

ResNetについて

ER17076 安井 理

目次

① 目的

② 概要

③ 方法

④ ResNetブロック

⑤ bottleneckアーキテクチャ

⑥ 今後

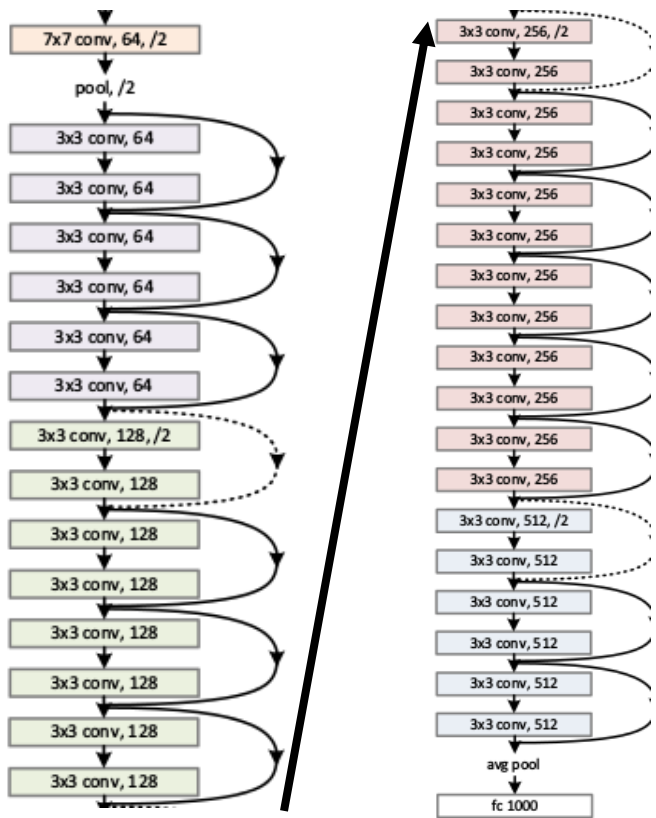
1. 目的

- ・6次元物体認識の論文を調べるうえでinceptionV4についての理解が必要
- ・inceptionV4には、ResNet(Residual Network)について学ぶ必要がある
- ・ネットワークを深くしたとき過学習を防ぐために用いられる

2. 概要

- ResNet (Residual Network) 残差を用いたネットワーク
- ニューラルネットワークの層を増やすモデル
- 層が多いほど高精度な検出が可能.単純に増やすだけではダメ

2. 概要

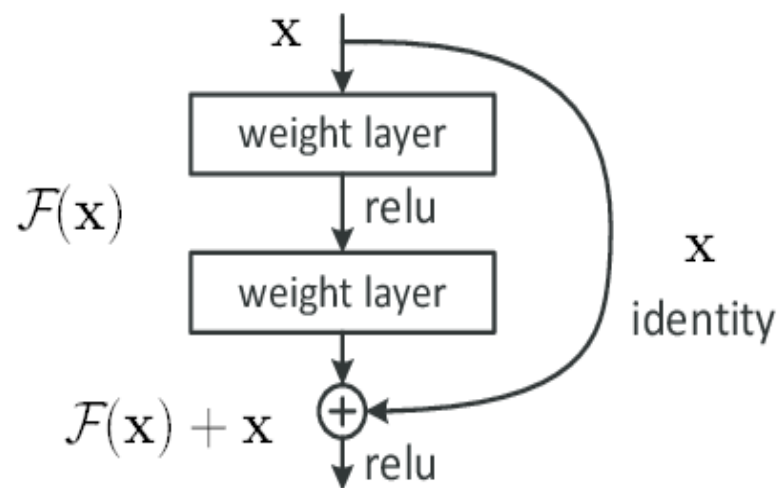


- ・最初は 7×7 の畳み込み
その後、 3×3 の畳み込みが続く
- ・以前のモデル(2012. Alexnet)では
 11×11 のフィルタから小さくしていく手法
- ・ResNetでは 3×3 のフィルタを
5回繰り返すと 11×11 と同等
- ・計算量は小さい.
 $3 \times 3 \times 5 = 45$
 $11 \times 11 = 121$

3.方法

- ・一部層で、関数(最適出力)を学習するのではなく入力参照
- ・層の入力を参照した残差関数を学習することで最適化
- ・ $H(x)$ を学習してほしい関数とすると、入力との差分: $F(x) = H(x) - x$
- ・ $H(x) = F(x) + x$ を学習するように再定義

