

幼児向け折り紙作品の創作支援システム

鶴田直也[†] 三谷純^{†,††}
金森由博[†] 福井幸男[†]

1枚の正方形の紙を折ることで形を作り出す「折り紙」は、日本に古くから伝わる遊びであり、我々の多くが幼少期に体験するものである。近年では、折り紙の数理に関する研究の成果により、複雑な折り紙作品を創作するための技術が発達してきている。しかし、まだ手指を正確に動かす能力が十分に発達していない幼児らを対象とした、少ない手順で簡単に折れる折り紙作品の創作については、これまでに十分な研究がされてこなかった。そこで本稿では、4回以下の折り操作で作ることができる単純な折り紙作品を、新しく創作するためのシステムを提案する。本システムは、あらかじめ4回までの折り操作で作ることができる折り紙の形を列挙し、それをデータベースに蓄えておく。ユーザが動物の顔など、折り紙で作りたい形をシステムに入力すると、それに類似した形をデータベースから検索し、折り方を提示する。近年の計算機の性能の向上により、このような力づくのアプローチでも目的を達成することが可能となった。幼児向けの折り紙作品に見られる「目の追加」を実装するなど、より実用的なシステムを構築した。

A system for generating new origami pieces for kids

NAOYA TSURUTA,[†] JUN MITANI,^{†,††} YOSHIHIRO KANAMORI[†]
and YUKIO FUKUI[†]

Origami is one of Japanese traditional plays which creates various shapes by simply folding a square sheet of paper. Most of Japanese experienced folding origami in childhood. Techniques for creating complex origami pieces have been developed based on studies of mathematics of origami in recent years. However, the creation of simple origami pieces made with a few folds which are required for children who don't have ability to precisely control their fingers has not been studied. In this paper, we propose a new system for generating new simple origami pieces they can be folded by four or less steps. Our system enumerates all possible origami shapes which can be folded by four steps, and then stores them in a database in advance. When the user inputs the shape he or she wants to create, such as face of an animal, the system retrieve origami pieces similar to the input. Then the folding process is displayed. This brute-force approach became possible by recent progress of performance of common PCs. We have made our system more practical by optionally allowing users to lay "eyes" on origami pieces, as often seen in origami pieces for children.

1. はじめに

1枚の正方形の紙を折ることで形を作り出す「折り紙」は、日本に古くから伝わる遊びであり、我々の多くが幼少期に体験するものである。しかし、単なる遊びとして捉えられるだけでなく、折り紙を折る操作と平面幾何学の密接な関係から、長い間数学者による研究の対象にもなってきた。これらの折り紙の数理に関する研究の成果により各種の設計技法が考案され、近年では複雑な折り紙作品を創作するための技術が発達

してきている。また、ものを折りたたむ、という操作は、ものをコンパクトにして収納するための技術に関係するため、工学的な観点からの研究も進められている。このように、幼児向けの遊びと考えられがちな折り紙も、現在では多方面から注目され、それぞれの分野で研究が進められている。その一方で、折り紙に初めて触れるであろう幼児たちのための折り紙については、ほとんど研究が行われていない。まだ手指を正確に動かす能力が十分に発達していない幼児らでも折り紙を楽しめるようにするには、図1に示すような簡単に作れる折り紙作品の存在が望まれるが、それらの創作には確立した手法が存在せず、経験が豊富な折り紙作家の手によって、試行錯誤を伴う発見的なアプローチがとられている。

[†] 筑波大学大学院 システム情報工学研究科
University of Tsukuba

^{††} JST ERATO
Japan Science and Technology Agency, ERATO

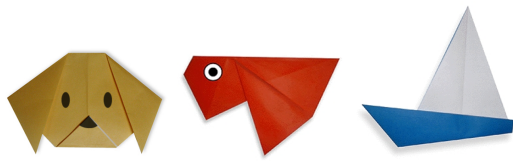


図 1 作品例¹⁾ (左から「いぬのかお」, 「きんぎょ」, 「ヨット」)
Fig. 1 Examples of simple origami pieces¹⁾. (From left, dog face, goldfish, yacht)

そこで本稿では、4 回以下の折り操作で作ることができる単純な折り紙作品を、新しく見いだすためのシステムを提案する (4 回という回数を選択した理由は第 3 章で述べる)。4 回以下で作れる形は極めて限定されるが、それでもあらゆる折り方を考慮すれば、そのバリエーションは膨大な数となる。その中に含まれるであろう、我々がまだ作品として見いだしていない形を見つけだし、幼児向け作品の創作を支援することを目標とする。

