

## 特徴点の照合に基づくベクタ画像検索システムの試作

清野 達也<sup>†</sup>, 林 貴宏<sup>††</sup>, 尾内 理紀夫<sup>††</sup>

<sup>†</sup> 電気通信大学大学院 電気通信学研究科 情報工学専攻 〒182-8585 東京都調布市調布ヶ丘 1-5-1

<sup>††</sup> 電気通信大学 電気通信学部 情報工学科 〒182-8585 東京都調布市調布ヶ丘 1-5-1

内容に基づくベクタ画像検索システムを試作した。検索対象のベクタ画像はロゴマークなどの抽象図形である。本システムは、ベクタ画像の概形を重みつき特徴点の集合により表現し、画像間類似度を特徴点間のPTD (Proportional Transportation Distance) により定義する。特徴点はベクタ画像の輪郭の中で曲率変化の大きな点で与えられ、曲率変化の大きさに応じた重みが付加される。約1100枚のベクタ画像を用いた実験から、特徴点の数が少なく曲率変化の大きな曲線を含むベクタ画像ほど検索がしやすくなることを確認した。

## A prototype system for vector image retrieval with feature point matching

TatsuyaKIYONO<sup>†</sup> TakahiroHAYASHI<sup>††</sup> RikioONAI<sup>††</sup>

<sup>†</sup> Department of Computer Science, Graduate School of Electro-Communications, The University of Electro-Communications 1-5-1 Chofugaoka, Chofu-shi, Tokyo, 182-8585 Japan

<sup>††</sup> Department of Computer Science, The University of Electro-Communications 1-5-1 Chofugaoka, Chofu-shi, Tokyo, 182-8585 Japan

This paper presents a prototype system for content-based vector image retrieval. The targets of the system are abstract figures such as logotypes. The system represents a vector image as a set of weighted feature points and defines similarity between two vector images using PTD (Proportional Transportation Distance) between the weighted feature points of the two images. The system extracts points which have large curvature change as feature points and varies weights given to the feature points depending on curvature change. We have evaluated the efficiency of the system using about 1100 vector images. The experimental results have shown that the system has high precision on search for vector images whose outlines are composed of points which have high curvature changes.

### 1 はじめに

近年、Web上にはAdobe FlashやSVGなどのベクタ画像が広く普及してきた。そのため、ベクタ画像に対して所望の画像を検索する技術の需要が高まっている。しかし、ベクタ画像を対象とした検索の研究はあまり行われていない。

本研究は、ベクタ画像を対象としたCBIR(Content-Based Image Retrieval)の検索システムの提案、評価を目的とする。提案システムは、ユーザが入力したスケッチ画像から類似画像を検索する。画像の類似度は、画像中の形状変化が大きい部分の特徴点として抽出を行い、これら特徴点の照合に基づいて計算した。

以下、2章で関連研究について述べる。3章でベ

クタ画像の概要、4章でベクタ画像同士の類似度計算方法を説明する。5章では試作システムを用いて、性能を実験した。6章をむすびとした。

### 2 関連研究

CBIRの代表的な例として、QBIC[1]やVisualSEEk[2]が挙げられる。これらのシステムの画像類似度は、複数個の特徴量を用いて特徴ベクトルを構成し、特徴ベクトル間のユークリッド距離に基づいて計算される。特徴量の例として、色ヒストグラムや低解像度の輪郭情報が用いられる。これらの特徴量はラスタ画像に対して設計された物が多く、ベクタ画像に対して適用するにはラストライズ(ラスタ画像への変換)が必要となる。そのため、ベクタ画像の利点である解像度に対するの

画質劣化が起きないという性質や、形状の重なりを保持できる性質が損なわれてしまう。本研究では、ベクタ画像に適した類似度計算法を用いる。

### 3 ベクタ画像の概要

ベクタ画像は、要素図形の集合によって画像を表すという特徴がある。ベクタ画像の要素図形には、線や円、多角形などが用いられる。複雑な輪郭をベクタ画像で表現するには、要素図形の一つであるパスと呼ばれる図形が用いられることが多い。パスはセグメントと呼ばれる部分曲線の列で構成されている。ベクタ画像フォーマットの1つであるSVGでは、1つのセグメントは3次ベジェ曲線により表現される。セグメントの始点と終点をアンカーポイントと呼ぶ。連続する2つのセグメントの最初のセグメントの終点と次のセグメントの始点は一致している。またパス中の最初のセグメントの始点と最後のセグメントの終点一致している場合、そのパスを閉じたパスと呼ぶ。

