

# キーワード

圧縮の種類	可逆圧縮 非(不)可逆圧縮
圧縮方法	エントロピー(平均情報量) 符号化(ランレングス, ハフマン, etc.)
画像形式 (フォーマット)	BMP, JPEG, GIF, PNG, TIFF, JPEG2000, PICT, MPEG, etc.

## 圧縮の起源・歴史

モールス通信, 暗号化

# モールス通信の歴史

1832年 サミュエル・F・B.モールス モールス符号の考案

1851年 アルファベット、数字、記号等を表現する現在国際的に共通に使用されている欧文モールス符号の制定

明治時代 欧文モールスにならって  
日本語(和文)モールス符号が考案される

1999年 日本の業務通信からモールスがなくなる。  
GMDSS [Global Maritime Distress and Safety System]  
[海上における遭難及び安全に関する世界的な制度]

現在 アマチュア無線の世界で健在

# モールス符号とは

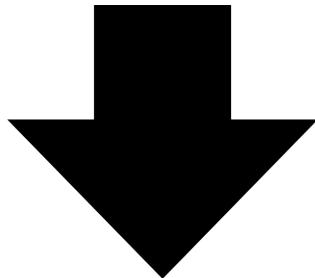
## ●モールス符号の規則

モールス符号は”長点”、”短点”、”空白”の三つから構成される。

短点の長さ1に対して、空白1，長点3の長さをもつ

文字間の空白は短点の長さの3倍

単語間の空白は短点の長さの7倍



種類	符号語	符号長
長点	—	3
短点	·	1
空白		1
文字間の空白		3
単語間の空白		7

規則正しい符号の長短のON/OFF組み合わせだけで、文字をあらわすことができる

→デジタル通信方式の元祖といえる



# モールス信号

A(5回繰り返し)

トツー トツー トツー トツー

B(5回繰り返し)

ツートト ツートト ツートト ツートト

C(5回繰り返し)

ツートツート ツートツート ツートツート ツートツート

モールス符号は、音のパターンで覚える！

モールス通信にチャレンジ？

# 大局的な画像の形式

- ・ビットマップ形式
- ・ベクタ形式

## 画像のフォーマット

JPEG, GIF, BMP, PICT,  
PNG, TIFF, JPEG2000,  
MPEG, etc.

# 画像の形式

## 【ビットマップ(ラスタ)画像】

(ビットマップグラフィックス)

色のついた点(ドット)の羅列として画像を表現する形式

風景や人物などの写真を表現するのに適している

## 【ベクタ(ベクトル)画像】

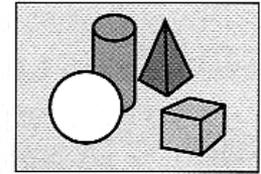
(ベクタグラフィックス)

点とそれを結ぶ線や面の方程式のパラメータ, および塗りつぶしなどの描画情報によって画像を表現する形式

図形や記号などのように線で輪郭が描かれた画像に適している



風景写真など



図形や記号など

ビットマップ画像が向いている

ベクトル画像が向いている

「ペイント系」ソフトで編集する

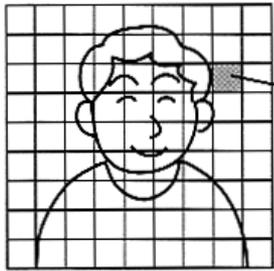
「ドロー系」ソフトで編集する



ビットマップ画像とベクトル画像を使い分けるといことは、結局ソフトを使いわけるといことなんだね

# ビットマップ画像

## カラー画像の表現法



赤の濃度値

緑の濃度値

青の濃度値

1つひとつの画素に、赤・緑・青のそれぞれの濃度値の情報をもたせる

## 通常の画像表現



赤	7	赤	14
緑	15	緑	14
青	21	青	10
赤	20	赤	14
緑	14	緑	14
青	22	青	55
赤	13	赤	21
緑	45	緑	14
青	14	青	14

それぞれの画素に赤・緑・青の濃度値が割り当てられているんだ



## インデックス表現



7	8		
4	3	17	26
6	2	5	10
9	18	11	32
22	65		

画素の濃度値のかわりに、その画素の色番号を割り当てているんだね



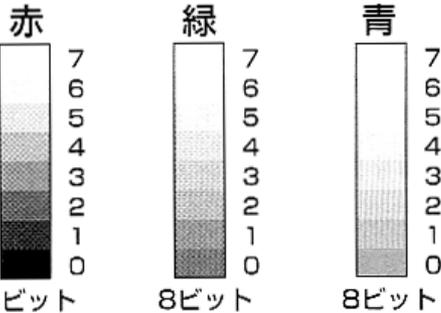
## カラーテーブル

番号	( 赤 , 緑 , 青 )
0	( 255, 250, 254 )
1	( 254, 249, 250 )
②	( 250, 240, 247 )
3	( 249, 235, 242 )
⋮	⋮
254	( 2, 14, 8 )
255	( 0, 0, 0 )

これなら、色の番号だけ記録すればいいからデータの量が少なくてすむよ

よく出てくる色をカラーテーブルにしておく

## 24ビットカラー



=24ビット

表現できる色 2の24乗=約1,677万色!!

