

数理・データサイエンス教育強化拠点コ ンソーシアム 模擬講義：「3-5 認識」

東京大学大学院 情報理工学系研究科
数理・情報教育研究部門
准教授 山肩洋子

担当講義

- 「メディアプログラミング入門」を担当
- 音、テキスト、画像の情報処理を座学＋演習
- おもに情報分野を専門としない学部・大学院が受講
 - 図や写真、実例を交えて紹介
- 254名が学務システムに登録、初回課題提出198名
- 講義資料・動画・演習教材（Jupyter notebook）を公開中
（東京大学 数理情報教育研究センターWebページよりe-learning教材>[数理・データサイエンス関連教材](#)>メディアプログラミング入門 山肩 洋子 准教授）
- コンソーシアム教材担当
 - [リテラシーレベル教材](#)：「4-6 画像解析」
 - 応用基礎レベル「3. AI基礎」のうち「3-2. AIと社会」「3-5. 認識」

私の遠隔講義環境

セカンドスクリーン

- スライドの流れを確認
- 演習 (Colaboratory) を実行

Zoomのチャットで
質問受付

- カメラと音はこちらを使用
- 配信映像を確認

- 授業はipadでスライド画面を共有
- Apple pencilで書き込みながら話す

演習@Colaboratory

- 予習教材をJupyter notebook形式で配布
 - 旧バージョンは当センターのe-learning教材「メディアプログラミング入門」よりダウンロード可能
- Colaboratoryのアドバンテージ
 - 学生それぞれのPC環境に依存しない
（ローカル環境構築を遠隔でサポートするのはほぼ不可能！）
 - GPUなど高性能な計算リソースを利用できる
 - 学習済みモデルのダウンロード等（ときには400MB超え！）で学生の通信量を消費しない（ローカルストレージも消費しない）
- 課題もColaboratory上でコーディングさせ、そのJupyter notebookを提出させる

3-5 認識

東京大学 数理・情報教育研究センター
2021年5月1日

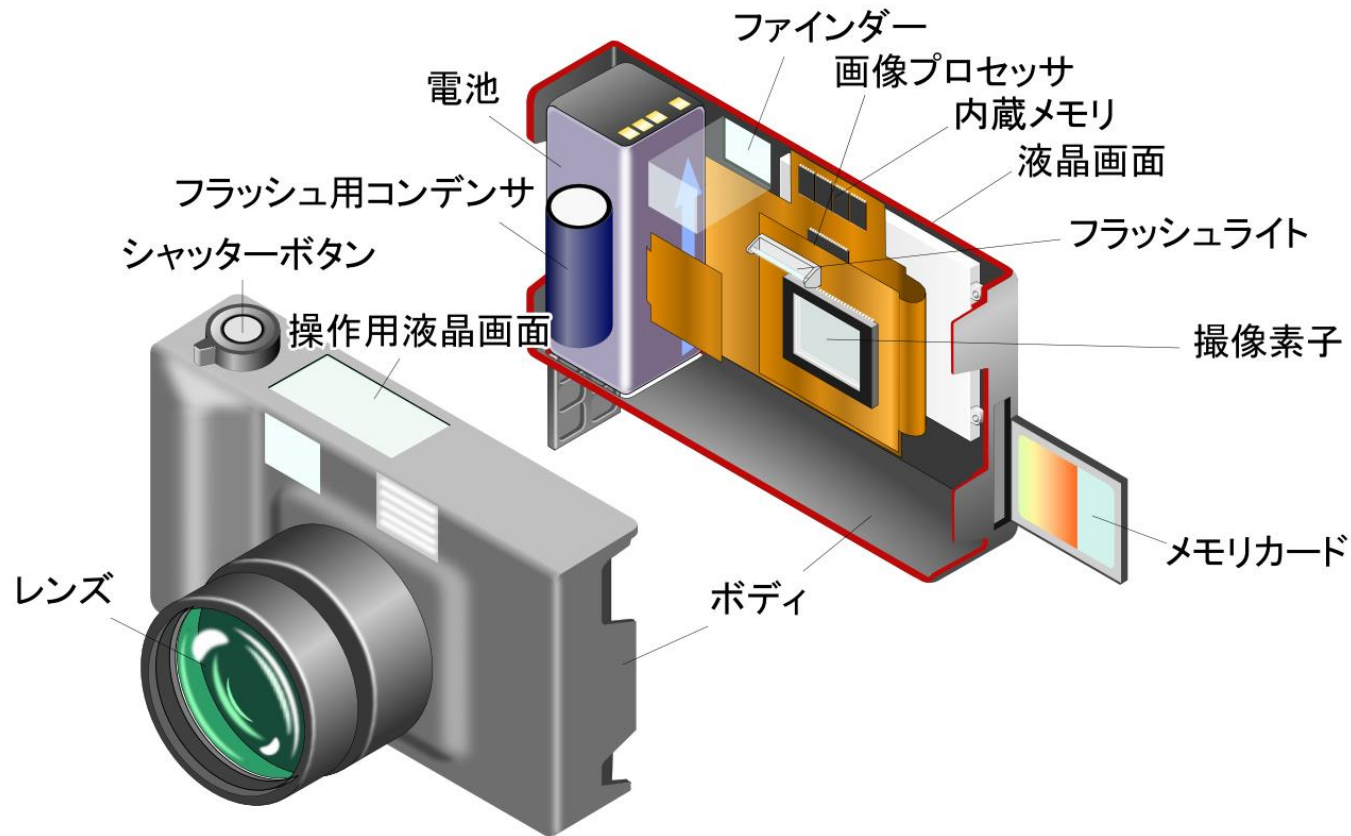
本教材の目次

1. AIにおける認識とは	4
2. 認識技術の活用事例	1 2
– 画像認識の活用事例	1 2
– 音声認識の活用事	2 4
3. 画像処理・認識技術	2 9
– デジタル画像の表現	2 9
– 二次元デジタルフィルタによる画像処理	3 6
– 深層学習による画像認識	4 8
4. 音響処理・音声認識技術	5 6
– デジタル音の表現	5 6
– 周波数解析	6 4
– 音声認識	7 8
5. 認識技術の応用	8 6

3.画像処理・認識技術

3.1 デジタル画像の表現

デジタルカメラの仕組み



ref: 千葉憲明著、『カメラの常識のウソ・マコト』、講談社、2004年6月20日第1刷発行、[ISBN 4062574462](#), 森枝卓士著、『デジカメ時代の写真術』、NHK出版、2003年7月10日第1刷発行、[ISBN 4140880740](#), 津軽海渡、木村誠聡著、『図解雑学 デジタルカメラ』、ナツメ社、2002年12月18日発行、[ISBN 4816334092](#)
ref. (2020/4/3): Wikimedia commons: File:Digital camera cut model 1.PNG (https://commons.wikimedia.org/wiki/File:Digital_camera_cut_model_1.PNG) [CC BY-SA 3.0](#)

アナログ画像とデジタル画像の表現

画像とは x 軸と y 軸からなる平面座標系で定義される2変数関数 f または F

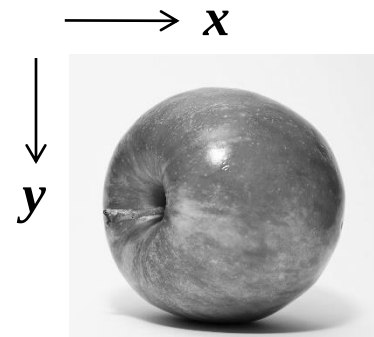
アナログ画像の表現

$$z = f(x, y)$$

x, y, z は0以上の**実数**

(x, y) は平面上の位置座標

z は (x, y) における光の強さ・明るさ



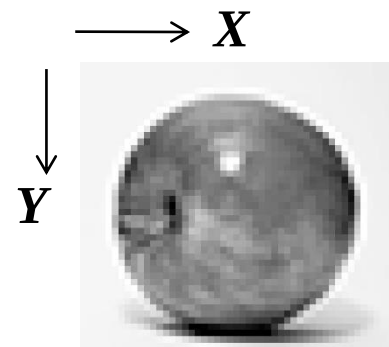
デジタル画像の表現

$$Z = F(X, Y)$$

X, Y, Z は0以上の**整数**

(X, Y) は平面上の格子点

Z は格子点 (X, Y) における光の強さ・明るさ



1つの格子点を画素またはピクセル(pixel)と呼ぶ
色の濃さ(Z の値の大きさ)を輝度または濃淡値と呼ぶ

ref. (2020/4/3): Wikimedia commons:
File:Red Apple.jpg [CC BY 2.0](https://commons.wikimedia.org/wiki/File:Red_Apple.jpg)
https://commons.wikimedia.org/wiki/File:Red_Apple.jpg

