

スケッチした pictorial symbols の深層学習のための 学習形状生成手法における洗練化の効果

佐藤 信¹

概要: スケッチした pictorial symbols の深層学習 (deep learning) に用いるための既提案の学習形状生成手法について、学習形状の洗練化の効果を検討する。既提案手法では、少数の形状を基準とし、手描き風の類似形状を生成する。そして、生成した形状の特徴に基づき、学習形状の集合を反復的に洗練化する。特徴の学習には深層学習を用いる。本稿では、形状分類の精度および統計的検定を用いて洗練化の効果を検討する。生成した形状を学習することにより、ストロークの数および順序に依存しない、多様な手描き形状によるインタフェースを構築することが可能である。

Effects of Refinements in Learning Shape Generation Method for Deep Learning of Sketched Pictorial Symbols

MAKOTO SATOH¹

Abstract: This paper discusses about effects of learning shape refinements in a proposed learning shape generation method for deep learning of sketched pictorial symbols. In the proposed method sketchy shapes similar to base shapes are generated from a small set of base shapes. Then the generated learning shapes are iteratively refined based on shape features. The shape features are learned using a deep learning method. In this paper the effects of the refinements are discussed in terms of shape classification accuracy and a statistical test. Human computer interfaces using diverse sketchy shapes, with free stroke-number and stroke-order, will be able to be constructed with machine learning of the generated shapes.

1. はじめに

本稿では、深層学習 (deep learning) に用いる学習形状を生成するための既提案手法 [15] について、学習形状の洗練化の効果を検討する。生成した形状を用いて、スケッチした pictorial symbols [17] を深層学習することが目的である。

私たちの身の周りでは、特定の言語に依存せずに情報を直感的に伝達することを目的として、多くの pictorial symbols が用いられている。そのことにより pictorial symbols は、視覚的な言語としての役割を担っているといえる。例えば、案内図あるいは各種のダイアグラムにおいて用いられる簡潔な図形要素などの一般的なものから、UML ダイアグラムに用いられる図形要素あるいは音符などのようなある特定

の分野に関連するものまで、非常に多くの種類の pictorial symbols が存在する。それらの pictorial symbols を深層学習などのアルゴリズムにより取り扱うことができれば、多くの役に立つシステムを創造することが可能となる。そのようなことを目的として、pictorial symbols の深層学習に用いる学習形状を生成するための研究をおこなっている。

既提案手法では、予め与えた少数の形状を基準とし、基準形状に多様な類似性をもつスケッチ風の学習形状 (ラフ類似形状) を生成する。そして、生成した形状の特徴を深層学習を用いて学習し、形状特徴に基づいて学習形状の集合の反復的な洗練化をおこなう。本稿では、その洗練化の効果、形状分類の精度、および統計的検定手法である McNemar 検定 [16] を用いて検討する。

これ以降の構成について、簡単に説明する。2 節では、関連研究との比較をおこない、既提案手法および洗練化の

¹ 岩手大学
Iwate University, Ueda, Iwate 020-8551, Japan

効果の評価手法について概要と特徴を述べる。そして、3節では、深層学習を用いて学習形状の集合を洗練化するための既提案手法および洗練化の効果の評価手法について述べる。実験結果および検討について4節で述べる。そして最後に、5節で本稿のまとめと今後について述べる。

2. 関連研究と評価手法の概要

2.1 手描き形状およびストロークの多様性

始めに、スケッチした形状を分類するためのアルゴリズムの設計において考慮されるべき点について述べる。

形状の多様性 自然画像に含まれる物体の形状と比較すると、スケッチした形状は形状の変化が大きいといえる。そのため、多様性のある類似形状を分類することが可能である必要がある。

ストロークの多様性 ヒューマン・コンピュータ・インタラクション (HCI) のためのインタフェースとしてスケッチした形状を用いる場合には、形状を描くストロークの数または順序が多様であることへの対応が可能である必要がある。

スケッチした形状が多様である原因として、人手により描いた形状であることを挙げることができる。そしてそれに加えてスケッチした形状に特有の原因として、Eitz 等 [5] が述べているように、形状を描く作者による対象の捉えかたが、最終的に描かれる形状に大きく影響を与えるということがある。

次に、ストロークの数および順序の多様性について述べる。紙と鉛筆を用いて形状を描く場合には、ストロークの数または順序について気にせず、自由に形状を描くことが多い。一方、ジェスチャの分類のような HCI のインタフェースのためのストローク形状の分類アルゴリズムでは、pictorial symbols の一種である矩形などの簡単な形状の分類をおこなう場合であっても、ストロークの数または順序に制約を設ける場合が多い。スケッチ形状の分類を用いたインタフェースをユーザフレンドリにするためには、紙と鉛筆で描くように自由に描ける必要がある。しかしながら実際には、ストロークに制約のないアルゴリズムを設計することは難しいことである。