$n-1$ 次ベジェ曲線は、 $n$ 個の制御点によって構成されるパラメトリック曲線である。 $b_1, b_2, \dots, b_n$ を制御点としたとき、ベジェ曲線は次式において $t$ を0から1に連続的に変化させたときの点 $p(t)$ の軌跡により定義される。

$$p(t) = \sum_{i=1}^n b_i J_{n-1,i-1}(t) \quad (0 \leq t \leq 1) \quad (1)$$

$J_{n,i}(t)$ はバーンシュタイン基底関数と呼ばれ、次の式で定義される。

$$J_{n,i}(t) = \binom{n}{i} t^i (1-t)^{n-i} \quad (2)$$

ベジェ曲線はその定義から、始点が第1制御点、終点が第 $n$ 制御点となる。また、第1制御点と第2制御点を結ぶ直線と第 $n$ 制御点と第 $n-1$ 制御点を結ぶ直線を方向線と呼ぶ。Fig. 1に3次ベジェ曲線の例を示す。第1制御点 $b_1$ と第4制御点 $b_4$ を結ぶ曲線が3次ベジェ曲線である。 $b_1$ と $b_2$ を結ぶ直線と $b_4$ と $b_3$ を結ぶ直線は方向線と呼ばれる。

## 4 提案システム

### 4.1 概要

本システムは、ユーザが入力した手書きスケッチをクエリとしてデータベース内のベクタ画像を検索し、類似するベクタ画像の上位10件を出力する。

ベクタ画像の検索は、手書きスケッチとベクタ画像との間の類似度に基づいて行われる。しかし、

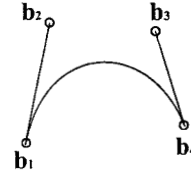


Fig. 1 3次ベジェ曲線の例

手書きスケッチとベクタ画像の表現の違いから、類似度を直接計算するのは困難である。そこで手書きスケッチをベクタ画像へ変換することで表現を統一し、ベクタ画像間の類似度を計算し、これを手書きスケッチとベクタ画像との間の類似度と定義する。

ベクタ画像間の類似度を計算するために本システムでは、ベクタ画像の輪郭に注目する。ベクタ画像の多くは、パスによって構成されているため、有限個のアンカーポイントによって輪郭の概形が表現できる。この性質を利用し、アンカーポイントをその輪郭を特徴付ける点(特徴点)とする。本システムはこの特徴点の位置に着目し、ベクタ画像間の類似性を評価することで検索を実現する。

しかし特徴点の位置のみを用いて類似性を評価した場合、その特徴点の重要度が考慮されないという問題が生じる。例えば曲率変化の小さな特徴点を取り除いたとしても、特徴点が表す概形はあまり変形しない。このことから曲率変化の小さな特徴点は、検索結果への影響が小さくなるように重要度は低く設定されるべきである。また曲率変化の大きな特徴点でも、その特徴点を含む輪郭が画像全体の大きさに比べて無視できる場合は、その特徴点の重要度は低く設定されるべきである。このように特徴点の位置だけでなく、曲率変化や輪郭の大きさを考慮するため、重要度(重み)を付加することを考える。本システムでは重みを付加した特徴点を、重み付き特徴点と呼ぶ。

### 4.2 手書きスケッチからベクタ画像への変換

ユーザがマウスなどのポインティングデバイスで入力した手書きスケッチは、その軌跡を表す座標列によって与えられる。手書きスケッチをベクタ画像へ変換するために、座標列をパスへ変換する。この処理は、座標列を3次ベジェ曲線に近似するステップとセグメントの分割を行うステップを交互に繰り返すことで実現する。Fig. 2にセグメント分割の例を示す。 $S_0$ における白点は手書きスケッチ

