

機械学習 ニューラルネットワーク(2)

管理工学科
篠沢佳久

資料の内容

- ニューラルネットワーク(2)
- 畳み込みニューラルネットワーク
 - 空間フィルタリング処理(1年生の復習です)
 - 畳み込み層, プーリング層
- 早慶戦により一回休講になってしまったため, 講義できなかった内容です.
- 資料のみ掲載しておきます.

畳み込みニューラルネットワーク

空間フィルタリング処理(情報学基礎の復習)

畳み込み層

プーリング層

空間フィルタリング処理①

- 各画素について、その画素周辺の $N \times N$ 画素の小領域と、 $N \times N$ の空間フィルタとの積和を行なう
- 入力画像を f 、空間フィルタを h とした場合、下記の式に基づいて変換後の画素値 g を求める

$$g(i, j) = \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} f(i+k, j+l)h(k, l)$$

空間フィルタリング処理②

3×3の空間フィルタ h

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

$$40 \times 0 + 60 \times (-1) + 40 \times 0 + 80 \times (-1) + 110 \times 5 + 80 \times (-1) + 100 \times 0 + 100 \times (-1) + 100 \times 0 = 230$$

積和

変換する画素

	40	60	40
	80	110	80
	100	100	100

入力画像

周辺の3×3の小領域 f

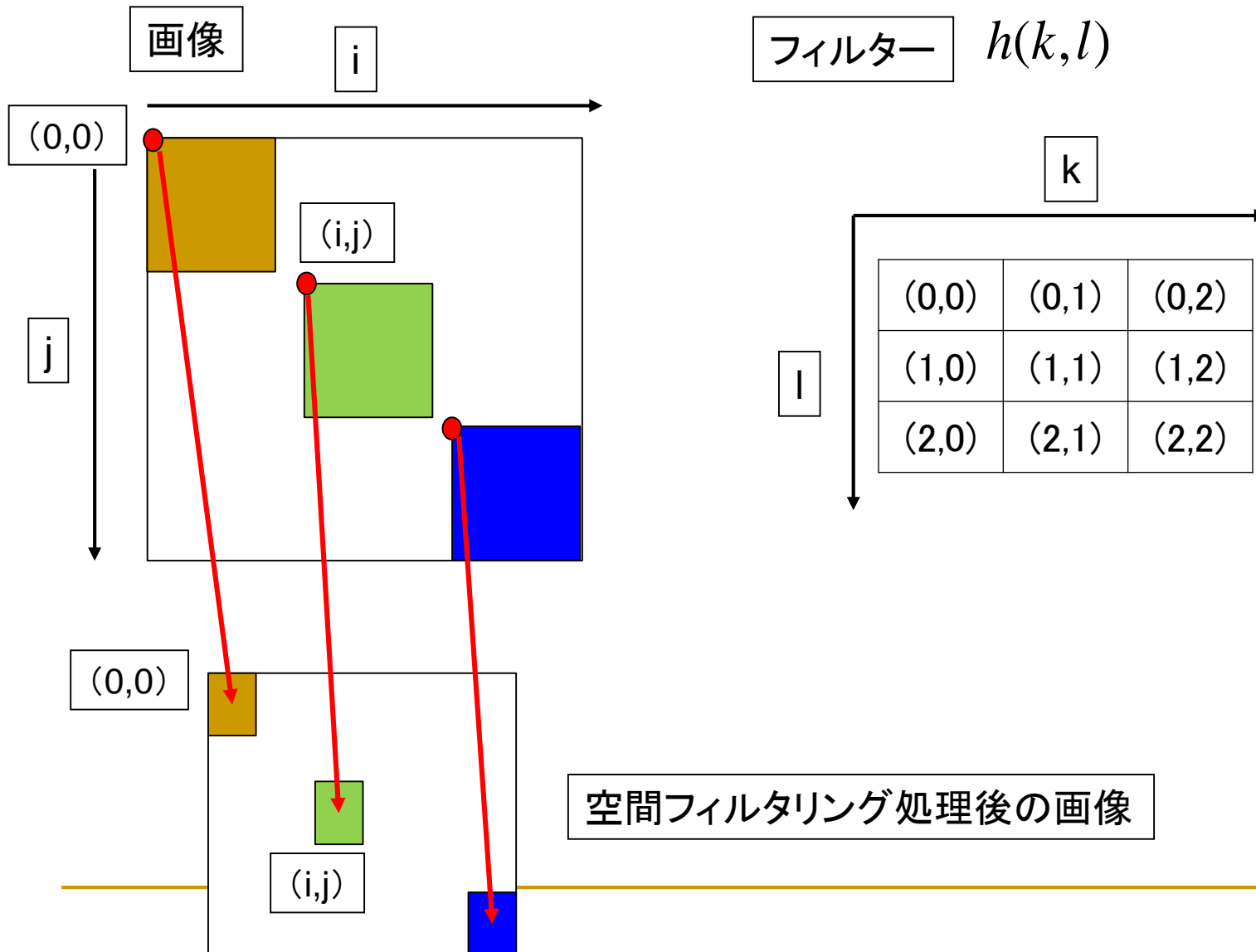
変換後の画素値 g

		230	

出力画像
 (i, j)

以上の処理を全ての画素で行なう

空間フィルタリング処理③



空間フィルタリング処理③

入力画像 $f(x,y)$

2	4	1	3	5
3	2	6	2	8
1	0	3	4	2
6	2	1	7	5
5	3	2	5	6

平滑化フィルタ h

$$h(k,l) = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

出力画像 $g(x,y)$

$$(2+4+1+3+2+6+1+0+3)/9$$

2.444444	2.777778	3.777778
2.666667	3	4.222222
2.555556	3	3.888889

$$(3+4+2+1+7+5+2+5+6)/9$$

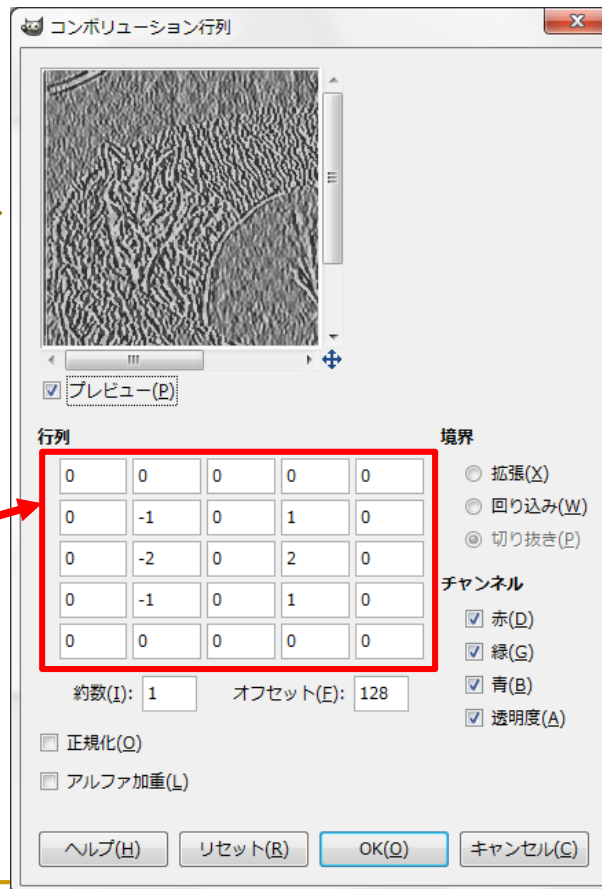
元の画像よりも小さくなってしましますが、同じ大きさにすることも可能です(パディング)

空間フィルタリング処理④

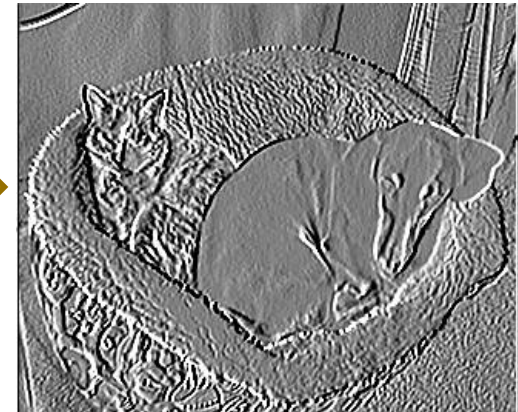
元画像



GIMPの場合



フィルタリング後の画像



フィルタを指定する

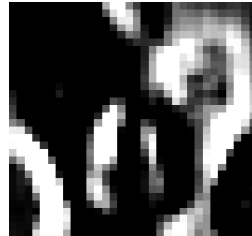
$$h(k,l) = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

空間フィルタリング処理⑤

元画像

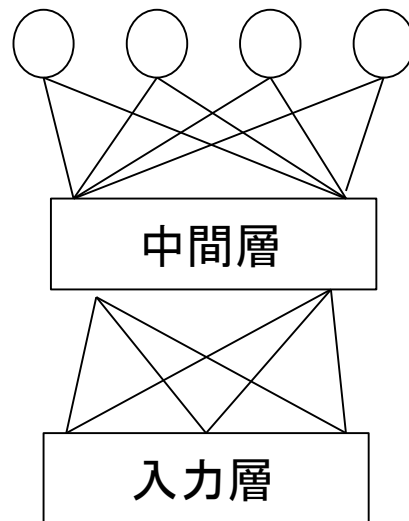
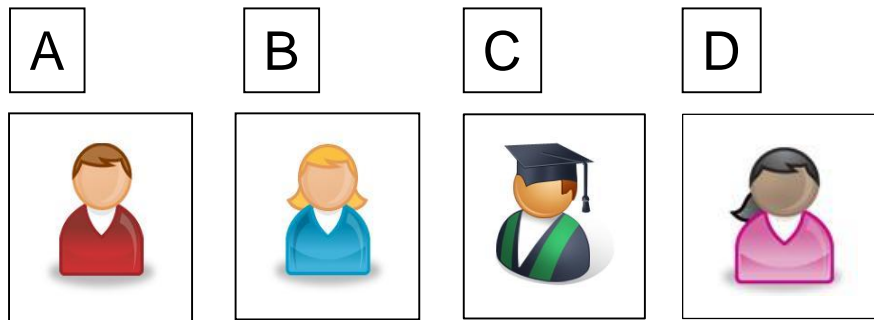