少数ではあるが、Hammond 等 [8] のようにストロークの数または順序の多様性についての制約を緩和したアルゴリズムが提案されている。それらは、ストロークの数または順序の多様性を取り扱うために複雑なアルゴリズムとなっている。そして、形状の多様性に対応するための処理を形状の認識過程においておこなっているために、対話的な用途に用いるには計算量が大きいアルゴリズムとなっている。そのようなアルゴリズムでは、認識過程での計算量を減少させ、形状認識の速度を改善することが課題とい

える。

これらの理由により、実用的なアプリケーションでは、ストロークに制約を設けたアルゴリズムを用いることにより計算量を減少させることが多い。そして、そこで用いられる形状認識のための特徴量は、アルゴリズムの設計者が着目した特徴についての特徴量であることが多い。そのため、その特徴量が対応できる範囲では効率的に形状認識が可能であるが、特徴の多様性あるいは変化に対しては頑健性が低いといえる。

ここまで述べたことを念頭に、形状およびストロークの多様性への対応が可能な手法について述べる。形状の変化の多様性に対応するためには、分類する形状の特徴を学習することが可能な手法が有効であるといえる。特に、形状の特徴に追加がある場合、または、特徴が変化した場合であっても、新しい特徴を学習することにより頑健な分類が可能となる。そして、ストロークの数および順序に制約を設けずにストロークの多様性に対応して形状を分類するためには、ストロークをレンダリングした画像を分類に用いる手法が考えられる。

ストロークによる形状画像を学習により認識するための研究には、Kara 等 [11] がある。ここでは、認識する各形状クラスについて1つの形状を入力として与え、形状認識をおこなう手法を提案している。局所的な幾何学的特徴量を用いることにより、Hausdorff 距離などを用いた場合と比較すると、形状の移動、回転および均一なスケールングに対して頑健な形状認識が可能であることが示されている。しかし、形状の非均一なスケールングに対しては頑健な形状認識は実現できていない。また、各形状クラスについて1つの形状のみを学習していることから、形状の多様性についての頑健性が低いといえる。

2.2 学習アルゴリズムの性能評価

学習アルゴリズムの性能を評価するために、多くの評価手法が提案され用いられている。複数の評価手法が存在することにより、アルゴリズムの性能を多角的に検討することが可能となっている。

学習アルゴリズムの一種である分類アルゴリズムの性能評価には、テストデータに対する分類精度を用いることが多い。それに加えて、統計的検定をおこなうことにより、分類精度の頑健性を検討することがある。統計的検定をおこなうことにより、テストに用いていないデータに対して同様の分類精度が得られるのかどうかについて、研究者が共通に理解している統計的な概念を用いて客観的に検討することが可能となる。Wasserstein [22] で述べられているように、どのように統計的検定を用いるのかは重要なことであり、多角的に検討をおこなうなかのひとつとして、統計的検定を客観的に用いることが大切である。

Dietterich [3] では、分類アルゴリズムおよび分類器の性

能を比較することを目的として、複数の統計的検定手法について比較をおこなっている。そこでは、分類アルゴリズムの性能比較に用いるのに適当な統計的検定手法として、Dietterich [3] で提案している検定手法である 5x2cv paired *t* test および McNemar 検定 [16] を挙げている。どちらの手法を用いるべきかは、学習アルゴリズムの訓練のために必要な計算量に依存すると述べている。

Adrian 等 [2] は、コンピュータビジョン分野でのアルゴリズムについての性能評価手法について述べている。ここでは、アルゴリズムの頑健性の検討のために McNemar 検定を用いている。

2.3 形状の深層学習と学習形状 augmentation

スケッチした形状がもつ多様な変化に頑健な形状分類をおこなうためには、形状の学習が有効である。

Bishop[1], p.320 では、変形に対して不変 (invariant) な学習をおこなうための手法をいくつか挙げている。そのひとつとして、形状の変化に対応する形状変換をおこなった形状を用いて学習をおこなう手法について述べている。スケッチした形状の多様性を学習形状により表現するためには、大量の学習形状が必要となる。

Hinton 等 [9] が示すように、深層学習は、大量の学習形状から特徴を紡ぎだす性能に優れた学習手法である。そのため、形状を学習するために深層学習を用いる研究が盛んにおこなわれている。しかし、それらの研究の大部分は、手描き数字の画像あるいは風景画像のパッチなどを対象とするものである。一方、スケッチした形状を対象とする研究は少数である。例えば、Fu 等 [6] では、少数の与えられた線図形を基に生成した類似形状を学習形状として、スケッチした形状の深層学習をおこなっている。ここでは、Simard 等 [19] の提案による画像の弾性変形手法を用いて形状 augmentation(増補)をおこなうことにより、スケッチ風の類似形状を生成し学習形状として用いている。

