

手書き風文字生成を用いた 印鑑自動デザインシステム

——文字の感性情報処理の一例——

塩野 充

岡山理科大学 工学部情報工学科

〒700岡山市理大町 1-1

E-MAIL: shiono@ice.ous.ac.jp

あらまし 印鑑は文具店等にいけば、既製品が安価で購入できる。しかし、それらは同じ姓であれば、殆ど同じ印影であり、しかも、一般的には全て姓だけであり、フルネームのものはない。姓だけ、あるいはフルネームにせよ、自分だけの唯一性の高い印鑑を求めようとすれば、いきおい特注にならざるを得ない。特注した場合、どのような字体になるかは、出来上がった実物を受け取るまでは分からず、自分の感性に合った好みの字体であるという保証もない。本研究では、以前提案した非線型な幾何学的変換を用いた手書き風の文字パターンの生成技法を、印鑑の自動生成に応用し、更に乱数処理を施すことによって、極めて多彩で唯一性の高い印鑑画像を簡単に生成することを可能にした。本研究で提案する印鑑の自動生成システムは、これを自動刻印装置に接続することにより、自分独自の唯一性の極めて高い印鑑を自分の感性に合った好みの字体で、フルネームで生成することが可能となる。

和文キーワード

手書き風文字、印鑑、幾何学的変換、文字生成、乱数処理、感性

Automatic Design System of Seal-Imprint using Handwritten Style Character Generation Technique.

——An Example of Kansei Information Processing for Characters——

Mitsuru SHIONO

Department of Information & Computer Engineering,

Okayama University of Science

1-1 Ridai-cho, Okayama-shi, 700 Japan

E-MAIL: shiono@ice.ous.ac.jp

Abstract

A seal-imprint can be bought at usual stationery shop with low price. But all of those ready-made seal-imprints are made with same character shapes, and they include only family name. If you want a seal-imprint containing your first name, you must order it to the shop. If you order a seal-imprint, you cannot see its character shape until you get it in your hand. In this paper, an automatic seal-imprint design system that can generate characters of various shapes. This system uses a method of hand-written style character generation by non-linear geometrical transform proposed in my former paper. By this system, any seal-imprint can be easily designed according to owner's kansei.

英文 key words handwritten style characters, seal-imprint, geometrical transform, character generation, random numbers processing, kansei

1. まえがき

印鑑は銀行等の金融機関、官公庁、企業その他、日本の社会の多くの場面で非常によく使われており、日常生活にとって必要不可欠な要素となっている。印鑑は一般の文具店等にいはば、既製品が安価で購入できる。しかし、それらは同じ姓であれば、殆ど同じ印影であり、印鑑の持つ使命の一つであるセキュリティは極めて低い。しかも、全て姓だけであり、フルネームのものはない。姓だけ、あるいはフルネームにせよ、自分だけの唯一性（ユニークネス）の高い印鑑を求めようとすれば、いきおい特注にならざるを得ない。特注した場合、セキュリティの高さは保持できるとしても、どのような字体になるかは、出来上がった実物を受け取るまでは分からず、自分の感性に合った気に入る字体であるという保証もない。

本論文で提案する手書き風の印鑑の自動生成システムは、これを自動刻印装置に接続することにより、自分独自の唯一性の極めて高い印鑑を自分の感性に合った好みの字体で、フルネームで簡単に生成することが可能となる。本論文の骨子は以前筆者が提案した非線型な幾何学的変換を用いた手書き風の文字パターンの生成方式⁽¹⁾を、印鑑の自動生成に応用した点である。さらに、乱数処理を施すことによって、極めて多彩で唯一性の高い印鑑画像が簡単に誰にでも生成することが可能である⁽²⁾。

2. 文字パターンの表現

文字パターンのデータ表現には種々の方法があるが、代表的には、ドット文字パターン、ベクトル文字パターン、輪郭線パターンの3種類であろう。ベクトル文字パターンは文字線を短い線分で折れ線近似して、文字をベクトルの集合として表す方法で、全てのベクトルの始点と終点を記憶していればよい。XYプロッタやドラフタ等の、線を描く出力装置で文字を表示する場合によく用いられる。1本1本のベクトルが短いほど、滑らかな文字線が描ける。本研究においては、文字パターンの幾何学的変換を行うに当たり、最も変形を行いやすい文字表現方式として、ベクトル文字パターンを使用した。ベクトル文字パターンを用いると、1本のベクトルの始点はその前につながっている前のベク

トルの終点と一致し、終点は次につながっているベクトルの始点と一致している。従っていかなる座標変換を受けても、元の文字線が切れ切れになったりして連続性が損なわれることはあり得ないので、本方式に適している。ベクトル表現された文字パターンを次式で表す。

$$H_c = \{ (x_m^{(1)}, y_m^{(1)}), (x_m^{(2)}, y_m^{(2)}) \mid m = 1 \sim W_c \} \quad (1)$$