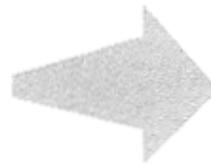
これだけ色の数があると、自然なカラー画像が表現できるね



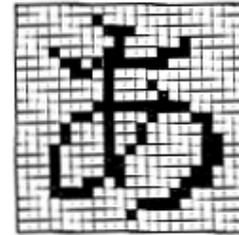
# ベクタ画像

拡大・縮小，回転，移動  
などの操作を行っても  
画質が劣化することはない。  
表示する都度に再計算が  
行われる。

## ビットマップ画像

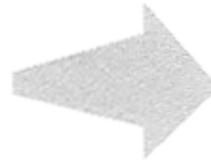


拡大



拡大すると  
ギザギザが  
めだつ

## ベクトル画像



拡大



拡大しても  
ギザギザに  
ならない

## ベクトル画像のしくみ



たとえば、円の図形を表  
すときは、「中心 (A,B)  
・半径C・色は青」とい  
うように表現するのね

図形を点の集まりではなく、数式の形で表現している

代表的なベクタ画像  
文字を表示・印刷する  
際に使うフォントデータ

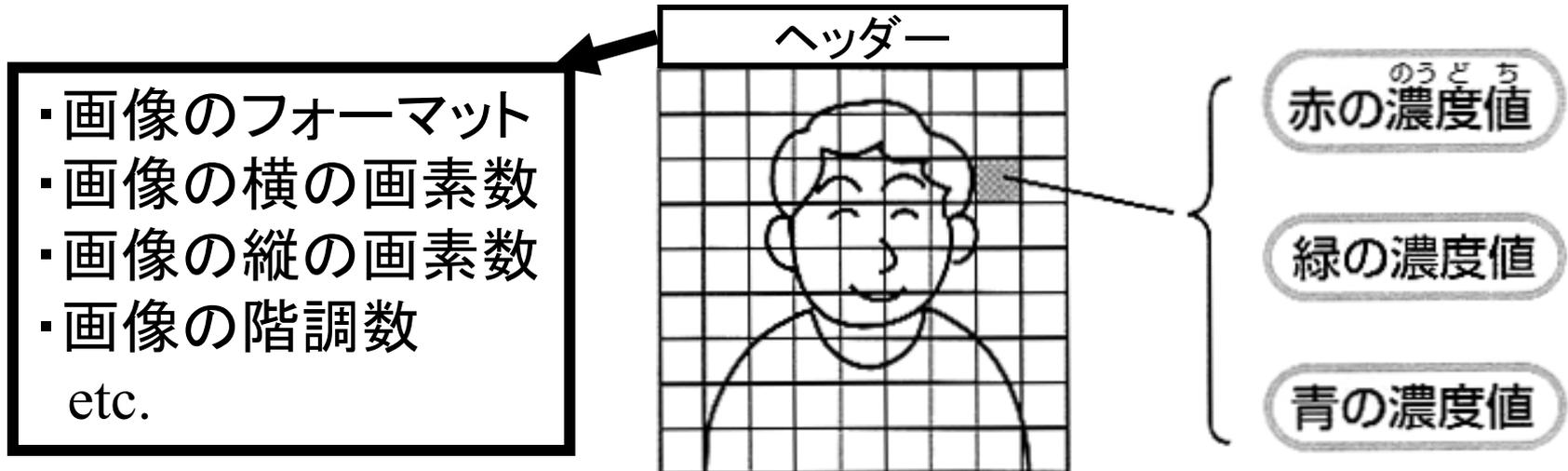
# 大局的な画像の形式

- ・ビットマップ形式
- ・ベクタ形式

## 画像のフォーマット

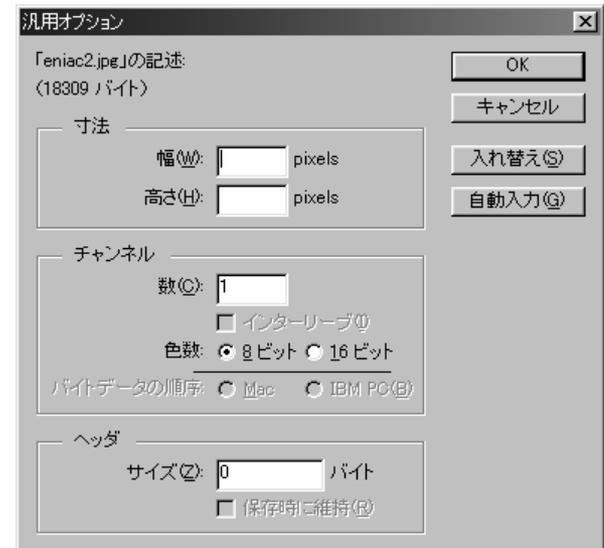
JPEG, GIF, BMP, PICT,  
PNG, TIFF, JPEG2000,  
MPEG, etc.