形状 augmentation とは、学習形状の個数をアルゴリズムを用いて増加させることにより、学習形状の不足を補うことである。Eitz 等 [5] が示すように、学習形状の収集には多くの時間と労力を必要とする。そのため、学習形状を収集するための労力を削減することを目的として、形状 augmentation をおこなうことがある。形状学習のために augmentation をおこなっている研究の例としては、Fu 等 [6], Krizhevsky 等 [12], Schmidhuber[18] がある。

2.4 既提案手法および評価手法の概要

既提案手法 [15] では、スケッチした pictorial symbols の深層学習に用いるための形状を生成することを目的として、学習形状の集合の洗練化をおこなう。少数の与えられた線図形に形状 augmentation をおこなうことにより、ラフ類似形状を学習形状として生成している。ラフ類似形状

とは、基準とする形状に多様な類似性をもつスケッチ風の形状である。形状 augmentation には、Simard 等 [19] の提案による画像の弾性変形手法を拡張した手法 [13], [14] を用いている。形状を構成する各形状グループに異なった弾性変形をおこなうことにより、より多様性のあるスケッチ風の類似形状を生成することが可能であるように拡張をおこなっている。例えば、拡張によりストロークの交差の形状を変化させることが可能となっている。

本稿でおこなう既提案手法 [15] での学習形状の集合の洗練化の効果の検討では、形状分類の精度、および統計的検定手法である McNemar 検定 [16] を評価尺度として用いる。McNemar 検定は、対応のあるカテゴリカル変数の分析のために用いられる統計的検定手法である。ここでは、洗練化する前の訓練形状を用いた分類器と洗練化した後の訓練形状を用いた分類器とを McNemar 検定を用いて分析する。本稿での洗練化の効果の評価手法は、Dietterich [3] での分類アルゴリズムまたは分類器の性能評価の 9 つの分類に対応付けて考えると、分類器が 2 つ与えられて十分なテストデータがある場合 ([3] での 3 番目の場合) に変更を加えた場合に該当する。

2.5 適用分野

既提案手法を用いると、少数の pictorial symbols の形状を基にして、学習形状をアルゴリズムにより生成することが可能となる。そのことから、大量の学習形状を必要とする深層学習のような機械学習のために有用な手法であるといえる。

スケッチした pictorial symbols の深層学習による形状分類を HCI に用いることにより、ユーザフレンドリなアプリケーションの創造が可能になると考えられる。例えば、幾何学形状あるいはソフトウェアなどの設計では、設計の洗練化を繰り返す必要がある。スケッチ・インタフェースにより設計の洗練化が可能であれば、設計者は洗練化の本質的な部分に集中することが可能になる。ソフトウェアの設計においては、リファクタリングのための機能が IDE において提供されることにより、どこをどのようにリファクタリングすべきであるかという本質的な点に設計者が集中できるようになっている。同様に、スケッチ・インタフェースを用いてソフトウェアの構造の洗練化が可能になれば、ソフトウェアの構造のどこをどのように洗練化すべきなのかという本質的な部分に集中できるようになることが期待できる。

また、与えられた少数の形状から訓練形状を生成することが可能なので、分類するスケッチした形状のカスタマイズが容易になる。そのため、ある特定の分野において用いられる図形形状の分類に用いることなどが考えられる。

3. 深層学習を用いた学習形状の洗練化の効果

3.1 学習形状の洗練化アルゴリズム

これから評価をおこなう既提案のアルゴリズム [15] について概要を説明する. このアルゴリズムは, スケッチした pictorial symbols の深層学習に用いる学習形状の集合を生成し洗練化するためのものである. アルゴリズムは, 以下の部分により構成される.

類似形状の生成 少数の pictorial symbols の形状を基にして, 多様な類似性をもつ学習形状を生成する. 類似な形状として, 派生基準形状 (佐藤 [13]) およびラフ類似形状 (佐藤 [14]) を生成する.

学習形状の洗練化 深層学習を用いて生成した形状の特徴の学習をおこない, 学習形状の集合を洗練化する.

派生基準形状は, 基準形状の各部分形状に線形の形状変換をおこない生成した形状である. ラフ類似形状は, 基準形状または派生基準形状の各部分形状に非線形の形状変換をおこない生成した形状であり, 基準形状に多様な類似性をもつスケッチ風の形状である. ラフ類似形状の生成には, Simard 等 [19] の手法を拡張した弾性変形手法 [15] を用いている.

