A Proposal for an Accelerated Model of Ensemble Deep Learning in Image Classification

Jian PIAO*, Mingzhe JIN**

(Received April 4, 2021)

Convolutional neural network (CNN) models have become the basis for deep learning in image classification because of their superior performance. Ensemble learning is effective in increasing the accuracy of CNN models. However, ensemble learning requires greater training and implementation time than a CNN model. In this paper, we present a split convolutional ensemble model that costs less in terms of training and implementation time, with same accuracy as that of the original ensemble method. This method can save up to 50% of time compared with the original ensemble method.

Key words：CNN, image classification, ensemble learning, training and implementation time

キーワード：畳み込みニューラルネットワーク，画像分類，集団学習，訓練時間，予測時間

画像分類におけるアンサンブル深層学習の加速化モデルの提案

朴　健，金　明哲

1. はじめに

画像分類技術は,医学や工業など様々な領域で使用されている.従来の画像分類では,人手で設計した特徴量をサポートベクターマシン(Support vector machine, SVM)のような機械学習モデルで分類する方法が多く使われていたが,今は Krizhevsky(2012)により提案された深層学習をベースとする畳み込みニューラルネットワーク(Convolutional neural network, CNN)が広く使用されている[1].

また,CNN モデルを用いる際に,正解率を上げるため,アンサンブル学習(Ensemble learning)方法がよく用いられる.アンサンブル学習は,複数のモデルの出力を統合・組み合わせ,正解率を向上させる機械学習方法である.Ciresan(2012)は CNN に基づいたアンサンブル学習モデルを画像分類に応用することを試みた[2].実験では,いくつかの CNN モデルを作成し,それぞれの CNN から出力された予測値の平均値を予測値の結果とした.その結果,ベースラインより高い正解率を得た.

しかし,CNN モデルは深層学習モデルの一種であるため,計算時間が一般的な機械学習方法より長く,アンサンブル学習の方法を適用することで,訓練時間と予測時間が更に増えるという欠点がある.訓練時間が長すぎると複数のモデルの試行や,ハイパーパ

* pjtmdpj223@gmail.com
**Culture and Information Science, Doshisha University, Kyoto
mjin@mail.doshisha.ac.jp

ラメーターのチューニングなどが難しい. また, 予測時間が長すぎると, 自動運転のようなリアルタイムタスクへの応用が困難になる.

予測時間を削減させることを目的とした研究として, Hinton et al. (2015)は一つの計算量が多いモデル, 或いは複数の計算量が少ないモデルが含まれているアンサンブル学習モデルを相対的に計算量が少ない1つのモデルに圧縮させるDistillingという方法を提案した[3]. また, Liu et al. (2017)はCNN構造で多く用いられるバッチ正則化操作を用いて寄与の度合が低い部分を除く方法を提案した[4].

訓練時間を減らすことを目的とした研究として, Huang et al. (2017)は一つのモデルを訓練する時間内でm個のモデルの重みを取得するSnapshotという方法を提案した[5]. また, Zhang et al. (2020)はSnapshot のいくつかの欠点を改良した Snapshot Boosting という方法を提案した[6]. モデルの訓練時間と予測時間は両方とも重要なものであるため, 共に減らすことが望まれているが, 先行研究は訓練時間か予測時間のいずれかに着目している.

本論文では, CNNのアンサンブル学習モデルである分裂畳み込みニューラルネットワークという構造を提案する. 提案したモデルはベースラインの正解率を保ちながら, 訓練時間と予測時間を短縮することを目指す.

また, アンサンブル学習モデルは各モデルの出力を統合するため, 各モデルから異なる特徴が学習され, 異なる出力を求めることが望まれる. 本論文ではモデルの多様性を高める手法を検討する.

2. 分裂畳み込みニューラルネットワーク

本章では,分裂畳み込みニューラルネットワークを提案し,モデルの多様性を高める手法を紹介する.

2.1 分裂畳み込みニューラルネットワークの構造

アンサンブル学習モデルは, m 個の CNN バックボーンを結合して一つの学習モデルを構築する. その際, m個のバックボーンの浅い層を訓練することが必要であり, 計算時間が長くなる. これはアンサンブル学習モデルの効率が悪い原因の一つである.