最初に思いつくアプローチは、意図した形を正方形から折りだすためには、どのようにすればよいかを幾何学的な観点から考察することであろう。しかし、これまでに考案された折り紙の設計技法は何回もの折り工程によって実現されるものであり、限られた折り回数で実現するための確固とした手法を確立することは、極めて難しい問題であることが予想される。

そのため、本稿で提案するシステムでは、4 回までの折り操作で作ることができる折り紙の形をすべて列挙し、それをあらかじめデータベースに蓄えておくことを行う。ユーザが、動物の顔など、折り紙で作りたい形をシステムに入力すると、システムはそれに類似した形をデータベースから検索し、折り方を提示する。一見すると、力づくのアプローチであり、理論的でないように見えるかもしれないが、近年の計算機の性能の向上によって実現可能となった、目的達成のための現実的なアプローチである。取りうるパターンをすべて列挙することで、意図した形を最も適切に再現する折り方を見つけ出すことが可能となる。なお、このような実現方法に「創作」という言葉は適切でないかもしれないが、ユーザの視点に立てば、今までに知られていない折り方で、形を創り出す方法を模索するためのツールとして捉えることができるため、「創作支援」という言葉を使用することとした。

さらに、折ってできる「形」だけでなく、幼児向けの折り紙作品に見られる「目や鼻の追加」の仕組みも組み込むことで、より実用的なシステムを構築した。提案システムを用いることで、幼児向けの新しい折り紙作品が数多く創り出されることが期待される。

続く第 2 章では関連研究について述べ、3 章では対象とする折り紙について述べ、4 章で提案手法の詳細を述べる。5 章で結果を示し、6 章で今後の展望を述べたのち、7 章でまとめを述べる。

2. 関連研究

紙を切ったり貼り合わせたりすることなく、折るだけで目的の形を作ることは難しい問題である。鶴や兜のような伝承的な折り紙作品は、先人の試行錯誤により創り出されてきたものと考えられるが、このようなアプローチでは昆虫や恐竜のような複雑な形を創り出すことは困難である。そのため、折り紙の数理に基づく新しい創作技法が模索され、その結果として 20 世紀後半以降、複雑な折り紙を理論的に設計するための技術が生み出されてきた。1979 年に発表された複雑でリアルな「悪魔」の作品²⁾は、折り紙設計の技術が導入されて創られたものであり、尻尾や角、指先に至る細部まで再現された形状は、それまでの折り紙の概念を覆した。この設計技法は、正四角形の領域に折り紙の基本構造を含む三角形を詰め込むというものである。また、これと類似した手法に、円形の領域を展開図上に配置する Circle Packing という名称の技法があり、これらは文献 3) に体系的にまとめられている。既定の手続きで、目的の構造を持った形を折り紙で作ることができるため、そのアルゴリズムを組み込んだ、折り紙専用の設計支援ソフトウェアも公開されている⁴⁾。このソフトウェアでは、目的の形の構造を、線分のネットワーク (グラフ理論で用いられる「木構造」として入力することで、その構造を折るための展開図が出力される。出力から得られるものは、基本構造だけであるため、細部の作り込みは人の手にゆだねられる。折り紙を格子状に区切り、その上に四角形領域を配置していく Box Pleating と呼ばれる設計技法も存在するが³⁾、その基本的な考え方は Circle Packing と同じである。これらの技術が依って立つ数学的な背景は、「折り」に関する約 400 もの参考文献を網羅した大著 5) にまとめられている。特に近年では、計算機を活用した折り紙の研究も増え、従来の折り紙とは一線を画す作品も登場するようになっている。以降で、計算機を用いた折り紙研究の例をいくつか紹介する。Betaman は、折り紙デセレーション (特定のパターンを敷き詰めて平坦に折り畳むもの) の展開図を自動生成するソフトウェア Tess⁶⁾ を作成した。Tachi は、CG の世界で一般的に用いられるメッシュモデルで表現される立体的な形を、1 枚の紙で折るための展開図を自動生成する手法を提案した⁷⁾。これは、表面に現れない不要

な部分を立体の内側に襷として折り込むことで実現している。一方で Mitani は、この襷を外側に折りだすことで、軸対称な立体形状をした折り紙を簡単なアルゴリズムで設計できることを示した⁸⁾。既存の折り紙作品の形を計算機の中にモデル化するためのインタフェースに関する研究には、宮崎、古田、三谷らの研究がある^{9)~11)}。このように、折り紙に関する研究の発展には目覚ましいものがあるが、本研究で取り上げる、幼児を対象とした簡易な折紙作品の創作についてはまったく注目されていない。これまでに紹介した研究のほとんどが、制作の難しさに大きく影響する「折り回数」に着目していないためである。少ない折り回数で、幼児が楽しめる形を創りだすことは、折り紙の裾野を広げる上で大切なテーマであり、それをどのように実現するかはまだ未解決な問題である。

