

非均一相似形状を学習したオートエンコーダを用いた 形状補間

佐藤 信¹

概要: 補間しようとする曲線の特徴を表現する 2 次元空間を用いた曲線の補間手法を提案する。提案では、非均一相似変換を用いて、補間しようとする曲線に形状が類似な曲線を生成する。そして、生成した曲線を学習データとして、オートエンコーダを用いて曲線の特徴を表現する空間を生成する。これにより、曲線を表現する多次元ベクトルと、その特徴を維持する 2 次元ベクトルとの写像が得られる。この 2 次元ベクトルを、曲線補間のための形状変形の制御ベクトルとして用いることが可能である。

Shape Interpolation Using Autoencoders Trained on Non-uniform Similar Shapes

MAKOTO SATOH¹

Abstract: This paper presents a method for curve interpolation using the two-dimensional space which represents the features of interpolated curves. In the method, first the shapes similar to interpolated curves are generated. Then the feature space of the curves is generated by autoencoders using generated curves as learning data. This enables the mapping of the multi-dimensional vectors which represent the curves to the two-dimensional vectors which represent the curve features. These two-dimensional vectors will be used as the control vectors for shape deformation in shape interpolation.

1. はじめに

本稿では、曲線の形状特徴を表現する 2 次元特徴ベクトル空間を生成するための手法、および、生成した特徴ベクトル空間を用いて曲線の形状を補間するための手法を提案する。対象とする形状は、平面上の Bezier 曲線で表現された形状である。

提案手法では、ディープオートエンコーダを用いた機械学習により、曲線を表現する多次元ベクトル空間と 2 次元特徴ベクトル空間との写像を求める。機械学習での学習データとしては、非均一相似変換を用いて生成した、補間しようとする曲線に形状が類似な曲線を用いる。

手法の特徴は、曲線を表現する多次元ベクトルを、形状の特徴を維持するように 2 次元ベクトル空間に写像することが可能なことである。与えられた形状の特徴を維持する

ように生成された 2 次元空間でのベクトルを、制御ベクトルとして用いることにより、直感的な操作により補間形状を生成することが可能である。

これ以降の構成について、簡単に説明する。2 節では、関連研究について説明をおこなう、そして、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、ディープオートエンコーダを用いた形状補間の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 節で説明する。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. 関連研究と提案手法の概要

2.1 関連研究

提案手法では、データの次元削減を用いて形状の補間をおこなうので、始めに、高次元データの次元を削減するための研究について述べる。

高次元データを、可能な範囲でその特徴を維持したまま

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

で、低次元空間に写像することを目的として、多くの手法が提案されている。Sarveniazi [12] は、それらの手法について紹介をしている。Hinton 等 [6] は、ディープオートエンコーダ (deep autoencoder) を用いて、高次元データの次元を削減するための手法を提案している。そこでは、Restricted Boltzmann Machine(RBM) により、ディープオートエンコーダを構成している。そして、画像データを用いて、入力画像に近い出力画像を得るように学習することにより、ディープオートエンコーダが次元削減の性能にすぐれていることを示している。RBM を、実数値の入力データに対して用いる場合には、工夫が必要である。Chen 等 [2] は、実数値を入力として用いるために、Continuous Restricted Boltzmann machine(CRBM) を提案している。

また、データの次元を削減する場合に、高次元データのどのような特徴を維持するべきであるのかは、次元を削減した低次元データをどのような目的に用いるのかに依存する。van der Maaten [13] は、高次元データの次元を削減して2次元空間に配置することにより、データの特徴を可視化することを目的とする非線形手法を提案している。線形の次元削減手法では、類似度の小さいデータを遠くに配置することが可能であるが、類似度の大きいデータが近接しすぎる場合がある。van der Maaten [13] では、その点について改良がおこなわれている。

次に、形状の特徴を把握して形状を操作するための研究について述べる。

形状の特徴を可能な限り維持したままで、形状を操作するための研究には、Fowler 等 [4], Welch 等 [14], Igarashi 等 [8], および、佐藤等 [10], [11] がある。これらの研究では、それぞれの手法で形状の特徴を表現するための基準を提案している。そして、与えられた変形のための制約条件を満たすように、その基準量の変化が最小になるように形状を操作している。