4. ResNetブロック



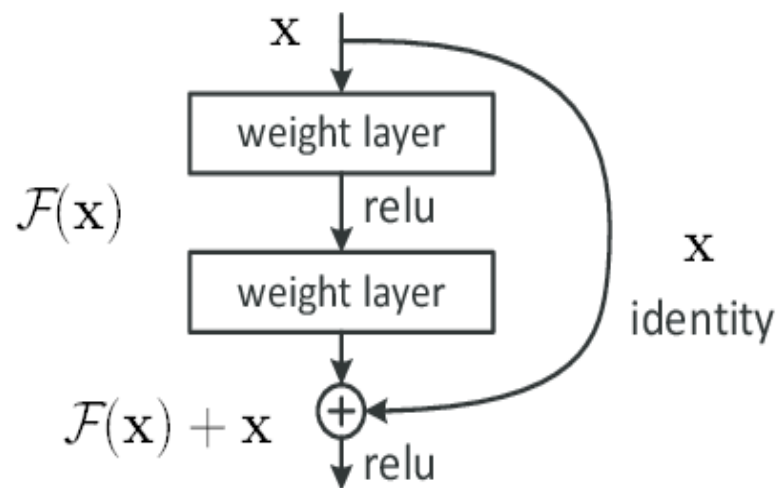
残差 $\mathcal{H}(x) - x$ が最適になるよう学習

計算も複雑でない

逆誤差伝播も可能なので実装も容易

<https://arxiv.org/abs/1512.03385>

4. ResNetブロック



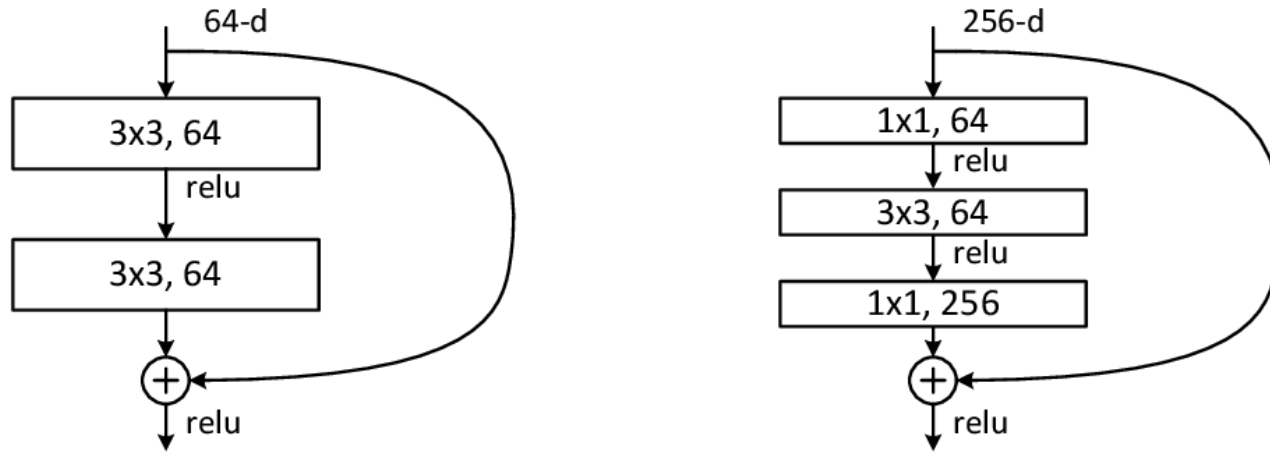
x, y がブロックの入カベクトル出カベクトル

残差の関数は $F = W_2 \sigma(W_1 x)$ で表される
(σ はReLU関数)

x と F は同じ次元でなければならない

<https://arxiv.org/abs/1512.03385>

5. bottleneckアーキテクチャ



深いネットワークを構成する場合、ブロックを2層ではなく3層に拡張

1×1 , 3×3 , 1×1 の3層にすることで、2層のときの同等の計算コスト

7. 今後

- inceptionV4について学ぶ
- 実装について考える

4. 参照

<https://arxiv.org/pdf/1602.07261v1.pdf>

https://deepage.net/deep_learning/2016/11/30/resnet.html

<https://www.kumilog.net/entry/resnet-paper>