

CNC による 2 値文字画像認識

Binary Character Recognition Using Cascaded Neuro-Computational Model

○平野 悠子¹, 保谷 哲也²

*Yuko Hirano¹, Tetsuya Hoya²

Abstract: Cascaded neuro-computational model (CNC) is a neural network model originally proposed for speech recognition. In this paper, we investigate the applicability of CNC to image recognition and report the simulation results of binary character recognition using CNC.

1. 概要

画像の分類および認識の分野に 2 値画像認識がある。本研究では 2 値文字画像の認識を行う上で、cascaded neuro-computational model(CNC)モデルにハミング距離の概念を取り入れたモデルを使用して実験を行い、認識率およびデータ量を PNN と RAN(修正版)を用いた場合と比較した。元来 CNC は音声認識の分野で提案された心理学的アプローチを用いたニューラルネットワークモデルである。

2. 2 値画像認識

2 値画像認識とは、2 値すなわち 1 と 0 または黒と白のみで表される 2 値画像の認識である。2 値画像の認識では、通常、元画像の正規化や特徴抽出を行った後、画像間の距離または類似度を計算することにより分類を行う。そのような距離の計算法として主に、1) 2 画像間のある範囲で一致する 1 の個数を加算して求める内積を用いる手法、2) 一致しない 1 と 0 の合計の個数を加算して表すハミング距離^[1]を用いる手法、および 3) 1 と 0 の一致や不一致に対し乗算を用いる相関係数を用いる手法、の 3 つがある^[2]。このような距離に対し加算や乗算を用いたモデルによって分類する^[2]。

3. PNN と RAN(修正版)

PNN^[3]を 2 値画像に用いた場合、画像全体を 1 つのデータとし 1 ピクセルごとの差(ハミング距離と等しい)を用いた RBF

$$h_i = \exp\left(-\|x - c_j\|_2^2 / \sigma^2\right) \quad (1)$$

の加算平均によりテストデータを分類する、といったことが考えられる。一方、RAN の修正版^[4]では PNN と同じ構造および RBF が用いられている。この場合は PNN と異なり、学習データを入力する際に、現時点でのモデルを用いて正しく分類できる学習データに対応するユニットは構築されない。

4. CNC(cascaded neuro-computational model)

CNC は神経構造を基にした 3 層構造のニューラルネットワークである^[5]。Figure 1 のように CNC の Layer1 は RBF ユニット、Layer2 は単語候補ユニット、Layer3 は単語ユニットから構成される。CNC を用いた 2 値画像認識では、まず 1 つの画像を一定の方法で分割し、分割されたデータ 1 つを”1 まとまり”とする。Layer1 では 1 まとまりを 1 つのユニットとするが、学習時に既存のユニットとの RBF を計算し、発火したものは Layer1 ユニットとして新たに追加されない。一方、Layer2 のユニットは 1 画像の該当するデータに対し最大発火するユニットに付加されるラベルの集合から構成されるが、学習時に既存のユニットで正しい認識結果になるものは Layer2 ユニットとして新たに追加されない。Layer3 のユニットは出力としてあり得る全ての分類結果に対し 1 つずつ構成され、Layer2 からの出力を各分類結果ごとに集計することにより算出される。

5. ハミング距離を適用した RBF

本研究では、画像を CNC で認識するため、以下のように従来の CNC における Layer1 の RBF を改良した：データを読み込むとき、従来の CNC では 1 データごとに計算を行うので 1 または 0 を 1 データとして読み込むことになる。しかしながらこのような手法では距離を適切に計算することが出来ないので、1 または 0 のデータの集合を、各桁が 1 または 0 をとる 2 進数の集合に変換する必要がある。本論文では、このように変換し 1 データにしたものを 2 値ブロックと呼ぶ。そのような変換は、1 と 0 の画像をブロックにより分割し 2 進数として扱うことで行う。なお、実験ではこの 2 値ブロックのサイズを(1×4, 1×8, 2×2, 4×4)の 4 種類を用いて比較した。また、従来の CNC で用いられる式(1)の代わりに、本研究では式(2)を用いる。なお、式(2)において D(a,b)は a と b のハミング距離を意味する。

$$h_i = \exp\left\{-\left(\sum_j D(x, c_j)\right)^2 / \sigma^2\right\} \quad (2)$$

1 : 日大理工・院(前)・数学 2 : 日大理工・教員・数学

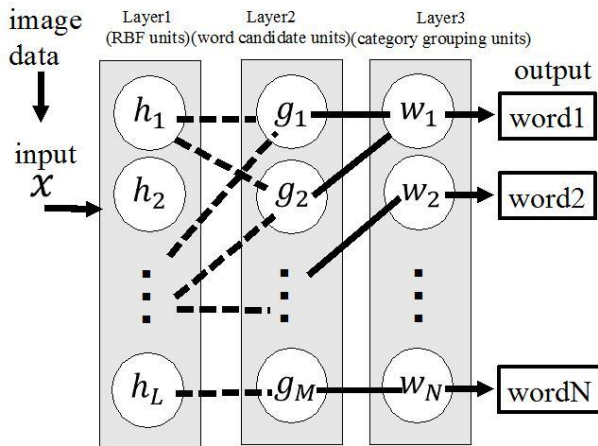


Figure 1. Cascaded Neuro-Computational Model

6. シミュレーション実験

実験では Optdigits-orig(The University of California at Irvine(UCI) Machine Learning Repository で配布されている 32×32 のサイズの 2 値数字画像 0~9 のデータセットであり, サイズと傾きの両方について正規化が成されている)^[6]を用いた. また, CNC のパラメータ θ は 0.7 で固定し, σ は[7]を参考に $0.6 \times \sigma \max$ を上限とした.