デジタル画像は正方形タイルの集まり

横縦方向の刻みが細かい⇒解像度が高い



各ピクセルの階調の刻みが細かい
⇒ビット深度が深い

ref. (2020/4/3): Wikimedia commons:
File:Red Apple.jpg [CC BY 2.0](https://commons.wikimedia.org/wiki/File:Red_Apple.jpg)
https://commons.wikimedia.org/wiki/File:Red_Apple.jpg

解像度の高低による画像の違い

ref. (2020/4/3): Wikimedia commons:
File:Red Apple.jpg CC BY 2.0
https://commons.wikimedia.org/wiki/File:Red_Apple.jpg

25x23



50x45



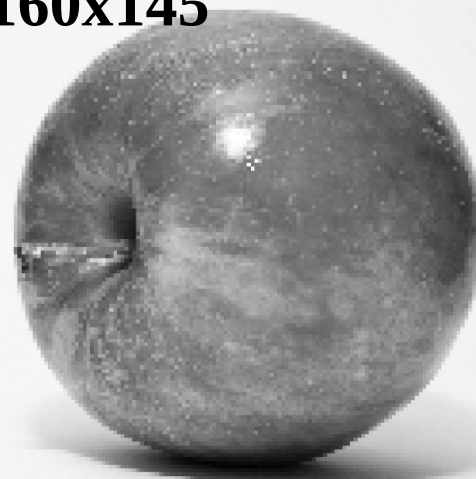
低解像度



84 × 76



160x145

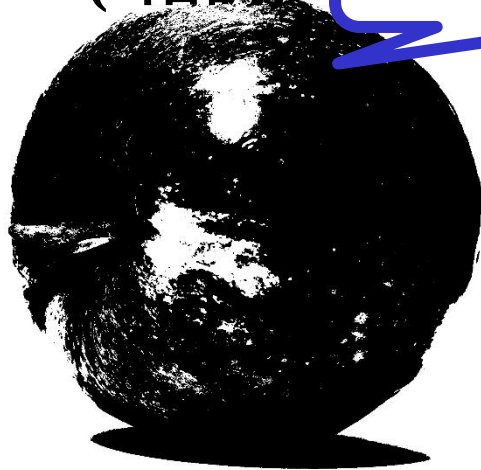


高解像度



ビット深度の深さによる画像の違い

1bit (2階調):

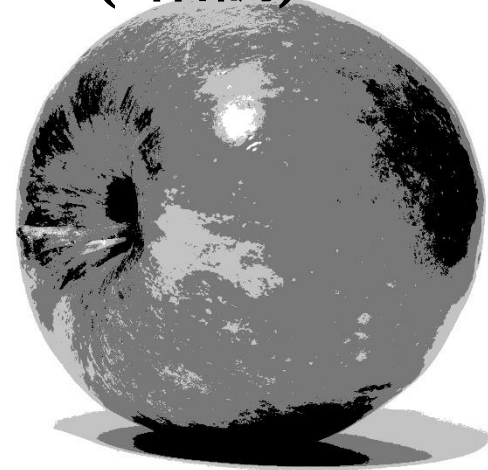


二値画像 (binary image)
とも呼ぶ

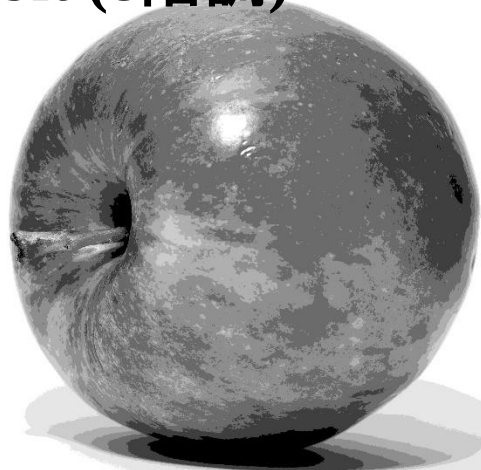
低ビット深度



2bit (4階調)



3bit (8階調)



高ビット深度



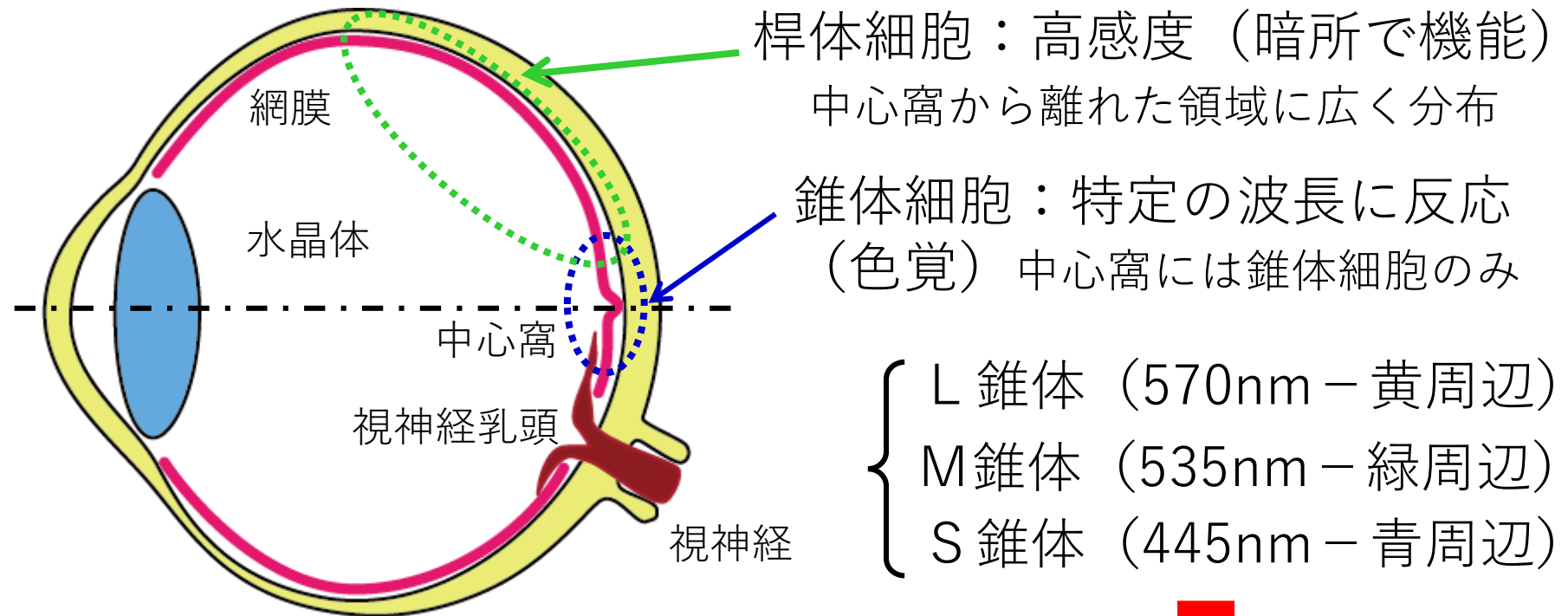
8bit (256階調):