# 画像データの格納形式



## RAW (ロー)

- ・ビットマップ形式の画像データのみを保有している画像  
(ヘッダーを付加する場合もある)
- ・最も汎用性が高く、OSやハードウェアにほとんど依存しない。



# JPEG (ジューペグ)

(Joint Photographic Experts Group)

- ・Web上で使用される2大画像フォーマットのひとつ↑
- ・写真のエキスパート集団が中心になって策定した画像フォーマット
- ・8～32ビットの色数で画像を格納
- ・圧縮アルゴリズム (JPEG圧縮):  
DCT (離散コサイン変換, ランレングス, ハフマン符号化, etc.)  
(画像のピクセルを固まりとしてとらえてその変化を記録する圧縮アルゴリズム)
- ・色が離散している写真や自然画などに適している
- ・一般的に非可逆圧縮  
(厳密には可逆圧縮, 非可逆圧縮どちらも可能であるが,  
非可逆圧縮のほうが圧倒的に多く用いられている)
- ・2022年7月にForgent社が突然特許権の保持を宣言 ×  
→サブマリン特許

# GIF (ジフ) (Graphic Interchange Format)

- Web上で使用される2大画像フォーマットのひとつ ↓
- 米CompuServe社が開発した画像フォーマット
- 1～8ビットの色数(2～256色)で画像を格納
- LZW圧縮を使用 (米UNISYS社がLZWの特許を所有, 2003年に失効)  
(出現する頻度の高い情報をまとめるアルゴリズムの一種)
- ベタ塗りの多い画像(アイコン, イラストなど)に高い圧縮率を示す
- 可逆圧縮

## インタレースGIF

画像を格納する際、ラインを飛び越しながらピクセルを格納する方式。Webブラウザ上でモザイクがかかったような見えかたをする。

## アニメーションGIF

複数の画像をまとめて格納する。Webブラウザからはいわゆるパラパラマンガ方式で、アニメーションが見える

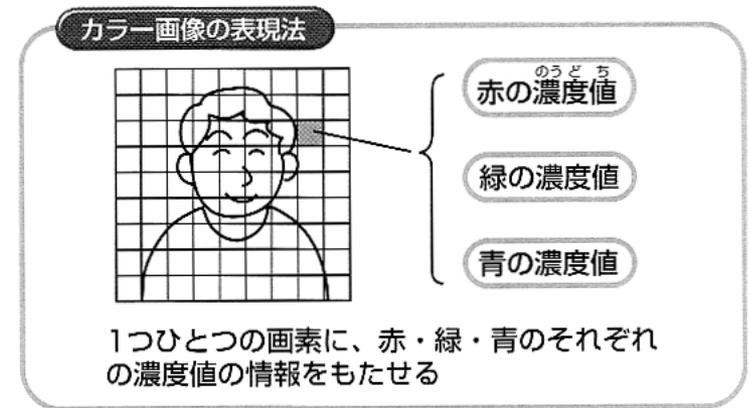
## トランスペアレント

GIF画像のなかの特定の1色を透過させる。Webブラウザからは指定された色が抜けているように見える。

# BMP (ビーエムピー) (BitMaP)

- ・Windowsが標準でサポートしている画像形式
- ・白黒(2値)の画像～フルカラー(1677万7216色)まで扱える
- ・基本的には無圧縮

BMPのヘッダーをつけ、ビットマップデータを最後尾から格納する形式(ほとんどRAW形式)  
圧縮はオプションで可能だが、圧縮率は高くない



# PICT (ピクト)

- ・Apple社のMacOSが標準でサポートしている画像形式の一つ
- ・MacOSのグラフィックス処理エンジンである32bit-QuickDrawが利用する画像フォーマット。
- ・ベクトルデータ, ビットマップデータの両方を格納可能
- ・フルカラー画像, データ圧縮にも対応

# TIFF (ティフ) (Tagged Image File Format)

- ・Aldus社(現Adobe社), Microsoft社などのスキャナメーカーが規格した画像フォーマット
- ・異なるOSやパソコン機器での画像ファイル交換が目的
- ・画像ファイルの先頭にタグと呼ばれる識別子を付けるのが特徴
- ・汎用性が高い ・一般的には無圧縮(LZW形式, PackBits形式で圧縮可能)

# PNG (ピング) (Portable Network Graphics)

- ・WWW上で広く使われることを目指して開発された(GIFの代り)
- ・圧縮アルゴリズムはdeflation方式(GIFの問題を避けるため)  
LZ77とハフマン符号化の2段階の圧縮を行なう方式 ←ファイル圧縮形式Zipで利用
- ・フルカラーの自然画を劣化無しで圧縮できる
- ・可逆圧縮 ・各ピクセル毎に透明度を指定できる

# データ(画像)圧縮

情報の余分な部分を取り除いて、データ量を減らすこと

↓  
冗長性

## 圧縮のメリット

通信時間や効率の向上

データ保管の効率の向上

# 可逆 vs 非可逆

可逆圧縮  
(Lossless)