アルゴリズムの理解を確実にするために, 上記の学習形状の洗練化について補足する. 既提案のアルゴリズムの目的は深層学習に用いるための学習形状を生成することである. そのことを目的として既定案のアルゴリズムでは, アルゴリズムでの処理として, 形状特徴を把握するために深層学習を用いる. そして, その形状特徴に基づいて学習形状の洗練化をおこなう.

なお, pictorial symbols は, SVG 形式 [21] で表現されることが多いことから, SVG 形式で表現した pictorial symbols をアルゴリズムの入力としている. そして, SVG 形式を構文解析することにより基準形状を生成している. 生成された基準形状は, SVG 形式の構文により決定される部分形状により構成される. 詳細については, [14], [13], [15] において述べられている.

[15] で提案した学習形状の集合の洗練化手法を, アルゴリズム 1, 2 に示す. それぞれ, 全体の流れ, および, 学習形状の更新手法の説明である.

3.2 洗練化の効果の評価

これ以降で用いる表記について説明する. アルゴリズム 1 を用いて洗練化する前と後の学習形状の集合を, それぞれ, \mathcal{L}_B そして \mathcal{L}_A とする. そして, \mathcal{L}_B または \mathcal{L}_A を用いて訓練した分類器を, それぞれ, \mathcal{C}_B または \mathcal{C}_A とする.

洗練化の効果の評価手法を, 以下に示す.

Algorithm 1 Learning shape refinement

Given a small set of shapes represented in SVG format.
Parse each of the shapes into base shape set (\mathcal{B}), composed of Bezier curves that represent partial shapes.
Generate a derived shape set \mathcal{D}_c , composed of Bezier curves, from \mathcal{B} .
Generate a roughly similar shape set \mathcal{R}_c as image data, using partial elastic deformation, from \mathcal{B} and \mathcal{D}_c .
Construct a training shape set \mathcal{T} for training DNN (deep neural network) with $\mathcal{B}, \mathcal{D}_c$ and \mathcal{R}_c .
Set a maximum iteration count of the following loop to N .
 $n \leftarrow 0$
while $n < N$ **do**
 Train DNN with the training set \mathcal{T} .
 Updated training set \mathcal{N} is generated using trained DNN.
 $n \leftarrow n + 1$
 $\mathcal{T} \leftarrow \mathcal{N}$
end while
The training set \mathcal{T} has been refined.

Algorithm 2 Updating training set in algorithm 1

Generate a derived shape set \mathcal{D}_n , composed of Bezier curves, from \mathcal{B} .
Generate a roughly similar shape set \mathcal{R}_n as image data from \mathcal{D}_n .
Construct a test set \mathcal{V} for testing DNN with \mathcal{D}_n and \mathcal{R}_n .
Make a empty shape set for correctly classified shapes: \mathcal{C} .
Make a empty shape set for misclassified shapes: \mathcal{M} .
for all the shapes in \mathcal{V} **do**
 Classify the shape using the trained DNN.
 if the shape is correctly classified **then**
 Add the shape to shape set \mathcal{C} .
 else if the shape is misclassified **then**
 Add the shape to shape set \mathcal{M} .
 end if
end for
Update training set \mathcal{T} with $\mathcal{B}, \mathcal{D}_c, \mathcal{R}_c$, and \mathcal{M} .

Step1: 学習形状の洗練化 与えられた少数の pictorial symbols の形状を基に, 学習形状の集合を生成し, 洗練化する (\mathcal{L}_B および \mathcal{L}_A). ここでは, アルゴリズム 1 を用いる.

Step2: 分類器の訓練 \mathcal{L}_B または \mathcal{L}_A を訓練形状として分類器を訓練する (\mathcal{C}_B または \mathcal{C}_A).

Step3: テスト形状の分類 同じテスト形状の集合を用いて, \mathcal{C}_B および \mathcal{C}_A によるテスト形状の分類をおこなう.

Step4: 分類精度の計算 Step3 での分類結果を用いて分類精度を計算する.

Step5: McNemar 検定 そ Step3 での分類結果について McNemar 検定により統計的検定をおこなう.

ここで, これ以降でおこなう統計的検定の枠組みについて述べる. 学習形状の洗練化の効果を評価しようとしていることから, 検証する仮説は以下ようになる.

