

非均一相似形状補間を用いた形状学習データ生成

佐藤 信¹

概要: 機械学習により形状を認識するために、少数の形状を基準として、アルゴリズムにより学習データを生成するための手法を提案する。提案手法では、非均一相似変換を用いて、基準形状の類似形状を生成する。そして、非均一相似変換を用いて、基準形状とその類似形状を形状補間することにより学習データを生成する。生成した補間形状を用いることの機械学習への効果について検討をおこなう。

Shape Training Set Generation Using Interpolation of Non-uniform Similar Shapes

MAKOTO SATOH¹

Abstract: This paper presents a method to algorithmically generate training sets from a few shapes for shape recognition. In the method, similar shapes with a base shape are generated by using non-uniform similarity transform. Then training sets are generated by using non-uniform similar shape interpolation between the base shape and similar shapes. Additionally, we discuss the effect of using generated interpolating shapes on a machine learning algorithm.

1. はじめに

本稿では、機械学習を用いて形状を認識するために、少数の形状を基準としてアルゴリズムにより学習データを生成するための手法を提案する。コンピュータを用いて形状を認識するためのアルゴリズムの開発は、重要な課題である。そのため多くの研究がおこなわれ、さまざまなアルゴリズムが提案されている。それらのアルゴリズムを大別すると、形状パターンなどのルールをアルゴリズムの一部としてプログラムに組み込む手法、または、機械学習により学習データから形状特徴を取得する手法に分類できる。本研究は、後者の機械学習を用いた形状認識のための手法である。

機械学習を用いた形状認識のための研究の一分野として、手描きストロークによる形状を認識することを目的とする研究がある。手描きストロークによる入力形状には、不明確な部分が含まれる可能性があるため、その点を考慮したアルゴリズムの開発が必要となる。例えば、Eitz 等 [3] は、

形状の分類ごとに形状を例示しておいて、スケッチの専門的な技術を習得していない人間に手描きでスケッチを描いてもらい、そのデータを用いて形状認識に関する実験をおこなっている。そこでは、人間が形状を認識する場合と、機械学習のアルゴリズムを用いて形状を認識する場合について比較をおこない、どちらの場合においても誤認識が発生することが示されている。そして、その誤認識には、形状の不明確性を原因とするものだけでなく、意味的な不明確性を原因とするものがあることについて指摘している。

このような手描き形状の不明確性を克服して形状を認識するためには、ユーザまたは適用分野ごとにカスタマイズした形状認識のための機械学習のパラメータを用いることが有効である。しかし、一般的には、機械学習のアルゴリズムは大量の学習データを必要とするので、学習データの収集には多くのコストを必要とする。そのため、少数のデータを基にして学習データを生成し、その学習データを用いて機械学習のパラメータを求めるための研究がおこなわれている。

少数の形状データを基にして生成した形状データを用いて機械学習をする手法については、Fu 等 [5] の研究がある。

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

ここでは、基準とする線画の画像を変形することにより、学習データを生成している。また、Nonnemaker 等 [10] は、フォントの形状認識において、アルゴリズムにより生成した学習データの妥当性について研究している。ここでは、実際に収集した学習データを用いて学習した場合と、アルゴリズムにより生成した学習データを用いて学習した場合では、形状認識率がほぼ等しいということが示されている。

本稿では、非均一相似性制約 [8] を用いて、Bezier 曲線で表現した少数の基準とする形状を基にして、それに類似した形状を生成して、それを機械学習の学習データとして用いるための既提案の手法 [6] を拡張したアルゴリズムを提案する。そのアルゴリズムの拡張点は、基準とする形状に類似した形状を補間する形状を生成する点である。補間形状の生成を容易にするために、曲線通過点の座標値の平均値を制約することにより、類似した形状を生成する手法を提案する。また、生成した形状学習データを、次元縮退 [11] を用いて分類することにより、生成した形状学習データの形状類似性を機械学習のアルゴリズムで取り扱うことが可能であることを示す。

2. 平均位置制約を用いた非均一相似形状生成

2.1 非均一相似変換

提案の学習データの生成手法では、基準とする形状の類似形状として非均一相似形状を生成する。このとき、非均一相似変換を用いるので、非均一相似変換について簡単に説明する。なお、詳細については、佐藤等 [8], [9] に詳しい説明がある。