圧縮して元に戻した画像が、圧縮前の画像と完全に一致する

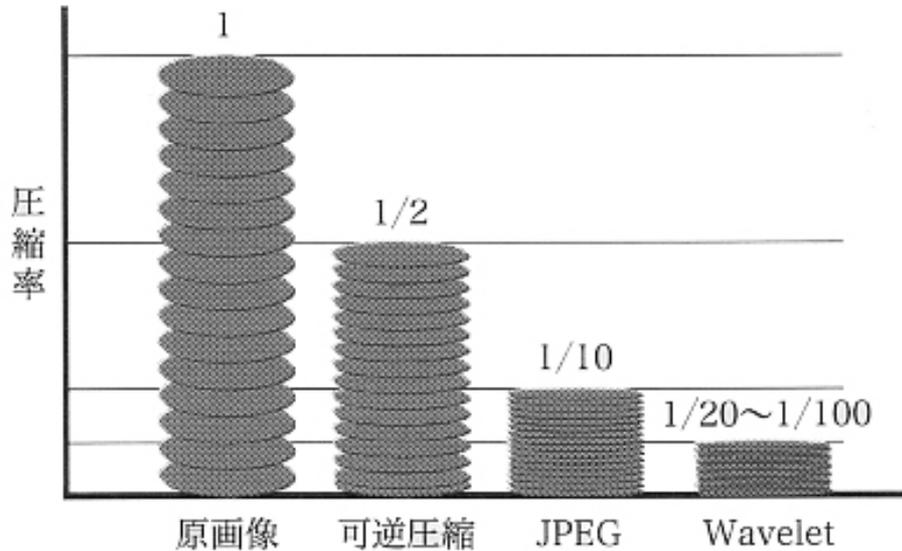
非可逆圧縮  
(Lossy)

圧縮して元に戻した画像が、圧縮前の画像と完全には一致しない

デジタル画像では、非可逆圧縮のほうがよく使われる。

→画像には人間の目では取り除いても気づかない成分がある。

# 圧縮率



インターネットでは  
JPEG, GIF, PNG形式の画像  
が利用されている。

医用画像では  
**JPEG圧縮**と  
**Wavelet圧縮**が  
主に用いられて  
いる。

	可逆圧縮	非可逆圧縮
圧縮率	1/2~1/4	1/10~1/100
圧縮形式 符号化	LZW entropy符号 算術符号	DCT Wavelet
画像形式	GIF, TIFF PNG...	JPEG JPEG2000

# LZW圧縮 (Lempel Ziv Welch)

1977年にAbraham Lempel氏とJacob Ziv氏が「LZ77」アルゴリズムとして発表した辞書圧縮という手法を、Welch氏が改良した方式

既に読込まれたデータの中からもっとも長い一致文字列を探し (もし発見できた場合に) その位置と一致長で置き換える方法

## 【辞書圧縮の例： 英文圧縮】

元データ

That algorithm is very simple. ....  
..... algorithm .....  
.....  
algorithm .....  
..... algorithm .....

algorithm → 9 bytes

圧縮データ

That algorithm is very simple. ....  
..... 6,9 .....  
.....  
6,9 .....  
..... 6,9 .....

6,9 → 3 bytes

6 bytes節約

# LZW圧縮の特徴

- ・出現する頻度の高いものが多いければ高い圧縮率が得られる.
- ・出現頻度の計算は走査方向に依存する.



(アンチエイリアシングあり)  
1321 bytes



(アンチエイリアシングなし)  
622 bytes

# JPEG圧縮

ジェイベグ  
JPEG=Joint Photographic coding Experts Group



JPEGは、<sup>が そう あっしょく</sup>画像圧縮の方式を決めたグループの名前なんだ

可逆・非可逆どちらにも対応している

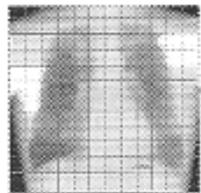
が

高圧縮率を実現するために非可逆圧縮のほうが圧倒的によく使われている。

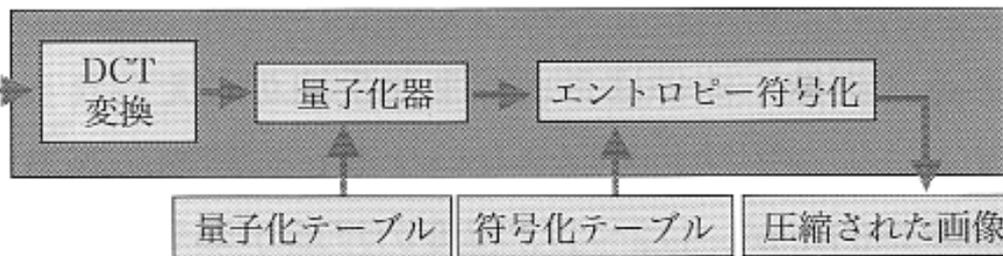
Why?