最後に、形状の学習に関連する研究について述べる。

形状を学習するための学習データを生成するための研究には、Fu 等 [5] がある。与えられた形状の画像データに類似な形状の画像データをアルゴリズムにより生成して、それらの生成した画像を学習データとするための研究である。また、佐藤 [9] では、Bezier 曲線で表現された基準の形状を与えて、その基準の形状に類似な形状を生成して学習データとするための手法を提案している。

与えられた形状に類似な形状を生成するために、機械学習の手法を用いる研究には、Campbell 等 [1] がある。アルファベットの文字ごとに、複数の書体のフォントデータを学習することにより、学習データとして用いたフォントの形状を補間する形状を生成する手法を提案している。

2.2 提案手法の概要と適用分野

2.2.1 概要

本研究では、機械学習を用いて、Bezier 曲線で表現された形状を補間する形状を生成するための手法を提案する。

アルゴリズムは、3段階により構成される。始めの段階では、非均一相似変換を用いて、与えられた基準の形状に類似な形状を生成する。これらの形状データを、以降の段階での形状学習データとする。次の段階では、ディープオートエンコーダを用いて、形状学習データを学習することにより、形状データを表現する低次元空間を生成する。最後の段階では、生成した低次元空間上の点から、学習した形状データの次元に等しい次元のベクトルをデコードすることにより、学習した形状に類似な補間形状を生成する。

ここで用いるディープオートエンコーダでは、van der Maaten [13] において提案されているパラメトリック tSNE と同様のプリトレーニングをおこなう。van der Maaten [13] では、RBM を用いているが、本研究では実数値データを用いるために、CRBM(Chen 等 [2]) を用いてプリトレーニングをおこなう。そして、その学習パラメータを用いて、Hinton 等 [6] と同様のディープオートエンコーダを構成する。

なお、非均一相似変換とは、Fowler 等 [4] の提案した曲線の形状洗練化 [3] のための手法を拡張したものである。非均一相似変換を用いると、平面上の Bezier 曲線で表現した基準形状を、曲線の通過点などの変形のための制約条件を満たしながら、可能な限り基準形状の特徴を維持して形状を洗練化することが可能である。

2.2.2 適用分野

提案手法を用いると、形状データを低次元空間で表現することが可能となる。このことにより、形状の操作を必要とする多くの分野への適用が可能であるといえる。特に、生成した低次元空間が形状の特徴の関連を表現することから、与えられた形状を補間する、類似な形状を直感的な操作で生成するためのユーザインタフェースとして用いることが可能である。

3. ディープオートエンコーダを用いた形状補間

3.1 アルゴリズム

形状補間手法の各段階を、以下に示す。

Step1:学習データ生成 与えられた形状を基準として、それに類似な形状を、非均一相似変換を用いて生成する。
Step2:形状学習 Step1 で生成した形状データを学習データとして、ディープオートエンコーダを用いて形状を学習する。これにより、学習した形状データとその2次元空間での表現とを写像するパラメータが得られる。
Step3:補間形状生成 Step2 で学習したオートエンコー

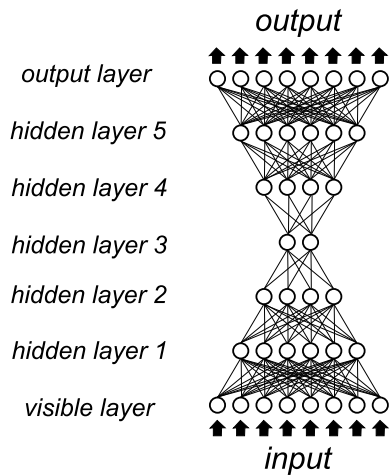


図 1 ディープオートエンコーダの例
Fig. 1 An Example of deep autoencoder.

ダのデコード部を用いて、学習データを補間する類似な形状を生成する。

本節のこれ以降では、Step2,3 について説明する。

3.2 形状学習

ここで用いるディープオートエンコーダとは、複数の隠れ層を持つニューラルネットで構成したオートエンコーダのことである。オートエンコーダを用いる学習では、入力層に与えるデータと同様の出力データを出力層から得るようにニューラルネットのパラメータを学習する。それにより、学習データの特徴を捉えようとするものである。図 1 に、ディープオートエンコーダの構成例を示す。