一般的にグレースケール
画像といえばこれ

色情情報の獲得

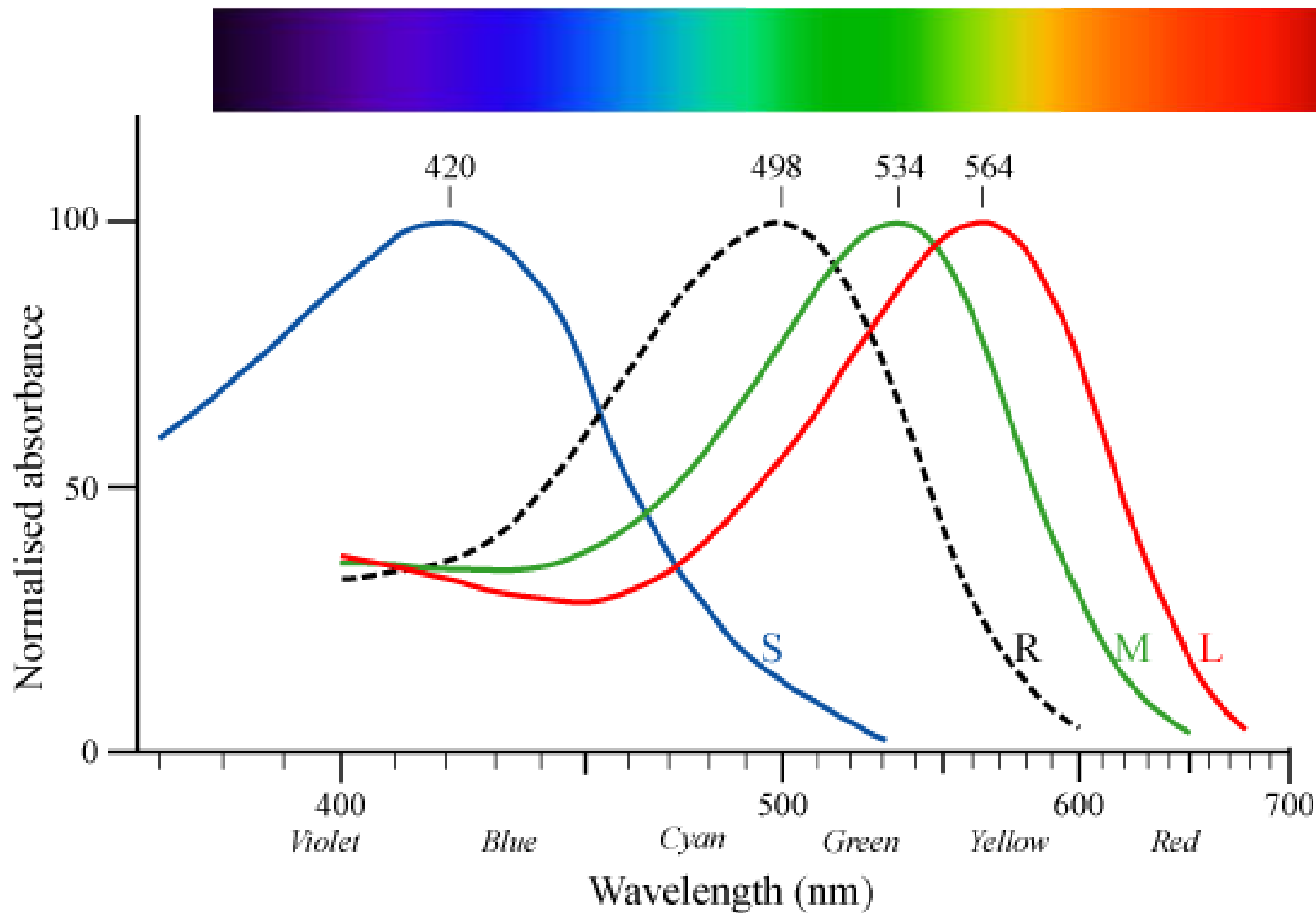
光と色を知覚する細胞



3色型色覚の人が
知覚する色情報は 3 次元

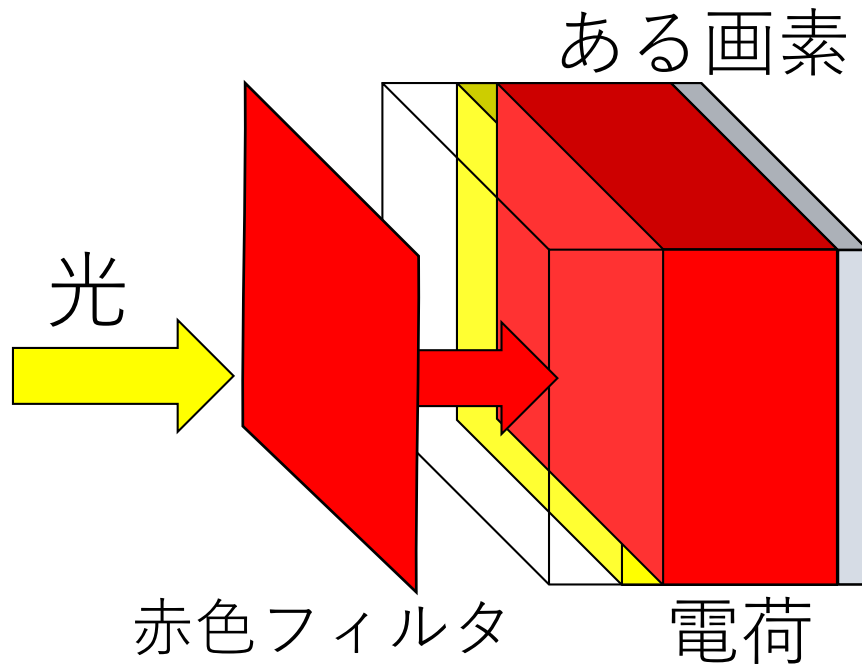
人間の錐体細胞

リテラシーレベル教材
「画像解析」より



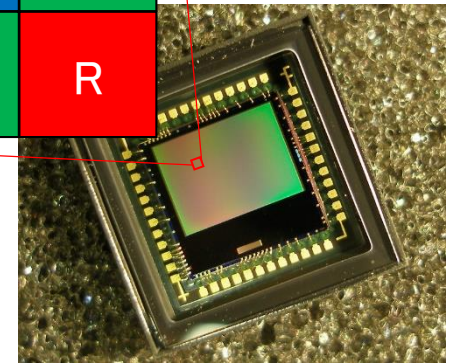
撮像素子による「色」の計測

リテラシーレベル教材
「画像解析」より



B	G	B	G
G	R	G	R
B	G	B	G
G	R	G	R

ベイヤー配列



- 撮像素子は光の強さしか計測できない
→ 撮像素子の各画素の前にカラーフィルタを設置
- 赤色フィルタは赤い波長の光のみを通過
→ 赤色の波長の光量を測定

デジタル画像（24bitフルカラー）の表現

R 1画素あたり8bit
(256階調)



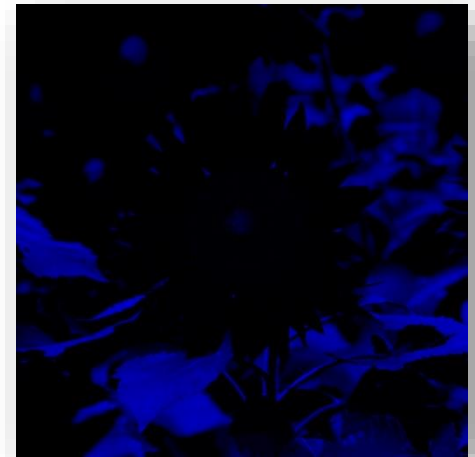
+

G 1画素あたり8bit
(256階調)



+

B 1画素あたり8bit
(256階調)