8×8ブロック

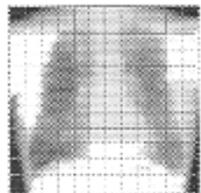
DCTによるエンコーダ



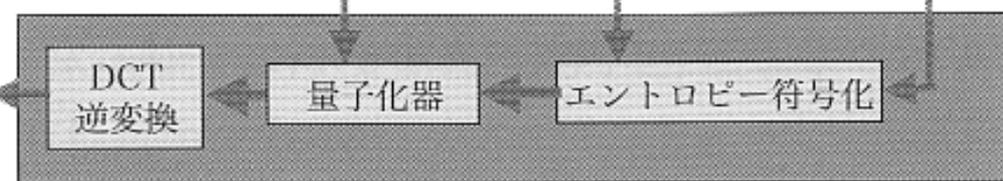
原画像



8×8ブロック



復元画像



DCTによるデコーダ

人間の目の性質をうまく利用して、見た目がそれほど変わらないように圧縮しているため

# JPEG圧縮のしくみ①

JPEG圧縮 (不可逆) のしくみ (STEP1)



元の画像

明るさの成分

色の变化の成分

色の变化の成分のデータを減らす

色の变化の成分

人間の目は、色の变化にはあまり敏感じゃないからデータを減らしても大丈夫なのね



人の目の性質

明るさの変化 : 敏感

色の变化 : 鈍感

# JPEG圧縮のしくみ②

人の目の性質

低周波成分: 敏感, 高周波成分: 鈍感

JPEG圧縮の手順



元の画像

STEP1

色の変化の  
データを削る

STEP2

高周波成分の  
データを削る

人間の目は、画像の高周波成分  
には敏感ではないから、データを削っても大丈夫なんだ



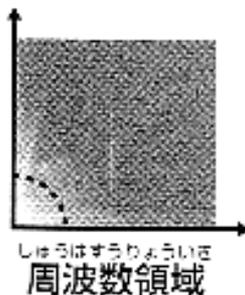
# JPEG圧縮のしくみ③

## DCT: 離散コサイン変換 (Discrete Cosine Transform)

どうやって高周波成分を削るか



DCT



ディーシーティ  
DCTはフーリエ  
変換の兄弟のよ  
うなものなのね

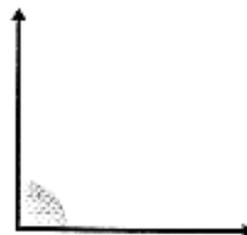
色の変化成分を  
削った画像



高周波成分  
をカット!



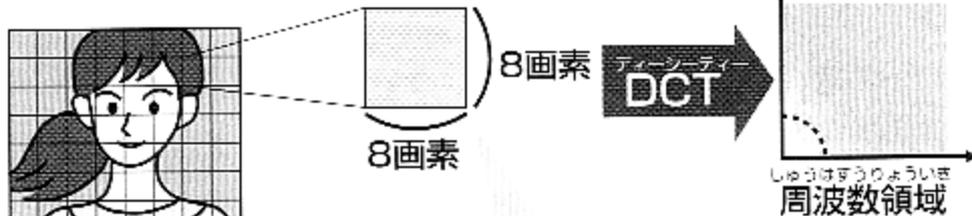
逆DCT



画像を表示する  
ときは逆DCTを使う

この状態でデータ  
を保存する

# JPEG圧縮 の性質



計算時間の節約のため、8×8画素のブロックごとに高周波成分を取り去る



高周波成分をたくさん削ってしまえばデータが小さくなるけど、画質も悪くなるよ



元の画像



20分の1に圧縮



30分の1に圧縮

## 人の目の性質

1/10に圧縮  
1/20に圧縮  
1/30に圧縮

ほとんど分からない  
少し画質が落ちた  
かなり画質が落ちた



JPEG圧縮では、圧縮の割合を自由に設定できるのね

# JPEG vs GIF



1957 bytes



620 bytes



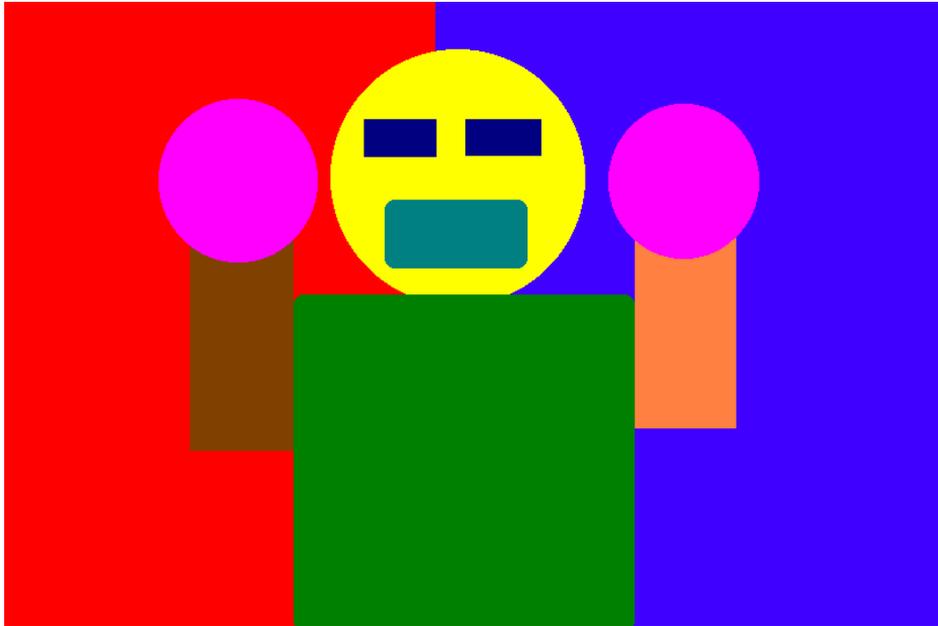
88.1 Kbytes



227 Kbytes

# BMP, JPEG, GIF

イラスト(べた塗り画像)