非均一相似変換とは、平面上の Bezier 曲線を、曲線の通過点などの形状変形のための制約条件を満たしながら、可能な限り基準とする形状の特徴を維持して形状を洗練化 [4] するための線形変換であり、以下の特徴をもつ。

洗練化形状の範囲 基準形状の特徴を維持しながら、その類似形状、直線形状そして鏡映対称形状の類似形状に形状洗練化が可能である。

類似性の基準 形状の類似性の基準として、形状類似性変数 [8] を用いる。形状類似性変数とは、基準曲線の制約点での 2 つの導関数ベクトルの始点を一致させてできるベクトル形状について、その形状を相似変換した形状と実際の変換形状との差を表現する変数であり、スケーリング、回転そして移動について不変量である。

最適化 形状類似性変数を変数として、形状変形のための線形制約条件を作成する。そして、その制約条件のもとで、形状類似性変数の値を最小化することにより、形状を洗練化する。この計算手法は、曲線形状を洗練化するための手法に関する Bartels 等 [1], [2] の研究を発展させたものである。

非均一性 制約点ごとの相似変換の倍率、回転そして移動量は、最適化計算により決定されるので、非均一で

ある。

2.2 平均位置制約

非均一相似変換は、曲線の表現形式変換、そして、最適化計算の 2 つの段階で構成される。提案する平均位置制約のために必要となる、非均一相似変換の各段階での変更点について、以下で説明する。

ここでは、平面上の n 次 Bezier 曲線で構成される形状について、その曲線上の m 個の点 $L_{iv}(u_v)$ ($0 \leq v \leq m-1$) について、その類似形状の曲線の $L_{iv}(u_v)$ に対応する点の平均位置を点 N に制約する場合について説明する。ここで、 i はセグメント番号、そして、 u は曲線パラメータである。また、説明のために、 n 次 Bezier 曲線の第 i セグメントの曲線パラメータ u での曲線の k 階導関数 $Q_i^{(k)}(u)$ ($k \geq 0$) を、

$$\sum_{j=0}^n V_{i,j} B_{i,j}^{(k)}(u) = Q_i^{(k)}(u) \quad (1)$$

と表現する。ここで、 $V_{i,j}$ は Bezier 制御点、そして $B_{i,j}^{(k)}(u)$ は基底関数である。

2.2.1 曲線表現形式変換

ここでは、Bezier 曲線の表現形式の変換について説明する。表現形式変換後の曲線は、形状類似性変数、そして、形状類似性変数以外の形状を決定するための変数により表現される。その形状類似性変数以外の変数として、平均位置座標を制約する変数 N を用いる手法を以下に示す。

(1) 式から、曲線の通過点 $P_i(u)$ は、

$$\sum_{j=0}^n V_{i,j} B_{i,j}^{(0)}(u) = P_i(u) \quad (2)$$

である。(2) 式から、平均位置 N を求めるための、基準形状の曲線上の点 $L_{iv}(u_v)$ に対応する類似形状の曲線上の点 $K_{iv}(u_v)$ は、

$$\sum_{j=0}^n V_{i_v,j} B_{i_v,j}^{(0)}(u_v) = K_{iv}(u_v) \quad (3)$$

である。(3) 式を用いて、平均位置座標を制約する変数 N は、

$$\frac{1}{m} \sum_{v=0}^{m-1} K_{iv}(u_v) = N \quad (4)$$

となる。

ここで、通過制御点、形状類似性制御点に関する方程式、曲線接続のための方程式 ([8], [9] 参照)、そして、(4) 式による平均位置を制御する点に関する方程式を組み合わせ、曲線を表現する連立方程式を作成する。なお、一意に形状を決定することが可能なように、表現する曲線の自由度にあわせて連立方程式を構成する方程式を用意する。次に、この連立方程式を行列を用いて表現する。

