

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

佐藤 信¹

概要: スケッチした形状の深層学習 (deep learning) のために、少数の基準形状から学習形状を生成するための手法を提案する。提案手法では、学習形状の洗練化のために、深層学習により学習した形状の特徴を用いる。始めに、SVG 形式により表現された pictorial symbols から基準形状および派生基準形状を生成する。次に、それらの形状に多様な類似性をもつラフ類似形状を生成し、その形状特徴を深層学習を用いて学習する。学習した特徴に基づいて学習形状に含まれる形状を更新することにより、学習形状の集合の洗練化をおこなう。

Learning Shape Refinement Using Deep Learning for Learning Sketched Pictorial Symbols

MAKOTO SATOH¹

Abstract: This paper presents a method for generating learning sets from a small set of base shapes for deep learning of sketched pictorial symbols. In the method, shape features learned using deep learning are used for refining generated shape sets. First base and its derived base shapes are generated from pictorial symbols represented using SVG format. Then roughly similar shapes which have diversity-carrying similarities, are generated, and the shape features of the generated shapes are learned using deep learning. The learning shape sets are refined by updating the shapes included in the generated shapes based on the learned features.

1. はじめに

本稿では、スケッチした pictorial symbols [13] を学習することを目的として、予め与えた少数の基準形状に基づき学習形状を生成するための手法を提案する。その特徴は、以下のとおりである。既提案手法 [10], [11] を基にして、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、深層学習 (deep learning) を用いて、生成した形状の特徴を学習し、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。

私たちの身の周りでは、多くの pictorial symbols が用いられている。例えば、案内図および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および音符

などのように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。その特徴は、特定の言語に依存せずに、情報を直感的に伝達することが可能なことである。これにより、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。このような pictorial symbols を機械学習により学習するための学習形状を生成することが、本研究の目的である。

これ以降の構成について、簡単に説明する。2 節では、関連研究について比較をおこない、提案する手法の概要と特徴について述べる。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手法を提案する。提案手法の実装と実験結果の検討について 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。

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

2. 関連研究と提案手法の概要

2.1 手描き形状の多様性

計算機を用いてスケッチした形状を認識する場合には、自然画像などの物体の形状を認識する場合と比較して、認識する形状の変化が大きいということを考慮する必要がある。形状に変化が生じるのは、人手により描いた形状であることが原因であるが、それに加えてスケッチした形状に特有の原因として、形状の作者が描く対象をどのように捉えているのかということが、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を挙げるができる。そのような形状の変化の多様性に対応するためには、認識する形状を学習することが可能な手法が有効であるといえる。

Kara 等 [8] では、認識しようとする形状クラスについて1つの形状を入力として与え、その形状を基に形状認識をおこなう手法を提案している。その手法を用いると、形状の移動、回転および均一なスケーリングに対して頑健な形状認識が可能である。しかし、形状の非均一なスケーリングに対しては頑健な形状認識は実現できていない。

2.2 形状変化に頑健な学習

スケッチした形状がもつ多様な変化に対応するためには、形状を学習することが有効である。特に、変形に対して頑健な学習である必要がある。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に、以下の手法について述べている。

手法 1: 学習形状 認識しようとする形状の変化に対応する形状変換をおこなった学習形状を用いて学習をおこなう。

手法 2: 形状特徴 認識しようとする形状の変化に対して不変な形状特徴を用いて学習をおこなう。

手法 3: 学習モデル 認識しようとする形状の変化に対して不変な学習をおこなうことが可能な学習モデルを用いて学習をおこなう。

手法 1 は、原理的に分かりやすい手法であるが、大量の学習形状が必要になる手法である。特に、変形の範囲に対応した十分な学習形状を用意する必要がある。**手法 2** は、適切な形状特徴を選択できる場合には、効率的な手法である。しかし、学習した形状特徴が変化した場合には、形状特徴を再選択することが必要になる可能性がある。**手法 3** は、適切な学習モデルを選択出来る場合には、有効な手法である。[1] では、多層のニューラルネットを用いる手法について述べている。しかし、その学習において過学習を避けるためには、十分な個数の学習形状が必要となる。

2.3 形状の深層学習と学習形状 augmentation

深層学習を用いて形状を機械学習するための研究が盛んにおこなわれている。それらのうちで形状画像を対象とする研究では、手書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が多数をしめる。

