Binarized-DCNN による識別計算の高速化

EP11040 神谷龍司　　指導教授：藤吉弘亘，山下隆義

1. はじめに

Deep Convolutional Neural Network(以下 DCNN)[1] は，音声認識や画像認識のベンチマークにおいてトップの認識性能を達成したことが報告されている．畳み込み層や全結合層において実数同士の演算を多大に行う必要があるため，識別処理に時間を要するという問題がある．そこで，本研究では DCNN の畳み込み処理と全結合処理に，近似計算を導入することで識別の高速化を実現する．

2. DCNN と計算量

DCNN は畳み込み層，プーリング層，全結合層から構成されるニューラルネットワークである．識別時は，まず入力画像に重みフィルタの畳み込み処理を施して特徴マップを得る．プーリングは，特徴マップの小領域から最大値を新たな特徴マップに配置する．全結合層では，結合重みとの内積計算を行い，softmax 関数により識別判定を行う．

DCNN は畳み込み層や全結合層において，実数同士の演算を行うため識別処理に膨大な時間を要する．畳み込み層を 3 層，全結合層を 1 層，フィルタサイズを 13 × 13，全結合層のユニット数を 400 としたとき，畳み込み層の内積計算は 8,782,592 回，全結合層の内積計算は 55,200 回必要となる．

3. Binarized-DCNN

本研究では，バイナリ演算による近似計算を導入することで識別処理を高速化した Binarized-DCNN を提案する．畳み込み処理では，重みフィルタを重み付き矩形フィルタの線形和で表現することで，高速化を実現する．全結合処理では，結合重みに対してベクトル分解法を適用して，バイナリ演算による近似計算をすることで高速化を実現する．

3.1. 畳み込み処理の高速化

畳み込み処理は，入力画像に対して重みフィルタを畳み込むため，膨大な処理時間を必要とする．そこで，畳み込みで使用する重みフィルタを D-BRIEF[2] で用いられたフィルタ近似アルゴリズムにより矩形フィルタに分解する．重みフィルタは矩形フィルタの線形和で表現できる (図 1 参照)．これにより，矩形フィルタに積分画像を用いることで高速な畳み込み処理が可能となる．

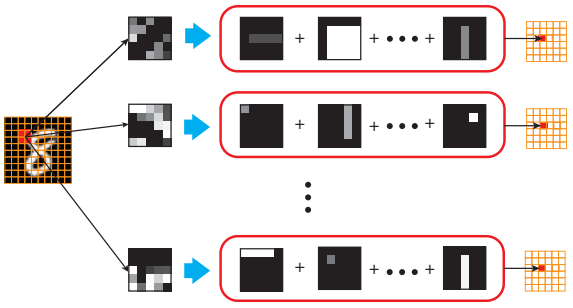

図 1：フィルタ近似アルゴリズムを導入した畳み込み処理

3.2. 全結合処理の高速化

全結合層の j 番目のユニットにおける出力は，入力 $\mathbf{x} = \{x_1, x_2, ..., x_I\}^{\mathrm{T}}$ と結合重み $\mathbf{w}_j^{\mathrm{T}} = \{w_{1j}, w_{2j}, ..., w_{Ij}\}$ の内積計算により得られる．入力 $\mathbf{x}$ と結合重み $\mathbf{w}$ は共に実数であるため，内積計算に処理時間を要する．そこで，全結合層の全てのユニットにおける結合重みに対してベクトル分解法 [3] を適用する．全結合処理にベクトル分解法を適用するには，ユニットへの入力 $\mathbf{x}$ をバイナリ値で表現する必要がある．そこで，学習時にシグモイド関数を傾きを制御するパラメータを用いてステップ関数に近づけることで，識別時の出力をバイナリ化する．結合重み $\mathbf{w}$ に対してベクトル分解法を適用することで図 2 のように二値基底行列 $\mathbf{M} = \{\mathbf{m}_1, \mathbf{m}_2, ..., \mathbf{m}_k\}^{\mathrm{T}}$ と実数のスケール係数ベクトル $\mathbf{c} = \{c_1, c_2, ..., c_k\}^{\mathrm{T}}$ に分解できる．ここで，k は基底数を示している．分解後は，バイナリ演算となるため，高速な識別計算が可能となる．

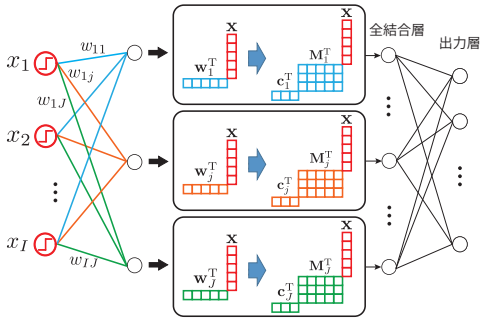

図 2：ベクトル分解法を適用したときの全結合処理

4. 評価実験

Binarized-DCNN の有効性を評価実験により示す．

4.1. 実験概要

本実験では，畳み込み処理と全結合処理の認識率と処理時間を調査する．データセットは畳み込み処理に Labeled Faces in the Wild Dataset(LFW)，全結合処理に CIFAR-10 を用いる．LFW は画像サイズ 100 × 100，DCNN の構造は畳み込み層 3 層，全結合層 1 層使用し，13 × 13 のフィルタサイズを全層において使用する．また，CIFAR-10 は画像サイズが 32 × 32，畳み込み層 3 層，全結合層 1 層使用し，全層において 5 × 5 のフィルタサイズを使用する．

4.2. 実験結果

畳み込み処理と全結合処理の実験結果を示す．

・畳み込み処理

図 3 に畳み込み処理における認識率と処理時間の比較を示す．Binarized-DCNN では，矩形フィルタ数が 20 枚のとき，約 2.0 倍の高速化を実現した．

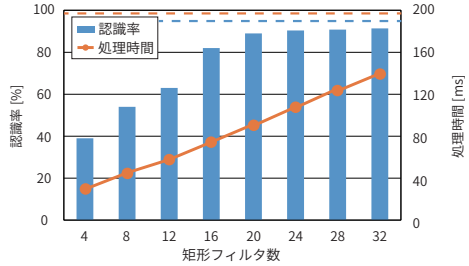

図 3：畳み込み処理の矩形フィルタ数による比較

・全結合処理

図 4 に全結合処理における認識率と処理時間の比較を示す．図 4 より，基底数 10 の場合において約 7.1 倍の高速化を実現した．

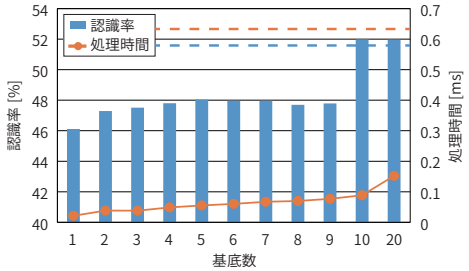

図 4：全結合層の基底数による比較

5. おわりに

Binarized-DCNN では，DCNN における畳み込み層に対して約 2.0 倍，全結合層に対して約 7.1 倍の高速化を実現した．今後はさらなる高速化を目標とする．

参考文献

[1] Y. Lecun, *et al.*, “Backpropagation applied to handwittern zip code recognition”, Neural Computation, 1989.

[2] T. Tomasz, *et al.*, “Efficient Discriminative Projections for Compact Binary Descriptors”, ECCV, 2012.

[3] S. Hare, *et al.*, “Efficient online structured output learning for keypoint-based object tracking”, CVPR, 2012.