$$U = MV \quad (5)$$

ここで、 V は $V_{i,j}$ を要素とするベクトル、 M は $B_{i,j}^{(k)}(u)$ または $B_{i,j}^{(k)}(u)$ に形状類似性を表現するための定数を演算した値を要素とする行列、そして、 U は表現形式を変換後の曲線を表現する変数を要素とするベクトルである。この (5) 式から、以下の曲線表現形式変換行列 M^{-1} を求める。

$$V = M^{-1}U \quad (6)$$

(6) 式は、形状類似性変数、そして、形状類似性変数以外の形状を決定するための変数により Bezier 制御点を表現している。

2.2.2 最適化計算

2.2.1 節で説明した曲線表現形式変換行列 M^{-1} が求められているとして、非均一相似形状を生成するための最適化計算について説明する。

Step 1 (2) 式および (6) 式を用いて、通過制御点に関する制約方程式を作成する。また、(1) 式および (6) 式を用いて、曲線接続に関する制約方程式を作成する。そして、(4) 式および (6) 式を用いて、平均位置制御点に関する制約方程式を作成する。これらは、 U を独立変数とする。また、追加の制約条件が必要な場合には、 U を独立変数として制約方程式を作成する。

Step 2 追加の形状類似性制御点が必要な場合は、 U を独立変数とした制約方程式を作成する。曲線の自由度の制限から、この制約は弱制約としなければならない。

Step 3 文献 [7] で述べている制約条件および (6) 式を用いて、 U を独立変数として形状を变形するための制約条件を作成する。

Step 4 作成した制約条件の線形連立方程式を、独立変数 U と追加の制約条件の弱制約変数を要素とするベクトルのノルムが最小値を持つように解く。

Step 5 (6) 式に U を代入して、Bezier 曲線の制御点を求める。

このように、Step 1 では、(5) 式を構成する方程式のうちの形状類似性制御点を表現する方程式以外の方程式を、(6) 式を用いて U を独立変数とするように変換したものを制約条件に含める必要がある。これにより、独立変数 U のうちの曲線通過点そして平均位置に関する制御変数などを定数に制約することになり、実質的に最適化されるのは、独立変数 U のうちの形状類似性変数、そして、弱制約の制約条件を追加した場合にはその弱制約変数となる。

3. 非均一相似形状の補間を用いた形状学習データ生成

3.1 形状学習データ生成アルゴリズム

ここでは、平均位置による制約、そして、乱数の生成範囲を制約点の近傍に制限しない 2 個の移動制約により、非均一相似変換を用いて、基準形状に類似な形状を生成する。そして、その生成した形状を補間することにより形状学習

データを生成するように、既提案の手法 [6] を拡張する。平面上の Bezier 曲線で表現した基準形状が与えられているとして、以下に概要を示す。

Step 1 類似形状の曲線通過点 $R_i (1 \leq i \leq 2)$ を制約するために、基準形状の曲線上の点 P_i を選択する。

Step 2 類似形状の曲線通過点の座標平均値 $N_j (1 \leq j \leq n)$ を制約するために、基準形状の曲線上の点 $L_{j,k} (1 \leq k \leq m)$ を選択し、その座標平均値 M_j を求める。そして、 M_j を移動した点 M'_j を求める。ここでの移動量は、形状をどのように洗練化するかにより決定する。

Step 3 点 P_i について、それぞれの移動制約点 R_i を確率的に生成する。

Step 4 直線 P_1P_2 を直線 R_1R_2 に相似変換するパラメータを用いて、直線 $P_1M'_j$ を相似変換する。その変換した直線の M'_j に対応する端点を、類似形状の曲線通過座標値の平均値の制約点 N_j とする。

Step 5 点 P_i を、それぞれ点 R_i に移動、そして、 $L_{j,k}$ に対応する類似形状の曲線通過座標値の平均値が点 N_j であるという制約条件のもとで、非均一相似変換を用いて類似形状を生成する。

Step 6 点 M_j と点 M'_j を補間する点を求め、平均位置制約補間点とする。

Step 7 移動制約補間点と Step 6 で求めた平均位置制約点について、Step 1 から Step 5 を繰り返す。

4. 実装と結果の検討

4.1 実装

提案アルゴリズムを、Java 言語により実装した。形状の表現には、平面上の 3 次 Bezier 曲線を用いた。曲線のグラフィックスデータとしての表現形式には、SVG を使用した。なお、図中の形状および印は、実装プログラムで生成したものを SVG 形式でファイル出力したものを EPS 形式に変換したものである。