カラー画像はRGBの
3枚の濃淡画像として
記録されている

=



1画素あたり24bit
(1677万色)
フルカラー

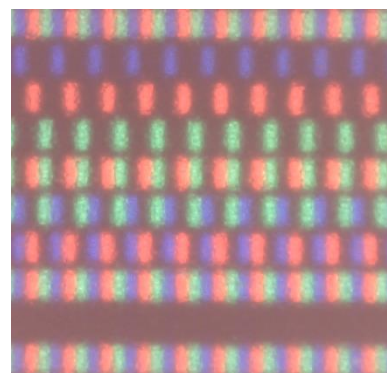
ディスプレイをマイクロカメラで見

リテラシーレベル教材
「画像解析」より

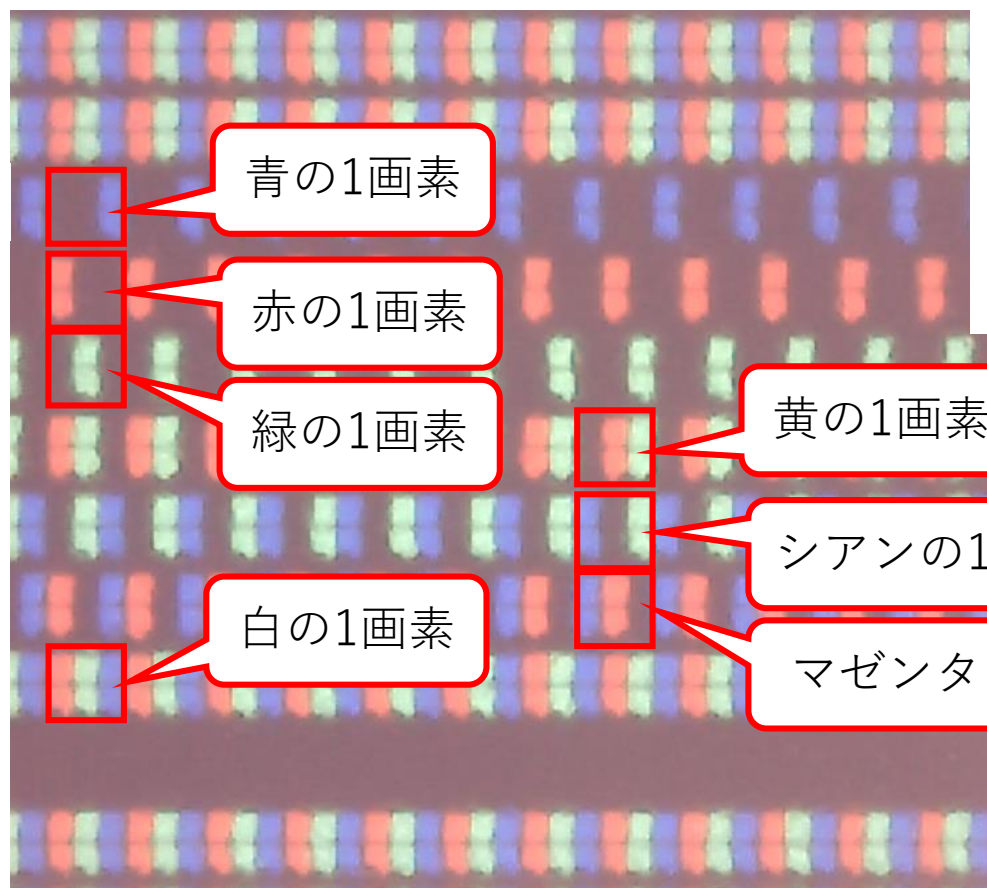
150倍マイクロカメラでディスプレイ表面を撮影



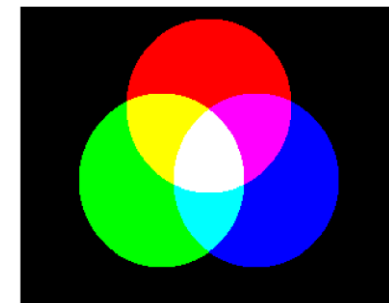
表示して撮影



モニタ (PHILIPS
HDR 600)



Let's note CF-SZ6



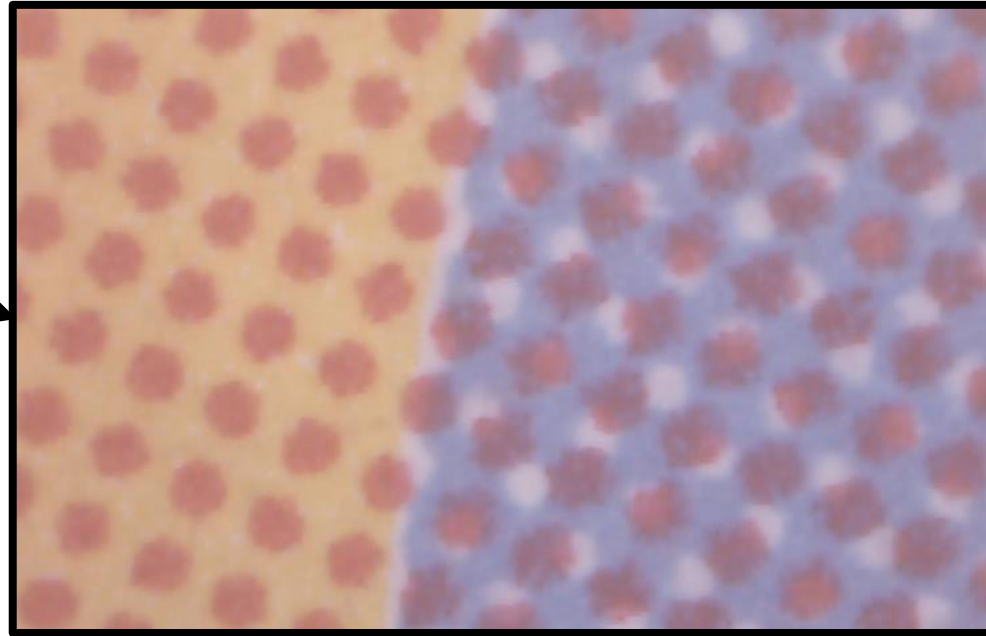
加色法 (RGB)

カラー印刷をマイクロカメラで見て

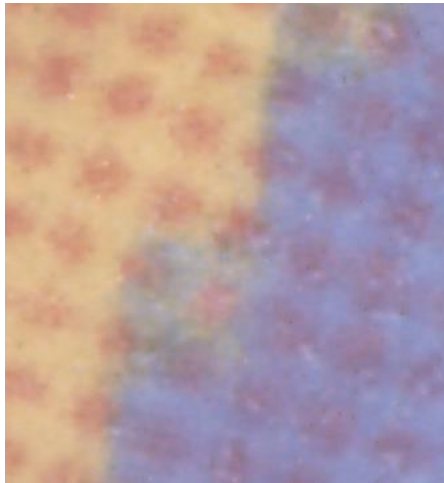
リテラシーレベル教材
「画像解析」より



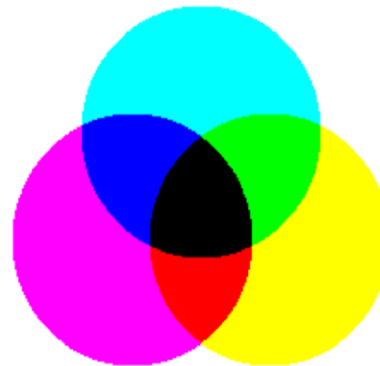
東京大学
THE UNIVERSITY OF TOKYO



印刷会社で印刷されたパンフレット



プリンタで印刷したもの
Fuji Xerox ApeosPort-VI



減色法 (CMYK)

3.画像処理・認識技術

3.2 2次元フィルタリングによる 画像処理

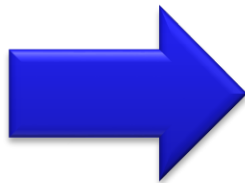
畳み込み演算による画像のフィルタリング

- デジタル画像を平面座標系において座標 (x, y) が決まると、その座標における濃淡値 $z=f(x,y)$ を返す関数と考える
(ただし、 x, y, z は0以上の整数)
- 入力画像 $f(x,y)$ に対し、何らかのフィルタを適用して、出力画像 $g(x,y)$ を生成することを考える

入力画像 $f(x,y)$



ケーススタディ:
画像を
ぼけさせる
フィルタとは?



出力画像 $g(x,y)$



2次元デジタルフィルタにおける畳み込み演算

$h(0,0)$	$h(0,1)$	$h(0,2)$
$h(1,0)$	$h(1,1)$	$h(1,2)$
$h(2,0)$	$h(2,1)$	$h(2,2)$

3×3のカーネル

$$g(n, m) = \sum_i \sum_j h(i, j) f(n + i, m + j)$$

出力画像の明るさを変えたくない場合は、

$$\sum_i \sum_j h(i, j) = 1$$

- 一般的にフィルタのカーネルは奇数×奇数画素の正方行列
- それぞれの画素を中心とし、その画素とその周辺からなる画素集合とカーネルとの行列積を、新しい画像の画素値とする処理を「**畳み込み (convolution)**」と呼ぶ
- **畳み込み**はDeep Learningによる画像処理の基本的な処理 (CNN=**C**onvolut**i**onal Neural Network)
- フィルタによって出力画像が様々に変化

2次元デジタルフィルタの適用

入力画像 $f(x,y)$

$f(0,0)$	$f(0,1)$	$f(0,2)$	$f(0,3)$	$f(0,4)$	$f(0,5)$
$f(1,0)$	$f(1,1)$	$f(1,2)$	$f(1,3)$	$f(1,4)$	$f(1,5)$
$f(2,0)$	$f(2,1)$	$f(2,2)$	$f(2,3)$	$f(2,4)$	$f(2,5)$
$f(3,0)$	$f(3,1)$	$f(3,2)$	$f(3,3)$	$f(3,4)$	$f(3,5)$
$f(4,0)$	$f(4,1)$	$f(4,2)$	$f(4,3)$	$f(4,4)$	$f(4,5)$
$f(5,0)$	$f(5,1)$	$f(5,2)$	$f(5,3)$	$f(5,4)$	$f(5,5)$

出力画像 $g(x,y)$

$g(0,0)$	$g(0,1)$	$g(0,2)$	$g(0,3)$	$g(0,4)$
$g(1,0)$	$g(1,1)$	$g(1,2)$	$g(1,3)$	$g(1,4)$
$g(2,0)$	$g(2,1)$	$g(2,2)$	$g(2,3)$	$g(2,4)$
$g(3,0)$	$g(3,1)$	$g(3,2)$	$g(3,3)$	$g(3,4)$
$g(4,0)$	$g(4,1)$	$g(4,2)$	$g(4,3)$	$g(4,4)$