ここで、 $(x_m^{(1)}, y_m^{(1)})$, $(x_m^{(2)}, y_m^{(2)})$ は第 m ベクトルのそれぞれ始点、終点を表す。又、 W_c は第 c カテゴリのベクトル文字パターン H_c を構成する全ベクトル数である。

3. 文字パターンの幾何学的変換

本システムで用いる幾何学的変換は以下の3種類である⁽¹⁾。

- ・ たる型（糸巻き）ひずみ
- ・ 台形ひずみ
- ・ 傾斜ひずみ

以下、これらについて説明を行う。

3. 1 たる型（糸巻き）ひずみ

たる型ひずみとは次式⁽³⁾で表現される非線型な変換により生じる幾何学的変形である。

$$\begin{aligned} x' &= \alpha_1 x \{ 1 + \beta_1 (x^2 + y^2) \} \\ y' &= \alpha_2 y \{ 1 + \beta_2 (x^2 + y^2) \} \end{aligned} \quad (2)$$

ここで、 (x, y) は元の座標位置、 (x', y') は変換後の座標位置を表す。又、 α_1 、 α_2 、 β_1 、 β_2 は変換の程度を決めるパラメータである。前のパラメータ α_1 、 α_2 は変換後の図形の全体の大きさを与えるパラメータである。後ろのパラメータ β_1 、 β_2 はこれらの値の違いによって、正方形の4辺の中央付近が膨らむたる型ひずみになったり、中央付近がへこむ糸巻きひずみになったりする。実際には式(2)の適用に当たっては、文字パターンの中心が x, y 座標の原点に来るように座標移動しておかなければならない。又、この式をそのまま適用すると変換は上下左右とも完全に対称な図形となり、手書き風の字体としては少し不自然な感を呈するので、文字パターンの中心を正確に座標原点におかずに、若干ずらす工夫を行っている。即ち、

$$\begin{aligned}
 x' - \gamma + \xi &= \alpha_1 u \{1 + \beta_1 (u^2 + v^2)\} \\
 y' - \gamma + \eta &= \alpha_2 v \{1 + \beta_2 (u^2 + v^2)\} \\
 u &= x - \gamma + \xi, \quad v = y - \gamma + \eta \\
 \gamma &= h/2
 \end{aligned}
 \tag{3}$$

とすることにより、中心位置を x 方向に ξ 、 y 方向に η だけ偏心させている。但し、偏心度 ξ 、 η をあまり大きくすると文字パターンがつぶれるので、文字パターンの外接矩形枠の 1 辺の長さの 2～3 割にとどめる。 h は画面の 1 辺の大きさである。

3. 2 台形ひずみ

更にもう 1 種類の非線型な幾何学的変換として、次式のような台形ひずみを用いている。

$$\begin{aligned}
 x' &= \{(d/a^2)y + 1\}x \\
 y' &= y
 \end{aligned}
 \tag{4}$$

パラメータ a は変換前の画面（正方形）の 1 辺の半分であり、 d は変換前の正方形と変換後の台形の上辺の差の半分を表し、変換の程度を決めるパラメータである。 $d > 0$ の場合は下辺より上辺の大きい下すばみの台形（逆台形）になり、 $d < 0$ の場合は上辺より下辺の大きい上すばみの通常の台形になる。このひずみを取り入れた理由は、例えば「日」という漢字を書いた場合、左右の 2 本の縦線は必ずしも平行に書かれるわけではなく、V 字形に近く、下すばみに書く人もいれば、A 字型に近く、上すばみに書く人もいることによる。このひずみは上記のたる型ひずみや糸巻きひずみとはまた、独立に発生し得ると考えた。

式（4）の適用に当たってはやはり文字パターン中心を座標原点に合わせる座標移動を必要とするが、この場合はたる型ひずみ（糸巻きひずみ）のときのような偏心操作は行っていない。たる型ひずみ（糸巻きひずみ）の印加にあたって既に偏心操作を行っているのでその上に更に偏心を行う必要はないと考えた。

3. 3 傾斜ひずみ

傾斜ひずみはスキューとも呼ばれ、マッチ箱の断面を押しつぶすときのような変形を表し、長方形が平行四辺形に変換される。これはアフィン変換の一部で線形変換であり、次式で表現

することができる。

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & s_1 \\ s_2 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}
 \tag{5}$$