BMP 1255 Kbytes  
JPEG 73 Kbytes  
GIF 7 Kbytes

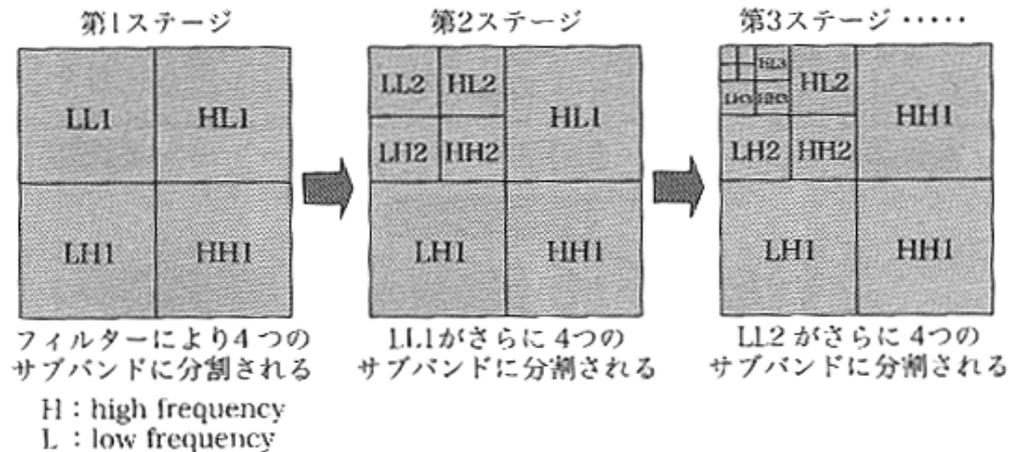
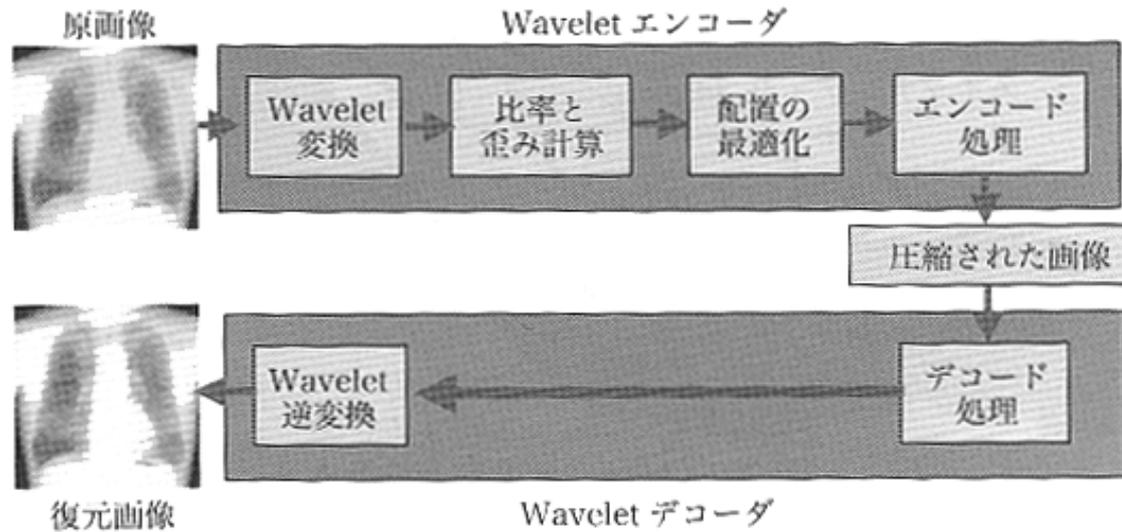
自然画