一方、スケッチした形状を対象とする研究は少数である。2.1 節において述べたように、スケッチした形状の学習においては、形状の変化の多様性に対応できることが重要である。そのためには、大量の学習形状を用いて、多様性をもつ類似な形状の特徴を学習することが必要となる。例えば、Fu 等 [4] では、スケッチした形状に形状 augmentation をおこなうことにより生成した、多様性のある類似形状を用いて学習をおこなっている。そして、生成した形状を、深層学習の学習モデルのひとつである Deep Belief Network (DBN) [7] を用いて学習している。深層学習は、Hinton 等 [6] が示すように、学習データの特徴を抽出する性能に優れた手法である。

スケッチした形状の学習に特有のことではないが一般に、機械学習においては、過学習の可能性を減少させるためには、十分な個数の学習データを用いて学習をおこなう必要がある。具体的には、学習誤差のバイアスとバリエーションとのトレードオフの関係 [5] を考慮して、学習をおこなうことが重要である。機械学習においては、学習アルゴリズムと同様に、学習データも重要な役割を果たすといえる [2]。

ところが、学習形状の収集には多くの時間と労力が必要となる。例えば、Eitz 等 [3] では、スケッチした形状の機械学習に用いるための形状データを、インターネットを利用して収集している。

一方、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、学習形状の augmentation (増補) がおこなわれることがある。学習形状の augmentation とは、学習データの個数をアルゴリズムを用いて増加させることにより、学習データの不足を補うことである。形状画像の学習データに augmentation をおこなっている例としては、Krizhevsky 等 [9]、Schmidhuber [15] がある。また、Fu 等 [4] は、少数の線図形を基にして、形状 augmentation を用いてスケッチ風の学習形状を生成している。そこでは、Simard 等 [16] の提案による画像の弾性変形手法を用いて、スケッチ風の学習データを生成している。

なお、弾性変形による微小な変形を用いて類似な形状を生成するという考え方は、Van Vaerenbergh, S 等 [17] の手法の基となっている考え方と共通するものがあるといえる。Van Vaerenbergh, S 等 [17] では、半教師あり学習による手書き数字画像の認識をおこなっている。そこでは、少数のラベル付き形状と多数のラベル無し形状を、学習形状として用いている。あるラベル無し形状を微小変化することにより、いづれかのラベル付き形状を生成可能であるという着想に基づいて、形状どうしの類似性の関係を表現す

るためのグラフを作成している。そして、そのグラフのリンクをたどることにより、あるラベル無し形状が、ラベル付き形状の所属するクラスのうちどのクラスに所属するのかを判定している。

2.4 提案手法の概要と適用分野

提案手法では、スケッチした pictorial symbols の機械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。

スケッチした形状は、2.1 節で述べたように、形状の変化に多様性があるといえる。そのため提案手法では、2.2 節での手法 1 の考えに基づいて、基準となる形状に形状変換をおこなった多様性のある類似な形状を学習する。その特徴は、以下に示すとおりである。

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 [10]) およびラフ類似形状 (佐藤 [11]) 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。

ラフ類似形状は、基準とする pictorial symbols に類似なスケッチ風の形状である。その生成には、Simard 等 [16] の手法を拡張した弾性変形手法を用いている。なお、Fu 等 [4] においても、少数の基準形状を基にして、類似な形状を生成するために Simard 等 [16] の手法を用いている。

また、pictorial symbols は、SVG 形式 [18] 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 を基準形状として用いる。

提案手法を用いると、少数の形状を基にして、機械学習のための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。そのことにより、深層学習のように、大量の学習形状を必要とする機械学習のために有用な手法であるといえる。また、認識しようとするスケッチした形状のカスタマイズが容易になることから、特定の適用分野において用いられる図形形状の認識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

3.1 アルゴリズムの各段階

学習形状の集合を洗練化するための手法を、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。概要を、以下に示す。

Step1: 基準形状の生成 与えられた少数の pictorial symbols の形状から、基準形状の集合 (\mathcal{B}) を生成する。ここで、pictorial symbols は、SVG 形式の形状ファイル

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 an empty shape set for correctly classified shapes: \mathcal{C} .
Make an 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} .