提案手法では、CRBM(Chen 等 [2]) を用いてディープオートエンコーダを構成する。[2] で提案されている CRBM のための活性化関数では、パラメータ a によりシグモイド関数の傾きをニューロンごとに調整をおこなう。図 2 に示すように、 a の値として、前向き伝搬と後ろ向き伝搬とで、それぞれ、 a_f と a_b というように異なる値を用いることが可能である。また、どの程度、学習を繰り返すのかについては、複数の方式が可能である。ここでは、次式で与えられる誤差 e の値が、予め与えた値より小さい場合に繰り返しを終了する。

$$e = \frac{1}{N} \sum_{n=1}^N \sum_i |v| (v_{n_i}^2 - \hat{v}_{n_i}^2). \quad (1)$$

ここで、 N 、ベクトル v と $\hat{v}$ は、それぞれ、CRBM の入力データ数、入力データベクトルとその推定値のベクトルである。ここでの推定値とは、CRBM でエンコードした値をデコードして得られる値のことである。なお、ディープオートエンコーダの学習はプリトレーニングと微調整の各段階により構成されるが、ここではプリトレーニングのみをおこなう。

CRBM で構成したディープオートエンコーダを用いた

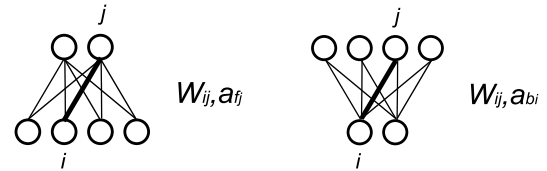


図 2 CRBM の例

Fig. 2 Examples of the parameters of CRBM. left: forward propagation through the edge from i to j , and right: backward propagation through the edge from j to i .

形状学習の各段階を以下に示す。

Step1:プリトレーニング オートエンコーダの隣接する 2 層のそれぞれについて、CRBM を訓練する。このとき、入力層と 1 番目の隠れ層により構成される、1 番目の CRBM の入力データには、形状データを用いる。そして、1 番目と 2 番目の隠れ層により構成される 2 番目の CRBM の入力データには、1 番目の CRBM の出力を用いる。残りの CRBM については、2 番目の CRBM と同様に、前段の CRBM の出力を入力として用いる。

Step2:形状学習 Step1 で訓練した CRBM のパラメータ w, a_f そして a_b を用いて、ディープオートエンコーダを構成する。エンコーダの構成には、 w そして a_f を用い、デコードの構成には w そして a_b を用いる。ここで、 w, a_f そして a_b は、それぞれ、ニューラルネットの各素子間の重み、前向き伝搬で用いるシグモイド関数の傾きを調整する係数、そして、後ろ向き伝搬で用いるシグモイド関数の傾きを調整する係数である。また、バイアスは、入力値が 1 である入力に対する値として取り扱うので、バイアス項の係数は重みに含めている。

3.3 補間形状の生成

3.2 節の手法で訓練したディープオートエンコーダを用いて、補間形状を生成する手法について述べる。

ディープオートエンコーダを構成する隠れ層のうちで、エンコード部の最後の層であり、そして、デコード部の最初の層でもある隠れ層の素子数が 2 であるとする。このような構成のディープオートエンコーダを、形状データを入力として 3.2 節の手法で訓練することにより、形状を表現する入力データと、2 次元空間での表現を対応づける写像が得られる。言い換えると、形状データの 2 次元空間での表現は、エンコード部の最後の層の要素の値として表現されていることになる。

これにより、2 次元ベクトルを制御ベクトルとして、それに対応する形状を得るためには、デコード部の最初の層に、ある値を入力し、それに対応するデコード部の出力を形状データとして取り扱えばよいということになる。

表 1 定数値を与えたパラメータ

Table 1 The parameters that are given constant values.

learning rate	momentum	weight cost	σ
each value ¹	0.5 ($iter. \leq 5$), 0.9 ($iter. > 5$)	0.00001	0

¹ This parameter is provided as a constant value in each computation.