本研究では, Fig. 1 に示すような主列と副列を並列させた分裂畳み込みニューラルネットワークを提案する. 主列は, アンサンブル学習モデルのバックボーンの一つであり, 副列に浅い層の学習結果を提供すると同時に深い層の学習を行う. 提案するモデルは, 副列では主列の浅い層の学習結果を引継ぎ, 深い層の学習を行うため, Fig. 2 に示すような従来のアンサンブル学習モデルより計算時間を削減することが期待できる. 同様に, 予測コストも減らすことができる.

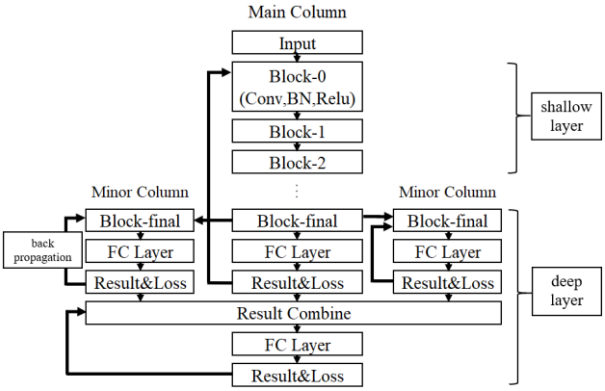

Fig. 1. Example network architecture for proposed method.

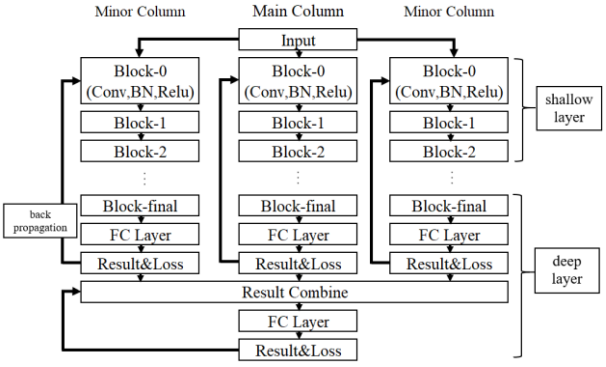

Fig. 2. Baseline architecture for ensemble learning.

2.2 モデルの多様性

アンサンブル学習においては, 各モデルから可能な限り異なる特徴量が得られること, 即ちモデルの多様性が高いことが望まれる. しかし, 分裂畳み込みニューラルネットワークは深い層で分裂してから各モデルの多様性を生み出すため, ベースラインとするアンサンブル学習モデルと比べて多様性が弱い可

能性がある.したがって,正解率を保つためにモデルの多様性を高める対策が必要である.

ニューラルネットワークを訓練する際には,ランダムにノイズを入れ,モデルを正則化する方法がよく用いられる.本論文では画像データに効果が良いと報告されたMixup(Zhang et al, 2017)を各副列に取り入れ,特徴量の多様性を高めることができるかどうかを検討する.Mixupはランダムに2つのデータを結合し,新しいデータとラベルを生成して訓練に用いる方法である [7].副列の入力特徴量に対してMixup方法を使用することは,各バックボーンが異なるデータを用いて訓練を行うことを意味する.このような方法は各バックボーンから取られた特徴量の多様性を高めることができると考えられる.

3. 実験

本章では,2章で述べた分裂畳み込みニューラルネットワーク,バックボーンの多様性を高めることが期待できる方法を用いて実験を行う.Ciresan(2012)が提案したアンサンブル学習方法は,現在最も多く使われている方法であるため,本論文ではベースラインとする.

実験には,Fig. 1 に示すモデルについて,上記の2.2 節で説明したモデルの多様性の有効性を確認するため, 二つの方法を用いる.一つは,ベースラインで用いた計算方法であり,もう一つは Mixup 正則化方法を導入した方法である.前者を提案方法 1,後者を提案方法2とする.