さらに, 実験では 2 値ブロック及び Layer1 の読み取り方(1 まとまりの形)により場合分けを行った. 2 値ブロックのサイズは全部で 4 通り(1×4, 1×8, 2×2, 4×4)あり, ブロックが正方形でない場合には向きを変え 2 方向から読み取るようにした. つまり, 2 値ブロックが正方形でない場合には, それぞれ正方形であるものの 2 倍のデータを読み込む. 一方, Layer1 の読み取り方は 9 通り(Table 1 参照)あり, 実験ではそれぞれ平行 1 行・垂直 1 行は圧縮方向と平行・垂直に 1 行分読み込み, $h \times h$ は正方形のブロック(一辺の長さ h)として読み込んだ. ここで, $h \times h(2 \text{ 回})$ とは $h/2$ ずつ右下に画像をずらして 2 回目の読み込みを行った場合を指す.

7. 実験結果

実験結果をまとめたものが Table 1 である. 全体的に読み込み回数を増やしたパターンが認識率が高くなっている. さらに, 認識率が高い 3 パターンの認識率とデータ量を PNN および RAN(修正版)とを比較したのが Table 2 である. なお, Table 2 の下 2 行は PNN および RAN(修正版)にも 2 値ブロックおよび式(2)を適用して実験した場合である. 実験では CNC を用いた場合, 認識率は PNN および RAN(修正版)より低くなった. 一方, データ量は最も認識率が良かった場合のデータ量(33448)と比較して, PNN の約 3/20, RAN(修正版)の約 1/5 のデータ量に削減できた. ただし, 2 値ブロック適用の RAN(修正版)と比較するとデータ量は逆に多くなった.

8. 結論

本研究では, CNC を用いた 2 値画像認識では, データ量を減少させつつ 90%以上の認識率の保持が可能であることが分かった. 今後は, ハミング距離以外を用いた RBF を用いた実験や, 2 値画像よりもデータ量が多いカラー画像への拡張について検討する予定である.

		平行1行	垂直1行	1×1	2×2	4×4	8×8	2×2(2回)	4×4(2回)	8×8(2回)
2方向	1×4	91.33	87.00	91.12	89.43	90.91	78.54	91.65	93.02	91.65
	1×8	92.81	81.71	92.60	93.34	86.89		92.71	93.66	
1方向	2×2	90.38	同左	86.79	84.99	76.64	80.87	87.00	86.26	84.04
	4×4	86.58	同左	84.04	76.43	82.35	90.17	86.79	85.31	

Table 1. 認識結果のまとめ

	読み込み方法		Layer1	Layer2			
	2値ブロック->Layer1	認識率	長さ	数	長さ	数	データ量
CNC	1×8->4×4(2回)($\sigma=18.2$)	93.66	16	232	72	413	33448
	1×8->2×2($\sigma=10.0$)	93.34	4	31	64	307	19772
	1×4->4×4(2回)($\sigma=15.3$)	93.02	16	9	108	493	53388
比較	PNN	98.94	1024	1934			1980416
	RAN	96.93	1024	151			154624
参考	PNN(2値:4×4)	99.15	64	1934			123776
	RAN(2値:4×4)	96.41	64	143			9152

Table 2. PNN および RAN との比較

9. 参考文献

[1] R.V.Hamming : “ Error detecting and error correcting codes, ” Bell Sys. Tech. Journal, vol.29, No.2, pp.147-160, 1950.

[2] Sung-Hyuk Cha, Sungsoo Yoon, and Charles C. Tappert : “ Enhancing Binary Feature Vector Similarity Measures, ” Journal of Pattern Recognition Research 1, pp.63-77, 2006.

[3] D.F.Specht : “ Probabilistic neural networks, ” Neural Netw., vol.3, pp.109-118, 1990.

[4] Tetsuya Hoya, Yoshikazu Washizawa : “ Simultaneous Pattern Classification and Multidomain Association Using Self-Structuring Kernel Memory Networks, ” IEEE Trans. Neural Networks, vol.18, No.3, pp.732-744, 2007.

[5] Tetsuya Hoya and Cees van Leeuwen : “ A cascaded neuro-computational model for spoken word recognition, ” Connection Science, vol.22, pp.87-101, Feb. 2010.

[6] Arthur Asuncion, David Newman : “ UCI Machine Learning Repository, ” 2007, Address : www.ics.uci.edu/~mlern.

[7] Tetsuya Hoya : “ On the Parameter Setting of A Network-Growing Algorithm for Radial Basis Kernel Networks, ” in proc. of 2016 Joint 8th International Conference on Soft Computing and Intelligent Systems and 17th International Symposium on Advanced Intelligent Systems, pp.355-359, 2016.