4.2 類似形状生成のための制約条件

提案アルゴリズムでの学習データの生成では、非均一相似性制約を用いて基準形状を類似形状に形状洗練化をする。この形状洗練化は、曲線の表現形式の変換、そして、表現形式を変換した曲線の形状洗練化のための最適化計算の 2 つの段階で構成される。各段階の詳細については、佐藤等 [8], [9] に説明がある。

ここでは、各段階の具体例として、図 1 の形状について、曲線の表現形式変換で用いる方程式と最適化計算で用いる制約条件を表 1, 2 に示す。この形状は、3 次 Bezier 曲線で表現した閉じた形状である。曲線の表現形式変換では、曲線形状のセグメント、そして直線形状のセグメントについて、それぞれ、曲線の通過位置または非均一相似性、そし

て直線形状の曲線の通過位置と直線形状との類似性を表現する方程式を用いている．それに加えて，曲線上の点の平均位置を表現する方程式を用いている．そして，曲線形状のセグメントどうしの接続，そして，曲線形状のセグメントと直線形状のセグメントの接続について，それぞれ， C^2 連続性，そして， C^0 連続性を表現するための方程式を使用している．最適化計算では，曲線形状のセグメント，そして直線形状のセグメントについて，それぞれ，曲線の通過位置，そして直線形状の曲線の通過位置を制約するための方程式を用いている．それに加えて，曲線上の点の平均位置を制約する方程式を用いている．そして，曲線形状のセグメントどうしの接続，そして，曲線形状のセグメントと直線形状のセグメントの接続について，それぞれ， C^2 連続性，そして， C^0 連続性を制約するための方程式を用いている．それに加えて，3 階導関数ベクトルのノルムを弱制約するための方程式を用いている．このようにして，最適化をする変数は，非均一相似性変数と直線形状からの差を表現する変数となる．

またこれ以降に示す例では，この基準形状を基にして類似形状を生成した．

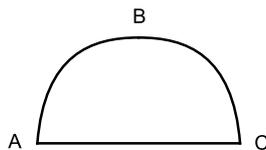
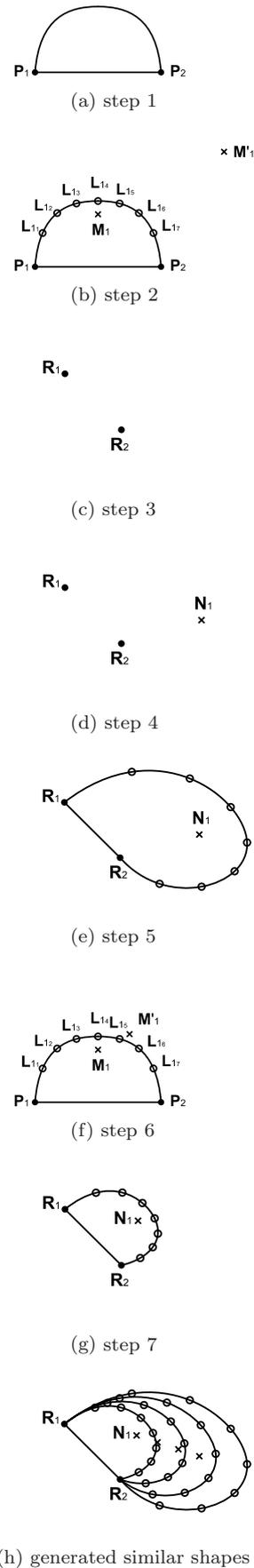


図 1 形状洗練化をする基準形状の例

Fig. 1 An example of a base shape to be refined.