として与えられているとする。その SVG ファイルを構文解析することにより、各形状の部分ごとの Bezier 曲線の集合として \mathcal{B} を生成する。

Step2: 派生基準形状の生成 \mathcal{B} から派生基準形状の集合 \mathcal{D}_c を生成する。 \mathcal{D}_c は、 \mathcal{B} を部分形状ごとに異なるパラメータを用いて形状変換をおこなうことにより生成した Bezier 曲線の集合である。

Step3: ラフ類似形状の生成 部分的な弾性変形を用いて、 \mathcal{B} および \mathcal{D}_c を基にして、多様な類似性をもつ粗く類似な形状 (ラフ類似形状) の集合 \mathcal{R}_c を画像データとして生成する。 \mathcal{R}_c は、スケッチ風の形状である。

Step4: 訓練形状の生成 $\mathcal{B}, \mathcal{D}_c$ および \mathcal{R}_c を用いて、DNN (deep neural network) のための訓練形状の集合 \mathcal{T} を生成する。

Step5: 訓練形状の洗練化 DNN を用いて \mathcal{T} の形状特徴を学習する。そして、学習した形状特徴を用いて、テ

スト形状のクラス分類をおこない、その結果に基づいて T を更新する。

Step5 を必要に応じて繰り返し、訓練形状を更新することにより訓練形状の洗練化をおこなう。

3.2 派生基準形状の生成

派生基準形状を生成するための手法には、[10] を改良したものをを用いた。改良点は、基準形状および派生基準形状の表現形式を変更した点である。[10] では、部分形状ごとの基準形状の画像データに形状変換をおこなうことにより派生基準形状の生成をおこなった。ここでは、部分形状ごとの基準形状の Bezier 曲線の集合に形状変換をおこなうことにより派生基準形を生成する。これにより、鮮明な形状の派生基準形状を生成することが可能となる。そして、派生基準形を基に生成するラフ類似形状についても、鮮明な形状を得ることが可能となる。

また、[10] では、スケーリングにより形状変換をおこなったが、ここでは、回転、スケーリングおよび移動変換により形状変換をおこなう。[10] と同様に、部分形状ごとに異なる形状変換パラメータを用いる。なお、形状の回転の中心は、部分形状の重心である。以下に、乱数を用いて派生基準形状を生成するための手法を示す。

Step1: 正規分布の乱数生成 形状変換の種類ごとに予め与えた正規分布のパラメータに基づき乱数を生成し、回転、スケーリングおよび移動変換のための形状変換パラメータとする。なお、予め生成する乱数の範囲を、形状変換の種類ごとに与えておいて、その範囲の値を用いる。

Step2: 形状変換 生成した形状変換パラメータを用いて形状変換をおこなう。

Step3: 変換形状のフィッティング 変換をおこなった形状を、予め与えておいた領域にフィッティングをおこなう。

3.3 ラフ類似形状の生成

基準形状および派生基準形状を基にして、多様な類似性をもつラフ類似形状を生成する。そのために、[11] で述べた部分的な弾性変形手法を用いた。3.2 節で述べた改良により、より鮮明な画像のラフ類似形状を得ることができる。

4. 実装と結果の検討

3 節で提案したアルゴリズムを実装し、その有効性について検討した。

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。



図 1 SVG ファイルから生成した基準形状

Fig. 1 Base shapes generated from SVG 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.

三角形および矩形を構成する部分形状の数は、3 および 4 である。三角形および矩形が所属する形状クラスを、それぞれクラス 1 および 2 とした。始めに、SVG 形式のファイルを構文解析することにより部分形状ごとの Bezier 曲線の集合を生成した。そして、生成した Bezier 曲線をラスターライズすることにより、これらの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ーライズした。

これらの基準形状を基にして、派生基準形状およびラフ類似形状を生成した。生成した派生基準形状の数は、各基準形状について 500 である。各基準形状および各派生基準形状について生成したラフ類似形状の数は、10 である。図 2 に、生成した派生基準形状およびラフ類似形状の一部分を示す。図の左半分が三角形であり、右半分が矩形である。それぞれの領域のなかでの最も左の列に示す形状が、派生基準形状である。それぞれの派生基準形状の右側の 10 個の列に示す形状が、その派生基準形状から生成したラフ類似形状である。3.2 節で述べたように、乱数を用いて決定したパラメータを用いて形状変換をおこなうことにより、基準形状から派生基準形状を生成した。乱数生成に用いたパラメータを表 1 に示す。形状変換の種類ごとに、表に示す平均値と標準偏差の乱数を発生した。そして、発生した値のうちで、表に示す最小値と最大値の間の値を用いて形状変換をおこなった。また、生成した形状を一意的に識別可能にするために、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、[11] において詳細が述べられている。