そして、学習形状の補間形状を得るための手順は以下のようになる。

Step1:エンコード 形状学習データに対応するエンコード部の最後の層の出力の2次元ベクトルを求める。

Step2:デコード Step1 で求めたベクトルを補間するベクトルをデコード部の最初の層に入力し、その出力として補間形状を得る。

4. 実装と結果の検討

3節でのアルゴリズムを実装して、提案手法の有効性を確認した。特に、ディープオートエンコーダの訓練での学習係数 (learning rate) と補間形状との関係について検討をおこなった。Hinton[7] では、ディープラーニングに用いるRBMの学習係数の調整方法について述べているが、ここで用いるCRBMにおいても同様の結果が得られるのかについて確認した。

4.1 訓練データ

学習に用いた形状は、平面上の3次Bezier曲線(1セグメント)により表現された形状である。そのBezier曲線の4個の制御点の値(x座標値とy座標値により構成されるので8個の値)を、[0,1]の範囲にスケーリングをおこなったものをディープオートエンコーダの訓練データとした。ディープオートエンコーダには、それ以外のデータは与えていない。

4.2 パラメータ

ここで、これ以降のCRBMの学習において予め定数値を与えたパラメータを、表1に示す。なお、表中の σ は、Chen等[2]の(3)式のパラメータである。本稿の実験では、CRBMの入力にノイズのないデータを用いるので、 $\sigma = 0$ とした。また、学習アルゴリズムにより学習したパラメータは、CRBMの各層を構成する素子間の結合を表す重み w (バイアスを含む)、活性化関数のスロープを調整するための係数 a_f, a_b である。 a_f, a_b は、それぞれ、前向き伝搬そして後ろ向き伝搬での活性化関数で用いる。

4.3 補間形状の生成

ディープオートエンコーダを用いて、形状を学習し、そ

の形状の補間形状を生成した。図3, 4, 5に生成した形状を示す。訓練に用いた形状を、第1行と第1列に示す(青色)。1行*i*列の形状と*i*行1列の形状は同一のものである。学習後のディープオートエンコーダを用いて訓練形状を再構成した形状を、対角線上のセルに示す(オレンジ色)。*i*行*i*列の形状は、1行*i*列の訓練形状(青色)を再構成したものである。左上の三角形の領域の形状は、学習後のディープオートエンコーダを用いて再構成した補間形状である(オレンジ色)。*i*行*j*列の形状は、*i*行*i*列の形状と*j*行*j*列の形状を補間した形状である。これらの訓練形状と補間形状を比較し易いように、右下の三角形の領域に、訓練形状(緑色)とその補間形状(オレンジ色)を示す。行と列の対応については、同様である。見易さを考えて、訓練形状の表示色を変更している。

補間形状の生成では、補間する訓練データをエンコードすることにより2次元空間での写像点を求め、その写像点の2点どうしについて、その中点をデコードして補間形状の制御点を求めた。

それぞれの実験で用いたディープオートエンコーダは、隠れ層が5層のものをを用いた。入力層と出力層を含めた各層の素子数は、8-6-4-2-4-6-8である。素子数が2である層までの前半部がエンコーダであり、素子数が2である層からの後半部がデコーダである。

図3, 4は、線形補間を用いて生成した訓練形状を学習したディープオートエンコーダを用いて生成した補間形状である。重みの学習係数は、それぞれ、0.5と1.0である。図5は、非均一相似変換を用いて生成した非均一相似形状を学習したディープオートエンコーダを用いて生成した補間形状である。重みの学習係数は、0.005である。そして、各ディープオートエンコーダのCRBMの学習を、(1)式の値が、0.0001より小さい値になるまで繰り返した。ただし、図5で用いたディープオートエンコーダの隠れ層の第2,3層で構成されるCRBMについては、予め与えた繰り返しの最大数である1,000,000回に達するまで学習を繰り返した。

4.4 検討

4.3節の結果から、訓練データを再構成した形状については、比較的良好な形状が得られることが分かる。また、一部分の補間形状は、補間形状として良好であるが、多くの補間形状は、補間をする形状の間にあるものの、補間をする形状のいずれか一方の形状に類似した形状であるので、補間形状として十分であるとはいえない。学習空間内でのそれらの形状の写像点の近傍では、クラスタリングをするための学習空間が生成されている可能性がある。

図3と4を比較すると、学習係数が小さいほうが、訓練データ以外の形状についても比較的良好なエンコードが可能なが多いことが分かる。このことは、Hinton[7]で

述べられている，ディープラーニングに用いるRBMの学習係数についての指摘と一致している．

図4では，補間の基準とする訓練データのいずれか一方に近い補間形状であることが多い．これは，形状をクラスタリングしているような結果といえる．

なお，訓練形状の2次元写像点の midpoint に対応する補間形状として，基準とする訓練形状からおおきく異なる形状が生成される例はなかった．さらに，それ以外の2次元空間に写像した点についても，デコードされる補間形状を検討する必要がある．