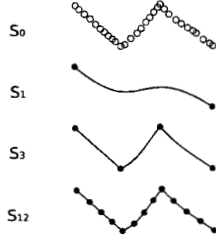


Fig. 2 セグメント分割の例

チによって与えられた座標列である。  $S_1, S_3, S_{12}$  はそれぞれ1個のセグメント, 3個のセグメント, 12個のセグメントを用いて  $S_0$  を近似したパスである。黒点はアンカーポイントを表す。  $S_1$  は,  $S_3$  や  $S_{12}$  に比べて明らかに  $S_0$  との誤差が大きい。セグメントの個数を増加させるほど誤差は0に近づいていく。  $S_3$  と  $S_{12}$  はいずれも元の座標列をうまく近似できているが,  $S_{12}$  は  $S_3$  に比べてアンカーポイントの数が多いため, 後述する類似度計算の計算量に大きな影響を与える。従って, セグメントの分割は適切なセグメント数で行われなければならない。

まず座標列が与えられたときに, それを1つの3次ベジェ曲線に近似する方法を述べる。その次にセグメントの分割を行う方法を述べる。

座標列  $V = \{v_1, v_2, \dots, v_n\}$  が与えられたとき, これを3次ベジェ曲線へ近似する方法を述べる。3次ベジェ曲線は,  $b_1, b_2, b_3, b_4$  の4つの制御点から構成される。3次ベジェ曲線への近似は, 始点  $v_1$  を制御点  $b_1$  に, 終点  $v_n$  を制御点  $b_4$  とし,  $b_2$  と  $b_3$  の推定を行う事で実現する。方向線の傾きをそれぞれ  $e_l, e_r$  とすると制御点  $b_2, b_3$  は次の式で表される。

$$\begin{aligned} b_2 &= b_1 + \alpha_l e_l \\ b_3 &= b_4 + \alpha_r e_r \end{aligned} \quad (\alpha_l, \alpha_r \in \mathbf{R}) \quad (3)$$

方向線の傾き  $e_l, e_r$  は次式を用いて推定する。

$$\begin{aligned} e_l &= \frac{m_L - v_1}{\|m_L - v_1\|} \\ e_r &= \frac{m_R - v_n}{\|m_R - v_n\|} \\ L &= \underset{s}{\operatorname{argmax}} (\|m_s - v_1\|) \\ R &= \underset{t}{\operatorname{argmax}} (\|m_t - v_n\|) \end{aligned} \quad (4)$$

ただし,  $m_s \in V_1, m_t \in V_n$  である。ここで,  $V_1, V_n$  は  $v_1, v_n$  から半径  $T$  の円を作り, その円

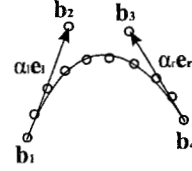


Fig. 3 3次ベジェ曲線の近似例

内に存在する  $V$  の要素の集合である。

係数  $\alpha_l, \alpha_r$  は Philip のアルゴリズム [3] により決定する。Fig. 3 にベジェ曲線近似の例を示す。図中の白点は, ユーザが入力したスケッチ (座標列) である。図中の矢印は, 入力された座標列から推定した方向線である。  $b_1$  と  $b_4$  を結ぶ曲線は, 方向線の推定によって決定した  $b_2, b_3$  を用いて近似した3次ベジェ曲線である。

次に, セグメントの分割について述べる。座標列  $\{v_1, v_2, \dots, v_n\}$  が与えられたとき, 座標列  $V_i = \{v_1, v_2, \dots, v_i\} (i \leq n)$  と, その近似3次ベジェ曲線  $p_i(t)$  との誤差  $E_i$  を次の式で定義する。

$$\begin{aligned} E_i &= \max \{ \|p_i(t_{i,j}) - v_j\| \mid v_j \in V_i \} \\ t_{i,j} &= \frac{\sum_{l=1}^j \|v_l - v_{l-1}\|}{\sum_{k=1}^i \|v_k - v_{k-1}\|} \end{aligned} \quad (5)$$