ガボールフィルター



フィルタリング後の画像

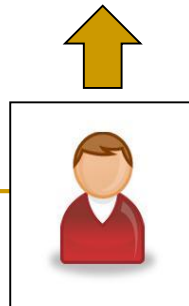
ニューラルネットワークでの顔画像認識



ニューロン数

出力層 → 4個
中間層 → 任意
入力層 → 顔画像の特徴数

入力画像



特徴として何を入力すべきか

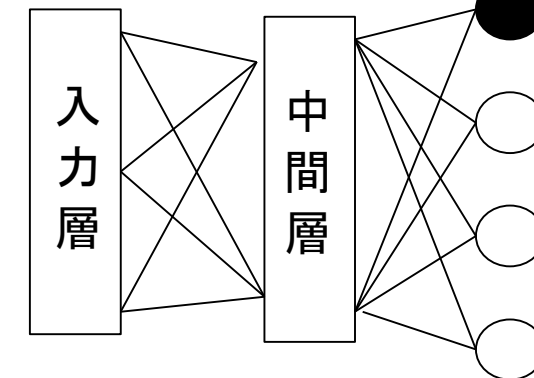
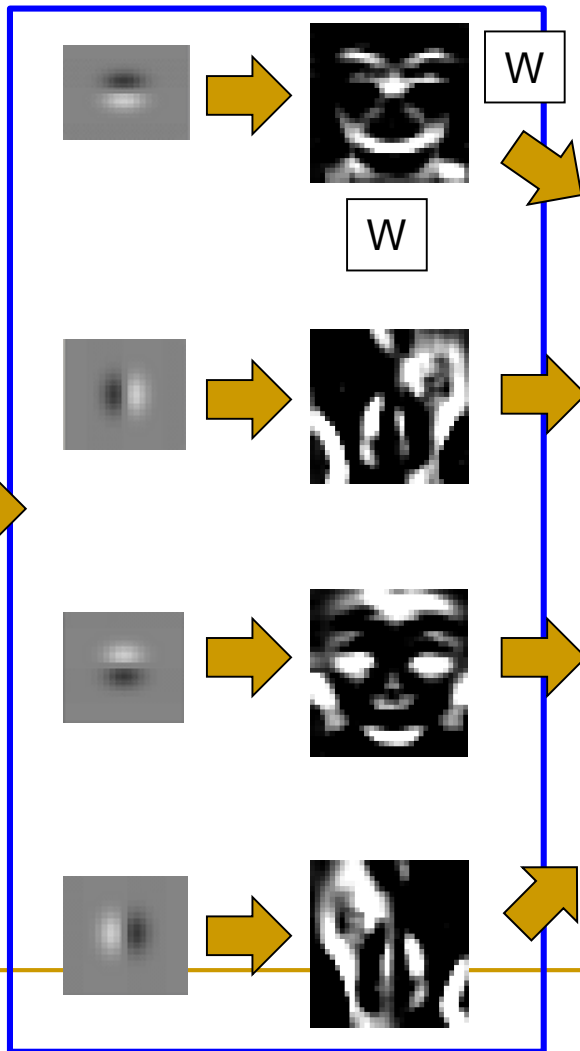
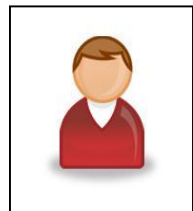
ニューラルネットワークでの顔画像認識

空間フィルタリング処理

フィルターは人手で設定

フィルターも結合係数として学習は可能か

入力画像

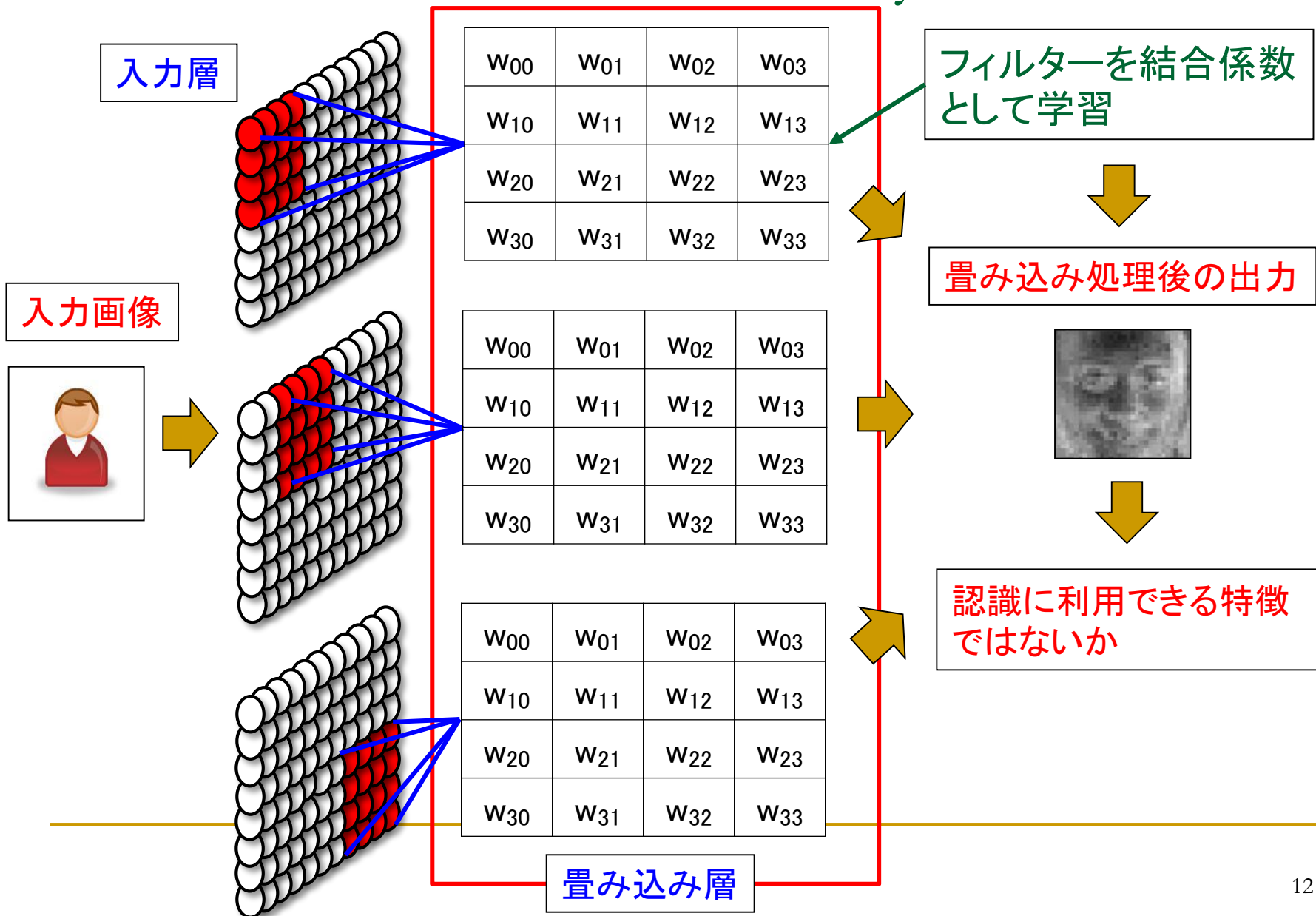


認識結果
「A」さん

入力層のニューロン数
($W \times W \times 4$) 個

特徴抽出処理

畳み込み層 (Convolution Layer)



畳み込み層 (Convolution Layer)

入力層

入力画像

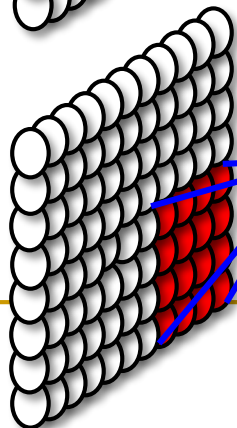
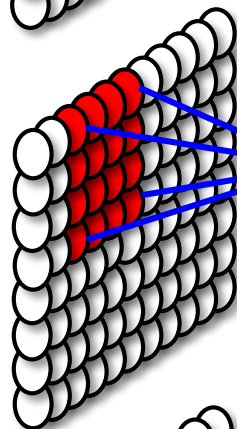
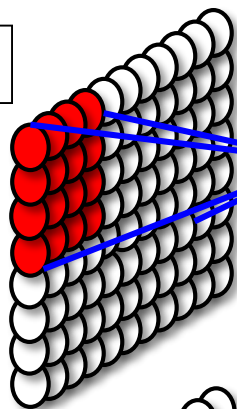
Y

X

$$f(x, y)$$

$$x = 0, 1, \dots, X - 1$$

$$y = 0, 1, \dots, Y - 1$$



W_{00}	W_{01}	W_{02}	W_{03}
W_{10}	W_{11}	W_{12}	W_{13}
W_{20}	W_{21}	W_{22}	W_{23}
W_{30}	W_{31}	W_{32}	W_{33}

W_{00}	W_{01}	W_{02}	W_{03}
W_{10}	W_{11}	W_{12}	W_{13}
W_{20}	W_{21}	W_{22}	W_{23}
W_{30}	W_{31}	W_{32}	W_{33}

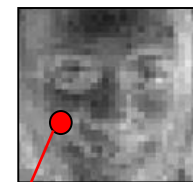
W_{00}	W_{01}	W_{02}	W_{03}
W_{10}	W_{11}	W_{12}	W_{13}
W_{20}	W_{21}	W_{22}	W_{23}
W_{30}	W_{31}	W_{32}	W_{33}

フィルター (結合係数)

$$w(p, q)$$

$$p = 0, 1, \dots, N - 1$$

$$q = 0, 1, \dots, N - 1$$



$$g(x, y) = \sum_{p=0}^{N-1} \sum_{q=0}^{N-1} f(x+p, y+q) w(p, q)$$

畳み込み層

パディング

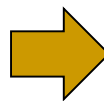
入力画像 $f(x,y)$

2	4	1	3	5
3	2	6	2	8
1	0	3	4	2
6	2	1	7	5
5	3	2	5	6

$$h(k,l) = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

出力画像 $g(x,y)$

2.444	2.778	3.778
2.667	3.000	4.222
2.556	3.000	3.889

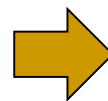


画像の大きさが小さくなる

周囲を0(ゼロパディング)

0	0	0	0	0	0	0
0	2	4	1	3	5	0
0	3	2	6	2	8	0
0	1	0	3	4	2	0
0	6	2	1	7	5	0
0	5	3	2	5	6	0
0	0	0	0	0	0	0

$$(0+0+0+0+2+4+0+3+2)/9$$



1.222	2.000	2.000	2.778	2.000
1.333	2.444	2.778	3.778	2.667
1.556	2.667	3.000	4.222	3.111
1.889	2.556	3.000	3.889	3.222
1.778	2.111	2.222	2.889	2.556
0.889	1.111	1.111	1.444	1.222

画像の大きさは変わらない

ストライド①

ストライドが1の場合

2	4	1	3	5	2	1
3	2	6	2	8	3	4
1	0	3	4	2	1	2
6	2	1	7	5	2	4
5	3	2	5	6	1	7
3	4	1	6	7	0	3
1	4	0	8	5	2	8

入力画像 $f(x,y)$

$$h(k,l) = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$



2.444	2.778	3.778	3.333	3.111
2.667	3.000	4.222	3.778	3.444
2.556	3.000	3.889	3.667	3.333
3.000	3.444	4.444	4.333	3.889
2.556	3.667	4.444	4.444	4.333

出力画像 $g(x,y)$

ストライド②

ストライドが1の場合

2	4	1	3	5	2	1
3	2	6	2	8	3	4
1	0	3	4	2	1	2
6	2	1	7	5	2	4
5	3	2	5	6	1	7
3	4	1	6	7	0	3
1	4	0	8	5	2	8

入力画像 $f(x,y)$

$$h(k,l) = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$



2.444	2.778	3.778	3.333	3.111
2.667	3.000	4.222	3.778	3.444
2.556	3.000	3.889	3.667	3.333
3.000	3.444	4.444	4.333	3.889
2.556	3.667	4.444	4.444	4.333

出力画像 $g(x,y)$

ストライド③

ストライドが2の場合

2	4	1	3	5	2	1
3	2	6	2	8	3	4
1	0	3	4	2	1	2
6	2	1	7	5	2	4
5	3	2	5	6	1	7
3	4	1	6	7	0	3
1	4	0	8	5	2	8

入力画像 $f(x,y)$

$$h(k,l) = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$



2.444	3.778	3.111
2.556	3.889	3.333
2.556	4.444	4.333

出力画像 $g(x,y)$

ストライド④

ストライドが2の場合

2	4	1	3	5	2	1
3	2	6	2	8	3	4
1	0	3	4	2	1	2
6	2	1	7	5	2	4
5	3	2	5	6	1	7
3	4	1	6	7	0	3
1	4	0	8	5	2	8