本稿で提案する手法は、「目的とする形の輪郭」を入力に用いるが、このように形から折り方を導出するアプローチは、すでに Shimanuki ら¹²⁾ によって試みられている。Shimanuki らは、イラストからスケルトン（骨格構造）を抽出し、その構造を折りだすための展開図を生成する手法を提案した。すでに挙げた Circle Packing の手法に近く、目的の構造は得られるが、その最終的な形を決めるのは、折り手にゆだねられているため、やはりこの手法も、図 1 に示すような幼児向けの折り紙を創作するために用いることはできない。

3. 対象とする折り紙

幼児向けの折り紙に求められる特徴を知るために、簡単に作れる作品を紹介した書籍 1) に掲載されている作品 64 点を調査した。その結果、それらの作品は平均して約 8 回の折り操作で作ることができ、最も折り回数の少ない作品は 3 回の折りだけでできていた（図 1 右の「ヨット」）。折り回数が多い作品でも、その手順には作品に丸みをつけるための「カドを少しだけ折る」といった操作が含まれることが多いため、作品の概形はより早い段階で決定される。以上より、4 回の折り操作でも十分に作品の表現が可能であると考えられる。本研究では、冒頭で述べたように、可能な折り操作をすべて列挙するというアプローチを取るが、そのバリエーションの数は、折り回数に応じて爆発的に増加するため、現代の計算機での実現可能性という観点からも、対象とする折り紙の折り回数は少なければ少ないほどよい。詳細は 4 章で述べるが、4 回までの折り操作に限定すれば、すべて列挙するという力づくのアプローチでも、現在の計算機で十分対応可能であった。幼児にとっても、少ない折り回数で形が作れることが

好ましいと考えられる。以上のことを踏まえ、本研究では「4 回以下の折り操作で作れる形」を対象とすることとした。

ところで、紙を折る操作には「カド A をカド B に重ね合わせるようにして折る」というように参照点が存在する折り方と、「このくらいを目途にして適当に折る」という、参照点の無い曖昧な折り方（折り紙の用語では「ぐらい折り」と呼ばれる）が存在する。後者を含めると 1 回の折り操作にも無限のバリエーションが存在することになるため、本システムでは参照要素（点と辺）を用いる次の 3 種類の折り方だけを対象とする。

- 点と点を通る直線で折る
- 点と点を重ねるように折る
- 辺と辺を重ねるように折る

参照要素の存在する折り方の種類は、過去の研究から折り紙の公理としてまとめられ、全部で 7 通り存在することが知られている¹³⁾。今回挙げた 3 つの折り方も、この公理の中に含まれる。残りの 4 つの公理は「2 点 p_1, p_2 と 2 本の直線 l_1, l_2 が与えられたとき、 p_1 を l_1 上に重ね、かつ p_2 を l_2 上に重ねる折り方」のように一般的に用いられない折り方も含むため、これらは幼児を対象とした簡単な折り紙には適さないとして除外した。また、1 回の折り操作で複数の折り線を同時に折る「しずめ折り」「かぶせ折り」「中割折り」などの技法も存在するが、これも同様の理由から今回の対象からは外した。

書籍 1) の作品を見ると、幼児が楽しめるよう、作品のモチーフは動物であることが多く、最後に顔を描くものが多かった。細部まで作りこむのではなく、簡単に作れることを重視した結果と思われる。特に、完成後に目を描く作品が多く、調査したものの半数以上がこれに該当した。このことを考慮し、本システムでは、作りたい形を入力するときに、目や鼻などの顔のパーツの位置も指定できるようにした。

4. 提案システム

システム全体の概要を図 2 に示す。上側は計算機があらかじめ行う処理であり、下側はユーザが入力を与えた後の処理である。

本稿で提案するシステムの流れは次のとおりである。

- (1) 4 回以下の折り操作で得られる折り紙作品をすべて列挙しデータベースに格納する。
- (2) ユーザが目的とする形を入力する。入力輪郭を表す多角形であるが、オプションで目および鼻などのパーツも配置可能である。また、紙の

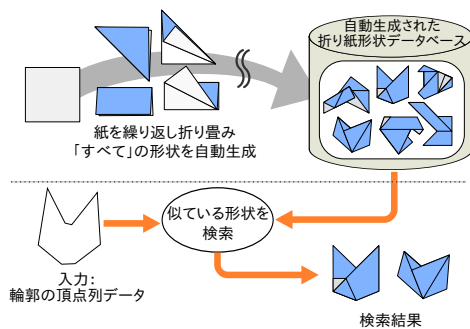


図 2 システム概要

Fig. 2 Overview of our system

表裏の色の違いを考慮した、2色の多角形の集合を入力とすることも可能である。

- (3) 入力された多角形に形が類似した折り紙作品を検索し、類似度順に複数の検索結果をユーザに提示する。入力に目や鼻のパーツが配置されている場合は、同じパーツを配置した結果を示す。
- (4) ユーザが選択した折り紙作品の折り手順を提示する。