ただし  $j \leq i$  である。ある定数  $E_{max}$  (許容誤差) に対し,  $E_i < E_{max}, E_{i+1} > E_{max}$  を満たすとき, 座標列の近似3次ベジェ曲線  $p_i(t)$  を1つのセグメントとして分割する。さらに, 残りの  $\{v_i, v_{i+1}, \dots, v_n\}$  を新たな座標列としてセグメントの分割をくりかえし行う。

### 4.3 重み付き特徴点抽出

重み付き特徴点の集合  $S$  は, 特徴点の座標  $p$  と重み  $w$  の対を用いて次式で定義される。

$$S = \{(p_1, w_1), (p_2, w_2), \dots, (p_n, w_n)\} \quad (6)$$

特徴点はアンカーポイントを用いる。このようにすることで, パスの始点と終点である「端点」と, 曲率の大きな「角点」を特徴点として抽出することが可能である。

パス内に2つの連続したセグメントが存在するとき, その2つのセグメント上のアンカーポイン

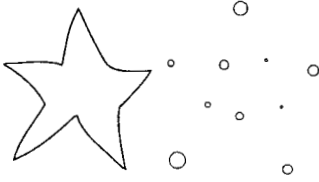


Fig. 4 重み付き特徴点

トを  $\mathbf{v}_{i-1}, \mathbf{v}_i, \mathbf{v}_{i+1}$  と定義する.  $\mathbf{v}_{i-1}, \mathbf{v}_i, \mathbf{v}_{i+1}$  は特徴点でもある. ここで,  $\mathbf{v}_{i-1}, \mathbf{v}_i$  を結ぶ3次ベジェ曲線の第1制御点, 第2制御点を  $\mathbf{b}_{i,1}, \mathbf{b}_{i,2}$  とし,  $\mathbf{v}_i, \mathbf{v}_{i+1}$  を結ぶ3次ベジェ曲線の第1制御点, 第2制御点を  $\mathbf{b}_{i+1,1}, \mathbf{b}_{i+1,2}$  としたときの  $\mathbf{v}_i$  の重み  $w_i$  は次式で定義される.

$$w_i = \text{ang}(\mathbf{v}_i) \cdot N(\text{dist}(\mathbf{v}_i)) \quad (7)$$

$$\text{ang}(\mathbf{v}_i) = \max\{c_{v,i}, c_{b,i}\}$$

$$c_{v,i} = c(\mathbf{v}_{i-1} - \mathbf{v}_i, \mathbf{v}_{i+1} - \mathbf{v}_i)$$

$$c_{b,i} = c(\mathbf{b}_{i,2} - \mathbf{v}_i, \mathbf{b}_{i+1,1} - \mathbf{v}_i)$$

$$c(\mathbf{a}, \mathbf{b}) = \frac{1 + \frac{\langle \mathbf{a}, \mathbf{b} \rangle}{\|\mathbf{a} - \mathbf{b}\|}}{2}$$

$$N(t) = \frac{t}{1+t}$$

$$\text{dist}(\mathbf{v}_i) = \|\mathbf{v}_{i-1} - \mathbf{v}_i\| + \|\mathbf{v}_i - \mathbf{v}_{i+1}\|$$

ここで関数  $\text{ang}$  はアンカーポイントの鋭さを表す. 関数  $\text{dist}$  は前後のアンカーポイント間の距離を表す. また閉じたパスでない場合 ( $\mathbf{v}_1 \neq \mathbf{v}_n$ ),  $\mathbf{v}_1$  と  $\mathbf{v}_n$  を引数としたときの関数  $\text{ang}$  と関数  $\text{dist}$  は次式で定義される.

$$\begin{aligned} \text{ang}(\mathbf{v}_1) &= 1 \\ \text{ang}(\mathbf{v}_n) &= 1 \\ \text{dist}(\mathbf{v}_1) &= \|\mathbf{v}_1 - \mathbf{v}_2\| \\ \text{dist}(\mathbf{v}_n) &= \|\mathbf{v}_{n-1} - \mathbf{v}_n\| \end{aligned} \quad (8)$$

Fig. 4 に手書きスケッチから重み付き特徴点を抽出した図を示す. 右図の白点の中心が特徴点の座標, 白点の大きさが特徴点の重みである. 図から曲率変化の大きな点は, 大きな重みが付加された特徴点として抽出されていることが分かる.