帰無仮説 \mathcal{L}_B と \mathcal{L}_A には統計的に差がない。
 対立仮説 \mathcal{L}_B と \mathcal{L}_A には統計的に差がある。

ここで、洗練化の前と後との学習形状について直接的に比較するのではなく、洗練化の前と後との学習形状を用いて訓練した各分類器の性能を統計的に比較することを考える。その場合に検証する仮説は以下のようになる。

帰無仮説 \mathcal{C}_B と \mathcal{C}_A には統計的に差がない。
 対立仮説 \mathcal{C}_B と \mathcal{C}_A には統計的に差がある。

なお、McNemar 検定の枠組みにあわせるために、テスト形状の集合として同一の集合を用いて分類をおこなった結果について McNemar 検定をおこなう。

4. 実装と結果の検討

3 節で提案した手法を用いて、形状の洗練化の効果について検討した。

4.1 学習形状の生成

pictorial symbols の例として図 1 に示す形状を用いた。三角形そして矩形を構成する部分形状の数は、それぞれ 3 そして 4 である。三角形そして矩形が所属する形状クラスを、それぞれクラス 1 そして 2 とした。始めに、これらの形状の基となる形状が定義されている SVG 形式ファイルの構文解析をおこない、部分形状ごとに Bezier 曲線の集合(基準形状)を生成した。そして、その Bezier 曲線をラスタライズすることにより、図 1 に示す画像形状を生成した。なお、形状の画像サイズは 30×30 ピクセルである。その画像の中央の 20×20 の領域に形状をラスタライズした。

各基準形状から、派生基準形状およびラフ類似形状を生成した。各基準形状から生成した派生基準形状の数は、それぞれ 1500 である。各基準形状そして各派生基準形状から生成したラフ類似形状の数は、それぞれ 10 である。生成した派生基準形状およびラフ類似形状の一部分を、図 2 に示す。図の左半面が三角形であり、右半面が矩形である。各領域の最も左側の列に示す形状が、派生基準形状である。各派生基準形状の右側の 10 列に示す形状が、その派生基準形状から生成したラフ類似形状である。派生基準形状の生成においておこなう形状変換のパラメータの決定には乱数を用いた。乱数の生成に用いたパラメータを表 1 に示す。表に示す平均値と標準偏差の乱数を生成し、それらのうちで表に示す最小値と最大値の間の値を、各形状変換のパラメータとして用いた。また、生成した形状を一意に識別可能にするために、形状にインデックスを付与した。

なお、部分的な弾性変形を用いたラフ類似形状の生成については、[14] において詳細が述べられている。



図 1 SVG 形式ファイルから生成した基準形状
Fig. 1 Base shapes generated from SVG format files: the left shape(shape class 1) is a triangle composed of 3 shape groups, the right shape(shape class 2) is a rectangle composed of 4 shape groups.

表 1 派生基準形状の生成で用いたパラメータ (乱数生成)

Table 1 Parameters for generating derived base shapes (random number generation)

transformation	mean	sigma	minimum	maximum
translation	0.0	1.0	-2.0	2.0
scaling	1.0	1.0	0.5	1.5
rotation ^a	0.0	1.0	-1.0	1.0

^a The angle is in radian.

4.2 学習形状の洗練化

形状特徴の学習には、DNN(deep neural network)を用いた。DNN の各層の要素数は、900, 196, 196, 196, 10 である。900 および 10 が、それぞれ入力層および出力層の要素数である。要素数が 196 である 3 つの層が、隠れ層である。クラス数は 2 であるが、出力層の要素数は 10 とした。

DNN の学習は、始めに各層ごとに事前学習をおこない、それからファインチューニングをおこなった。事前学習は、訓練形状を用いた教師なし学習である。そして、ファインチューニングは訓練形状と訓練ラベルを用いた教師あり学習である。DNN の事前学習には、自己符号化器(autoencoder)を用いた。なお、事前学習には、ドロップアウトを用いた。ドロップアウト率は 0.1 とした。DNN の学習で用いた訓練形状およびテスト形状には、4.1 節で生成した形状から選択した形状を用いた。繰り返し 1 での、訓練形状およびテスト形状の数は、それぞれ 4400 および 2200 である。各繰り返しで新たに選択するテスト形状は、2200 である。そして、各繰り返しで共通のテスト形状の数は、4422 である。それらの一部分を、図 3 に示す。なお、4.1 節で付与した形状のインデックスが重複しないように、訓練形状およびテスト形状を選択した。

4.3 分類精度および McNemar 検定による洗練化の効果の検討

学習形状の洗練化の効果を評価するために、洗練化の各段階での形状により訓練した分類器の分類精度について検討する。測定に用いた形状は、各洗練化段階で用いた訓練形状、テスト形状、および各洗練化段階に共通のテスト形状である。なお、4.2 節で述べたインデックスが重複しな