4.3 形状学習データ生成の例

図 2 は，基準形状 (図 2(a)) から類似形状 (図 2(h)) を生成する例である．図 2(a,b,c,d,e,f,g) は，3 節でのアルゴリズムの各段階である．図 2(h) は，3 節でのアルゴリズムを繰り返して生成した形状である．図 2(a) では，基準形状の曲線上の移動制約点 P_1, P_2 () を選択している．図 2(b) では，基準形状の曲線上の点 $L_{j,k}$ ($j = 1, 1 \leq k \leq 7$) () を選択し，その座標平均値 M_1 そして $M'_1(x)$ を求めている．図 2(c) では，点 P_1, P_2 について，それぞれの移動制約点 R_1, R_2 () を確率的に生成している．図 2(d) では，3 節 Step4 の手法を用いて，類似形状の曲線通過座標値の平均値の制約点 $N_1(x)$ を求めている．図 2(e) では，点 P_1, P_2 を，それぞれ点 R_1, R_2 に移動，そして， $L_{1,k}$ に対応する類似形状の曲線通過座標値の平均値が点 N_1 であるという制約条件のもとで，非均一相似変換を用いて類似形状を生成している．ここでの 印は，生成した類似形状で $L_{1,k}$ に対応する点である．この 印の点の座標値の平均値が， N_1 となるように類似形状を生成している．図 2(f) では，補間平均位置制約点を求めている．図 2(g) では，補間平均位置制約点にあわせて類似形状を生成している．図 2(h) は，



(h) generated similar shapes

図 2 形状学習データ生成の例

Fig. 2 An example of similar shape generation steps.

補間平均位置制約点にあわせた類似形状生成を繰り返して生成した類似形状である。これにより、制約条件を満たしながら、基準形状に類似な形状を生成可能であることがわかる。

4.4 生成した形状学習データの曲率プロットの例

学習データとして生成した類似形状の曲率プロットの例を、図 4.4 に示す。図 4.4(a) が基準形状である。この基準形状から、図 4.4(b) の類似形状を生成している。図 4.4(c) は、図 4.4(a,b) の形状の曲率プロットである。それぞれの形状を表現する曲線を長さについて等分割する点での曲率値を計算している。図 4.4(d,e,f) と図 4.4(g,h,i) についても同様である。これらの形状について、それぞれの基準形状とその類似形状の曲率の変化を比較すると、曲率が同様の変化をしていることがわかる。このことから、これらの生成した類似形状は、基準形状の形状特徴を維持しているといえるので、基準形状を機械学習するための学習データとして適当である。

4.5 曲率特徴ベクトルの次元縮退の例

生成した形状学習データの曲率特徴ベクトルを次元縮退した 2 次元特徴ベクトルを、散布図としてプロットしたものを、図 4, 図 5 に示す。図 4.4(a,b,c) と同様の形状洗練化をした基準形状とその類似形状を Type 1, 図 4.4(d,e,f) と同様の形状洗練化をした基準形状とその類似形状を Type 2, そして、図 4.4(g,h,i) と同様の形状洗練化をした基準形状とその類似形状を Type 3 としている。それぞれの形状について、形状を表現する曲線を長さについて 100 等分する点での曲率値を要素とする特徴ベクトル (101 次元) を、2 次元特徴ベクトルに次元縮退 [11] している。図 4(a,b,c) は、それぞれ、Type 1 と Type 2, Type 2 と Type 3, そして Type 1 と Type 3 について次元縮退したものをプロットしている。図 5(a,b) は、Type 1, Type 2 そして Type 3 を次元縮退したものである。また、プロットした点の近傍に、その特徴ベクトルに対応する形状を表示している。ここで用いた次元縮退では、2 次元曲率特徴ベクトルをプロットした 2 点間の距離が小さいほど、その点に対応する形状が類似した特徴をもっていることを表わしている。そのことを考慮して、これらのプロットした点の位置とその対応する形状を比較すると、次元縮退した曲率特徴ベクトルが形状の特徴を表現していると考えることができる。これにより、生成した形状は、機械学習の学習データとして用いることが可能であることがわかる。

5. おわりに

機械学習を用いて形状を認識するために、対話的に入力した少数の形状をもとにして学習データを生成するための既提案の手法 [6] を拡張したアルゴリズムを提案した。

提案手法の特徴は、非均一相似性制約 [8] を用いて、基準形状から類似形状を生成することである。その特徴をいかして、基準とする形状に類似な形状を補間する形状を生成するようにアルゴリズムを拡張した。そのとき、補間形状の生成を容易にするために、曲線通過点の座標値の平均値を制約するための手法を提案した。また、生成した形状学習データを、次元縮退 [11] を用いて分類することにより、生成した形状学習データの形状類似性を機械学習のアルゴリズムで取り扱うことが可能であることを示した。

