ノート

ディープラーニングを用いた画像認識

福嶋　祐一*　石澤　賢太*　大野　宏*

Image recognition using deep learning

FUKUSHIMA Yuichi*，ISHIZAWA Kenta*　and　OHNO Hiroshi*

1.　緒　　言

近年，人工知能の一分野であるディープラーニングが注目を集めており，産業利用が盛んになっている。特に，画像認識分野では従来技術よりも高い認識精度を得られることもあり，実用化が進んでいる。県内企業からのニーズも高く，注目されている技術分野である。

自社製品や装置にディープラーニングを組み込む場合，無償で公開されているツールを利用することで，その導入費用を抑えることができる。しかし，最近，多くのツールが公開され，どのツールを使用すべきかを判断しにくいのが現状である。

本研究では，タイヤに記載されているメーカ名とタイヤサイズの認識を題材とし，無償で公開されているツールである Caffe と YOLO を用いて認識を試みた。また，認識率を上げるためディープラーニングの学習用データを工夫した。

2.　ディープラーニングのツール

2.1　ディープラーニングとツールの概要

ディープラーニングの基となるニューラルネットワークは，生物の脳神経網を模した構造をしており，1943年にMcCullochとPittsが単一の神経細胞の動作を計算機上でモデル化したところから始まった。[1]1998年には，誤差逆伝搬法による学習を取り入れた畳み込みニューラルネットワーク（CNN：Convolutional Neural Network）がLeCunらによって提案された。[2]その後もアルゴリズムが改良され，2011年頃から有用な研究成果が報告され始めた。

この頃から徐々に，研究用に開発されたディープラーニングのツールが無償で公開されるようになった。それまでは，研究者が一からプログラムをする必要があったが，これらのツールを誰でも使えるようになり，研究開発に取り組みやすくなった。代表的なツールを表1に示す。今回はC++で開発され処理速度が速く，高精度の認識を期待できるCaffeとYOLOを使用した。

2.2　Caffe

Caffe[3]は，カルフォルニア大学バークレー校で開発されたツールである。画像認識分野でディープラーニングが注目され始めた頃から広く使われているが，近年は後発のツールに押され利用者は減っている。

特徴としては，学習とテストが簡単で，高速かつ高い精度で画像認識が可能なことである。

表 1　ディープラーニングツール（無償）

ツール名	言語	開発元
Caffe	C++ Python	カルフォルニア大学
Chainer	Python	Preferred Networks
TensorFlow	Python (C++)	Google
Keras	Python (C++)	Google
Pytorch	Python	Facebook
YOLO	C++	J. Radmon氏
Neural Network Console	—	SONY

*　中越技術支援センター

また，インターネット上に多数の参考事例が公開されており情報を得やすい。ただし，CaffeではCNNで画像を認識するため，1枚の画像で認識できる対象は1つだけである。そのため，画像中に複数の対象物がある場合，画像処理を行い複数の対象物をそれぞれ1枚の画像に切り分ける前処理が必要になる。

2.3 YOLO

YOLO（You Only Look Once）[4]は，一般物体検出と呼ばれる手法で，物体の認識と同時にその物体が位置している領域も同時に検出することが可能である。さらに，Caffeとは異なり，1 枚の画像に複数の対象物が存在しても，その領域と対象物の認識が可能である。YOLOは，一般物体検出の手法の中でも高速な処理が可能であり，開発者が改良を加え認識精度も高くなっている。

一般物体検出では，物体が位置している領域を検出するために，認識対象物の学習用画像に加えて，画像の中でその対象物がどこに位置しているかという領域の情報も必要になる。図1に学習時に必要なクラス名と領域を記載したテキストファイル例を示す。左からクラス番号，対象の中心位置の座標，対象の領域の大きさを示す。座標，大きさとも画像サイズを1とした時の値を示す。

3. 画像の取得

タイヤの文字はタイヤ表面の凹凸で表されているため，3次元センサ（ラインレーザとカメラ）で表面形状を取得し画像化した。試作した装置の外観を図2に示す。ターンテーブル上に模型のタイヤを載せ，ラインレーザをタイヤに当て斜めに設置したカメラで画像を取得する。タイヤを回転させながら連続して画像を取り込み，その稜線を明るさに変換して画像化したものを図3に示す。タイヤの表面は曲面になっているため地の色が場所により変わる。この影響を除去する動的2値化を利用してして得た2値画像のう