入力画像 $f(x,y)$

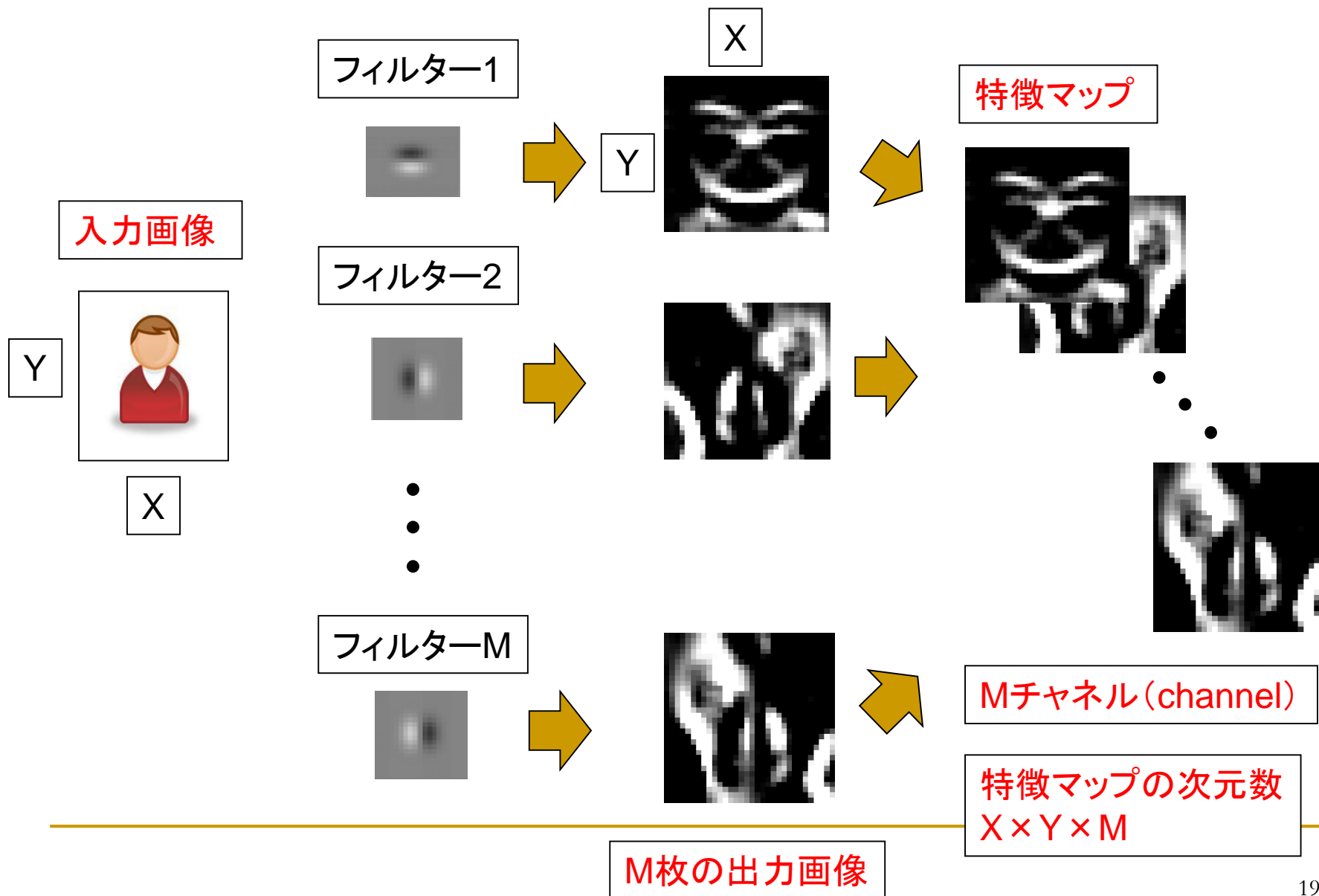
$$h(k,l) = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$



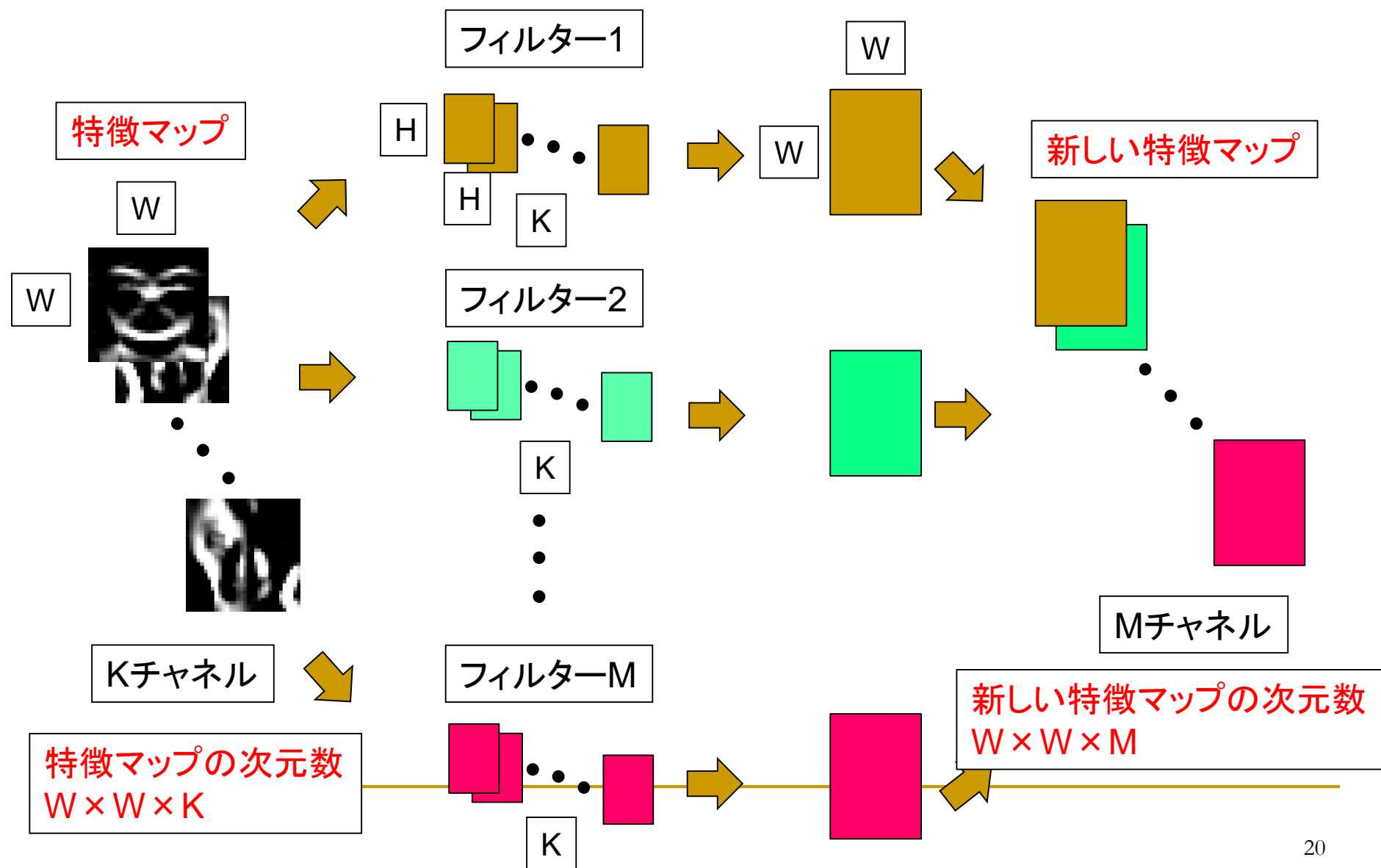
2.444	3.778	3.111
2.556	3.889	3.333
2.556	4.444	4.333

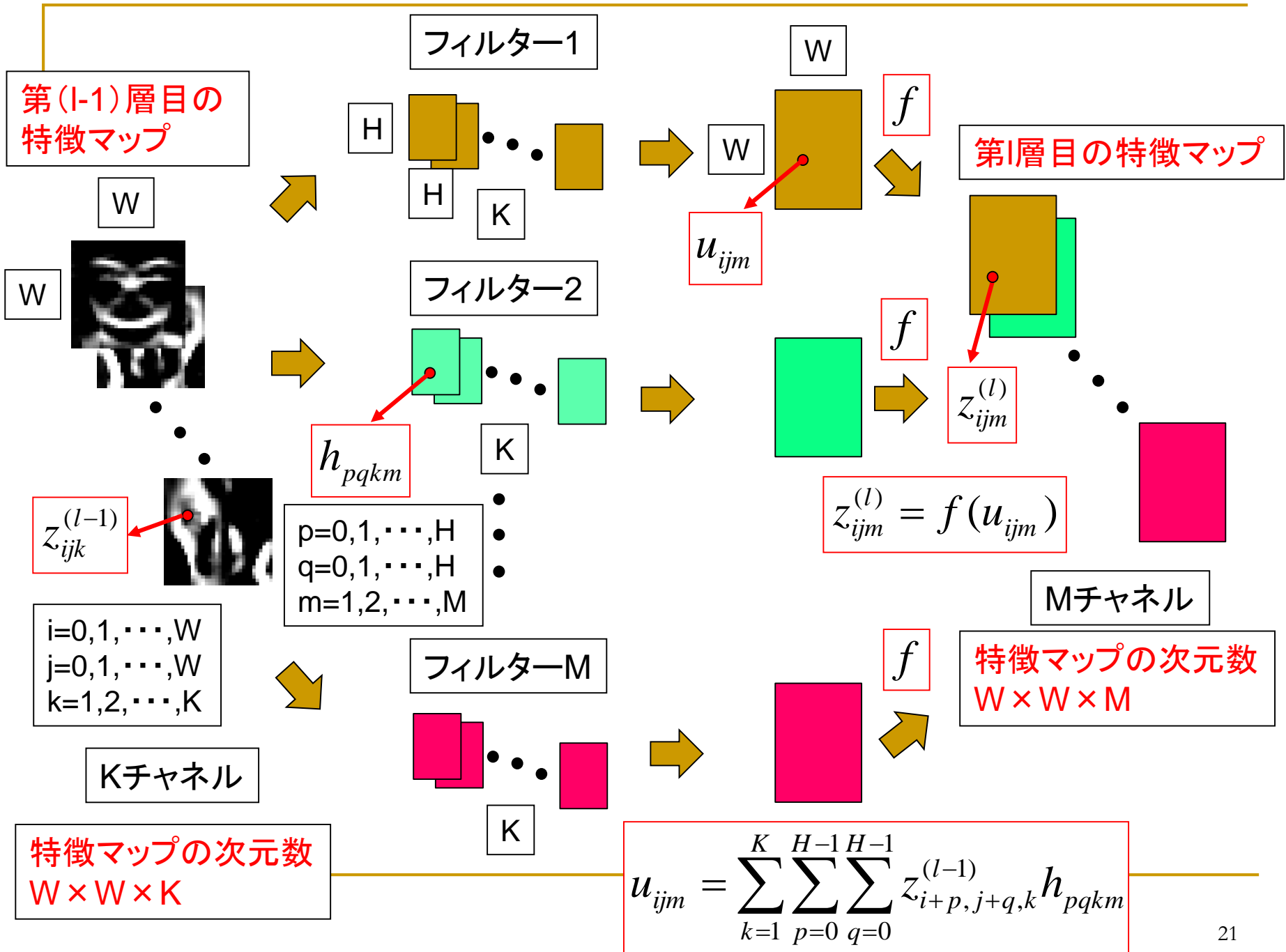
出力画像 $g(x,y)$

特徴マップ



特徴マップの畳み込み処理①





特徴マップの畳み込み処理②

$$u_{ijm} = \sum_{k=1}^K \sum_{p=0}^{H-1} \sum_{q=0}^{H-1} z_{i+p,j+q,k}^{(l-1)} h_{pqkm}$$



$$u_{ijm} = \sum_{k=1}^K \sum_{p=0}^{H-1} \sum_{q=0}^{H-1} z_{i+p,j+q,k}^{(l-1)} h_{pqkm} + b_{ijm}$$

