

スパース性に基づく正則化を導入した DenseNet

日大生産工 ○種田 祥吾 日大生産工 山内 ゆかり

1. まえがき

物体認識において Convolution Neural Network(CNN)[1]が機械学習としてよく用いられている.近年,CNNはハードウェアとネットワークの改善によりさらに深い層での学習が可能になり,新たな深いネットワークが学習可能となったため,学習精度は通常のCNNよりも向上した.しかし,CNNの層が深くになるにつれて,入力値や,勾配に関する情報がネットワークの最後または最初に到達するまでに消失する可能性があることが分かった.この問題に対処し,深い層での学習が可能となったネットワークが Gao Huangらが提案したDense Convolutional Network(DenseNet)[2]である.Gao Huangらは層を層の間の情報の流れを最大にするために前の層すべての入力の値を後続層すべてに渡した.渡した値を連結することで情報の保存,維持ができるようになった.本研究では,DenseNetの畳み込み部分の改良を行うことで既存のDenseNetより精度が向上することを目指す.

2. 従来研究

2.1 畳み込みニューラルネットワーク

畳み込みニューラルネットワークは畳み込み層,プーリング層,全結合層の3種類の層で構成されている.畳み込み層ではカーネルを用いて特徴量を抽出する.プーリング層では画像の圧縮を行うことでレイヤーの複雑さを軽減する.畳み込み層とプーリング層を繰り返し,最後に全結合層を経由して結果が出力される.

2.2 DenseNet

DenseNetはネットワークの再利用を行うことで,学習が容易となり,パラメータ効率の高い凝縮モデルを生成することができる.Dense BlockとTransition Layerの2つを用いて構成される.Figure 1にDenseNetの簡単なモデルを示し,概要を説明する.まず,畳み込みを行いチャンネル数を増やす.次にDense Block内で畳み込みを行い,チャンネル数を増加させる.この時,畳み込みをする前にバッチ正規化と活性化関数であるReLUを使用する.Dense Blockの後にはTransition Layer内で,チャンネルの圧縮およびAverage Poolingを行い,出力サイズを半分にする.基本はDense BlockとTransition Layerを繰り返していき,最後にDense Block を通った

後,Global Average Poolingを行い,全結合層を通り,予測値が出力される.

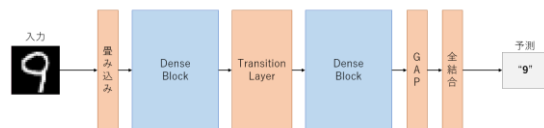


Figure 1 DenseNet のモデル

Dense Blockの構造についてFigure 2に示す.Dense Block内では入力値が成長率 k のチャンネル数に畳み込まれる. k チャンネルに畳み込まれた特徴マップは保存され,Dense Block内すべての入力に用いる.また,入力値は最初以外,Dense Block内の前層すべての値を結合した値となる. k チャンネルに畳み込んだ回数を l としたとき,Dense Blockを抜けた後の出力値は

$$x_l = H_l([x_0, x_1, \dots, x_{l-1}]) \quad (1)$$

となる.

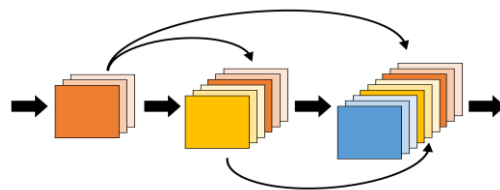


Figure 2 Dense Block のモデル

Transition Layerではモデルのコンパクトさを向上させるためにチャンネル数の圧縮を行う.圧縮率は θ ($0 < \theta \leq 1$)で表され,Dense Blockの出力値を θ 倍に圧縮する.その後,2*2 Average Poolingで出力サイズを小さくする.DenseNetは主にDense BlockとTransition Layerを繰り返すことで層を深くしていく.

最後にDense Blockを通った後,Global Average Pooling(GAP)を行う.GAPは通常のPoolingとは違い,1チャンネル内すべての値で平均プーリングをする.そしてプーリング後の値は全結合層を通り,最終的に予測値が出力される.