行列の積

$h(0,0)$	$h(0,1)$	$h(0,2)$
$h(1,0)$	$h(1,1)$	$h(1,2)$
$h(2,0)$	$h(2,1)$	$h(2,2)$

$$g(n,m) = \sum_{i=-1}^1 \sum_{j=-1}^1 h(i,j) f(n+i, m+j)$$

2次元デジタルフィルタの適用

入力画像 $f(x,y)$

$f(0,0)$	$f(0,1)$	$f(0,2)$	$f(0,3)$	$f(0,4)$	$f(0,5)$
$f(1,0)$	$f(1,1)$	$f(1,2)$	$f(1,3)$	$f(1,4)$	$f(1,5)$
$f(2,0)$	$f(2,1)$	$f(2,2)$	$f(2,3)$	$f(2,4)$	$f(2,5)$
$f(3,0)$	$f(3,1)$	$f(3,2)$	$f(3,3)$	$f(3,4)$	$f(3,5)$
$f(4,0)$	$f(4,1)$	$f(4,2)$	$f(4,3)$	$f(4,4)$	$f(4,5)$
$f(5,0)$	$f(5,1)$	$f(5,2)$	$f(5,3)$	$f(5,4)$	$f(5,5)$

出力画像 $g(x,y)$

$g(0,0)$	$g(0,1)$	$g(0,2)$	$g(0,3)$	$g(0,4)$
$g(1,0)$	$g(1,1)$	$g(1,2)$	$g(1,3)$	$g(1,4)$
$g(2,0)$	$g(2,1)$	$g(2,2)$	$g(2,3)$	$g(2,4)$
$g(3,0)$	$g(3,1)$	$g(3,2)$	$g(3,3)$	$g(3,4)$
$g(4,0)$	$g(4,1)$	$g(4,2)$	$g(4,3)$	$g(4,4)$

行列の積

$h(0,0)$	$h(0,1)$	$h(0,2)$
$h(1,0)$	$h(1,1)$	$h(1,2)$
$h(2,0)$	$h(2,1)$	$h(2,2)$

$$g(n,m) = \sum_{i=-1}^1 \sum_{j=-1}^1 h(i,j) f(n+i, m+j)$$

2次元デジタルフィルタの適用

入力画像 $f(x,y)$

$f(0,0)$	$f(0,1)$	$f(0,2)$	$f(0,3)$	$f(0,4)$	$f(0,5)$
$f(1,0)$	$f(1,1)$	$f(1,2)$	$f(1,3)$	$f(1,4)$	$f(1,5)$
$f(2,0)$	$f(2,1)$	$f(2,2)$	$f(2,3)$	$f(2,4)$	$f(2,5)$
$f(3,0)$	$f(3,1)$	$f(3,2)$	$f(3,3)$	$f(3,4)$	$f(3,5)$
$f(4,0)$	$f(4,1)$	$f(4,2)$	$f(4,3)$	$f(4,4)$	$f(4,5)$
$f(5,0)$	$f(5,1)$	$f(5,2)$	$f(5,3)$	$f(5,4)$	$f(5,5)$

出力画像 $g(x,y)$

$g(0,0)$	$g(0,1)$	$g(0,2)$	$g(0,3)$	$g(0,4)$
$g(1,0)$	$g(1,1)$	$g(1,2)$	$g(1,3)$	$g(1,4)$
$g(2,0)$	$g(2,1)$	$g(2,2)$	$g(2,3)$	$g(2,4)$
$g(3,0)$	$g(3,1)$	$g(3,2)$	$g(3,3)$	$g(3,4)$
$g(4,0)$	$g(4,1)$	$g(4,2)$	$g(4,3)$	$g(4,4)$

行列の積

$h(0,0)$	$h(0,1)$	$h(0,2)$
$h(1,0)$	$h(1,1)$	$h(1,2)$
$h(2,0)$	$h(2,1)$	$h(2,2)$

$$g(n,m) = \sum_{i=-1}^1 \sum_{j=-1}^1 h(i,j) f(n+i, m+j)$$

2次元デジタルフィルタの適用

入力画像 $f(x,y)$

$f(0,0)$	$f(0,1)$	$f(0,2)$	$f(0,3)$	$f(0,4)$	$f(0,5)$
$f(1,0)$	$f(1,1)$	$f(1,2)$	$f(1,3)$	$f(1,4)$	$f(1,5)$
$f(2,0)$	$f(2,1)$	$f(2,2)$	$f(2,3)$	$f(2,4)$	$f(2,5)$
$f(3,0)$	$f(3,1)$	$f(3,2)$	$f(3,3)$	$f(3,4)$	$f(3,5)$
$f(4,0)$	$f(4,1)$	$f(4,2)$	$f(4,3)$	$f(4,4)$	$f(4,5)$
$f(5,0)$	$f(5,1)$	$f(5,2)$	$f(5,3)$	$f(5,4)$	$f(5,5)$

出力画像 $g(x,y)$

$g(0,0)$	$g(0,1)$	$g(0,2)$	$g(0,3)$	$g(0,4)$
$g(1,0)$	$g(1,1)$	$g(1,2)$	$g(1,3)$	$g(1,4)$
$g(2,0)$	$g(2,1)$	$g(2,2)$	$g(2,3)$	$g(2,4)$
$g(3,0)$	$g(3,1)$	$g(3,2)$	$g(3,3)$	$g(3,4)$
$g(4,0)$	$g(4,1)$	$g(4,2)$	$g(4,3)$	$g(4,4)$

行列の積

$h(i,j)$ のとる値によって
出力画像が様々に変化

$$g(n,m) = \sum_{i=-1}^1 \sum_{j=-1}^1 h(i,j) f(n+i, m+j)$$

→ h の f に対する畳み込み演算と呼ぶ

平均値フィルタ (Mean filter)

画像をぼけさせる
フィルタ

入力画像(320x480)

3x3 平均値フィルタ

$1/9$	$1/9$	$1/9$
$1/9$	$1/9$	$1/9$
$1/9$	$1/9$	$1/9$



5x5 平均値フィルタ

$1/25$	$1/25$	$1/25$	$1/25$	$1/25$
$1/25$	$1/25$	$1/25$	$1/25$	$1/25$
$1/25$	$1/25$	$1/25$	$1/25$	$1/25$
$1/25$	$1/25$	$1/25$	$1/25$	$1/25$
$1/25$	$1/25$	$1/25$	$1/25$	$1/25$



鮮鋭化フィルタ

輪郭を際立たせ、鮮鋭にするフィルタ

- その画素と周辺の画素の値が違うとき、その画素の値を大きくする
- その画素と周辺の画素の値が同じ時、その画素の元の値をそのまま使う

3x3鮮鋭化フィルタ

$k=5$ とすると

下図のようになる

$-k/8$	$-k/8$	$-k/8$
$-k/8$	$1+k$	$-k/8$
$-k/8$	$-k/8$	$-k/8$

入力画像 (320x480)



エッジ抽出：ラプラシアンフィルタ

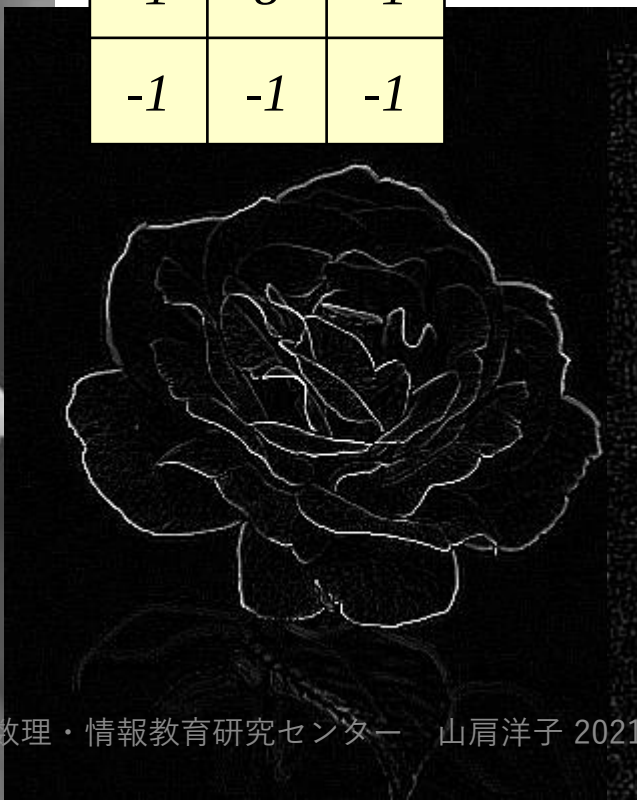
その画素の値が周辺の画素と違うほど大きな値とすることによって、
周辺との輝度が大きく変化する輪郭線を取り出す