3.1 実験設定

本論文では,CIFAR-10,CIFAR-100[8] とTiny-ImageNetの三つのデータセットを用いて,提案した分裂畳み込みニューラルネットワークの有効性を検証する.CIFAR-10 と CIFAR-100 は合計 60000 枚の32×32 ピクセルの画像であり,そのうち,50000 枚は学習データであり,残りの 10000 枚はモデルの正解率を測るためのテストデータである.CIFAR-10 には,10 種類(飛行機,車,鳥,猫,鹿,犬,蛙,馬,船,トラック)の画像が含まれている.各種類はそれぞれ5000枚の学習データと 1000 枚のテストデータに分かれているため,バランスの取られたデータである.CIFAR-100 は,CIFAR-10 と似たデータ形式であるが,100 種類の画像データがあり,種類ごとに 500 枚の訓練データと 100 枚のテストデータがある.Tiny-ImageNet は 200 種類のカラー画像であり,各種類には500枚の訓練データと50枚のテストデータが含まれている.画像のサイズは64×64 である.

深層学習モデルは大量の訓練データを要求するため,データを拡張する前処理が必要である.本実験では, ランダムに左右に反転した画像を拡大したうえでランダムクリップして用いる.CIFAR データを 40×40 のピクセルまで拡大し,ランダムにその中の 32×32 部分を切り取る.Tiny-ImageNet の場合では 80×80 まで拡張し,ランダムに64×64 の部分を切り取る.最後に,切り取った画像に対してランダムに左右反転を行い,標準化する.

CIFAR データにおいてはそれぞれ,20 層を持つResNet-20[9],各層のパラメーターを 8 倍にした 16 層を持つ WRN-16-8[10]と各層を 3 に区分けした 29 層を持つ ResNeXt-29-3d[11]バックボーンを用いる.データを訓練する際には各バックボーンの論文の通りに行った.

- ResNet: L2 正則化のλパラメーターを 0.0001,バッチサイズを128にする.学習率の初期値を0.1に設定する.訓練を 32000,48000 回繰り返した際,それぞれの学習率を0.1倍に調整して,合計64000回訓練する.
- WRN: L2正則化のλパラメーターを0.0005,バッチサイズを128にする.学習率の初期値を0.2に設定し,訓練を 23437,46875,62500 回繰り返した際,学習率を0.2倍にして,合計 78125 回訓練する.
- ResNeXt: L2正則化のλパラメーターを0.0005,バッチサイズを128にする.学習率は0.1から開始し,訓練を58593,87890回繰り返した際,学習率を0.1倍にして,合計 117187 回訓練する.

Tiny-ImageNet データセットにおいては,ResNet,WRN,ResNeXtのバックボーンのL2正則化のλハイパーパラメーターをそれぞれ 0.0001,0.0001,0.0005 にする.学習率の初期値は 0.1 に設定し,訓練を 120000,160000,200000 回繰り

返した際に学習率を 0.1 倍にして,合計 210000 回訓練する.Tiny-ImageNet のデータのサイズは CIFAR-10 と CIFAR-100 データの 2 倍であり,特徴量を取るためにより多くの層が必要とされるため,各種のバックボーンを ResNet-56,WRN-20-8,ResNeXt-37-4dにする.バックボーンの仕組みは変わらないが,CIFARデータのバックボーンをより深くしたものである.

GPU メモリを考慮した上で,各アンサンブル学習モデルの列数を 3 にした.より一般性を持つ結果を得るために,本論文では実験をそれぞれ 3 回行い,その平均値を分類の正解率とする.

3.2 正解率の結果

Table 1 にバックボーン,ベースライン,本論文で提案した二つの分裂畳み込みニューラルネットワークモデルの正解率を示す.各行における最も高い正解率を太字で示す.

Table 1 の正解率の第 1 列はバックボーン,第 2 列は Fig. 2 に示したベースラインである.第 3 列は提案方法 1,第 4 列は提案方法 2 である.提案方法 1 と提案方法2の違いは,それぞれMixup の方法の導入有無になる.

Table 1. Accuracy(%) using original backbone, baseline ensemble learning and proposed method.