以降ではそれぞれの詳細について述べる。

4.1 折り紙形状データベースの構築

まず、4.1.1 節で折り紙形状の生成手法を述べ、4.1.2 節では、重複した形状を取り除いてデータベースへ格納する方法を述べる。ここまでの処理でデータベースの構築は完了するが、検索を効率化するために、構築したデータベースに対してクラスタリングを行う(4.1.3 節)。

4.1.1 折り紙形状の生成

紙を折りたたんだ後の形を得るために、まず、折る場所を指定するための折り線の位置を決定する。ある状態における可能性のあるすべての折り線の位置は、3章で述べた3つの折り紙公理を、すべての参照要素の組み合わせに適用することで求めることができる。具体的には、2つの参照点を結ぶ線分、2つの参照点の垂直二等分線、および2つの辺の成す角の二等分線が候補となる。

続いて、このようにして求めた折り線の位置で紙を折る。提案システムでは、1つの折り線に対して次の3通りの折り方があるものとした。

- 谷折り
- 山折り
- 折り線をつけて戻す (以降の手順で折るときに参照できる点や線をつける操作)

紙が複数枚重なっているとき、谷折りの場合は一番上にある面から1枚ずつ順番に折り畳み、すべてを異なる折り紙作品として扱う。これは先行研究として

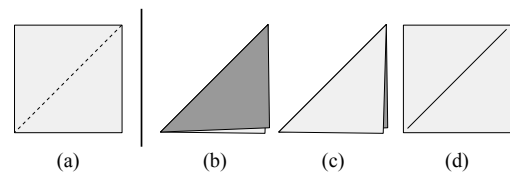


図 3 折り操作 (a) 折り線 (b) 谷折り (c) 山折り (d) 折り線をつけた状態

Fig. 3 Folding operations (a)Folding line (b)Valley fold (c)Mountain fold (d)Adding a crease line

行った折り図作成支援ツールの開発¹⁴⁾で、折り手順の予測の際に用いたアルゴリズムと同じであり、紙が内側に折り込まれるケースが除外されるため、紙が貫通するような形状が発生しない。山折りの場合は逆に、一番下にある面から順に折り畳む。「折り線をつけて戻す」操作は、すべての面に一度に折り線をつける。重なっている面の数を n とすると、1つの折り線から最大で $2n + 1$ 個の形状が生成される (図 3)。

4.1.2 データベースへの格納

前述の処理で折り紙の形状を網羅的に生成したのち、重複するものを除外してデータベースに登録する。ここでは、ある形に対し、各面の共通重心から各頂点までの距離の総和 (図 4) が等しいものを、「回転および反転して一致するもの」と判断して除外した。この方法で除外すると、輪郭は同じで面の重なり順だけが異なるものも除外されてしまうため、以降の手順において生成される形の候補が少なくなり、その結果として後述する紙の表裏の色の違いを利用した検索で適切なものが見つからなくなる可能性がある。しかし、この問題を解決するためには、紙を一旦開いて、展開図から面の重なり順が異なるパターンをすべて数え上げることが必要となる。異なる手順で同じ展開図が得られる可能性もあるため、それらの重複判定を含めると、データベースに格納される形のユニーク性を維持するのは、極めて難しい問題と言える。今回は、折った後に現れる輪郭の形に主眼を置いたためこの方法を採用したが、今後検討すべき点と言える。

なお、新しい形を格納する際には、1手前の形が格納されている場所の参照を保持するようにした。これを順に辿ることで、あとから折り手順を知ることが可能である。

4.1.3 クラスタリング

検索を高速化するために、構築したデータベースに対してクラスタリングを行い、あらかじめ似た形状をまとめておく。これにより、最初に代表となるいくつかの形状を検索し、その後、その形状に似たものの中から検索するという段階的な検索で、トータルを検索

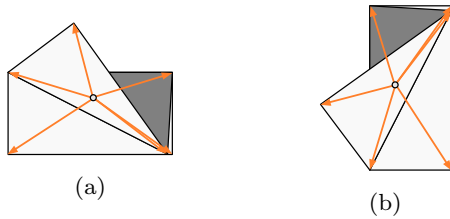


図 4 同一形状であるか否かの判定. (b) は (a) を 90 度回転させたものであり、どちらも矢印の長さの総和が等しい.