0	0.3593	0.2937	0.1453	0.3458
0	0.6093	0.3291	0.1453	0.3520
0	0.8281	0.2937	0.1437	0.3354
0	0.1250	0.7500	0.1343	0.3500
1	0.8281	0.2666	0.1312	0.3458
1	0.1156	0.6708	0.1343	0.3375
1	0.3656	0.6979	0.0843	0.3416
1	0.5875	0.6583	0.0765	0.3354

図1　学習に必要なテキストファイル例

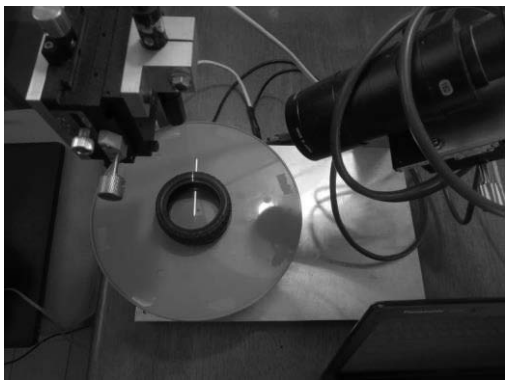

図2　試作した装置の外観

図3　タイヤの凹凸の画像化

図4　2値画像（タイヤサイズ）

図5　2値画像（メーカ名）

（a）chars74k の文字

（b）縁を凸状にした文字

図6　学習とテストに使用したデータ例

ち，タイヤサイズを図4に，メーカ名を図5に示す。この画像をディープラーニングを使って認識する。なお，本装置はカメラのレンズの倍率を変えることで，タイヤの微小の凹凸も測定可能である。

4.　学習と認識

4.1　データの準備

タイヤサイズとメーカ名を認識するために，chars74k[5]というデータセットを利用した。これは，0～9，A～Z，a～zの62クラスからなるデータセットで，1つの画像に1つの文字が書かれており，様々なフォントと手書き文字が用意されている。メーカ名の文字は，図5に示すとおり，文字の縁が凸状になっているためchars74kの文字と異なる。そのため，画像処理ライブラリOpenCVを使い，chars74kの文字を変形させて，縁が凸状になる文字を作成した。chars74kの文字を使い学習とテストに使用したデータ例を図6に示す。学習用にはchars74kの中でタイヤデータに近いものを100個選び，若干サイズを拡大縮小し，タイヤの文字は前後に傾いているので，傾けた画像を作成し，画像を水増しさせた。作成した画像のうち，半分を学習用に，残りの半分をテスト用の画像として使用した。メーカ名とタイヤサイズでは文字の形状が異なるため，学習とテスト用のネットワークモデルを別々に作成した。

ディープラーニングの学習・テスト用の画像データは，カメラで実際の画像を撮像して使う方法が一般的であるが，この作業は大変で時間がかかる。今回は，chars74kというデータセットを利用し，これらをタイヤ文字に近い形状にプログラムで変形させて使用し，省力化をはかった。

4.2　Caffe

Caffeで認識するためには，学習とテスト用の画像ファイル，学習とテストそれぞれの画像ファイル名とクラス名を記載したファイル，CNNのネットワーク構造を記載したファイル，学習回数や学習率などを記載したファイルを用意する。タイヤサイズ用の画像には0～9の数字の他に，「/」，「R」，「Z」の13種類の画像を用意した。また，メーカ名の学習用画像には，26個のアルファベットの大文字を使用した。

ネットワークは，手書き文字認識用のLenetを改良して使用した。これは，畳み込み層，プーリング層,全結合層が各2層ずつと比較的小規模にもかかわらず高い認識率が得られるネットワークで，入力画像サイズは28×28と小さい。ここでは128×128の画像を入力しているため，畳み込み層のサイズを大きくした。学習用画像に各クラス1000枚,テスト用画像に各クラス1000枚を用意した。これらの画像を使って学習を行い，定期的にテスト用画像で認識率を計算し，95%を目途に学習を繰り返した。