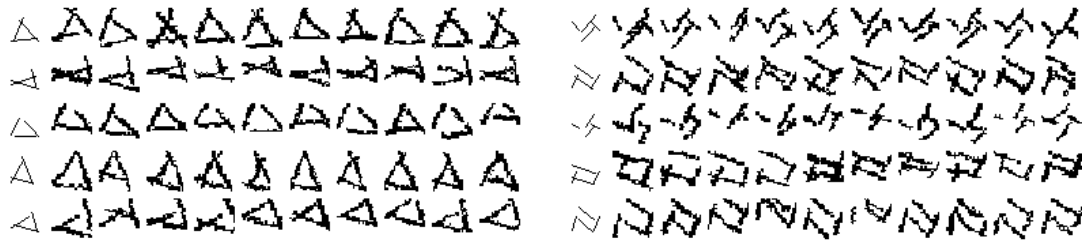


図 2 基準形状を基に生成した派生基準形状およびラフ類似形状の例

Fig. 2 Examples of derived base, and roughly similar shapes generated with base shapes: Lef-half plane; shapes generated from the class 1 base shape in Fig. 1. Right-half plane; shapes generated from the class 2 base shape in Fig. 1.

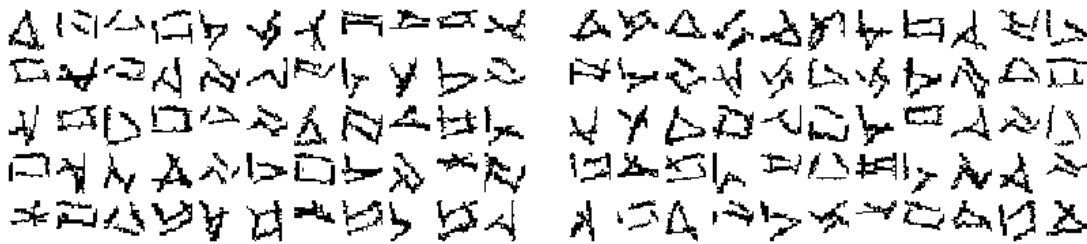


図 3 訓練形状およびテスト形状の例 (繰り返し 5)

Fig. 3 Examples of training and testing shapes(iteration 5): Lef-half plane; training shapes. Right-half plane; testing shapes.

表 2 洗練化の各段階での形状により訓練した分類器の分類精度

Table 2 Classification accuracy of classifiers trained on shapes in each refinement step.

classifier ^a	C_0	C_1	C_2	C_3	C_4
training accuracy	1.0	1.0	1.0	1.0	1.0
testing accuracy ^b	0.8991 (0.8976)	0.8877 (0.8953)	0.9132 (0.9003)	0.9241 (0.9175)	0.93 (0.9233)

^a The classifier C_i is trained on i -th refined training shapes. The classifier C_0 is trained on training shapes without refinement.

^b The accuracy is provided in parentheses, is tested using the same testing shapes.

表 3 洗練化の各段階での形状により訓練した分類器の McNemar 検定 (p 値) による評価

Table 3 Evaluation of classifiers, trained on shapes in each refinement step, using McNemar's test (p-value): The column C_j value on row C_i shows the p-value of McNemar's test used on the classification results of classifier C_i and C_j .

classifire ^a	C_0	C_1	C_2	C_3	C_4
C_0	—	0.7086	0.6444	2.553×10^{-4}	1.106×10^{-6}
C_1	0.7086	—	0.3873	4.558×10^{-5}	2.34×10^{-7}
C_2	0.6444	0.3873	—	1.498×10^{-3}	6.831×10^{-6}
C_3	2.553×10^{-4}	4.558×10^{-5}	1.498×10^{-3}	—	0.2508
C_4	1.106×10^{-6}	2.34×10^{-7}	6.831×10^{-6}	0.2508	—

^a C_i denotes the classifier trained on i -th refined training shapes. C_0 denotes the classifier trained on training shapes without refinement.



図 4 形状分類の結果の例 (繰り返し 5)

Fig. 4 Examples of classification results(iteration 5): Lef-half plane; correctly classified shapes. Right-half plane; misclassified shapes. Upper-half plane; class 1 shapes. Lower-half plane; class 2 shapes.