入力画像(320x480)



3x3 ラプラシアンフィルタ

-1	-1	-1
-1	8	-1
-1	-1	-1



5x5 ラプラシアンフィルタ

-1	-3	-4	-3	-1
-3	0	6	0	-3
-4	6	20	6	-4
-3	0	6	0	-3
-1	-3	-4	-3	-1



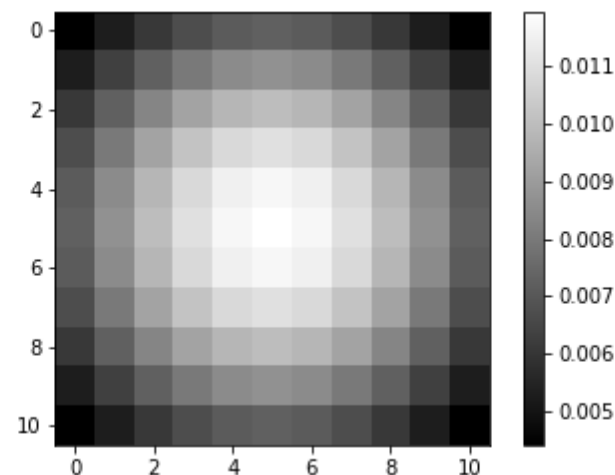
ガウシアンフィルタ

中心が最も値が大きく、
中心から離れるほど値が小さくなる
x軸方向、y軸方向それぞれの値の変化は
ガウス関数に従う

$$h(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

カメラのレンズによるボケを
幾何学的に再現したモデルと考えられる

11x11のガウシアンフィルタ



サイズ3×3、 $\sigma=2$ のフィルタ

0.102	0.115	0.102
0.115	0.131	0.115
0.102	0.115	0.102

サイズ5×5、 $\sigma=5$ のフィルタ

0.0369	0.0392	0.0400	0.0392	0.0369
0.0392	0.0416	0.0424	0.0416	0.0392
0.0400	0.0424	0.0433	0.0424	0.0400
0.0392	0.0416	0.0424	0.0416	0.0392
0.0369	0.0392	0.0400	0.0392	0.0369

ガウシアンフィルタ

- その画素の値の影響を最も強く受け、そこから離れた位置にある画素ほど影響を受けなくなる

入力画像(320x480)



ガウシアンフィルタ
サイズ: 5x5
 $\sigma=5$



ガウシアンフィルタ
サイズ: 11x11
 $\sigma=5$



3.画像処理・認識技術

3.3 深層学習による画像認識

深層学習による画像認識の幕開け

- クラウドソーシングによる大規模データセット：ImageNet
- ImageNet Large-Scale Visual Recognition Challenge (ILSVRC)
 - 1000クラス、学習120万枚、検証5万枚、テスト10万枚
 - マルチラベル：1枚の画像に複数ラベル+信頼度
- 2011年のILSVRCで優勝したモデルのエラー率は26%
→ 2012年にDeep Learningを使ったモデルが登場→15%に激減！

Easiest classes

red fox (100) hen-of-the-woods (100) ibex (100) goldfinch (100) flat-coated retriever (100)



tiger (100)



hamster (100)



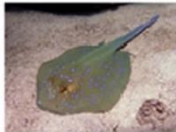
porcupine (100)



stingray (100)



Blenheim spaniel (100)



Hardest classes

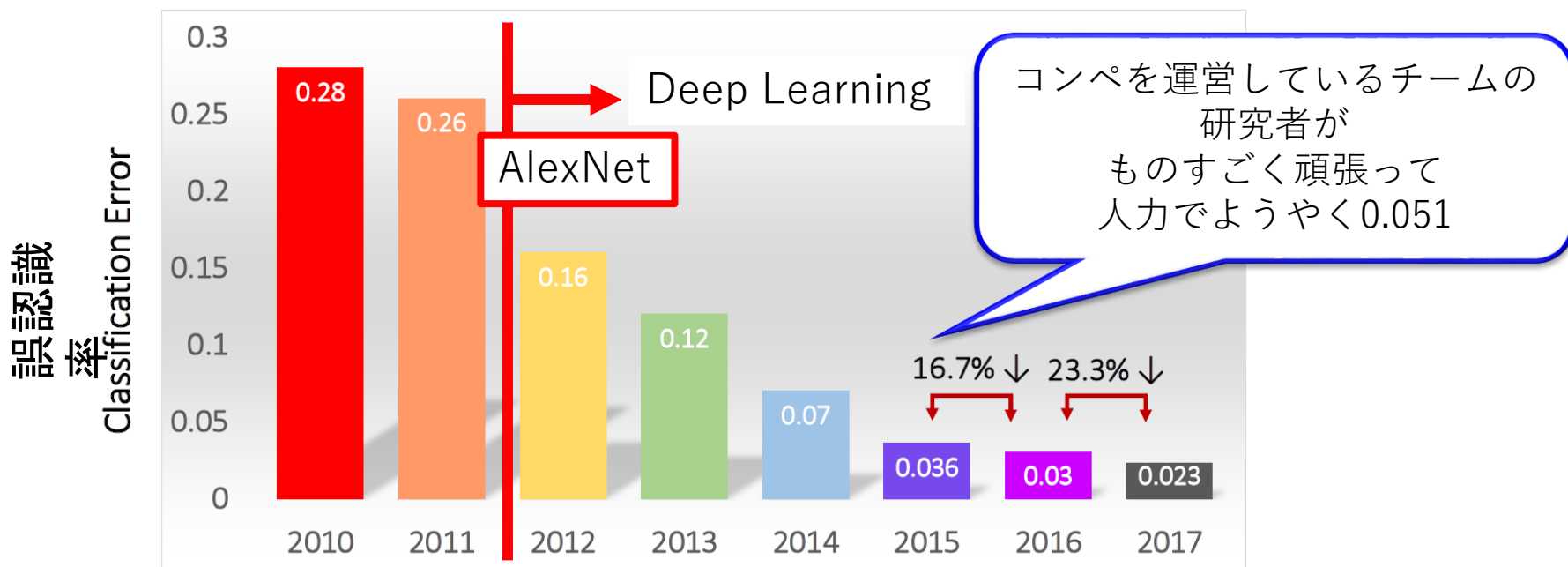
muzzle (71) hatchet (68) water bottle (68) velvet (68) loupe (66)



ref. Russakovsky, O., Deng, J., Su, H. *et al.* ImageNet Large Scale Visual Recognition Challenge. *Int J Comput Vis* 115, 211–252 (2015).

物体認識精度は人間を超えた!?

- ILSVRC2012で、トロント大学Geoffrey Hinton教授率いるグループが初めて多層ニューラルネットワークによる画像認識モデルを提案
- 主著者の名前をもじって**AlexNet**と呼ばれる
- 翌年から上位チームはすべてDeep Learningを採用
- 2015年について人間の精度を超えた



Excerpted from [ImageNet Large Scale Visual Recognition Challenge \(ILSVRC\) 2017 Overview](#)

Andrei Karpathy, "What I learned from competing against a ConvNet on ImageNet", Sep 2, 2014, <http://karpathy.github.io/2014/09/02/what-i-learned-from-competing-against-a-convnet-on-imagenet/>