閾値

第1層の特徴マップ

$$z_{ijm}^{(l)} = f(u_{ijm})$$

活性化関数

畳み込みニューラルネットワーク

LeNet(Y.LeCun,1989)

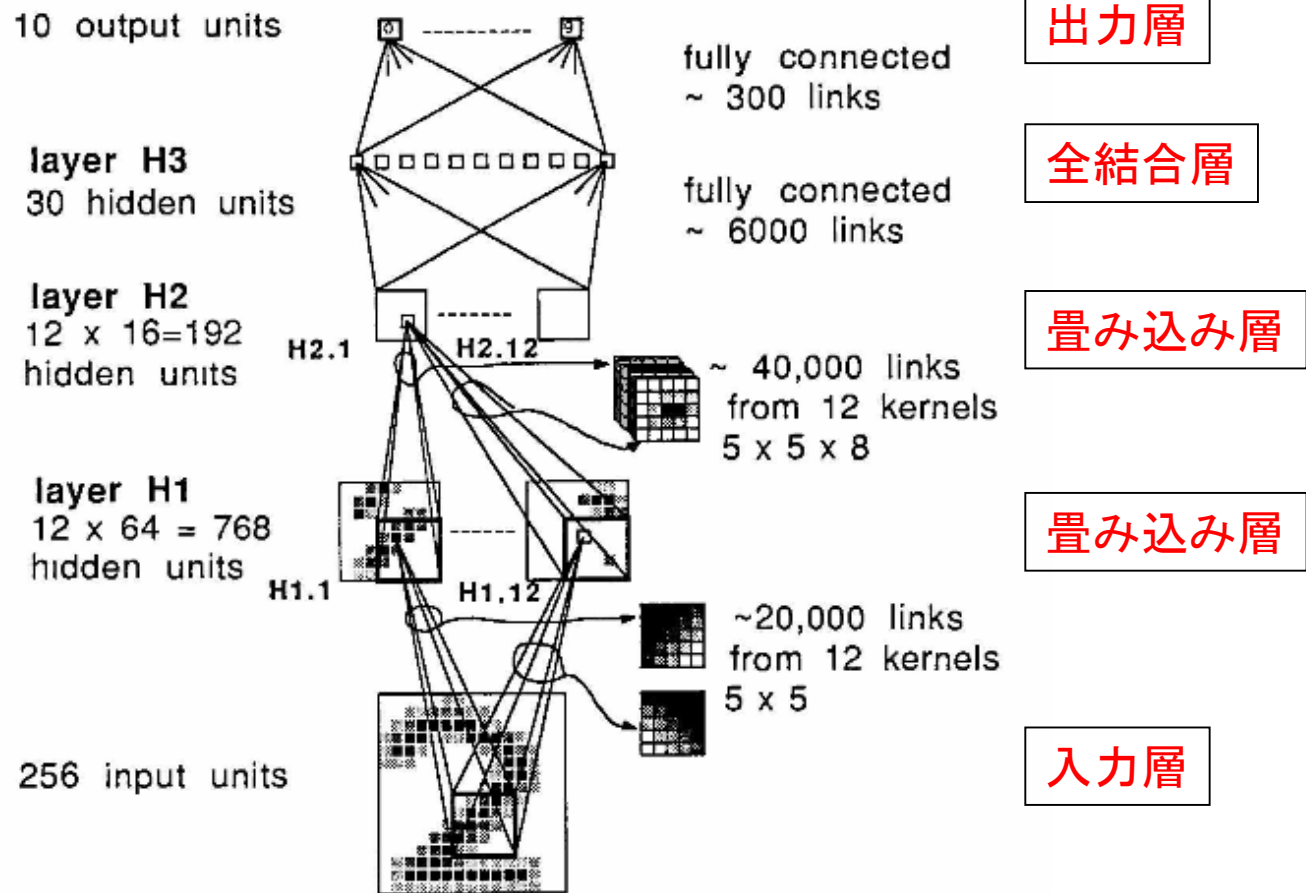
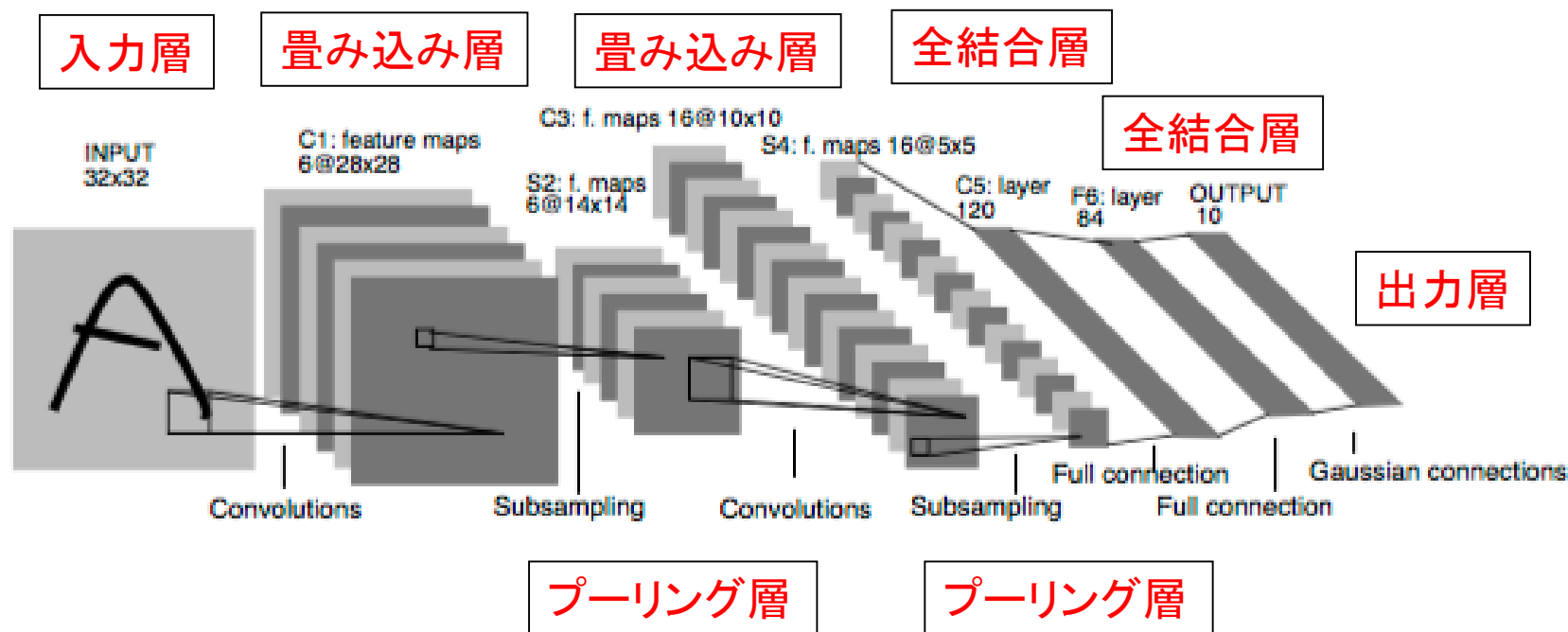


図: Y. LeCun, Backpropagation Applied to Handwritten Zip Code Recognition, 1989

畳み込みニューラルネットワーク

LeNet-5 (Y. LeCun, 1998)

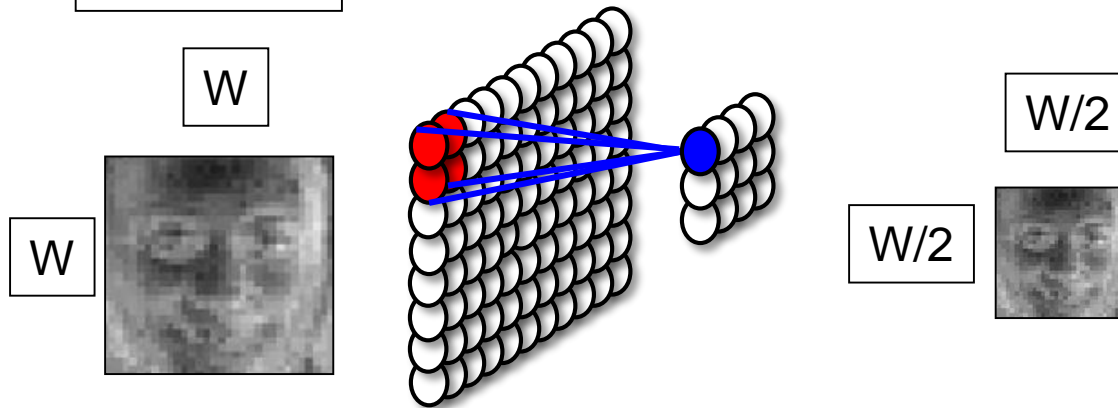
- 畳み込み層, プーリング層, 全結合層から構成



Y. LeCun, L. Bottou, Y. Bengio and P. Haffner, Gradient-Based Learning Applied to Document Recognition, Proceedings of the IEEE, Vol.86, No.11, pp.2278-2324, 1998

プーリング層 (Pooling Layer)

特徴マップ



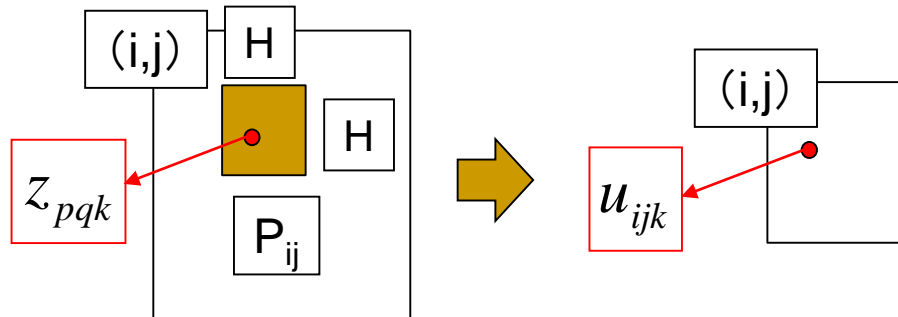
最大プーリング (大きさは 2×2 , スライドは2)

22	27	26	68	64	39	38	72
16	84	29	4	10	47	25	3
37	66	7	89	49	72	81	24
83	67	61	70	95	88	43	48
54	70	0	49	54	34	29	92
10	97	25	1	67	43	10	67
61	66	59	16	54	85	58	17
29	32	87	63	37	15	8	44



84	68	64	72
83	89	95	81
97	49	67	92
66	87	85	58

プーリング



最大プーリング

$$u_{ijk} = \max_{(p,q) \in P_{ij}} z_{pqk}$$

平均プーリング

$$u_{ijk} = \frac{1}{H^2} \sum_{(p,q) \in P_{ij}} z_{pqk}$$

Lpプーリング

$$u_{ijk} = \left(\frac{1}{H^2} \sum_{(p,q) \in P_{ij}} z_{pqk}^P \right)^{\frac{1}{P}}$$

