 - 0 = 1$$

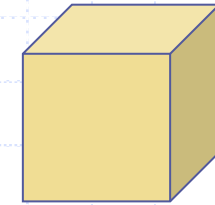


$$8 - 8 - 4 + 4 - 0 = 0$$



$$6 - 4 - 4 + 2 - 0 = 0$$

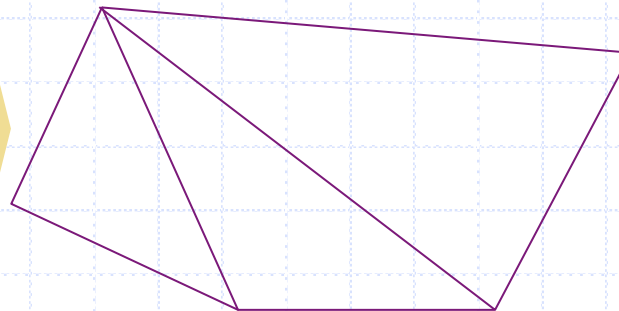
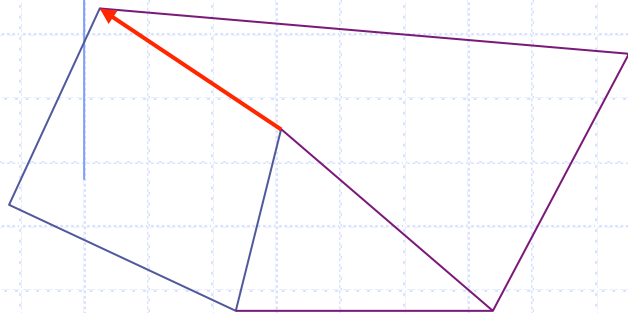
# 余談:オイラー数について



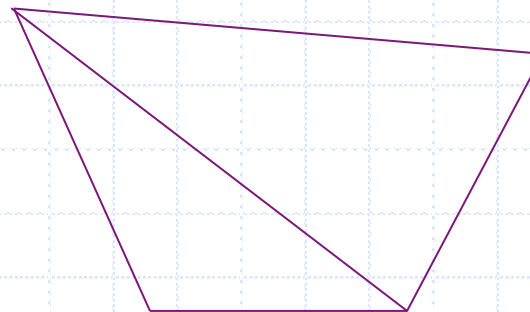
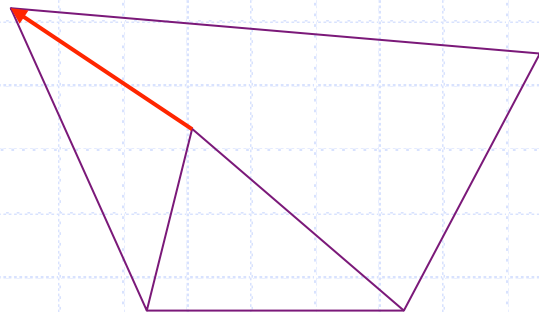
## ◆例:多面体の性質

■ (頂点数) - (辺数) + (面数) = 2 - 2 \* (穴数)

◆ 立方体:  $8 - 12 + 6 = 2$  四角錐:  $5 - 8 + 5 = 2$



両側が四辺形である辺の消去: 辺と頂点が1つずつ消える



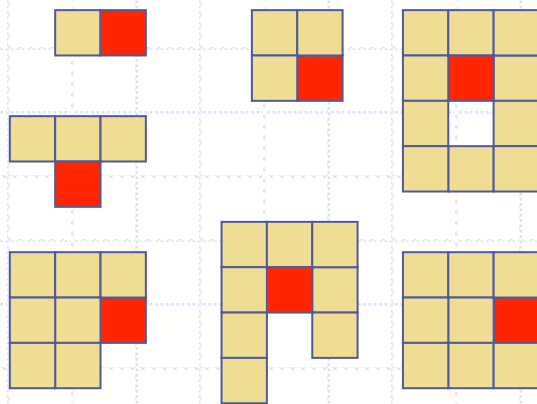
一方が三角形である辺の消去: 辺が2つ消え, 面と頂点が1つずつ消える

→ 最終的に4面体(オイラー数:2)に帰着

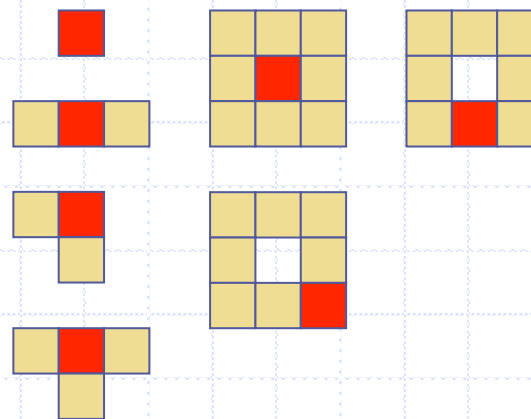
# 消去可能性

## ◆ 消去可能画素とは

- 画像全体の連結性が変化しない画素
- 位相構造を変化させずに図形を変換



消去可能



消去不可能

4近傍の場合

# 消去可能性の計算

$X_4$	$X_3$	$X_2$
$X_5$	$X_0$	$X_1$
$X_6$	$X_7$	$X_8$

◆ 以下の式が1のとき消去可能

◆ 4近傍の場合

$$\begin{aligned} & \blacksquare X_1 + X_3 + X_5 + X_7 \\ & \quad - X_1 X_2 X_3 - X_3 X_4 X_5 - X_5 X_6 X_7 - X_7 X_8 X_1 \end{aligned}$$

◆ 8近傍の場合

$$\begin{aligned} & \blacksquare \frac{X_1 + X_3 + X_5 + X_7}{-X_1 X_2 X_3 - X_3 X_4 X_5 - X_5 X_6 X_7 - X_7 X_8 X_1} \end{aligned}$$

◆ 8近傍: 4近傍の図と地を入れ替えて計算