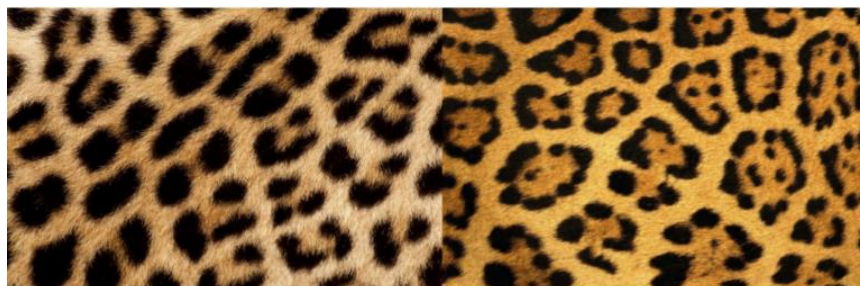
どんな問題を解いたのか？ ネコ科大型動物編

リテラシーレベル教材
「画像解析」より

これは何ですか？



ヒョウ？ ジャガー？



ヒョウ

ジャガー

模様から、これはヒョウです！

<http://image-net.org/challenges/LSVRC/2012/browse-synsets>

どんな問題を解いたのか？犬編

リテラシーレベル教材
「画像解析」より

これは何ですか？

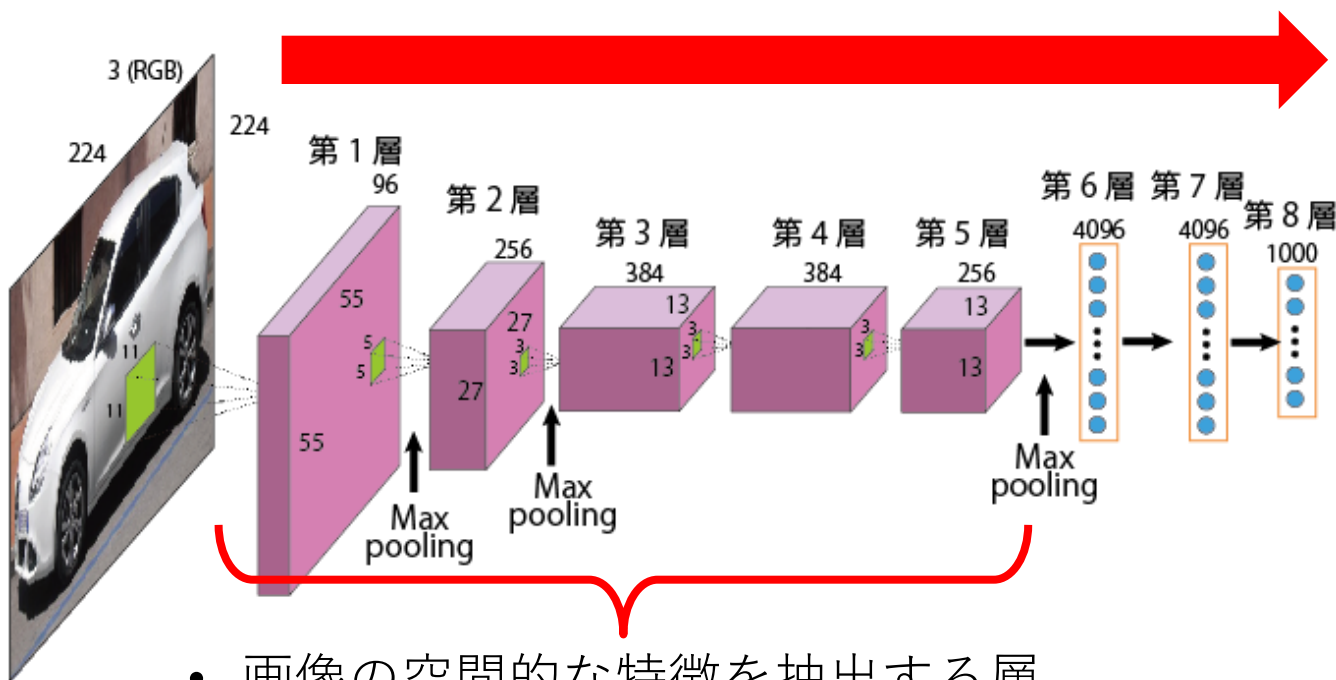


ダルメシアン、ニューファン
ドランドドッグ、シベリアン
ハスキー、ハンティングドッ
グ、ジャーマンシェパード、
イングリッシュシェパード
フレンチブルドッグ、マル
チーズ、グレイトマウンテン
ドッグ、パグ、テリア、
シェットランドシェパード

<http://image-net.org/challenges/LSVRC/2012/browse-synsets>

畳み込みニューラルネットワーク (CNN; Convolutional Neural Network)

- 代表的な物体識別モデル
- 実際には様々な実装がある（下図はAlexNet）



最終層にsoftmax
を適用した結果

0.00 goldfish
0.00 great white shark
0.00 tiger shark
0.00 hammerhead
0.01 electric ray
...
0.89 car
...
0.00 stinkhorn
0.00 earthstar
0.00 hen-of-the-woods
0.02 bolete
0.00 ear, spike
0.00 toilet tissue

- 画像の空間的な特徴を抽出する層
- 途中の層ではどうなっているか？

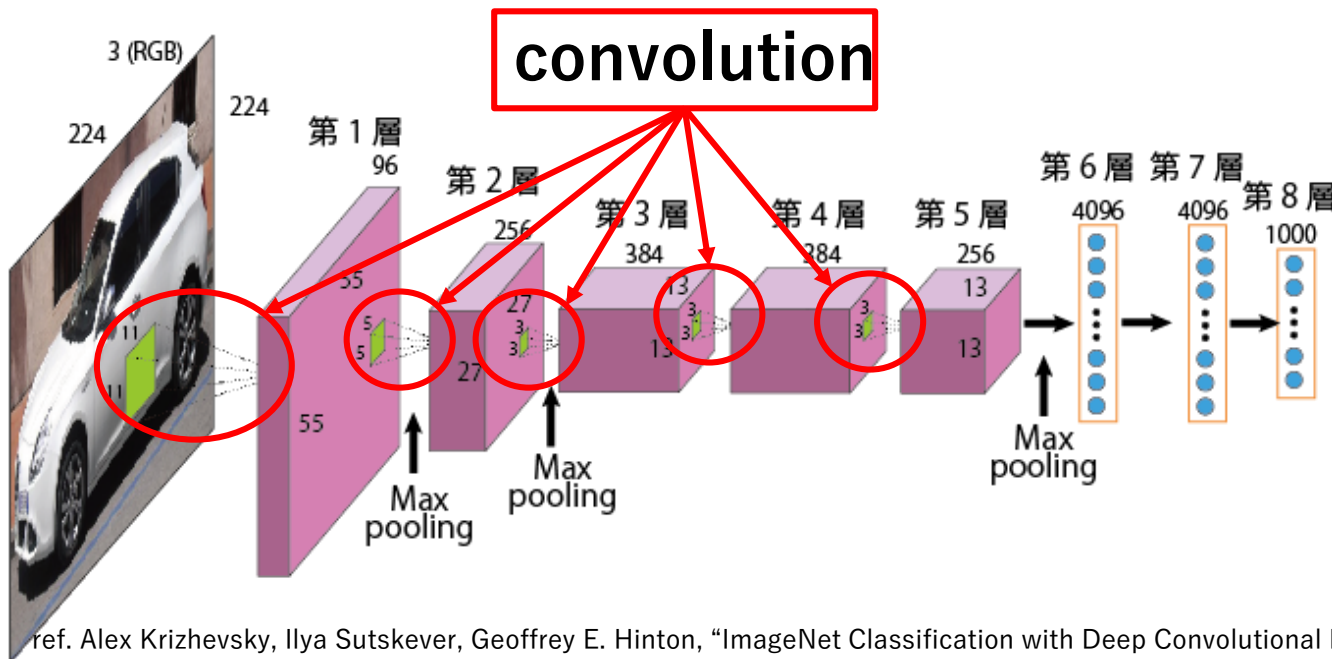
1000クラスのうちcarだけが高く、
残りがほぼ0のようなベクトルが出力

ref. Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", NIPS2012

ref. (2020/4/6): Wikimedia commons: File:" 10 Alfa Romeo Giulietta white Derivate cut.JPG [CC0](https://commons.wikimedia.org/wiki/File:%22_10_Alfa_Romeo_Giulietta_white_Derivate_cut.JPG)
https://commons.wikimedia.org/wiki/File:%22_10_Alfa_Romeo_Giulietta_white_Derivate_cut.JPG

畳み込みニューラルネットワークにおける 畳み込み演算

- 第1層から第5層までは、2次元デジタルフィルタで説明した畳み込み演算 (convolution) を行っている
- ただし、2次元デジタルフィルタではフィルタは人がデザインしていたのに対し、CNNではデータから学習により取得する



0.00 goldfish
0.00 great white shark
0.00 tiger shark
0.00 hammerhead
0.01 electric ray
...
0.89 car
...
0.00 stinkhorn
0.00 earthstar
0.00 hen-of-the-woods
0.02 bolete
0.00 ear, spike
0.00 toilet tissue