出力層

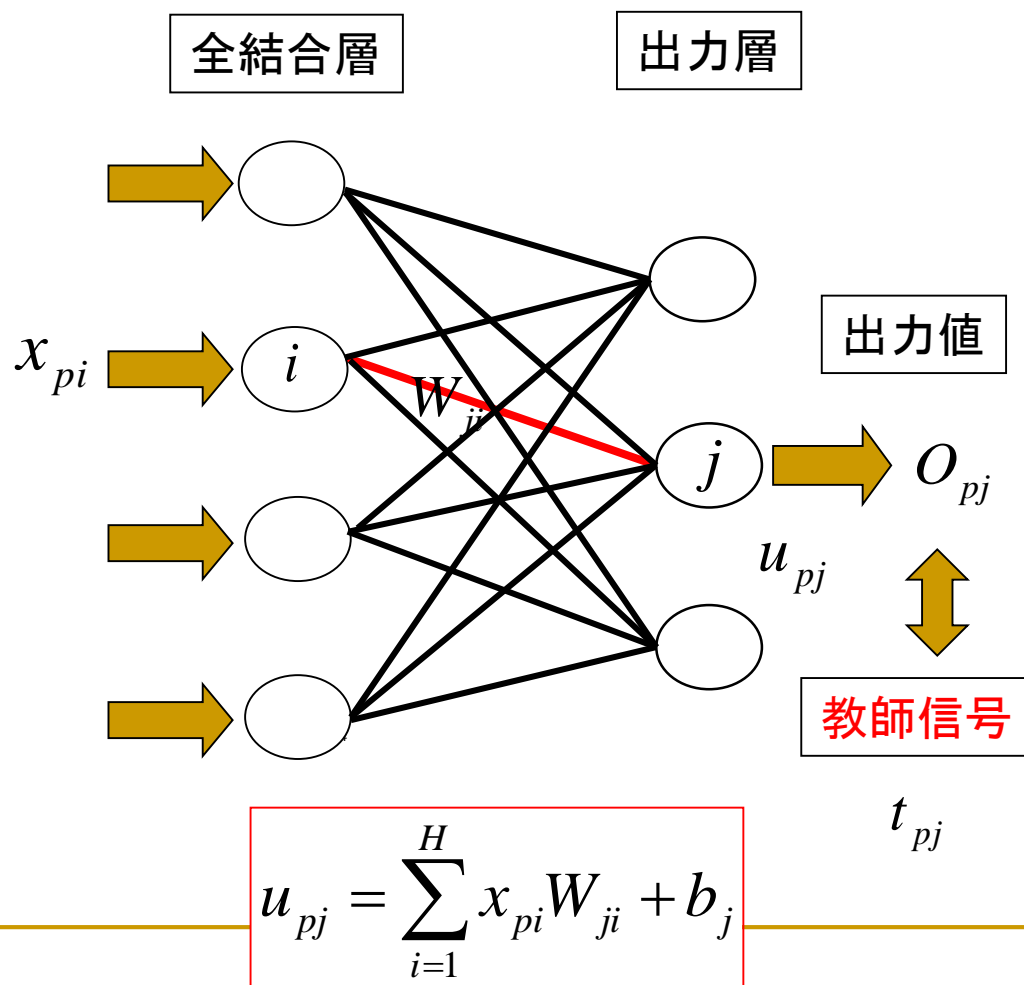
回帰の場合

$$O_{pj} = u_{pj}$$

クラス分類の場合

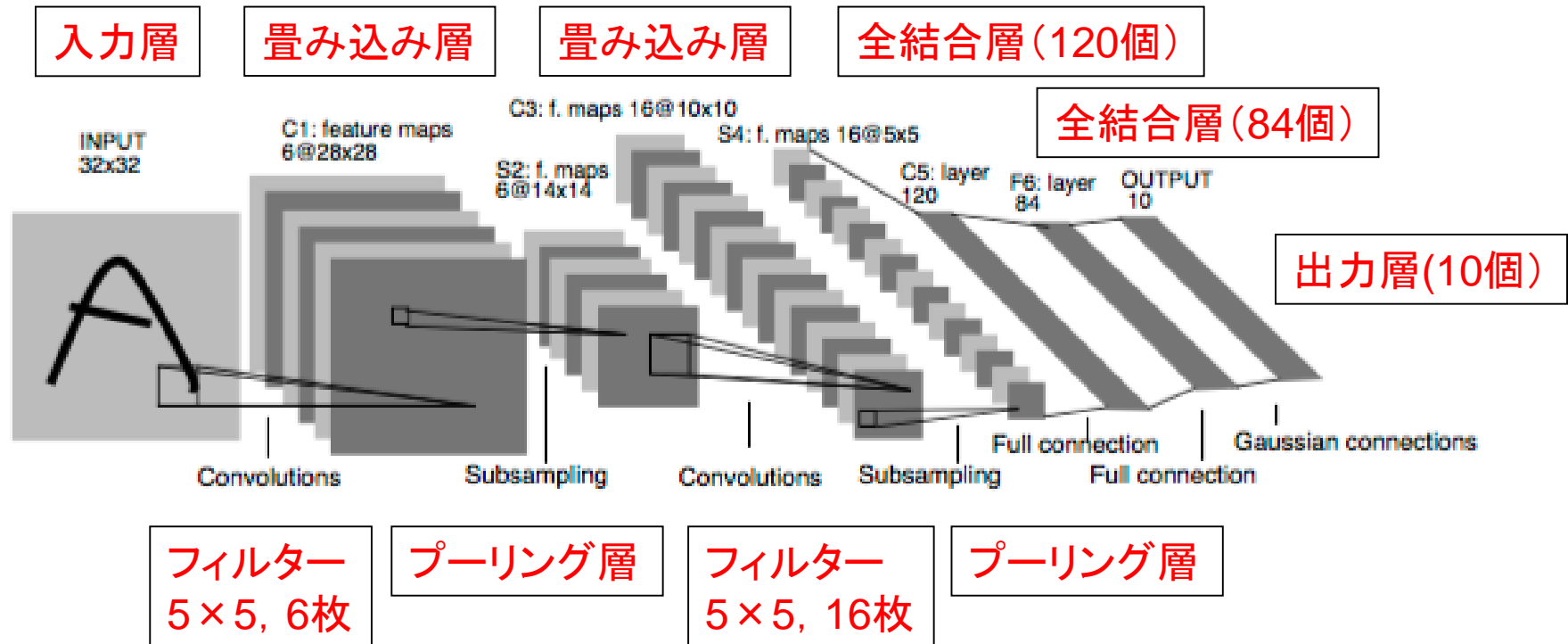
$$O_{pj} = \frac{e^{u_{pj}}}{\sum_{k=1}^K e^{u_{pk}}}$$

ソフトマックス関数



畳み込みニューラルネットワーク(LeNet5)

数字認識

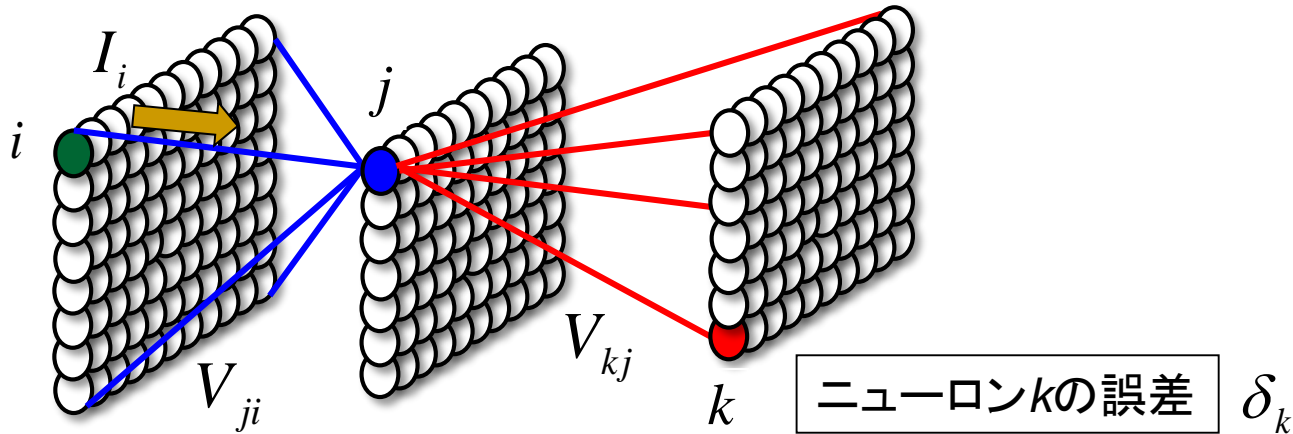


学習方法(誤差逆伝播則)

全結合の場合

ニューロン j の誤差

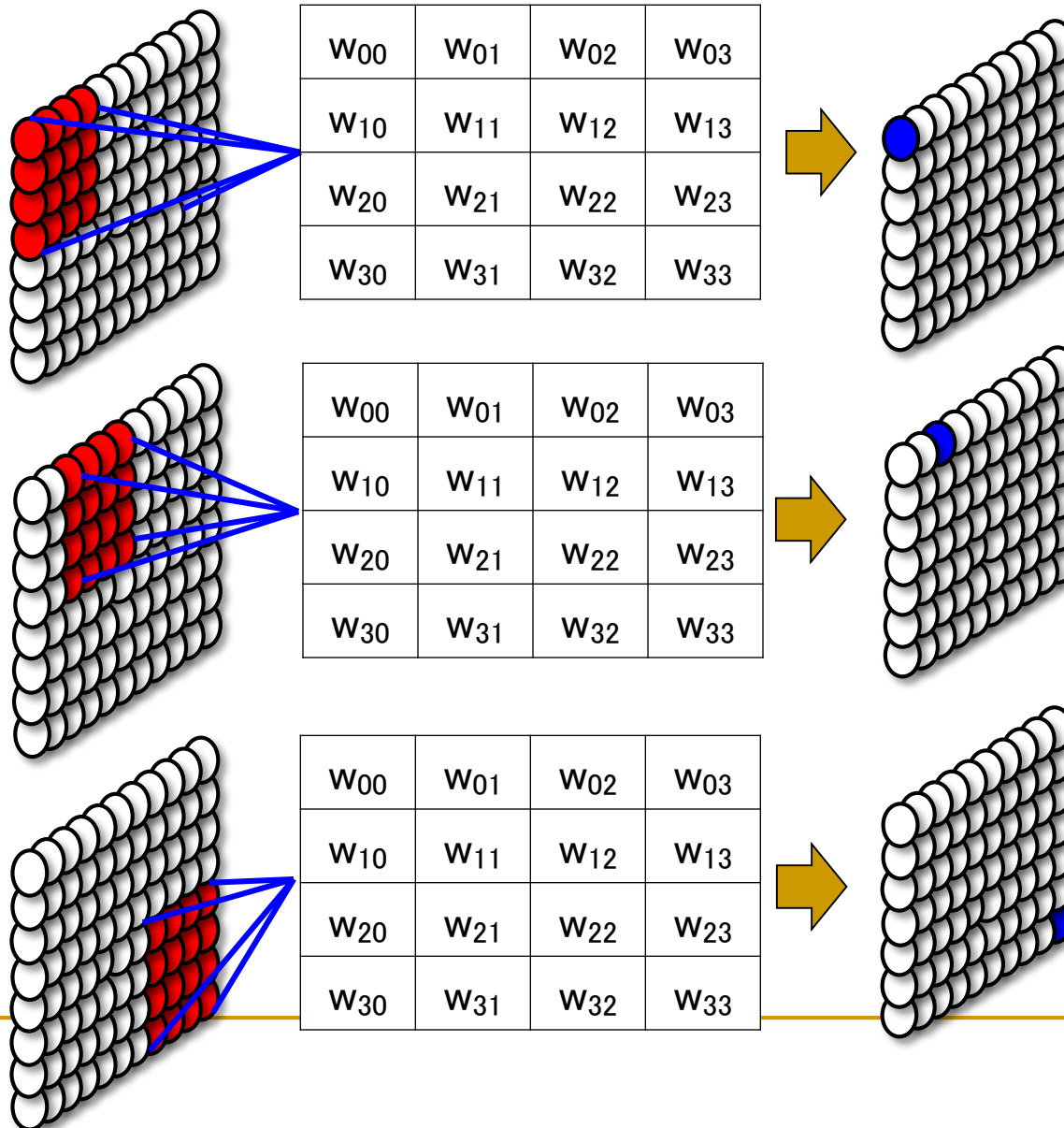
$$\delta_j = \left(\sum_{k=1}^m \delta_k V_{kj} \right) \frac{\partial f(U_j)}{\partial U_j}$$



$$V_{ji} \leftarrow V_{ji} - \alpha \frac{\partial E_p}{\partial V_{ji}} \quad \frac{\partial E_p}{\partial V_{ji}} = \delta_j I_i$$