補間形状を改善するためには，補間形状が十分でない形状の類似形状を生成して，それを訓練データとして，再学習をおこなうことが考えられる．その場合には，アルゴリズムにより類似形状を生成することが可能であることから，非均一相似変換が有効であるといえる．

5. おわりに

曲線の形状特徴を表現する2次元特徴ベクトル空間を生成するための手法，および，生成した特徴ベクトル空間を用いて曲線の形状を補間するための手法を提案した．

平面上のBezier曲線で表現された形状を基準として，それに類似な形状を，非均一相似変換を用いて生成した．それらの生成した形状を学習データとして，ディープオートエンコーダを用いた機械学習により，曲線を表現する多次元ベクトル空間と2次元特徴ベクトル空間との写像を求めた．そして，生成した2次元ベクトル空間上での訓練形状の対応点を基にして，訓練形状の補間点を求めた．その補間点をデコードすることにより，訓練形状の補間データを生成した．

提案の手法を用いて，Bezier曲線の2次元の特徴空間を得ることが可能であることを示すことができた．しかし，精度については，さらに改良が必要であるといえる．今後の課題には，精度の改良，計算時間の減少，そして，複雑な形状への適用などに関する研究を挙げることができる．

参考文献

- [1] Campbell, N. D. F. and Kautz, J.: Learning a Manifold of Fonts, *ACM Trans. Graph.*, Vol. 33, No. 4, pp. 91:1–91:11 (online), DOI: 10.1145/2601097.2601212 (2014).
- [2] Chen, H. and Murray, A.: Continuous restricted Boltzmann machine with an implementable training algorithm, *Vision, Image and Signal Processing, IEE Proceedings -*, Vol. 150, No. 3, pp. 153–158 (online), DOI: 10.1049/ip-vis:20030362 (2003).
- [3] 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).
- [4] Fowler, B. and Bartels, R.: Constraint-based curve manipulation, *Computer Graphics and Applications*,

- IEEE*, Vol. 13, No. 5, pp. 43–49 (online), DOI: 10.1109/38.232098 (1993).
- [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] 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.: A Practical Guide to Training Restricted Boltzmann Machines, *Neural Networks: Tricks of the Trade - Second Edition*, Vol. 7700, Springer, pp. 599–619 (2012).
- [8] Igarashi, T., Moscovich, T. and Hughes, J. F.: As-Rigid-As-Possible Shape Manipulation, *ACM SIGGRAPH 2005*, ACM, pp. 1134–1141 (2005).
- [9] 佐藤 信：非均一相似形状補間を用いた形状学習データ生成，情報処理学会研究報告，Vol. 2013-CG-150, No. 11, pp. 1–8 (2013).
- [10] 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，情報処理学会研究報告，Vol. 2011-CG-142, No. 12, pp. 1–6 (2011).
- [11] 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，Vol. 2012-CG-146, No. 34, pp. 1–6 (2012).
- [12] Sarveniazi, A.: An Actual Survey of Dimensionality Reduction, *American Journal of Computational Mathematics*, Vol. 4, No. 2, pp. 55–72 (2014).
- [13] 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).
- [14] Welch, W. and Witkin, A.: Variational surface modeling, *SIGGRAPH '92: Proceedings of the 19th annual conference on Computer graphics and interactive techniques*, New York, NY, USA, ACM, pp. 157–166 (online), DOI: <http://doi.acm.org/10.1145/133994.134033> (1992).