		Backbone	Baseline	ProMeth1	ProMeth2
CIFAR-10	ResNet-20	91.30	**93.18**	92.66	93.02
	WRN-16-8	95.25	**95.63**	95.24	95.60
	ResNeXt-29-3d	93.30	**94.63**	94.13	94.24
CIFAR-100	ResNet-20	64.50	**72.74**	71.62	71.62
	WRN-16-8	78.90	**81.48**	80.60	81.14
	ResNeXt-29-3d	74.01	**78.21**	77.40	77.15
Tiny-ImageNet	ResNet-56	48.59	56.70	56.69	**57.76**
	WRN-20-8	51.81	55.72	54.34	**56.64**
	ResNeXt-37-4d	48.69	57.36	56.79	**57.37**

表の中に示されている数値のみでは,その差異を判断することが難しい.そこで,Table 1 のデータについて,チューキーの HSD 検定 (Tukey honestly significant difference test)の多重比較を行う.チューキーの HSD 検定は,多群のデータにおける各 2 群間の平均値の差についての検定を行う方法である.そのグラフを Fig. 3 に示す.Fig. 3 からわかるようにベースラインと提案方法 2 の信頼区間の中心が横軸の座標 0 に最も近い.これはベースラインと提案方法 2 の差が最も小さいことを意味する.ベースラインと提案方法 1 の差はやや大きい.若干ではあるが Mixup 正則化を行う提案方法 2 の方がよい.ベースラインと提案方法 2 の正解率の t 検定の p 値は 0.9875 である.

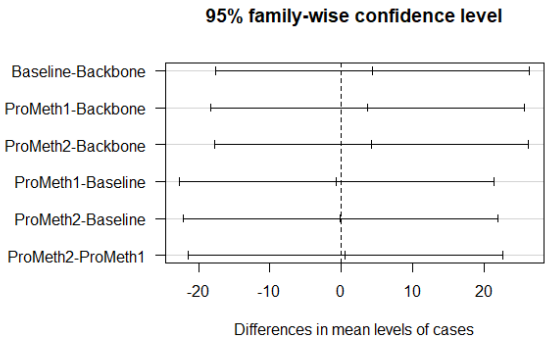

Fig. 3. Plot of Tukey-Honest significant differences.

3.3 時間の比較

本項ではモデルの訓練の時間と予測の時間について説明する.Table 2 では,分裂畳み込みニューラルネットワークと本論文でベースラインとしたアンサンブル学習モデルの訓練時間と予測時間を示す.各時間は 3 回の平均値である.比較のために,Table 3 に提案手法の計算時間をベースラインモデルの計算時間で割った値を示す.

Table 3 から,CIFAR-100 データセットと ResNeXt-29-3dバックボーンの組み合わせの場合,訓練時間と予測時間の減少量が最も多く,両方ともベースラインの 0.47 倍しかかからなかった.ベースラインモデルの計算時間と分裂畳み込みニューラルネットワークの計算時間に対して,対応ありの t 検定を行った結果,訓練時間の p 値は 0.0192(効果量は 0.52)であり,予測時間の p 値は 0.0012(効果量は 0.68)であり,提案した分裂畳み込みニューラルネッ

トワークとベースラインモデルの訓練時間と予測時間には有意の差があると判断できる.

Table 2. Computation time(s).

		Baseline Time		Proposed Method Time	
		Training	Implementation	Training	Implementation
CIFAR-10	ResNet-20	27.65	2.15	15.85	1.47
	WRN-16-8	149.50	9.64	80.75	5.39
	ResNeXt-29-3d	141.07	12.31	69.40	6.01
CIFAR-100	ResNet-20	33.03	2.14	17.32	1.30
	WRN-16-8	159.81	9.70	84.66	5.39
	ResNeXt-29-3d	140.16	12.40	65.25	5.77
Tiny-ImageNet	ResNet-56	1034.88	26.51	662.99	18.39
	WRN-20-8	399.09	11.66	268.95	8.08
	ResNeXt-37-4d	288.70	8.37	175.87	5.56

Table 3. Computation time ratio(proposed method/baseline ensemble learning).

		Training Time Ratio	Implementation Time Ratio
CIFAR-10	ResNet-20	0.57	0.68
	WRN-16-8	0.54	0.56
	ResNeXt-29-3d	0.49	0.49
CIFAR-100	ResNet-20	0.52	0.61
	WRN-16-8	0.53	0.56
	ResNeXt-29-3d	0.47	0.47
Tiny-ImageNet	ResNet-56	0.64	0.69
	WRN-20-8	0.67	0.69
	ResNeXt-37-4d	0.61	0.66

また, 訓練時間は予測時間より大幅減少する傾向が見られた. 深層学習モデルはフォワードプロパゲーション(Forward propagation)とバックプロパゲーション(Back propagation)の 2 つの段階に分けることができ, 訓練する際には 2 段階の計算がともに行われるが, 予測の際にはフォワードプロパゲーションのみを行うことが, 理由として考えられる. また, 計算時間の減少はバックボーンの計算量の増加につれて増える.