*二次元ですが、式は一次元にしたものとして考えます

畳み込み層の学習方法



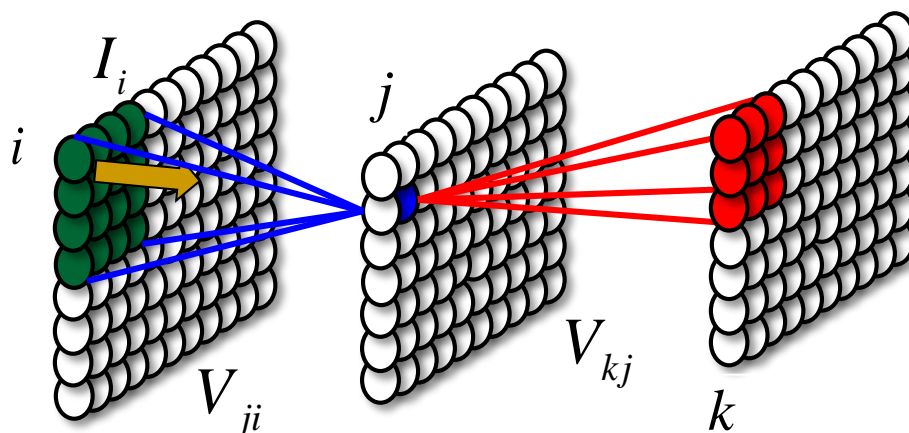
畳み込み層の学習方法(誤差逆伝播則)

畳み込み層の場合

ニューロン j の誤差

$$\delta_j = \left(\sum_{k=1}^m \delta_k V_{kj} \right) \frac{\partial f(U_j)}{\partial U_j}$$

結合しているニューロンの
誤差のみ逆伝播



ニューロン k の誤差 δ_k

畳み込み層

畳み込み層

畳み込み層

$$V_{ji} \leftarrow V_{ji} - \alpha \frac{\partial E_p}{\partial V_{ji}} \quad \frac{\partial E_p}{\partial V_{ji}} = \delta_j I_i$$

*二次元ですが、式は一次元にしたものとして考えます

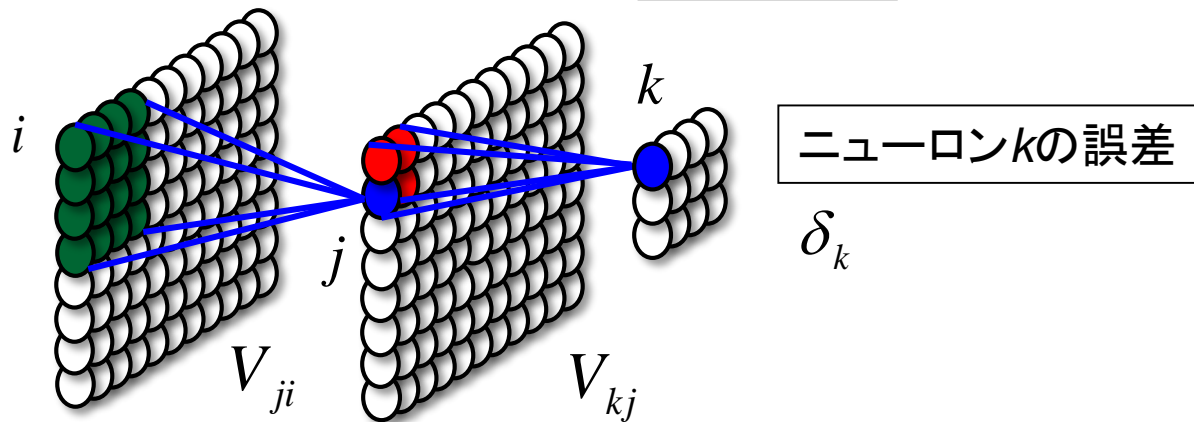
プーリング層の場合

プーリング層の場合

特徴マップ

プーリング層

新しい
特徴マップ

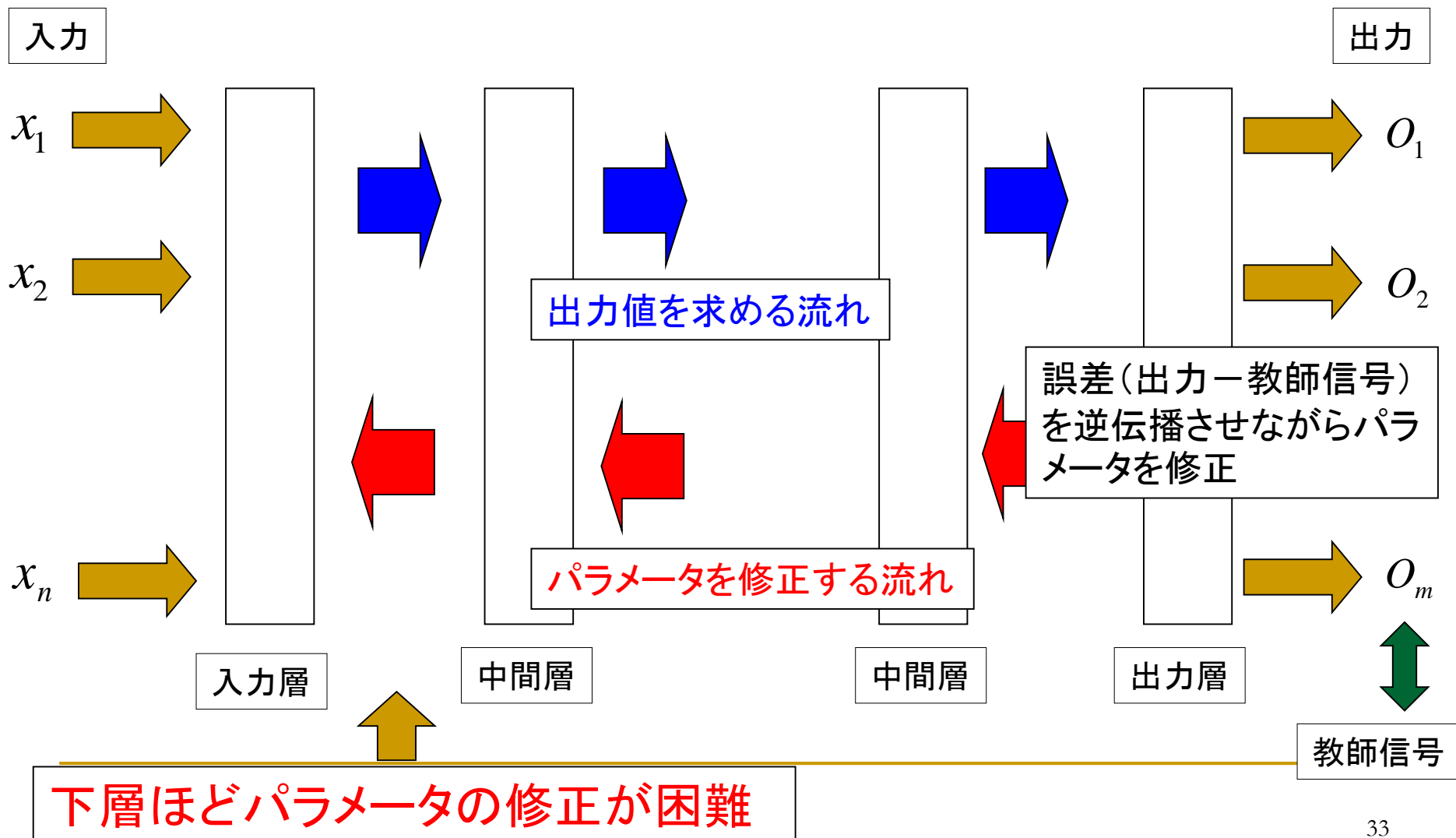


結合係数の学習はしない

プーリングの結果, j と結合しているニューロンに誤差を伝播させる

誤差逆伝播則の問題点

■ 勾配消失問題問題

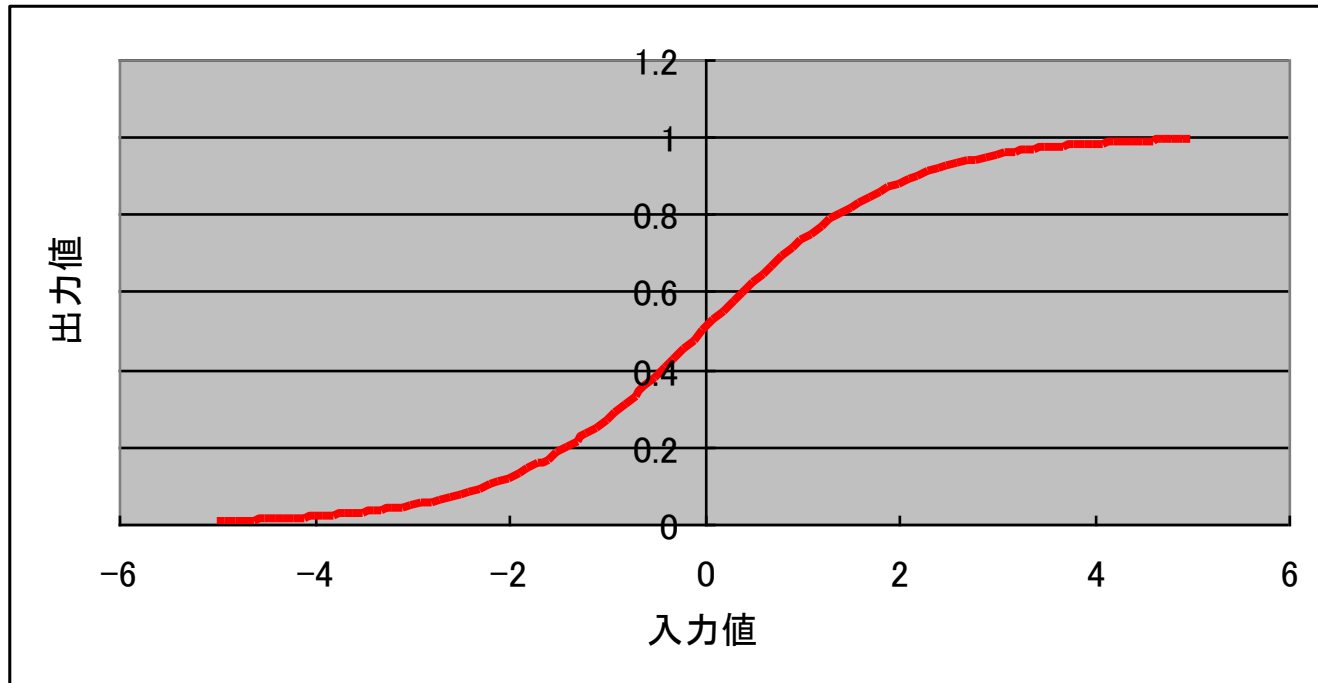


畳み込みニューラルネットワークの特徴

- 疎結合
- 勾配消失問題を解消するための手法の導入
 - 正則化
 - ドロップアウト
 - 活性化関数の改良 (ReLU関数)
- 多層化 (深層学習) が可能