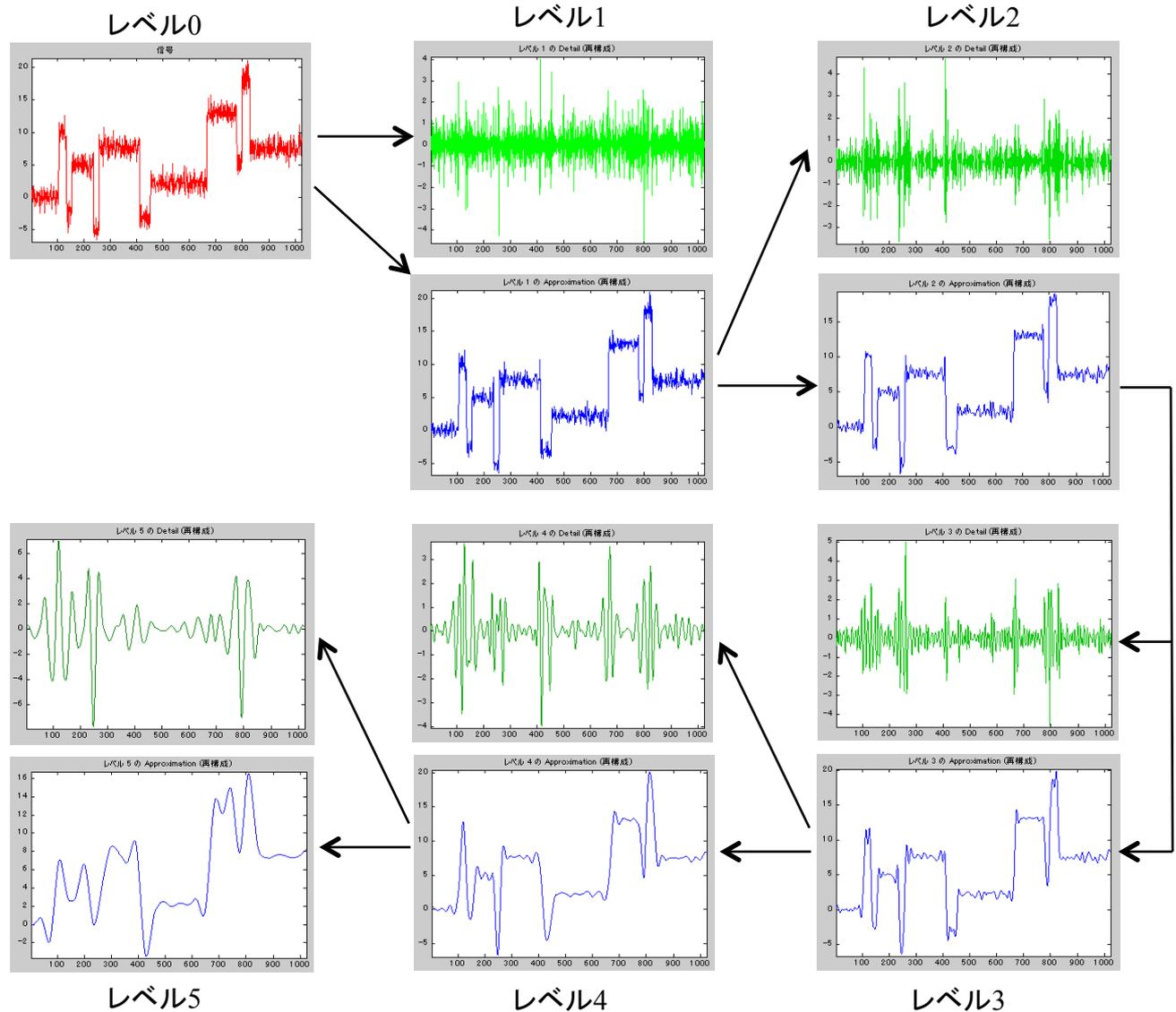
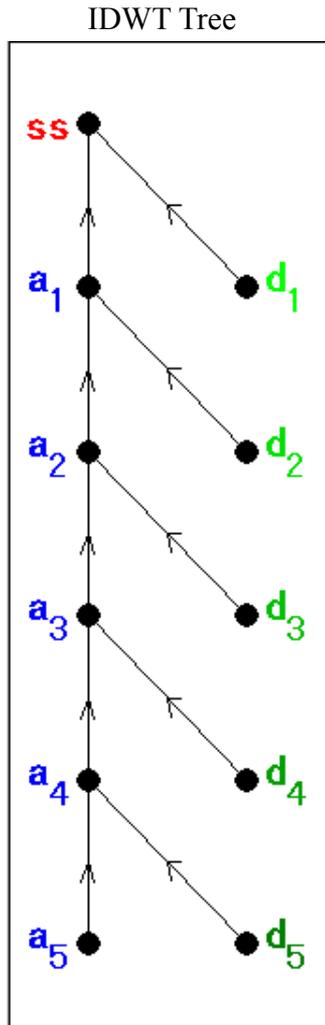
BMP 9217 Kbytes  
JPEG 570 Kbytes  
GIF 1066 Kbytes

# Wavelet圧縮

離散ウェーブレット変換 (DWT: Discrete Wavelet Transform)