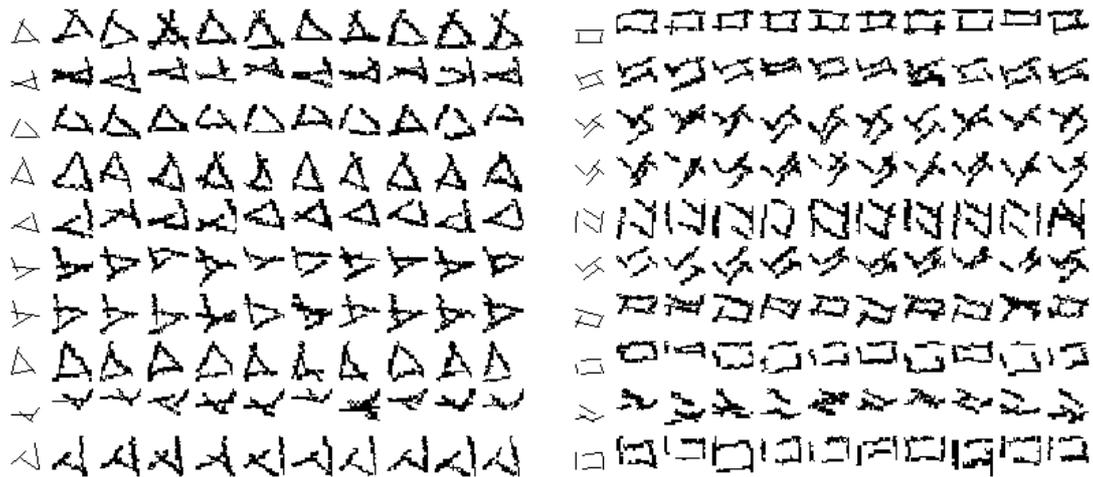


図 2 基準形状を基に生成した派生基準形状およびラフ類似形状の例

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.

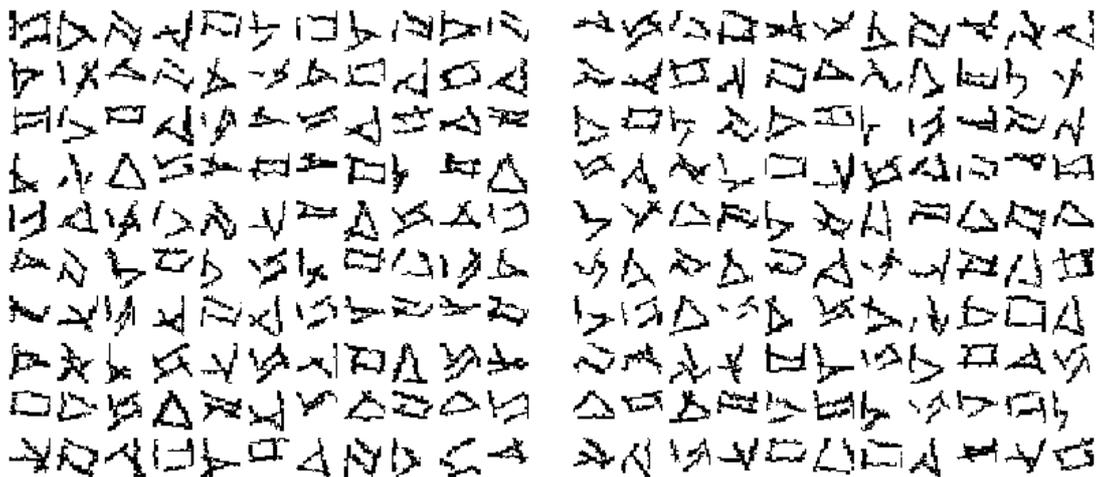


図 3 訓練形状およびテスト形状の例 (繰り返し 1)

Fig. 3 Examples of training and testing shapes(iteration 1): Lef-half plane; training shapes. Right-half plane; testing shapes.