この学習済みのモデルを使い，認識した結果を図7に示す。模型タイヤの画像を取得し，2値

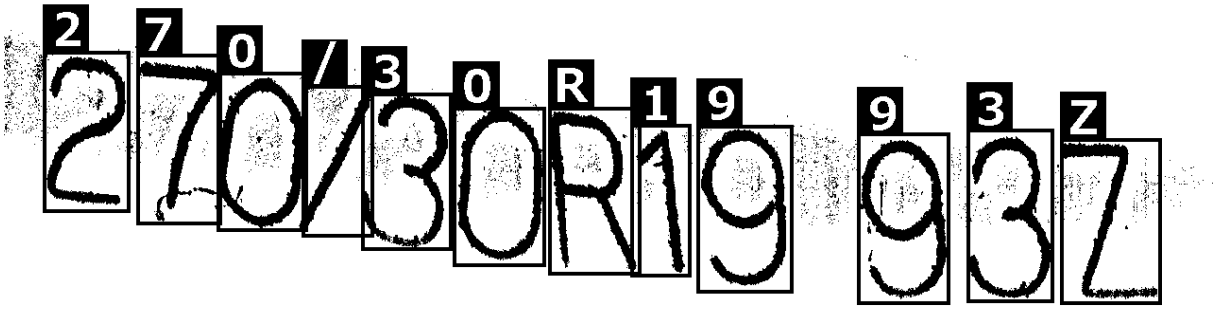

図 7　Caffe によるタイヤサイズの認識結果

図8　YOLOによるメーカ名の認識

化とラベリングを行い1文字ずつ切り出してCaffeで認識させた。文字の上の四角形に認識結果を表示したが，正しい結果が得られた。

4.3　YOLO

学習とテスト用にはCaffeと同じ画像を利用した。画像サイズは416×416とし，1枚の画像に9個の文字を入れ，これに合わせて先の図1に示すようなテキストファイルを作成した。Caffeと同様に，学習用画像に各クラス1000枚，テスト用画像に各クラス1000枚を用意した。これらの画像を使って学習を行い，テスト用画像の認識率が95%を超えるまで学習を繰り返した。

メーカ名を認識した結果を図8に示す。文字と文字の間隔が狭いせいか，2つの文字を1つの文字として認識してしまい，正しい認識が出来なかった。ただ，文字のある領域は四角形で正しく切り出されていたので，YOLOで切り出しを行いCaffeで認識するという方法も有効と思われる。

今回は模型のタイヤを対象にしたため，多数のサンプルで検証を行うことができなかった。実際には，より多くのデータで高い認識率が得られるか検証を行う必要がある。

5.　結　　言

（1）ラインレーザとカメラを使いタイヤ文字を画像化するセンサを試作した。

（2）ディープラーニングツールCaffeとYOLOを使い，タイヤサイズとメーカ名を認識させた。Caffeでは良い結果が得られた。

（3）既存のデータセットを元に画像処理ライブラリを使って実際のタイヤ文字に近いデータを作成し，学習と認識を行った。

参考文献

1）W. S. McCulloch and W. Pitts, “A logical calculus of the ideas immanent in nervous activity”, *Bulletin of Mathematical Biophysics*, Vol.5, No.4, 1943, pp.115-133

2）Y. LeCun, et.al, “Gradient-based Learning Applied to Document Recongnition”, *Proceedings of the IEEE*”, Vol.86, No.11, 1998, pp.2278-2324

3）Y. Jia, et.al, “Caffe: Convolutional architecture for fast feature embedding”, *Proceedings of the 22nd ACM international conference on Multimedia*, 2014, pp.675-678

4）J. Redmon, et.al, “You only look once: Unified, real-time object detection”, *The IEEE Conference on Computer Vision and Pattern Recongnition (CVPR)*, 2016, pp.779-788

5）T. E. de Campos, et.al, “Character recognition in netural images”, *Proceedings of the International Conference on Computer Vision Theory and Applications (VISAPP)*, 2009