4. まとめ

本論文では, CIFAR-10, CIFAR-100 と Tiny-ImageNet の三つのデータセットを用いて, 三つの代表的な ResNet, WRN, ResNeXt をバックボーンとして, 提案した分裂畳み込みニューラルネットワークとベースラインの比較実験を行った.

その結果, 正解率においては, 増加と減少の微細の変動があるものの提案した分裂畳み込みニューラルネットワークはベースラインモデルと有意の差がなく, 正解率を保つことができた. 訓練時間と予測時間の平均値は, 提案の方法はベースラインの 0.5600±0.0680, 0.6011±0.0855 倍であり, 大きく減少している. 減少量が最も多かったのは, ResNeXt バックボーンを用いた分裂畳み込みニューラルネットワークであり, ベースラインの約 1/2 しかかからなかった.

この結果により, 計算量がより多いバックボーンを使うことや, 副列の数を増やすことなどにより時間をさらに減少させることができると考える.

また, ランダム性を用いる正則化方法 Mixup は特徴の多様性を高め, 正解率の向上に有効であることが分かった. その理由は主列と異なる信頼水準を持つ出力を提供することができることにある.

以上のように, 提案した分裂畳み込みニューラルネットワークモデルを用いることによって, 正解率を保ちながら訓練時間と予測時間を短縮できることが実証された. 提案の方法は, 画像分類タスクだけではなく, 物体検知, 画像セグメンテーションのようなタスクへの応用にも期待できる.

本論文では, ベースラインと比較するために主列と副列から得られたベクトルを線形モデルで結合を行った. しかし, 結合の方法が正解率に影響を及ぼす可能性があるため, Self-attention[12] 方法の導入などが課題として挙げられる.

参考文献

1) A. Krizhevsky, I. Sutskever and G. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", *Advances in Neural Information Processing Systems* (NIPS), **25**, 1097–1105 (2012).

2) D. Ciresan, U. Meier and J. Schmidhuber, "Multi-Column Deep Neural Networks for Image Classification", *IEEE*

Conference on Computer Vision and Pattern Recognition (CVPR), 3642-3649 (2012).

3) G. Hinton, O. Vinyals and J. Dean, "Distilling the Knowledge in a Neural Network", *NIPS Deep Learning and Representation Learning Workshop* (2015).

4) Z. Liu, J. Li, Z. Shen, G. Huang, S. Yan and C. Zhang, "Learning Efficient Convolutional Networks Through Network Slimming", *Proceedings of the IEEE International Conference on Computer Vision* (ICCV), 2736-2744 (2017).

5) G. Huang, Y. Li, G. Pleiss, Z. Liu, J. E. Hopcroft and K. Q. Weinberger, "Snapshot Ensembles: Train 1, Get M for Free", *5th International Conference on Learning Representations* (ICLR) (2017).

6) W. Zhang, J. Jiang, Y. Shao and B. Cui, "Snapshot Boosting: a Fast Ensemble Framework for Deep Neural Networks", *Science China Information Sciences*, **63**, 112102 (2020)

7) H. Zhang, M. Cisse, Y. N. Dauphin and D. Lopez-Paz, "Mixup: Beyond Empirical Risk Minimization", *International Conference on Learning Representations* (ICLR) (2018).

8) A. Krizhevsky, "Learning Multiple Layers of Features from Tiny Images", *Master's thesis, Department of Computer Science, University of Toronto* (2009).

9) K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition", *2016 IEEE Conference on Computer Vision and Pattern Recognition* (CVPR), 770-778 (2016).

10) S. Zagoruyko and N. Komodakis, "Wide Residual Networks", *Proceedings of the British Machine Vision Conference* (BMVC), 87.1-87.12 (2016).

11) S. Xie, R. Girshick, P. Dollár, Z. Tu and K. He, "Aggregated Residual Transformations for Deep Neural Networks", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1492-1500 (2017).

12) V. Ashish, S. Noam, P. Niki, U. Jakob, J. Llion, N. Aidan, K. Łukasz and P. Illia, "Attention is All you Need", *In Advances in Neural Information Processing Systems*, 5998–6008 (2017).