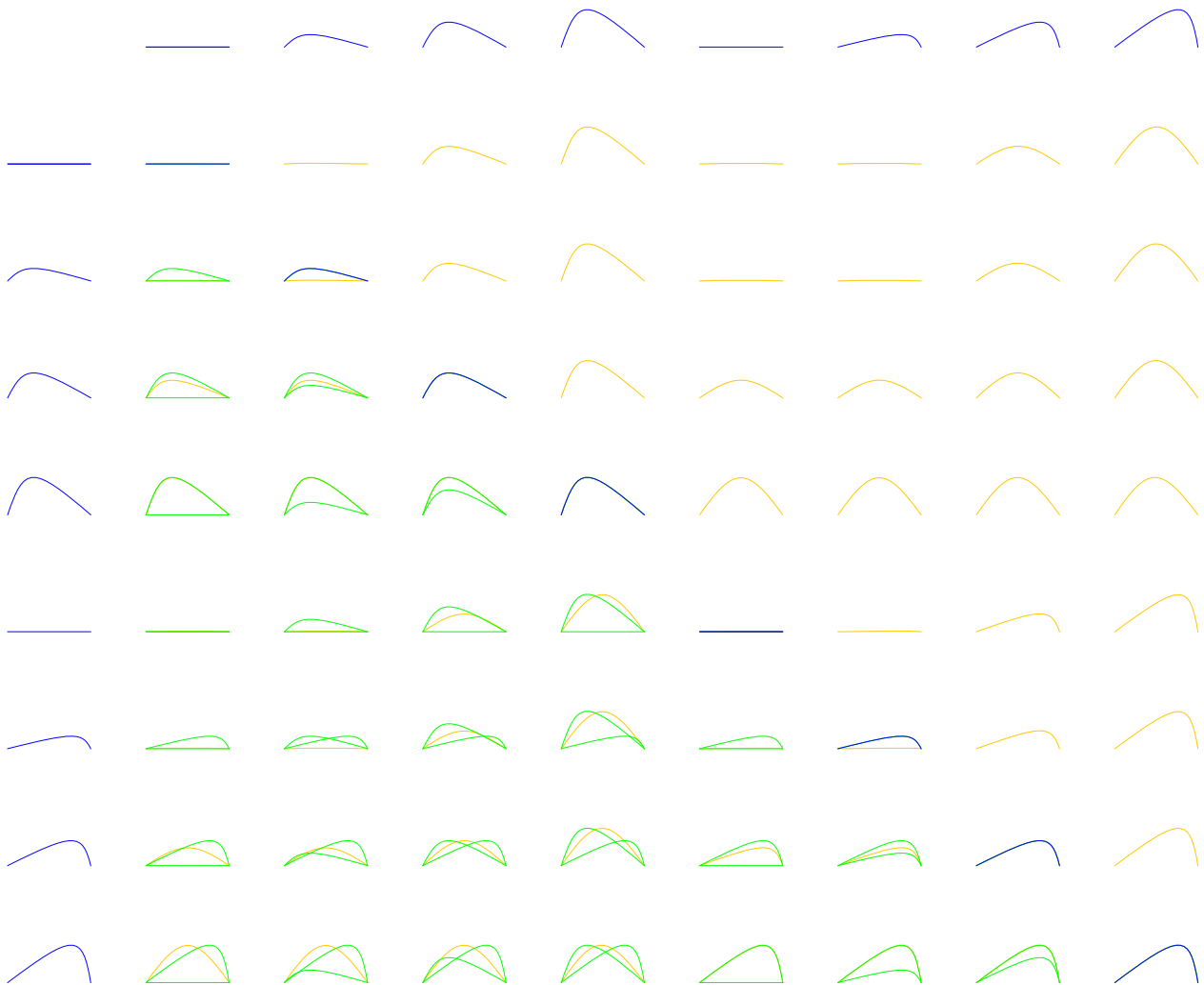


図 3 線形補間をした類似形状を用いた補間形状 (重みの学習係数 0.5)

Fig. 3 Interpolating shapes using linearly interpolated similar shapes (learning rate of weight 0.5). First row and left column: Training shapes are shown in blue. Diagonal cells: Reconstructed training shapes are shown in orange, training shapes are shown in blue. Upper right triangular cells: Interpolating shapes are shown in orange. Lower left triangular cells: Training shapes and interpolating shapes are shown together: training shapes in green, interpolating shapes in orange. The color of training shapes are changed so as not to impair easiness of view. The interpolating shape in row i and column j is interpolating the shapes in row i and column i , and row j and column j .