#### 4.4 PTDに基づく類似度計算

本システムではベクタ画像間の類似度を, ベクタ画像から抽出した重み付き特徴点を用いたPTD(Proportional Transportation Distance)[3]により定義する. PTD は, 線形計画問題の1つである輸送問題における総輸送コストの最適解と

して定義される. 輸送問題とは需要地と需要量の集合  $\mathbf{S} = \{(\mathbf{p}_1, w_1), (\mathbf{p}_2, w_2), \dots, (\mathbf{p}_n, w_n)\}$  と, 供給地と供給量の集合  $\mathbf{T} = \{(\mathbf{q}_1, u_1), (\mathbf{q}_2, u_2), \dots, (\mathbf{q}_m, u_m)\}$ , 需要地と供給地間の輸送コスト  $\text{cost}(\mathbf{p}, \mathbf{q})$  が定義されたときの総輸送コストの最小値を求める問題である. PTD は次式で算出可能である.

$$\text{PTD} = \frac{\min \left( \sum_{i=1}^n \sum_{j=1}^m f_{i,j} \cdot \text{cost}(\mathbf{p}_i, \mathbf{q}_j) \right)}{\sum_{i=1}^n w} \quad (9)$$

subject to

$$\begin{aligned} \sum_{j=1}^m f_{i,j} &\leq w_i \\ \sum_{i=1}^n f_{i,j} &\leq \frac{u_j \sum_{i=1}^n w}{\sum_{j=1}^m u} \\ \sum_{i=1}^n \sum_{j=1}^m f_{i,j} &= \sum_{i=1}^n w \\ f_{i,j} &\geq 0 \end{aligned}$$

$f_{i,j}$  は, 供給地  $\mathbf{q}_j$  から需要地  $\mathbf{p}_i$  への輸送量を表す. PTD を重み付き特徴点の照合に適用するために, まず需要地と供給地をそれぞれ, スケッチから得られる特徴点と検索対象画像から得られる特徴点の位置とし, 需要量と供給量は各特徴点の重みで定義する. 輸送コストは需要地と供給地間のユークリッド距離を用いる. PTD が小さいほど, 2つの画像間で類似していると考えることが出来る.

## 5 実験

### 5.1 実験方法

提案システムを試作し 1120 枚のベクタ画像を用いて有効性を評価した. 使用したベクタ画像は SVG 形式のロゴマークである. Fig. 5 に試作システムの画面を示す. ユーザは図中の上側にあるキャンバス領域に, スケッチ画像を描画する. その後試作システムは, スケッチ画像をクエリとしてデータベース内のベクタ画像の検索を行い, 画面の下側に検索結果上位 10 件のベクタ画像を表示する.

評価実験では, Fig. 6 上で示される 7 つのロゴマークを所望の画像 (正解例) として, 1 人の入力



Fig. 5 試作システム

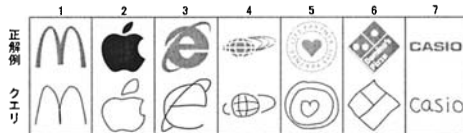


Fig. 6 所望のロゴマーク (正解例) とユーザが描いたスケッチ (クエリ)

者がこれらの画像をスケッチした。スケッチした画像 (クエリ) を Fig. 6 下に示す。これら7つのロゴマークは、入力者が過去に見てよく知っているものである。ただし、スケッチ画像を描画している間はロゴマークを見なかった。

### 5.2 実験結果

実験結果を Fig. 7 に示す。図は左から検索番号、入力したスケッチ画像 (クエリ)、入力者の所望の画像 (正解例)、試作システムによって出力された結果1位、2位、3位の順に示されている。また図の下側には重み付き特徴点の分布を示した。