4.2 形状特徴の学習

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. このうち, 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. 訓練形状およびテスト形状の数は, それぞれ 4422 および 2200 である. それらの形状の一部分を, 図 3 に示す. なお, 4.1 節で生成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.

学習したパラメータを用いて形状分類をおこなった結果を, 表 2 に示す. 表に示す値は, 全ての訓練形状およびテスト形状についての分類結果を基にした値である. 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. なお, 誤って分類されたクラス 1 の形状は, ここに示す形状が全ての形状である.

表 2 分類精度

Table 2 Classification accuracy

iteration	training shape	testing shape
1	1.000	0.901
2	1.000	0.915 (0.940) ^a

^a The accuracy is tested using the same testing shapes as iteration 1.

表 3 繰り返し 1 と同じテスト形状を用いての繰り返し 2 での分類結果の詳細

Table 3 Details of classification results at iteration 2, using same testing shapes as iteration 1.

classification results		counts of shapes
iteration 1	iteration 2	
true	true	1849
true	false	133
false	true	218
false	false	0

4.3 学習形状の洗練化

4.2 節でのテスト形状の分類結果に基づいて、アルゴリズム 2 により訓練形状の更新をおこなった。そして、更新した訓練形状を用いてアルゴリズムの繰り返し 2 での DNN の訓練をおこなった。

更新した DNN のパラメータを用いて形状分類をおこなった結果を表 2 に示す。繰り返し 2 で用いるテスト形状としては、繰り返し 1 で用いた訓練形状およびテスト形状と形状のインデックスが重複しないように選択した形状を用いた。表中のカッコ内は、繰り返し 1 で用いたものと同じテスト形状により形状分類をおこなった結果である。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 5 に示す。なお、誤って分類された形状は、ここに示す形状が全ての形状である。また、繰り返し 1 で用いたものと同じテスト形状により形状分類をおこなった結果の詳細を、表 3 に示す。

4.4 検討

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も約 0.9 の分類精度が得られることが分かる。ここで、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、一部分の形状は、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。

次に、洗練化をおこなった訓練形状を学習した場合（繰り返し 2）について、形状を分類した結果を検討する。表 2 および図 5 から、繰り返し 1 とほぼ同様の分類結果が得られることが分かる。表 2 のテスト形状の分類精度を繰り返し 1 と 2 について比較すると、分類精度が改善されていることが分かる。また、繰り返し 1 で用いたテスト形状について、繰り返し 1 と 2 とを比較すると、繰り返し 2 での分類精度が改善されていることが分かる。これは、繰り返し 1 で誤って分類されたテスト形状を繰り返し 2 で用いた学習形状に追加したことによる効果である。一方、表 3 からは、繰り返し 1 で正しく分類されたテスト形状が繰り返し 2 でどのように分類されるのかが分かる。それによると、繰り返し 1 で正しく分類された形状の大部分は繰り返し 2 でも正しく分類されているが、一部分の形状は繰り返し 2 では誤って分類されていることが分かる。しかしながら、表 2 に示すように、全体としては分類精度が改善していることが分かる。

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 を深層学習により学習することを目的として、予め与えた少数の基準形状に基づき学習形状を生成するための手法を提案した。

提案では、学習形状として用いるためのラフ類似形状を鮮明にするために、既提案手法 [10], [11] を拡張した。そして、ラフ類似形状を用いて学習形状を生成および洗練化するための手法を提案した。特徴は、深層学習により学習した形状特徴を用いて、学習形状を洗練化することである。

提案手法を実装することにより、SVG 形式で定義した pictorial symbols を基準形状として、多様な類似性をもつスケッチ風の形状を生成することが可能であることを示した。そして、生成した形状を深層学習することにより、学習形状を洗練化することが可能であることを示した。