いようにこれらの形状を選択した。測定した分類精度を、表 2 に示す。ここで、 C_i は、 i 回洗練化をおこなった形状を用いて訓練した分類器である。また、 C_0 は、洗練化をおこなう前の形状を用いて訓練した分類器である。

次に、学習形状の洗練化の効果の頑健性を評価するために、各分類器 C_i を用いた分類結果について、McNemar 検定により統計的検定をおこなった。表 3 に、McNemar 検定の p 値を示す。なお、ここでの各分類器 C_i による分類に用いたテスト形状の集合は、同一のものである。すなわち、テスト形状の集合を構成する形状を変更しないものを用いた。その共通のテスト形状の集合を用いて、分類器 C_i による分類の正誤および C_j による分類の正誤について McNemar 検定をおこない p 値を求めた。

正しく分類された形状および誤って分類された形状を図 4 に示す。

図 1 と 2 とを比較することにより、基準形状に類似な派生基準形状を生成することが可能であることが分かる。そして、派生基準形状を基にして、それに類似なラフ類似形状を生成することが可能であることが分かる。生成した派生基準形状およびラフ類似形状は、基準形状に類似な形状であり、多様な類似性をもつスケッチ風の形状であることが確認できる。

図 3 から、DNN の訓練形状およびテスト形状として、スケッチ風の多様な類似性をもつ形状学習データを生成できることが分かる。

表 2 に示す DNN を用いた形状分類の結果からは、多様な類似性をもつ形状を対象として、テスト形状に対して約 0.9 の分類精度が得られることが分かる。そして、訓練形状の洗練化により分類精度が改善していることが分かる。

括弧で囲んでいないテスト形状の分類精度は、形状のインデックスが重複しないように選択したテスト形状を用いたものである。そのことから、訓練形状を洗練化することにより、未知の形状に対する分類精度が改善しているといえる。また、括弧で囲んだテスト形状の分類精度は、同一の集合のテスト形状を用いたものである。

表 3 に示す McNemar 検定の p 値は、分類器 C_i と C_j に差がないと仮定した場合に、その分類結果が得られる度合いを表現している。 $i = j$ の場合には $p = 1$ であるのに対して、 $i \neq j$ の場合には p 値は小さい値となっている。

分類器の性能を評価するために統計的検定をおこなう場合に、有意水準としてどのような値を用いるべきかについては、十分な研究がなされているとはいえない。ここでは、 p 値を有意水準と比較せずに、 p 値について定性的に検討する。分類器 C_i と C_j についての p 値は、 i と j の差が大きいほど小さい値となっている。そのため、訓練形状の洗練化の回数の差が大きいほど分類器の性能の差が大きくなっているといえる。そのことに加えて、表 2 に示すように、訓練形状の洗練化の回数が大きいほど分類器の分類精度が改善されていることを考慮すると、訓練形状の洗練化の回数が多いほど訓練形状が改善されているといえる。

なお、ここでは、DNN を訓練する場合の乱数の影響については考慮していない。この点については、今後の検討が必要である。

次に、図 4 に示す分類された形状について具体的に検討する。正しく分類された形状については、大部分は適切な分類形状であると考えられる。しかしながら、一部の形状は、クラス 1 およびクラス 2 のどちらにも所属しない形状と考えるのが適切な形状であるといえる。これらの形状

については、訓練形状に含めないなどの、アルゴリズムの改良が必要である。誤って分類された形状については、正しいクラスに分類されるべき形状が多数含まれている。これらの形状については、訓練形状に追加する必要がある。

これらを総合して判断すると、さらに訓練形状を増加する必要があると考えられる。そして、訓練形状を増加することにより分類精度の改善が期待できるといえる。

5. おわりに

スケッチした pictorial symbols の深層学習 (deep learning) に用いることを目的とした既提案の学習形状の生成手法 [15] について、学習形状の洗練化の効果を検討した。

学習形状の洗練化を繰り返すほど、その形状を用いて訓練した DNN の分類精度が改善されることを示した。それに加えて、洗練化をおこなう前と後とでは学習形状の品質に統計的な差があるのかどうかについて、統計的検定を用いて検討した。検討をおこなうために始めに、洗練化の繰り返しの各段階での形状を用いて DNN を訓練し、テスト形状の分類をおこなった。そして、分類結果について McNemar 検定を用いて統計的検定をおこなった。それにより、洗練化の回数が大きいくほど、学習形状が改善されることを示した。

今後の課題には、既提案手法の複雑な形状への適用、および、スケッチした pictorial symbols の形状のクラス数を大きくしての深層学習に関する研究などを挙げることができる。

参考文献

[1] Bishop, C. M.: *Neural Networks for Pattern Recognition*, Oxford University Press, Inc., New York, NY, USA (1995).

[2] Clark, A. F. and Clark, C.: Performance Characterization in Computer Vision: A Tutorial, <http://peipa.essex.ac.uk/benchmark/tutorials/essex/tutorial.pdf>.

[3] Dietterich, T. G.: Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms, *Neural Comput.*, Vol. 10, No. 7, pp. 1895–1923 (online), DOI: 10.1162/089976698300017197 (1998).