提案手法は、機械学習の学習空間モデリングのためのデータを生成する手法として有効であるばかりではなく、学習空間のテストデータを生成する手法としても用いることが可能である。今後の課題には、学習データ生成の自動化のための研究を挙げることができる。

参考文献

- [1] Bartels, R. and Forsey, D.: Constraint Based Curve Manipulation, *Tutorial Notes: Splines in Computer Graphics prepared for Eurographics '94*, pp. 31–36 (1994).
- [2] Bartels, R. H. and Beatty, J. C.: A Technique for the Direct Manipulation of Spline Curves, *Graphics Interface 89*, pp. 33–39 (1989).
- [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] Forsey, D. R. and Bartels, R. H.: Hierarchical B-spline refinement, *SIGGRAPH '88: Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, New York, NY, USA, ACM, pp. 205–212 (online), DOI: <http://doi.acm.org/10.1145/54852.378512> (1988).
- [5] Fu, L. and Kara, L. B.: Neural network-based symbol recognition using a few labeled samples, *Computers & Graphics*, Vol. 35, No. 5, pp. 955 – 966 (online), DOI: 10.1016/j.cag.2011.07.001 (2011).
- [6] 佐藤 信：形状認識のための非均一相似性制約を用いた学習データの対話的生成，研究報告グラフィクスと CAD (CG)，Vol. 2012-CG-149, No. 19, pp. 1–7 (2012).
- [7] 佐藤 信，三輪譲二：接続の類似性制約を用いた曲線洗練化法，情報処理学会研究報告，Vol. 2009-CG-137, No. 7, pp. 1–6 (2009).
- [8] 佐藤 信，三輪譲二：導関数ベクトルの非均一相似性制約に基づく曲線洗練化法，情報処理学会研究報告-グラフィクスと CAD，Vol. 2011-CG-142, No. 12, pp. 1–6 (2011).
- [9] 佐藤 信，三輪譲二：平面曲線形状洗練化のための導関数ベクトルの非均一相似性制約を用いた鏡映対称変換，情報処理学会研究報告-グラフィクスと CAD，Vol. 2012-CG-146, No. 34, pp. 1–6 (2012).
- [10] Nonnemaker, J. and Baird, H.: Using synthetic data safely in classification, *Proceedings of SPIE - The International Society for Optical Engineering*, Vol. 7247 (2009).
- [11] van der Maaten, L.: Learning a Parametric Embedding by Preserving Local Structure, *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics (AI-STATS)*, pp. 384–391 (2009).

表 1 図 1 の基準形状を用いた学習データ生成で用いた方程式 (曲線表現形式変換)

Table 1 The equations used in training set generation with the base shape in Fig. 1 (rerepresentation).

components	equations (t : curve parameter)
the curve AB	positional constraints ($t = 0$), similarity constraints ($t = 0.5$) (1st. and 2nd. derivatives)
the curve BC	similarity constraints ($t = 0.5$) (1st. and 2nd. derivatives)
the curve AB and BC	a constraint on mean position (the mean value of 7 positions, curve AB ($t = 0.25, 0.5$ and 0.75)), and curve BC ($t = 0, 0.25, 0.5$ and 0.75)).
the straight line CA	positional constraints ($t = 0$), 1st. and 2nd. derivatives minimization ($t = 0.5$)
the connection (between the two curves)	C^2 continuity
the connections (between the straight line and the curve)	point connection

表 2 図 1 の基準形状を用いた学習データ生成で用いた方程式 (最適化計算)

Table 2 The equations used in training set generation with the base shape in Fig. 1 (optimization).

components	equations (t : curve parameter)
the curve AB	positional constraints ($t = 0$)
the curve AB and BC	a constraint on mean position (the mean value of 7 positions, curve AB ($t = 0.25, 0.5$ and 0.75)), and curve BC ($t = 0, 0.25, 0.5$ and 0.75)).
the straight line CA	positional constraints ($t = 0$), 3rd. derivatives minimization ($t = 0.5$)
the connection (between the two curves)	C^2 continuity
the connections (between the straight line and the curve)	point connection