3. 提案手法

DenseNetはDense Block内部の畳み込み部分でランダムにドロップアウトを行って,過学習を抑制している.本研究ではDense Block内部の畳み込み部分でドロップアウトするのではな

く,スパース化を行うL1正則化を取り入れる.L1正則化は損失関数 $L(x)$ を最小化するために罰則項として損失関数に加算する[3].式(2)にL1正則化の最小化問題を示す.

$$\min_x f(x) \triangleq L(x) + \lambda \|x\|_1 \quad (2)$$

$$w \leftarrow w - \eta \nabla_w J_{train}^{old}(w) - \eta \lambda \cdot \text{sign}(w) \quad (3)$$

パラメータの更新式を式(3)に変更する.式(3)を基に学習を行う. λ はハイパーパラメータであり $\lambda = 0.01$ に設定する. $\text{sign}(w)$ では w の符号を判断する.0より大きい場合1,0より小さい場合-1,0の場合0を出力する.パラメータが一定値より小さくなった時,パラメータは0となる.以降0となったパラメータは学習を行わない.

4. 実験および検討

本研究ではMnist[4]のデータセットの学習用画像3000枚とテスト用画像1500枚でDenseNetの従来手法と提案手法の精度を比較する.また,従来手法と提案手法を組み合わせた時の精度も比較する.それぞれバッチサイズ64でミニバッチ学習を行う.成長率を $k=4$ とし,Transition Layerのチャンネル圧縮のための圧縮率を $\theta = 0.5$ にし,チャンネルを半分にする.学習率は0.1に設定し,20epoch学習を行う.

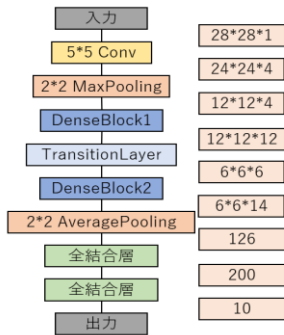


Figure 3 ネットワーク構成

Figure 3にネットワークの構成を示す.初めに 3×3 の畳み込みを行い, 2×2 の最大値プーリングでマップサイズを小さくする. Dense Block内では 1×1 のフィルタで $k \times 2$ チャンネルに畳み込み, 3×3 のフィルタで k チャンネルに畳み込む.従来研究では最後のGAPを通して全結合層へ値が伝わるが,本研究では低層での実験を行うため,チャンネル数は小さくなる.GAPの代わりに 2×2 の平均値プーリングを行う.

Table 1 実験結果

	train	test
Dropout	89.68	83.30
L1-regularization	90.32	83.94
Dropout+L1-regularization	89.84	83.02

Table 1に各手法の4回平均の実験結果を示す.提案手法を導入したことにより従来手法より汎化性能が向上することがわかった.また,従来手

法と提案手法を組み合わせる学習を行った.この手法では従来手法及び提案手法より精度が低くなった.精度が低くなった理由としてドロップアウトでの欠損とスパース性による情報削減により学習に必要なデータが足りなかったためだと考える.

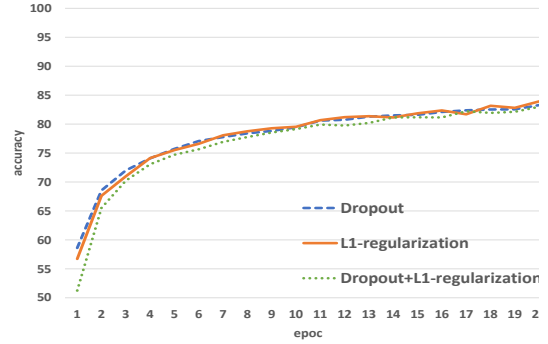


Figure 4 テスト時の学習曲線

Figure 4に実験した各手法のテスト時の学習曲線を示す.初めは従来手法の精度が良くなるが,学習していくうちに提案手法の精度が従来手法より高くなる時が多くなっていることがわかる.

5. まとめ

提案手法を導入したことにより,従来手法より良い結果を残すことができた.本研究で実装したDenseNetは低層であり,深層学習ほどの深さはないが,提案手法を導入することで従来手法より本質的な情報を得ることができたといえる.

参考文献

- [1] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton "ImageNet Classification with Deep Convolutional Neural Networks", Proc. NIPS (2012)
- [2] G. Huang, Z. Liu, L. Van Der Maaten and K. Q. Weinberger, "Densely Connected Convolutional Networks," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2261-2269, doi: 10.1109/CVPR.2017.243.
- [3] Mark Schmidt, Glenn Fung, Romer Rosaless, "Optimization Methods for l_1 -Regularization", University of British Columbia, Technical Report TR-2009-19.
- [4] "THE MNIST DATABASE of handwritten digits", <http://yann.lecun.com/exdb/mnist/>, (2022/10/11)