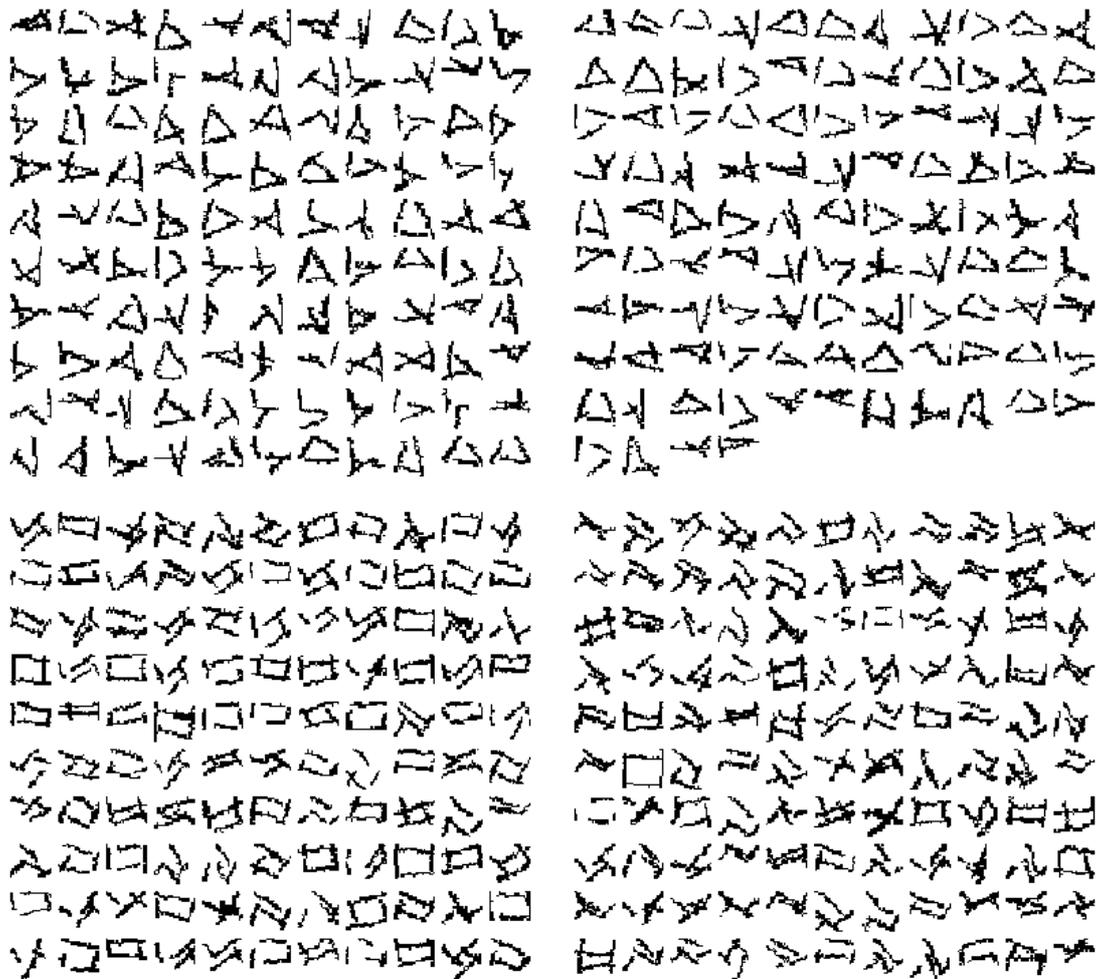


図 4 形状分類の結果の例 (繰り返し 1)

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

なお、類似形状を生成するためには、線形の形状変換および非線形の弾性変形を用いた。そのためのパラメータの調整方法については検討が必要である。また、学習形状の生成をどのように反復するのかについても、さらに検討が必要である。それ以外の今後の課題には、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] Edwards, C.: Growing Pains for Deep Learning, *Commun. ACM*, Vol. 58, No. 7, pp. 14–16 (online), DOI: 10.1145/2771283 (2015).
 [3] 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).
 [4] 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).
 [5] 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).
 [6] Hinton, G. E. and Salakhutdinov, R. R.: Reducing the Dimensionality of Data with Neural Networks, *Science*, Vol. 313, No. 5786, pp. 504–507 (2006).
 [7] 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).
 [8] 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).
 [9] 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).