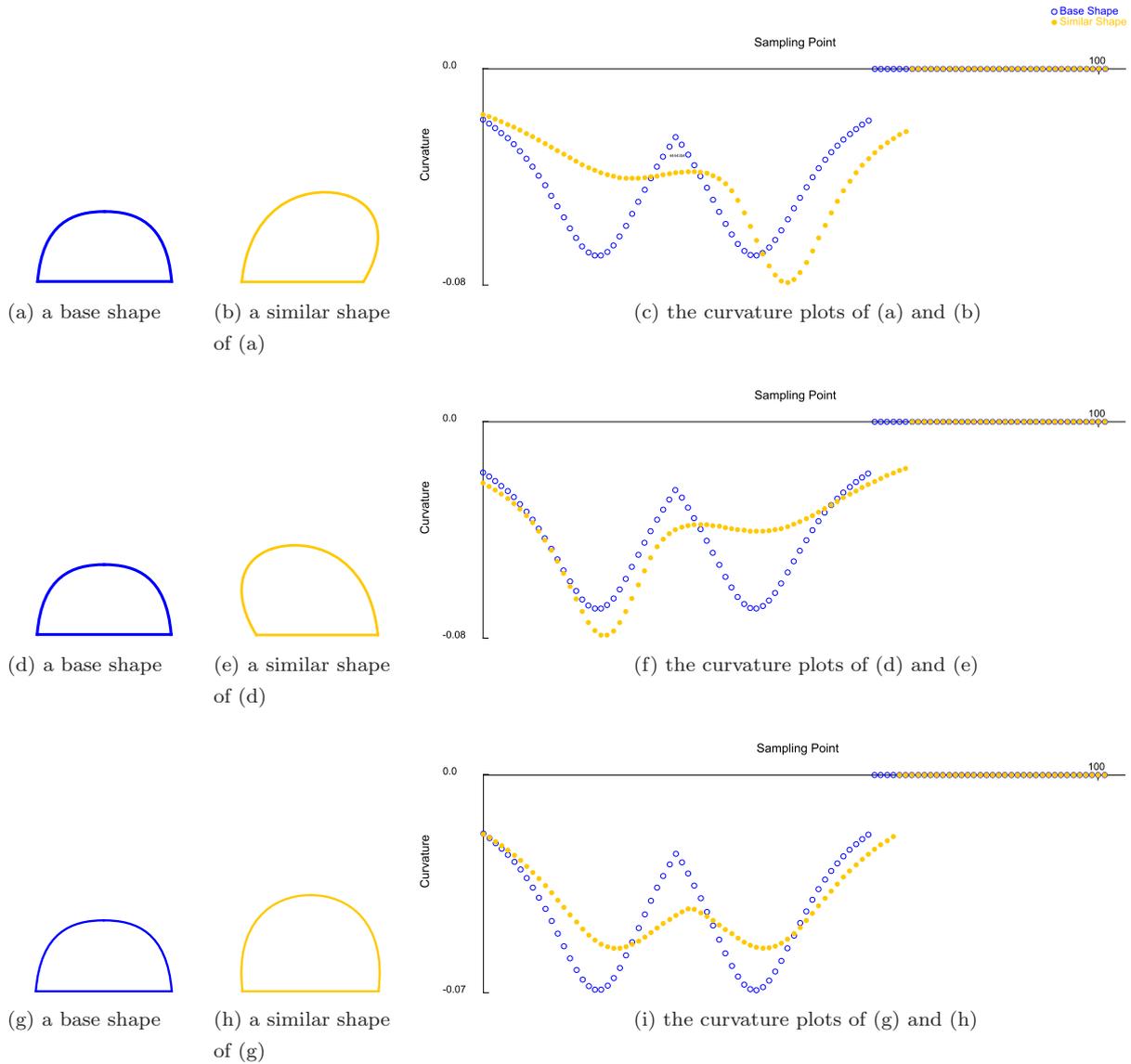


図 3 生成した形状学習データの曲率プロットの例

Fig. 3 Examples of the curvature plots of generated shapes.