Fig. 4 Comparison of two origami pieces. (b) matches to (a) by rotating 90 degrees. Both pieces have equal sum of the length of arrows.

時間を短縮することが可能となる.

まず、初期クラスタとして代表となる形状を 100 個ランダムに選択し、残りのデータを最も似ているクラスタに配属する（類似度の評価方法については 4.3 節で述べる）。すべての形状をクラスタ化した後、格納数の大きいクラスタから新しく代表となる形状を選び、再びクラスタリングを行う。それと同時に、サイズの小さいクラスタを取り除き、それに含まれていた形状を別のクラスタに分配する。このような処理を繰り返し行うことで、類似したデータをまとめることを試みた。

4.2 ユーザからの入力

図 5 の上段に入力画面を示す。ユーザは輪郭を表す多角形をマウスクリックにより入力する。これは各頂点を順番にクリックしていくことで行う。また、オプションで色の付く多角形領域を追加できる。これにより、折り紙の色のついた面が現れる部分を指定できる。さらに追加で、顔のパーツを配置できる。ツールバーにある目や鼻のボタンをクリックし、選んだパーツの位置をカーソルで指定し、マウスホイールによって大きさを調整できる。ここで配置したパーツが、4.4 節で述べる検索結果に重ねて描画される。今回の実装では、目のパーツを 3 つ、鼻と口のパーツを 2 つずつ用意したが、システムの起動時に画像ファイルから読み込まれるようになっているため、他の画像を用意すれば、新しいパーツとして追加することが可能である。

4.3 類似形状の検索

ユーザによって入力された輪郭多角形とオプションの色付き多角形領域の情報を元に、それに類似した折り紙作品を検索する。2 次元の形状マッチングの手法には、頂点と数式で表されたベクタ画像を対象としたもの¹⁵⁾ やラスタ画像を対象としたもの¹⁶⁾ など、過去に様々な手法が提案されているが、今回は独自の実装を行った。その理由は、折り紙形状データがベクタ情報を持っており、ラスタ化による情報損失は無駄が大

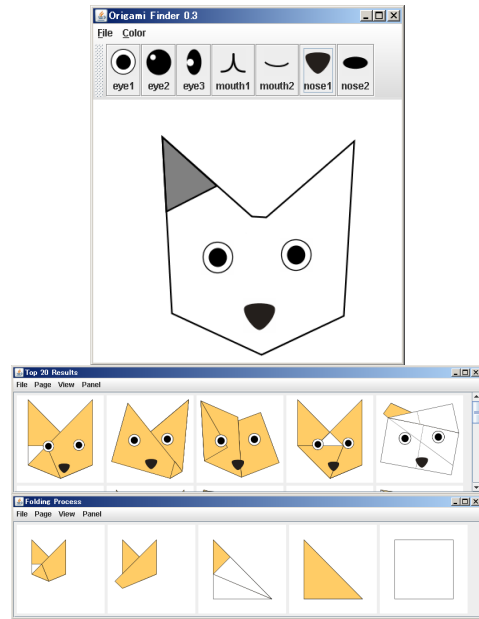


図 5 システム画面 (上: 入力ウィンドウ, 中: 結果表示ウィンドウ, 下: 折り手順表示ウィンドウ)

Fig. 5 Screen of our system (top: Input window, middle: Result window, bottom: Folding process window)

きく、また、色のついた領域の情報も含むため、既存のベクタ画像に対する手法も適用が難しいと判断したためである。実装した手法では、前処理として、形状の外接四角形の大きさでスケールを統一し、入力された形とデータベースに格納された折り紙形状の輪郭多角形および色付き領域の多角形を重ねた差分（排他的論理和）の面積が小さいほど類似度が高いものとした。比較の際には、配置の仕方（向き）の影響を軽減するために、一方を重心まわりに 10 度刻みで回転させながら比較し、最も小さい値をその類似度とした。この比較処理を各クラスタの代表形状に対して行い、最も似ているクラスタを決定したのちに、クラスタに含まれるすべての形状と比較を行った。最後に、値が最も小さいものから順に上位 20 個を出力するものとした。

4.4 検索結果の提示

検索を実行した後は、その結果の上位 20 個が図 5 中段のように表示される。ここで提示された結果画像から 1 つを選択してクリックすると、その折り紙形状を折るための手順が別ウィンドウで表示される（図 5 下段）。

検索前に目や鼻を配置した場合は、入力した多角形の重心から各パーツまでの距離を元に、検索結果にも同じパーツを重ねて描画する。重心からパーツまでの距離とパーツの大きさは、外接四角形で比べたときの比率を元に拡大または縮小され、結果画像に合うように表示される（図 6）。