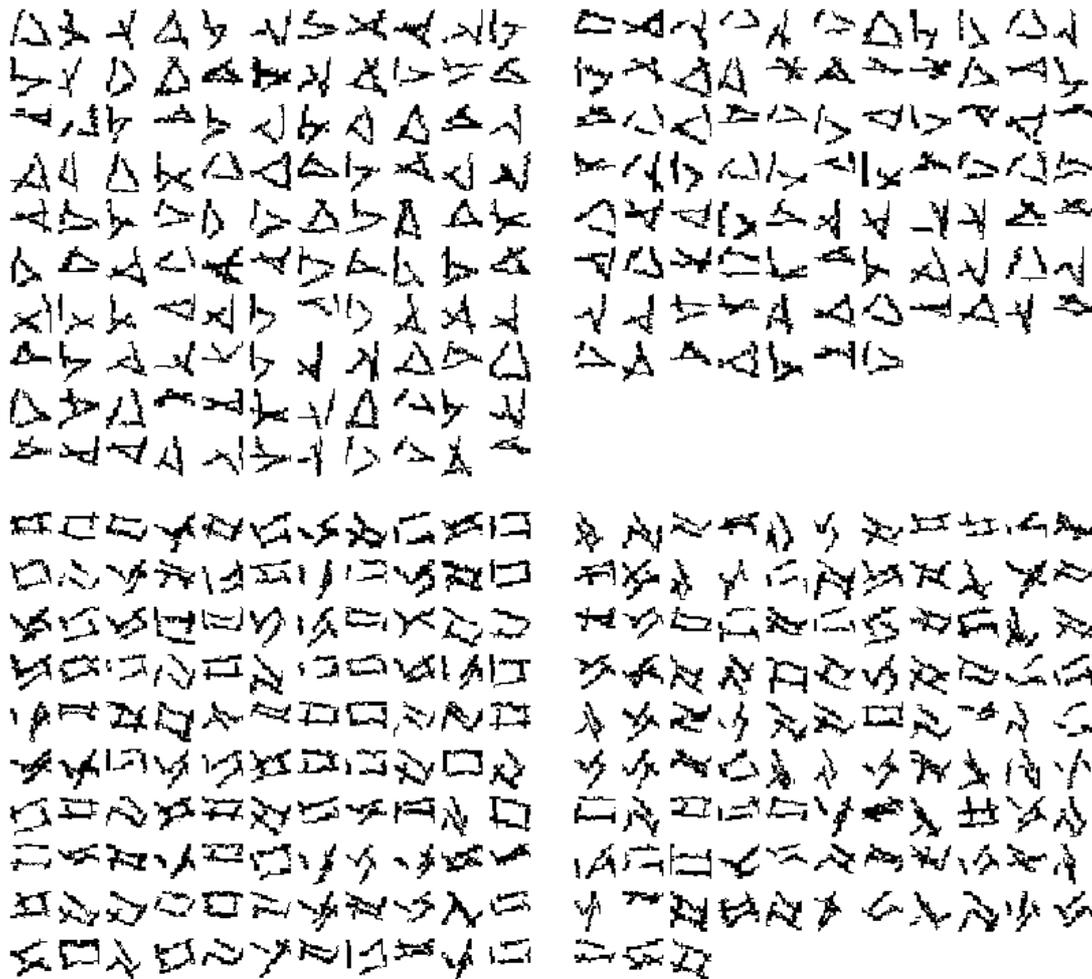


図 5 形状分類の結果の例 (繰り返し 2)

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

- [10] 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).
- [11] 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).
- [12] Mitchell, T. M.: The Need for Biases in Learning Generalizations, *Readings in Machine Learning* (Shavlik, J. W. and Dietterich, T. G., eds.), Morgan Kaufman, pp. 184-191 (1980).
- [13] Modley, R. and Myers, W.: *Handbook of Pictorial Symbols: 3,250 Examples from International Sources*, Dover pictorial archive series, Dover Publications (1976).
- [14] Paulson, B. and Hammond, T.: PaleoSketch: Accurate Primitive Sketch Recognition and Beautification, *Proceedings of the 13th International Conference on Intelligent User Interfaces*, IUI '08, New York, NY, USA, ACM, pp. 1-10 (online), DOI: 10.1145/1378773.1378775 (2008).
- [15] 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).
- [16] 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).
- [17] 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).
- [18] W3C: *Scalable Vector Graphics (SVG) 1.1 Specification (Second Edition)*, <http://www.w3.org/> (2011).