1, 2, 3, 4 番の実験結果は成功例である。成功例は、重みの大きな特徴点の位置が、クエリと正解例の間ではほぼ一致している。重みの大きな点は、「端点」や「角点」のように曲率変化の大きな点である。また、重みの小さな特徴点がクエリと正解例のどちらにも存在し、これらの位置は一致していない場合もある。これは、本システムでパス中のすべてのアンカーポイントの特徴点としていることによる。しかし、これらの重みが小さい特徴点は検索結果への影響は少ない。なぜならPTDの計算をする際、特徴点の重みは需要量もしくは供給量とされるので、需要量もしくは供給量が少ないと輸送コストが高くて全体のコストには影響を与えないためである。よって、曲率変化の大きな曲線を含む画像に対して、本システムは有効である。

	クエリ	正解例	検索結果1位	検索結果2位	検索結果3位
1					
2					
3					
4					
5					
6					
7					

Fig. 7 実験結果

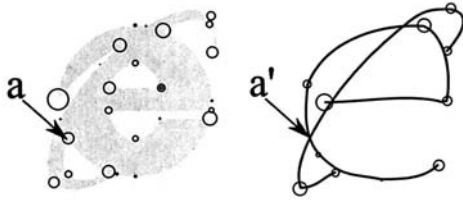


Fig. 8 3番の詳細図

5番の結果から、円形の画像の検索に失敗した。これはきれいな円をパスで表現すると、円周上の上下左右の4点がアンカーポイントとなる。しかし、スケッチは多少なりとも歪んだ円になるため特徴点が分散してしまう。その結果、類似度が低くなる。また6番のように、ユーザが菱形の形状をスケッチすると、円の特徴点分布とほぼ同じになり、検索結果の上位に円が出やすくなってしまいう問題がある。この問題を解決するためには、前処理で検索対象画像のパスに対してサンプル点を座標列として取り、セグメントの再分割を行う必要があると考えられる。

7番の結果から、特徴点が広範囲に分散している場合の検索に失敗していた。これは、PTDが特徴点の分布に対しての類似度を取るためである。つまり、特徴点が分散している画像はほとんど同じ類似度となってしまう。この問題に対し、特徴点を領域ごとに分割し、領域毎の特徴点の照合を行う事で解決することを現在考えている。

3番の結果から、スケッチの交点が特徴点として抽出されない問題がある。Fig. 8で示すように、正解例の特徴点  $a$  はスケッチでは点  $a'$  が対応すると考えられる。しかし、点  $a'$  は特徴点として抽出されていない。つまり、交点も特徴点として抽出できればより検索精度を向上できる可能性がある。

一般的にクエリの特徴点数は正解例よりも少ない。特に検索対象画像は接近して複数の特徴点が出ている事が多い。これはFig. 9で示すように、スケッチ画像は輪郭線が1つの線で表されているのに対し、検索対象画像は幅のある線状領域の輪郭として表されているためである。また角が丸い矩形は、角の付近で複数の特徴点に分かれる傾向がある。これらの特徴点は1つ1つの重みが小さくなる傾向があるため、近接する特徴点を統合するなどの後処理により、特徴点の増加を抑制する事が重要と思われる。



Fig. 9 6番のクエリと正解例上部の拡大図

## 6 おわりに

本稿では特徴点の照合に基づいたベクタ画像の検索システムを提案、評価した。実験の結果、曲率変化の大きな曲線を含む画像に対し提案システムが効率よく検索できる事を確認した。今後検索性能向上へ向け、特徴点の抽出方法、領域分割に基づく特徴点照合、交点の特徴点抽出、近接特徴点の統合などを行っていきたい。

## 参考文献

- 1) M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker, *Query by image and video content: the QBIC system Reference*, Addison Wesley Publishing Company, pp. 87-93, 1986.
- 2) J. R. Smith and S-F. Chang, *in Intelligent Multimedia Information Retrieval (M. T. Maybury, ed.)*, AAAI Press/MIT Press, 1997.
- 3) Panos Giannopoulos and Remco C. Veltkamp *A pseudo-metric for weighted point sets*, Proceedings of the 7th European Conference on Computer Vision-Part III, pp. 715-730, 2002.
- 4) Philip J. Schneider, *An Algorithm for Automatically Fitting Digitized Curves*, Graphics gems, no. 1, pp. 612-626, 1990.