# DWT: 多重解像度解析(1次元)

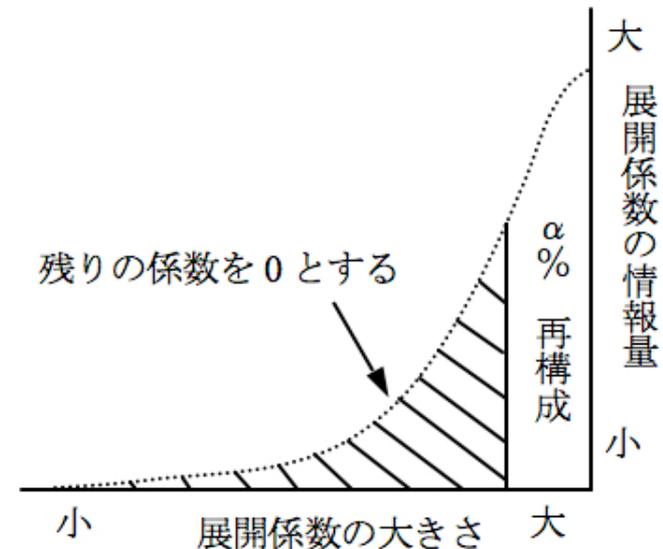


# DWTによるデータ圧縮

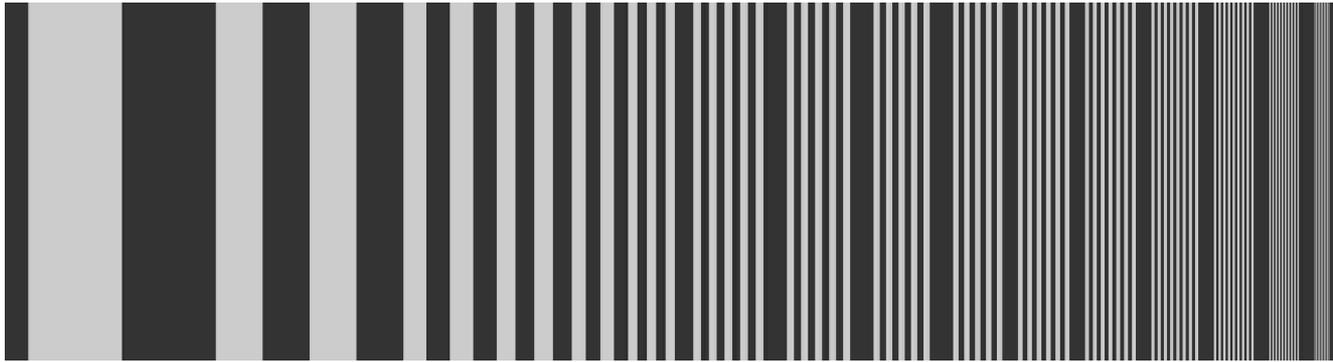
- DWTでは、絶対値の大きい順に上位数パーセントの展開係数にエネルギーが集中する
  - データ中に存在する頻度の高いパターン

## DWT圧縮の基本的な考え方

絶対値の大きい展開係数のみを残し、残りの係数を0にして再構成すれば、信号の特徴を保存したまま、少ないデータ量で信号が再構成される



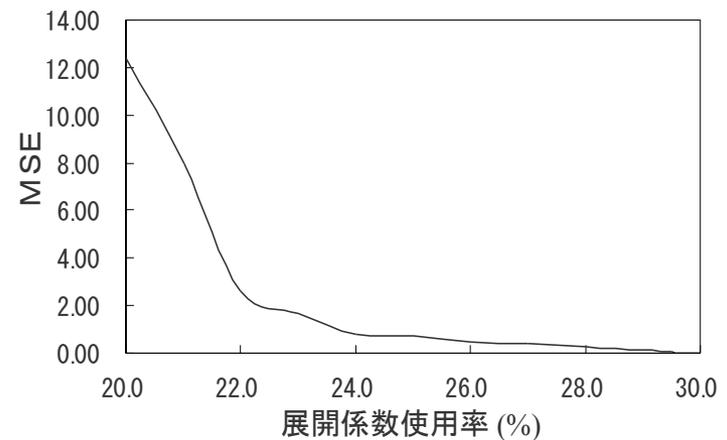
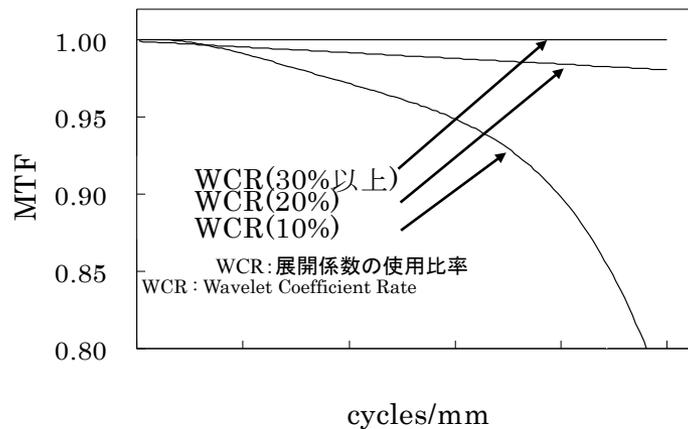
# DWTによる圧縮シミュレーション



0.05 LP/mm ~ 10.0 LP/mm

コンピュータ上で作成したシミュレーションテストパターン

DWT後, 展開係数の使用比率(WCR= $\alpha$ )を決めて, 再構成を行う



Original



WCR=30%



WCR=20%



WCR=10%



WCR=5%



WCR=2%



WCR=1%



WCR=3%



WCR=5%



WCR=7%



# JPEG2000

JPEGよりも高圧縮、高品質な画像圧縮が行なえる  
→医療画像においてJPEGに置き換わる可能性が高い？

JPEGでは高圧縮率(低画質)で保存したときに目立っていた**ブロックノイズ**(格子状ノイズ)や**モスキートノイズ**(水面の波紋状のノイズ)が、JPEG2000では発生しない

最も大きな違い	
	周波数領域での処理
JPEG	DCT(離散コサイン変換)
JPEG2000	<b>ウェーブレット変換</b>

急速な普及が予想されたが、現実的にはそれほど普及していない。 Why？

- ・Windowsが標準サポートしてない。(JPEGはOK)
- ・Webブラウザのサポートがいまいち。

JPEG

JPEG2000

1/120圧縮



# JPEG

## VS

# JPEG2000



1/60圧縮



1/30圧縮