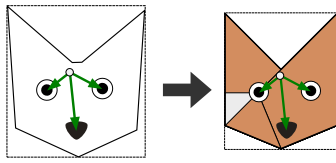


図 6 パーツの位置の決定

Fig. 6 Location of the parts

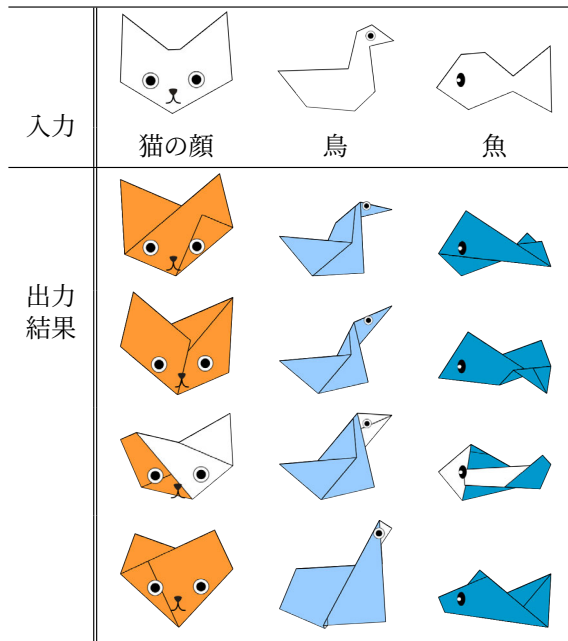


図 7 入力形状と検索結果

Fig. 7 Input and result of retrieval

5. 結果と考察

提案システムで構築したデータベースの詳細とクラスタリングの結果、入力としていくつかの図を与えたときの検索に要した時間と出力された結果を示す。さらに、目や鼻などの顔のパーツの表示について考察する。なお、システムの実装には Java と SQLite を用いて、Core2Duo 2.66GHz, 2GB RAM を搭載した PC 上で動作させた。

5.1 データベースの構築

折り紙形状を保持するデータベースの要素数とデータサイズを表 1 に示す。今回は 4 回までの折り操作としたため、4 回折りのデータ数には「折り線をつけて戻す操作」によるデータを含めなかった。表 1 から、折り回数が増えるに従ってデータ数が指数関数的に増加していることがわかる。4 回折りのデータベースをクラスタリングした結果、クラスタ数（代表となる形状の数）は 145 個で、クラスタに属する形状数は最低 10、最大 29252、平均 964 となった。4 回折りのデータは 139844 通りあり、データが各クラスタに均

表 1 折り回数とデータ数およびデータサイズ

Table 1 The number of folding steps, the number of items, and the size of data

折り回数	データ数	データサイズ	構築時間
1	4	4 (KB)	16 (ms)
2	37	11 (KB)	796 (ms)
3	1,505	468 (KB)	2046 (ms)
4	139,844	70.9 (MB)	136 (s)

等に分配されると仮定すると、クラスタ数が 365 個前後のときに検索時間が最も短くなる。この値を目安としてクラスタ数の調整を試みたが、極端に要素数の大きいクラスタが発生するなどして実現は難しかった。

5.2 検索結果

例題として入力した 3 つの多角形を入力し、検索を行った。それぞれの検索結果を図 7 に示す。結果に示した折り紙形状は、各入力に対して線形探索によって得られた上位 20 個の形状のうち、筆者らがより似ていると判断した 4 つである。

図 7 の結果からは入力に対して十分な精度が得られていると言える。入力形状と合わないような折り紙形状が出力される場合もあるが、最終的な判断はユーザが実際に見て行うため、極端に高い精度はなくても問題はないと思われる。上位 20 個程度まで絞り込めれば、人の目で容易に判断することができる。

5.2.1 検索時間

表 2 には各入力の検索に要した時間をまとめた。クラスタ化することで、検索時間を 1/100～1/700 程度に短縮できたことがわかる。しかし、クラスタ検索を行った結果には図 7 に示した線形探索の結果の一部しか含まれなかった。これは、入力形状に最も似ている形状が、入力形状に最も似ているクラスタに属していない可能性があるためである。このような最適解の見落としの問題を解決するためには、適切な形状が代表となるようにクラスタを改善する必要があるが、その方法については今後の課題である。

5.2.2 紙の表裏を考慮した検索

図 8 には紙の表裏を指定した入力と、輪郭だけで検索した場合で、最も似ていると判断した折り紙形状を示した。結果としては、共通して出力される形状はわずかで、異なる形状が多く出力された。図 8 に示した

表 2 検索時間（線形検索とクラスタ検索）

Table 2 Retrieval time (Linear and Cluster)