活性化関数

シグモイド関数

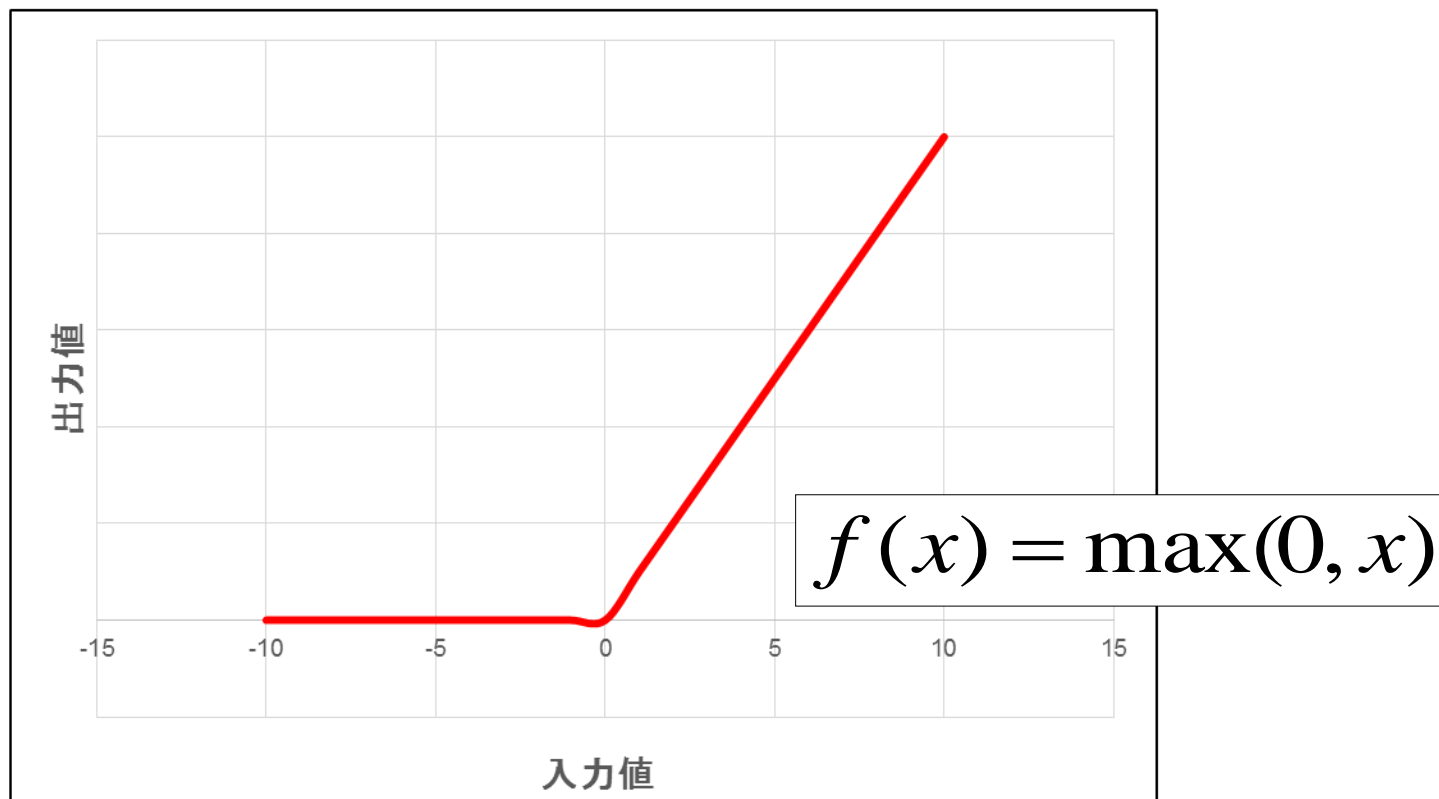


出力値の範囲

$$O_{i1}, O_{i2}, \dots, O_{im} \quad 0 \leq O_{ij} \leq 1$$

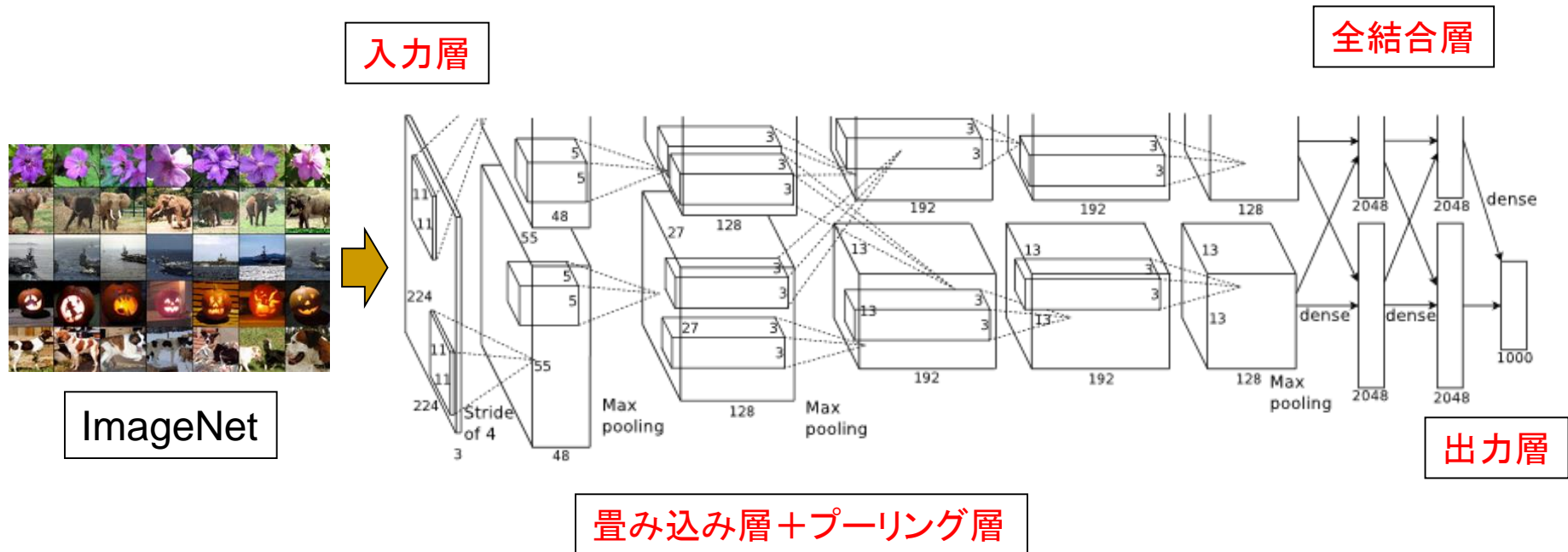
活性化関数

■ 正規化線形関数 (Rectified Linear Unit)



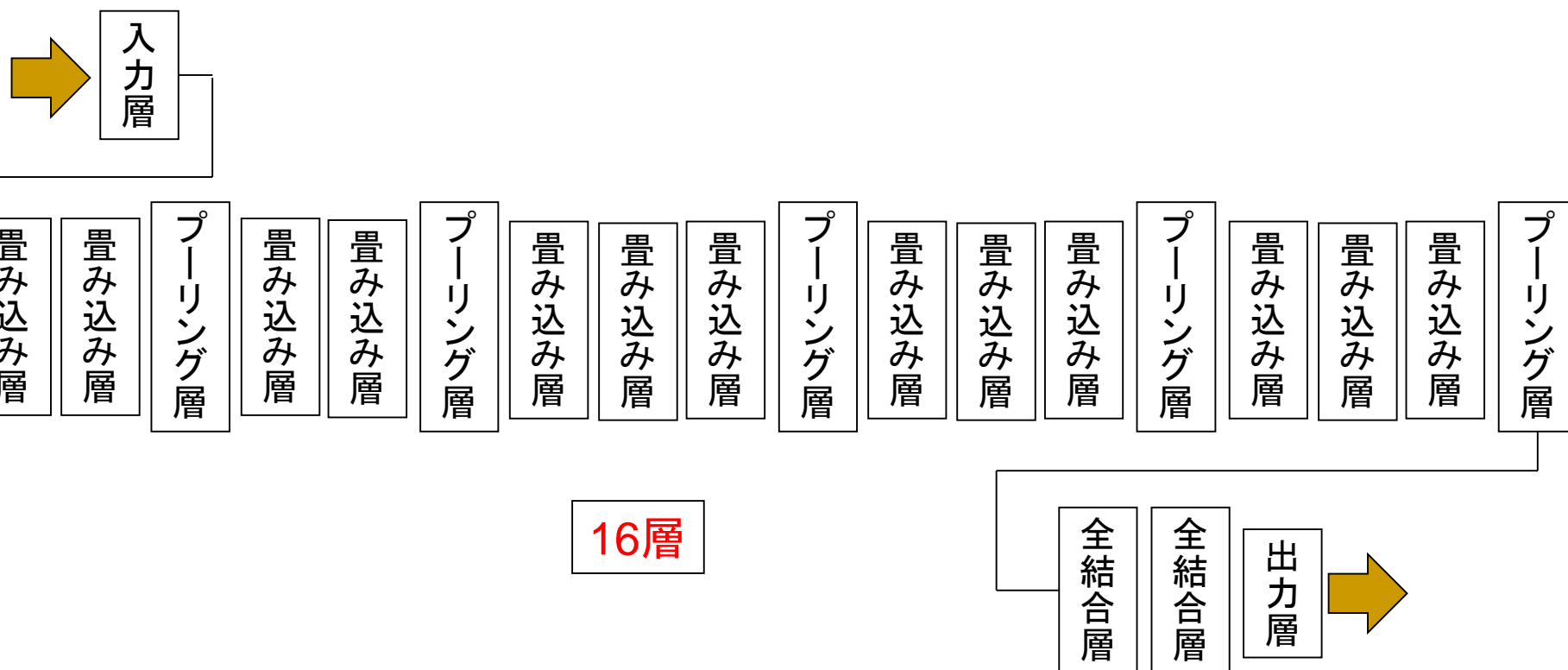
AlexNet (A. Krizhevsky, 2012)