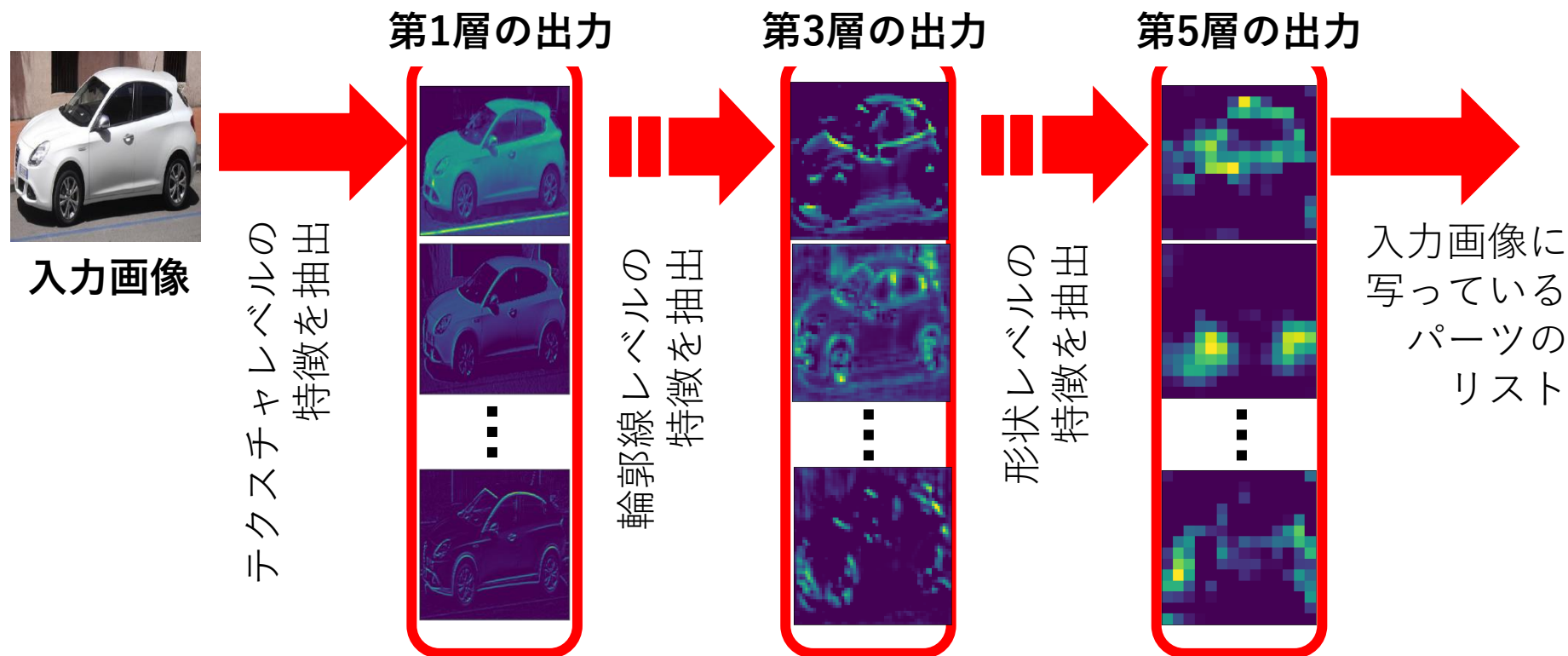
ref. Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", NIPS2012

ref. (2020/4/6): Wikimedia commons: File:" 10 Alfa Romeo Giulietta white Derivate cut.JPG [CC0](#)

https://commons.wikimedia.org/wiki/Category:CC-Zero?uselang=ja#/media/File:%22_10_Alfa_Romeo_Giulietta_white_Derivate_cut.JPG

CNNにおける画像のフィルタリング

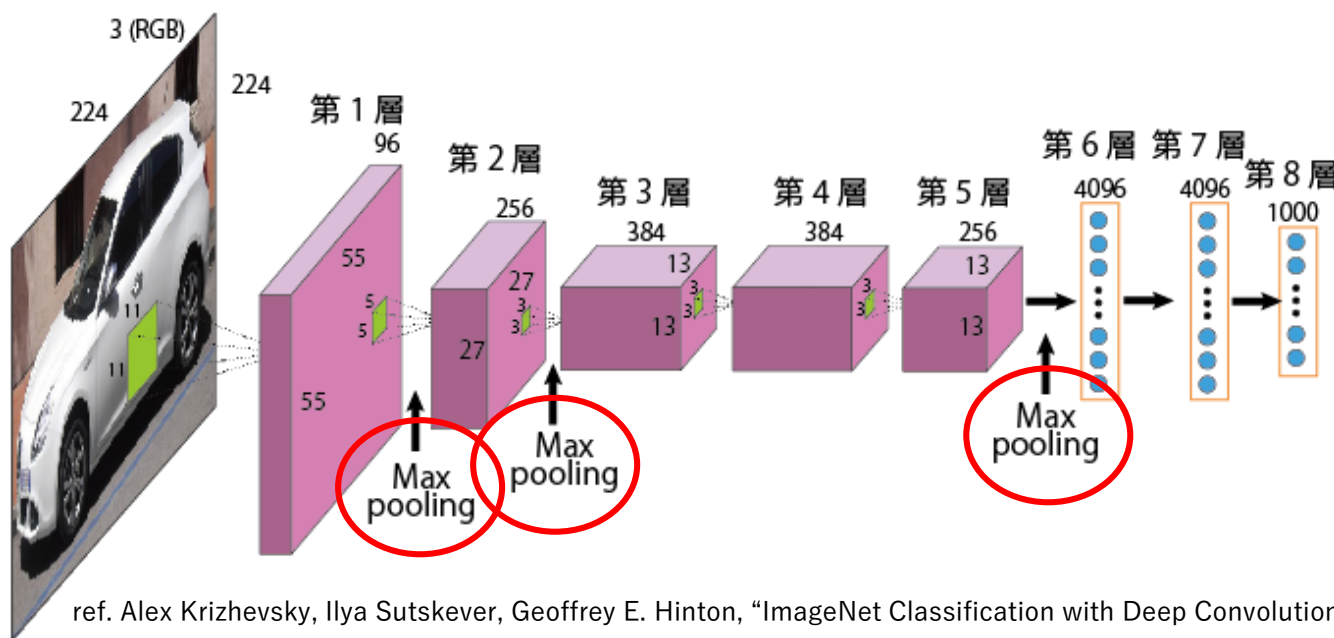
- ・ 層を経るごとににより具体的な形状の特徴を取り出していく
- ・ 第5層になると、「入力画像にどんなパーツが写りこんでいるか（実際には尤度分布）」がわかってくる
- ・ 「入力画像に写っているパーツのセット」が「他のクラスに比べ、車にありがちなパーツのセット」であるならば「車」と判別



モデルはKeras VGG16 pretrainedを使用

畳み込みニューラルネットワークにおけるプーリング (pooling)

- 画面を小さく区切り、各区間ごとに画素をまとめて1つの値にする
 - データのサイズを小さくする役割
 - 物体が写っている位置や角度の違いに頑健にする役割
- 最大値を取る場合 max pooling、平均値をとる場合 average poolingと呼ぶ



0.00 goldfish
0.00 great white shark
0.00 tiger shark
0.00 hammerhead
0.01 electric ray
...
0.89 car
...
0.00 stinkhorn
0.00 earthstar
0.00 hen-of-the-woods
0.02 bolete
0.00 ear, spike
0.00 toilet tissue

ref. Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", NIPS2012

ref. (2020/4/6): Wikimedia commons: File:" 10 Alfa Romeo Giulietta white Derivate cut.JPG [CC0](https://commons.wikimedia.org/wiki/File:%22_10_Alfa_Romeo_Giulietta_white_Derivate_cut.JPG)

https://commons.wikimedia.org/wiki/Category:CC-Zero?uselang=ja#/media/File:%22_10_Alfa_Romeo_Giulietta_white_Derivate_cut.JPG

深層学習はそれまでの画像処理と何が違ったのか？

- AlexNet (2012)登場より前の画像認識では、CNNにおける第1~2層の出力に相当する情報を使って認識していた
→ Deep Networkと対比してShallow networkと呼ばれる
- 2000年前後にいくつかの技術革新
 - **数学的解法**：勾配消失問題（層を深くすると学習が進まなくなる現象）に対する効率的な解決法の提案（1990年代後半）
 - **GPGPUの発展**：コンピュータグラフィックスの描画に用いられていたGPUをベクトル計算機とみなして気象や地震シミュレーション等、数値計算に利用（2006年NVIDIAがCUDA提供開始）
 - **Big Data時代の到来**：
Webで画像やテキストなどが大量に収集できるようになり、モデルの学習に使えるデータが爆発的に増加
- 様々な画像処理タスクに対する学習データセットが公開
 - 学習データを使わない自己教師あり学習（self-supervised learning）の研究も進められている