入力形状	線形検索 [sec]	クラスタ検索 [sec]
猫の顔	439	0.7
鳥	442	3.8
魚	439	0.6

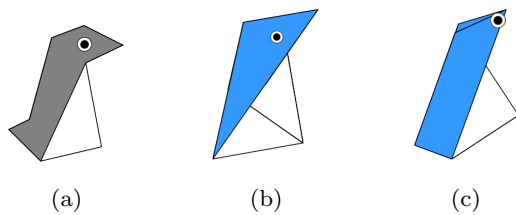


図 8 表裏を考慮した検索例 (a) 入力 (b) 輪郭のみを用いた検索結果 (c) 表裏を考慮した検索結果

Fig. 8 Result of retrieval with colors. (a)Input (b)Result by referring outline polygon (c)Result by referring outline polygon and colors.

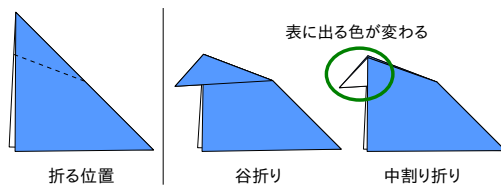


図 9 中割り折りによる色の変更効果

Fig. 9 Effects of “Reverse folding” for the appearance

(b) は表裏を考慮した検索結果の上位 20 個には含まれず、また、逆に (c) は輪郭のみの出力結果に含まれなかった。図 7 に示した例でも、一部の色を指定することで違った結果が得られると思われる。

表裏を考慮した検索の精度をさらに高めるためには、次の 3 つの方法が考えられる。

- 折り紙を裏返した場合も比較する
- 重複の削減手法を改良する
- 他の折り方を実装して候補を増やす

折り紙形状の中には、表から見た場合と裏から見た場合で、色の出ている部分が違う形状がある。そのため、反転した形状も比較すればそれだけで違う結果を得られる可能性がある。特に、折り紙においてよく用いられる「中割り折り」という技法（図 9）を実装することは、表裏を考慮した検索に非常に効果的であると考えられる。

5.2.3 顔のパーツの表示について

図 10 には目のある場合とない場合の画像を示した。目があることで、動物としての認識のし易さが大きく向上する。本研究で取り上げたような簡単な作品では、回転させるだけで違う作品に見えたり、何に見えるかが人によって異なる場合が多い。顔のパーツの表示はそういった曖昧さを解消するための重要な要素であると考えられる。今回は顔のパーツのみを指定する機能を実装したが、顔だけでなく、例えば「くるま」や「いえ」などの作品であれば窓を描く場合もある。そのため、顔のパーツの位置を指定するだけでなく汎用性の

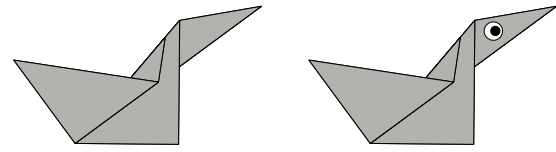


図 10 目の有無による違い

Fig. 10 Difference of appearance with/without an eye.



図 11 本システムで得られた結果の折り紙

Fig. 11 New origami pieces generated with our system

高いペイント機能があれば、検索結果での見やすさをより向上させることが可能である。

5.2.4 得られた折り紙作品

提案システムを用いて得られた結果の折り紙作品を図 11 に示す。これらは今までに作品として知られていなかった折り紙作品であり、本システムを用いることによって創りだされたものと言うことができる。すべてが 4 回以内の折り操作で作ることができる、簡単な作品である。

6. 制限と展望

提案システムでは折り紙形状のデータベースを構築する際に、折り操作の回数（4 回まで）と、折り方（折り紙の公理に基づく 3 通りの折り）に制限を設けた。提案手法では、データベースに存在する形状からユーザ入力に似たものを検索するため、データベースが大きければ大きいほど、より適切な形状を検索できる可能性が高くなる。上に挙げた制限を外すこと自体は難しくなく、そうすれば膨大な数の形状を生成することができる。

ただし、それに伴ってデータベースのサイズが膨らみ、検索時間が長くなることが問題である。前節で示したように、線形探索 1 回の計算時間は約 7 分であった。結果の精度と検索に要する時間はトレードオフの関係にあるため、対象とする形状の制限と、検索の効率化については、今後の課題である。

折り回数と生成される形状数は指数関数的に増加するため、折り回数を 1 回増やすだけで場合の数は大幅に増える。4 回折りのデータに、折り線をつける操作を加えたときのデータは約 22 万通りあり、5 回折っ