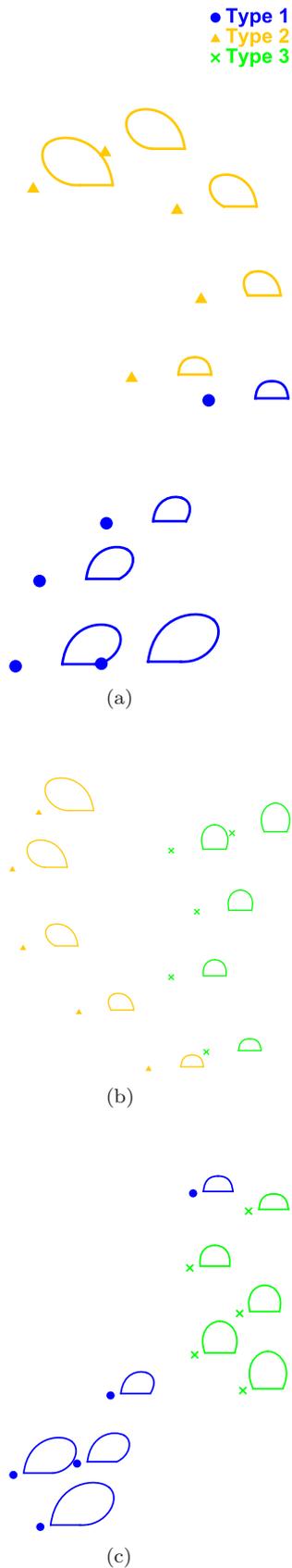


図 4 曲率特徴ベクトルの次元縮退の例 (1)

Fig. 4 Examples of the reduced curvature feature vectors of generated shapes (1). (a), (b), and (c) are respectively the scatter plots of reduced feature vectors of shape type 1 and 2, shape type 2 and 3, and shape type 1 and 3.

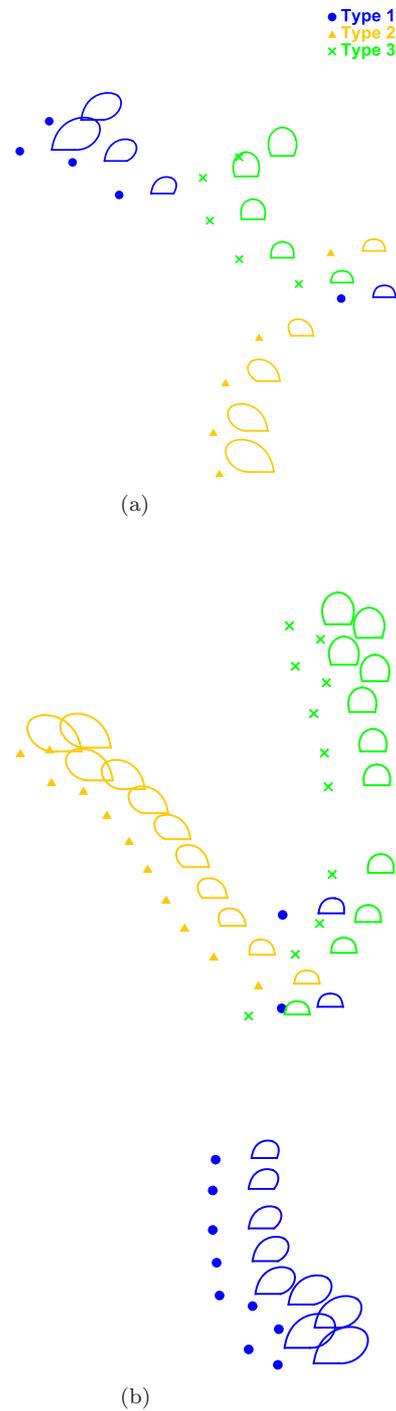


図 5 曲率特徴ベクトルの次元縮退の例 (2)

Fig. 5 Examples of the reduced curvature feature vectors of generated shapes (2). (a) and (b) are respectively the scatter plots of reduced feature vectors of 5 shapes for each shape type and 11 shapes for each shape type.