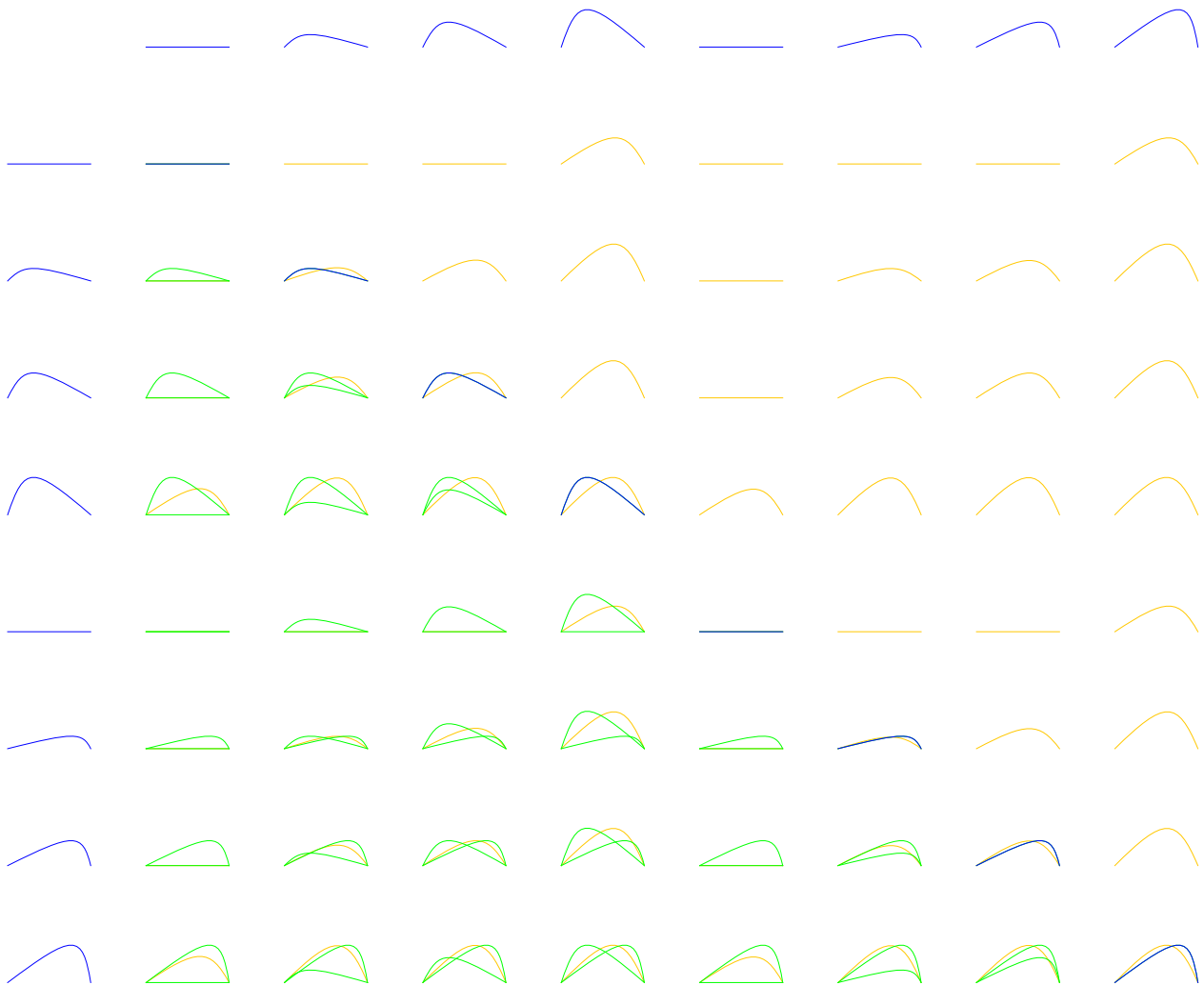


図 4 線形補間をした類似形状を用いた補間形状 (重みの学習係数 1.0)

Fig. 4 Interpolating shapes using linearly interpolated similar shapes (learning rate of weight 1.0). First row and left column: Training shapes are shown in blue. Diagonal cells: Reconstructed training shapes are shown in orange, training shapes are shown in blue. Upper right triangular cells: Interpolating shapes are shown in orange. Lower left triangular cells: Training shapes and interpolating shapes are shown together: training shapes in green, interpolating shapes in orange. The color of training shapes are changed so as not to impair easiness of view. The interpolating shape in row i and column j is interpolating the shapes in row i and column i , and row j and column j .



図 5 非均一相似形状を用いた補間形状

Fig. 5 Interpolating shapes using non-uniform similar shapes. First row and left column: Training shapes are shown in blue. Diagonal cells: Reconstructed training shapes are shown in orange, training shapes are shown in blue. Upper right triangular cells: Interpolating shapes are shown in orange. Lower left triangular cells: Training shapes and interpolating shapes are shown together: training shapes in green, interpolating shapes in orange. The color of training shapes are changed so as not to impair easiness of view. The interpolating shape in row i and column j is interpolating the shapes in row i and column i , and row j and column j .