た場合は、少なく見積もっても数千万のオーダーになることが予想される（厳密に場合の数を見積もるのは難しい問題で、数の推定方法も未解決問題である）。個々の形状は複雑になるため、データベースのサイズはギガバイトのオーダーになるだろう。

クラスタリングはこの問題を解決する手法の一つであるが、適切なクラスタリングが行われないと検索精度が低下するため、注意が必要である。また、検索手法をより高速なものに変更することも考えられる。データをあらかじめラスター画像に変換しておけば、既存の画像検索アルゴリズムを利用することができ、高精度な検索が行える可能性がある。したがって、複数の検索手法を実装し検索時間と精度を比較して、より折り紙データに適した検索手法を探すことも今後の課題の一つである。

7. ま と め

本研究は、幼児用の折り紙に着目し、ユーザが入力した形状を折り紙で表現する手法を提案した。提案手法では、折り紙のデータベースを自動で構築し、構築されたデータベースから入力に似た形状を検索する。既存の手法と異なり、できるだけ少ない折り回数で表現することを目的としているので、幼児でも簡単に折ることができる作品が得られるという特徴がある。

しかし、折り回数を増やすと扱うべき場合の数が爆発的に増えるため、これよりも難易度の高い（折り回数の多い）折り紙作品を対象とすることが難しいといった問題もある。第6章で述べたように、同じ折り回数であっても、形状のデータを増やすことはより良い結果を得ることにつながるが、検索時間とのトレードオフ問題があるため、これに対処しなければならない。クラスタリングはこの対応策の一つであり、適切なクラスタリングができれば、かなりの高速化が期待できる。

実験結果では、顔のパーツを描画することによる作品認識の向上についても触れた。幼児の折り紙において目や鼻などの付加情報が持つ意味は大きく、本研究では、このような点に着目したところに新規性があると考えている。認知科学の分野に含まれるが、これらのパーツの有無が具体的に対象物の認知にどの程度の影響を及ぼすのかも興味深いテーマである。

参 考 文 献

- 1) 新宮文明：おりがみしようよ！ - めだまシールつき (実用 BEST BOOKS), 日本文芸社 (2005).
- 2) 前川 淳, 笠原邦彦：ビバ！おりがみ, サンリオ,

- 〔新装版〕 edition (1989).
- 3) Lang, R.J.: *Origami Design Secrets: Mathematical Methods for an Ancient Art*, A K Peters Ltd, illustrated edition edition (2003).
- 4) Lang, R.J.: TreeMaker, <http://www.langorigami.com/science/treemaker/treemaker5.php4> (2006).
- 5) Demaine, E.D. and O'Rourke, J.: *Geometric Folding Algorithms: Linkages, Origami, Polyhedra*, Cambridge University Press, reprint edition (2008).
- 6) Bateman, A.: Tess: origami tessellation software, <http://www.papermosaics.co.uk/software.html>.
- 7) Tachi, T.: Origamizing Polyhedral Surfaces, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 16, No. 2, pp. 298–311 (2010).
- 8) Mitani, J.: A Design Method for 3D Origami Based on Rotational Sweep, *Computer-Aided Design and Applications*, Vol.6, No.1, pp.69–79 (2009).
- 9) Miyazaki, S., Yasuda, T., Yokoi, S. and ichiro Toriwaki, J.: An Origami Playing Simulator in the Virtual Space, *Journal of Visualization and Computer Animation*, Vol.7, No.1, pp. 25–42 (1996).
- 10) 古田陽介, 木本晴夫, 三谷 純, 福井幸男：マウスによる仮想折り紙の対話的操作のための計算モデルとインタフェース, 情報処理学会論文誌, Vol.48, No.12, pp.3658–3669 (2007).
- 11) Mitani, J.: Recognition, modeling and rendering method for Origami using 2D bar codes, *Origami 4*, pp.251–258 (2006).
- 12) Shimanuki, H., Kato, J. and Watanabe, T.: Constituting Origami Models from Sketches, *Pattern Recognition, International Conference on*, Vol.1, pp.628–631 (2004).
- 13) Lang, R.J.: Origami Geometric Construction, <http://www.langorigami.com/science/hha/origami-constructions.pdf>.
- 14) 鶴田直也, 三谷 純, 金森由博, 福井幸男：折り図作成を支援する手順予測インタフェースと次の手順候補に対するランク付け手法, 第9回 NICOGRAPH 春季大会 (2010).
- 15) Veltkamp, R.C.: Shape Matching: Similarity Measures and Algorithms, *Shape Modeling and Applications, International Conference on*, pp. 0188– (2001).
- 16) Smeulders, A.W., Worring, M., Santini, S., Gupta, A. and Jain, R.: Content-Based Image Retrieval at the End of the Early Years, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.22, pp.1349–1380 (2000).