- 8層の畳み込みニューラルネットワーク
 - ILSVRC2012において判定エラー率を25.8%から16.4%に改善



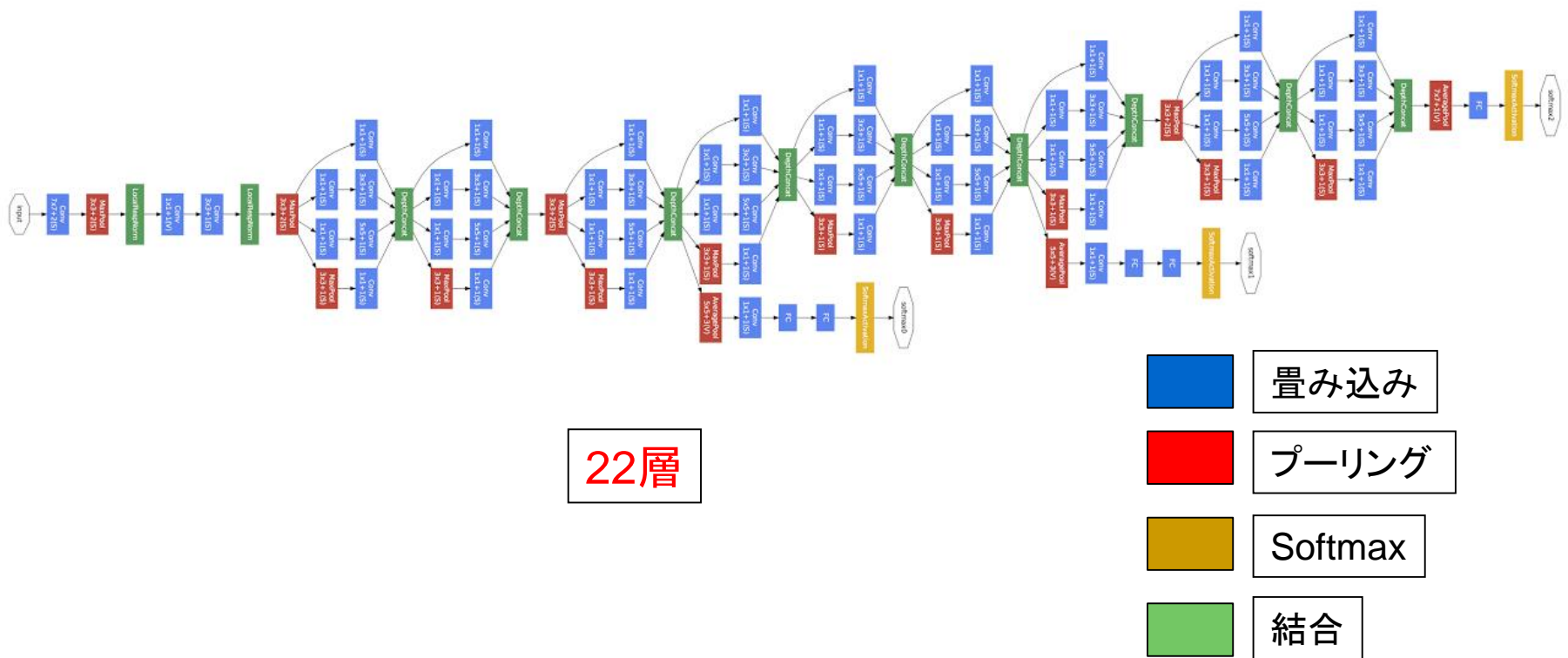
VGG (Visual Geometry Group)

- VGG-16 (ILSVRC2014)
 - 判定エラー率を7.3%に改善



GoogLeNet (C. Szegedy, 2014)

- GooLeNet (Inception-v3) (ILSVRC2014)
 - 判定エラー率を6.7%に改善



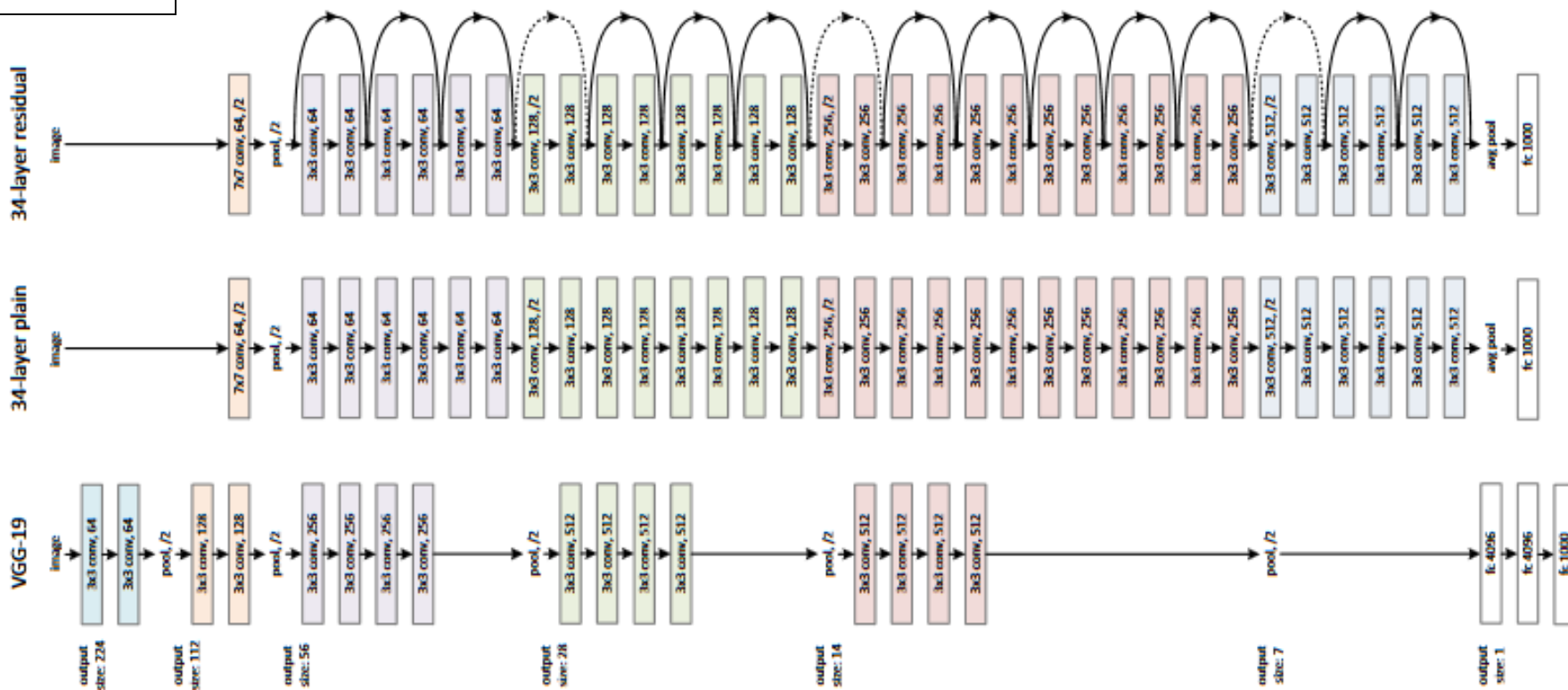
ResNet (Microsoft Research Asia, 2015)

■ ResNet (Residual Network)

- 判定エラー率を3.57%に改善

ResNet

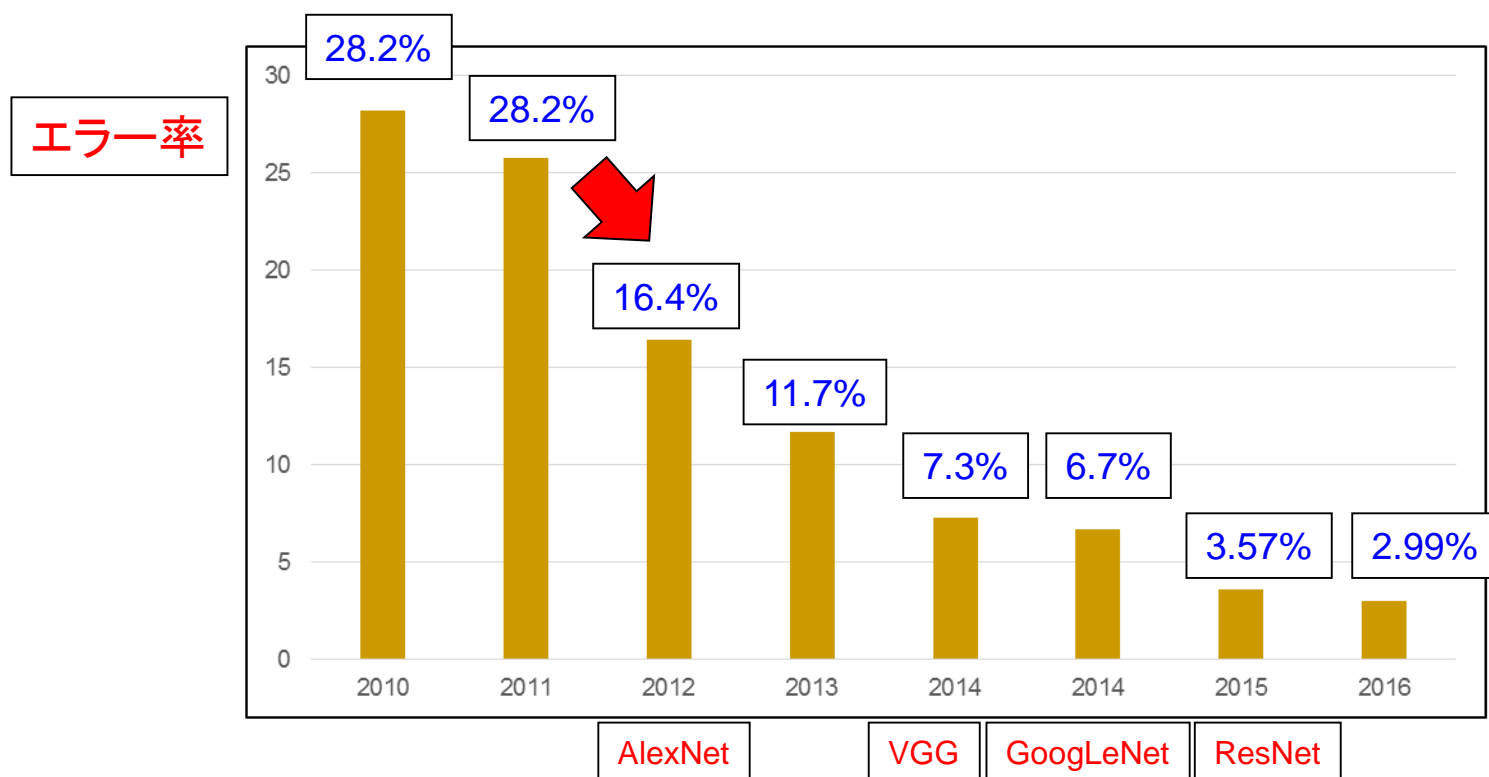
152層 (ILSVRC2015)



☒ : K. He, Deep Residual Learning for Image Recognition, 2015

大規模化と精度の向上

■ ILSVRCにおけるエラー率の向上



ILSVRC2013: ZFNET
ILSVRC2016: CUI Image

参考文献

- J.デイホフ:ニューラルネットワークアーキテクチャ入門, 森北出版(1992)
- P.D.Wasserman:ニューラル・コンピューティング, 理論と実際, 森北出版(1993)
- 岡谷貴之:深層学習, 講談社(2015)
- 人工知能学会:深層学習, 近代科学社(2015)
- 瀧雅人:これならわかる深層学習入門, 講談社(2017)
- 原田達也:画像認識, 講談社(2017)