[4] Edwards, C.: Growing Pains for Deep Learning, *Commun. ACM*, Vol. 58, No. 7, pp. 14–16 (online), DOI: 10.1145/2771283 (2015).

[5] Eitz, M., Hays, J. and Alexa, M.: How do humans sketch objects?, *ACM Trans. Graph.*, Vol. 31, No. 4, pp. 44:1–44:10 (online), DOI: 10.1145/2185520.2185540 (2012).

[6] Fu, L. and Kara, L. B.: Technical Section: Neural Network-based Symbol Recognition Using a Few Labeled Samples, *Comput. Graph.*, Vol. 35, No. 5, pp. 955–966 (online), DOI: 10.1016/j.cag.2011.07.001 (2011).

[7] Geman, S., Bienenstock, E. and Doursat, R.: Neural Networks and the Bias/Variance Dilemma, *Neural Comput.*, Vol. 4, No. 1, pp. 1–58 (online), DOI: 10.1162/neco.1992.4.1.1 (1992).

[8] Hammond, T. and Paulson, B.: Recognizing sketched multistroke primitives, *ACM Trans. Interact. Intell.*

Syst., Vol. 1, No. 1, pp. 4:1–4:34 (online), DOI: 10.1145/2030365.2030369 (2011).

[9] Hinton, G. E. and Salakhutdinov, R. R.: Reducing the Dimensionality of Data with Neural Networks, *Science*, Vol. 313, No. 5786, pp. 504–507 (2006).

[10] Hinton, G. E., Osindero, S. and Teh, Y.-W.: A Fast Learning Algorithm for Deep Belief Nets, *Neural Comput.*, Vol. 18, No. 7, pp. 1527–1554 (online), DOI: 10.1162/neco.2006.18.7.1527 (2006).

[11] Kara, L. B. and Stahovich, T. F.: An Image-based, Trainable Symbol Recognizer for Hand-drawn Sketches, *Comput. Graph.*, Vol. 29, No. 4, pp. 501–517 (online), DOI: 10.1016/j.cag.2005.05.004 (2005).

[12] Krizhevsky, A., Sutskever, I. and Hinton, G. E.: ImageNet Classification with Deep Convolutional Neural Networks, *Advances in Neural Information Processing Systems 25* (Pereira, F., Burges, C., Bottou, L. and Weinberger, K., eds.), Curran Associates, Inc., pp. 1097–1105 (2012).

[13] 佐藤 信: スケッチした pictorial symbols のディープラーニングのための派生基準形状を用いた学習形状の反復的生成, 情報処理学会研究報告, Vol. 2015-CG-161, No. 12, pp. 1–8 (2015).

[14] 佐藤 信: スケッチした pictorial symbols のディープラーニングのための部分的弾性変形を用いた形状学習データ生成, 情報処理学会研究報告, Vol. 2015-CG-160, No. 7, pp. 1–8 (2015).

[15] 佐藤 信: スケッチした pictorial symbols の学習のための深層学習を用いた学習形状の洗練化, 情報処理学会研究報告, Vol. 2016-CG-162, No. 3, pp. 1–8 (2016).

[16] McNemar, Q.: Note on the sampling error of the difference between correlated proportions or percentages, *Psychometrika*, Vol. 12, No. 2, pp. 153–157 (online), DOI: 10.1007/BF02295996.

[17] Modley, R. and Myers, W.: *Handbook of Pictorial Symbols: 3,250 Examples from International Sources*, Dover pictorial archive series, Dover Publications (1976).

[18] Schmidhuber, J.: Multi-column Deep Neural Networks for Image Classification, *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, CVPR '12, Washington, DC, USA, IEEE Computer Society, pp. 3642–3649 (online), available from <http://dl.acm.org/citation.cfm?id=2354409.2354694> (2012).

[19] Simard, P. Y., Steinkraus, D. and Platt, J. C.: Best Practices for Convolutional Neural Networks Applied to Visual Document Analysis, *Proceedings of the Seventh International Conference on Document Analysis and Recognition - Volume 2, ICDAR '03*, Washington, DC, USA, IEEE Computer Society, pp. 958–963 (2003).

[20] Van Vaerenbergh, S., Santamaria, I. and Barbano, P.: Semi-supervised handwritten digit recognition using very few labeled data, *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pp. 2136–2139 (2011).

[21] W3C: *Scalable Vector Graphics (SVG) 1.1 Specification (Second Edition)*, <http://www.w3.org/> (2011).

[22] Wasserstein, R. L. and Lazar, N. A.: The ASA's statement on p-values: context, process, and purpose, *The American Statistician*, pp. 1–17 (online), DOI: 10.1080/00031305.2016.1154108 (2016).