ここで、 s_1 は x 方向への傾斜量を、 s_2 は y 方向への傾斜量を与えるパラメータである。

このひずみを取り入れた主な理由は、右上り又は右下りの手書き癖を表現するためである。右下りの文字を書く人はあまり多くないが、右上りの文字を書く人は非常に多く、殆どの人が程度の差はあっても若干の右上りの傾向があるとと言っても過言ではない。右上りや右下りを表すために有効なパラメータは s_1 だけである。もう一つのパラメータ s_2 はあまり多くは見られないが右倒れ又は左倒れの手書き癖を表す。

式（2）～（5）を用いて、式（1）の文字パターン H_c を表す全てのベクトルの、始点 $A_m(x_m^{(1)}, y_m^{(1)})$ 、終点 $B_m(x_m^{(2)}, y_m^{(2)})$ について変換を施し、変換後のベクトルの始点 $A'_m(x'_m^{(1)}, y'_m^{(1)})$ 、終点 $B'_m(x'_m^{(2)}, y'_m^{(2)})$ を求める ($m = 1 \sim W_c$)。これより変換後のベクトル文字パターン H'_c は次式で表される。

$$H'_c = \{ (x'_m^{(1)}, y'_m^{(1)}), (x'_m^{(2)}, y'_m^{(2)}) \mid m = 1 \sim W_c \}
 \tag{6}$$

但し、 $(x'_m^{(1)}, y'_m^{(1)})$ 、 $(x'_m^{(2)}, y'_m^{(2)})$ は第 m ベクトルのそれぞれ始点、終点を表す。又、 H'_c の全ベクトル数 W_c は式（1）と同じである。

3. 4 文字の大きさの正規化

上記の 3 種類のひずみを印加して文字パターンに幾何学的変換を行うと、文字の大きさ（座標中での文字存在領域）が変化する。特に糸巻きひずみの場合には極端に大きさが変化する。そこで、たる型ひずみ（糸巻きひずみ）を印加する場合には実際には式（3）の大きさの係数 α_1 、 α_2 はいずれも 1 としておき、3 種類全ての幾何学的変換を施したあとの文字パターンを囲む外接矩形枠の大きさを求め、その矩形枠を所定の大きさの正方形に変換するという、大きさの正規化を行っている。

4. 印鑑画像への適用

以上の方法によって生成した手書き風の文字

列を印鑑の文字に適用する。ここでは、印鑑の形状は最も一般的な円（真円）だけとし、楕円や矩形の印鑑は除外して考える。簡単に考えれば、生成した名前の文字列を円で囲めばよいわけであるが、実際には種々の操作が必要である。円で囲むためには、生成した文字列全体を囲む矩形の縦と横がほぼ同じ大きさである方が望ましい。印鑑の文字列は一般的には縦書きであるので、2文字や3文字などからなる名前を、各文字を正方形で出力したのでは、全体が縦に細長い矩形になる。それを円で囲むと、円の中の左右に広い空白の部分ができて、間延びした感じになってしまう。やはり、印鑑の円の中はバランスよく文字が詰まっている方が望ましい。また、本方式の特徴として、フルネームの印鑑が手軽にできる点がある。この場合、姓の下に名を続けると余計に文字列が長くなり、バランスの悪いものになってしまう。そこで、姓と名は2行に分けて表示するものとする。実際の手彫りの印鑑もこのようにしている場合が多い。そして、全体の矩形をできるだけ正方形に近くするために、次式によって各文字の大きさの調整を行う。

$$\begin{aligned} r_{ya} &= (r / m_a) \mu \\ r_{yb} &= (r / m_b) \mu \\ r_{x1} &= r \\ r_{x2} &= (r / 2) \mu \end{aligned} \quad (7)$$

ここで、 r は前述した大きさの正規化によって正方形に正規化された文字の1辺の大きさである。 r_{ya} は姓を表す各文字の縦方向の大きさで、 m_a は姓の字数である。 r_{yb} は名を表す各文字の縦方向の大きさで、 m_b は名の字数である。 r_{x1} と r_{x2} は各文字の横方向の大きさで、 r_{x1} は1行表示の場合、 r_{x2} は2行表示の場合である。1行表示は姓のみ、あるいは名をみの印鑑の場合である。2行表示は姓と名の表示の場合、あるいは、2文字の姓などを横書きで表示したい場合（銀行印でそのようにする場合がある）である。 μ ($0 < \mu \leq 1$)は文字と文字の間に隙間をあけるための係数で、 $\mu = 1$ の場合、隙間がなく、文字は上下又は左右でくっついた状態になる。本論文の実験では $\mu = 0.95$ とした。式(7)によって各文字を表示すると、文字集合全体を囲む矩形はほぼ正方形

となって、円の中に納まりやすくなる。

このようにして氏名の文字集合をほぼ正方形に近い矩形領域で表示したあと、その矩形の中心から円を描く。そして、円からはみ出した文字の部分は消去する。円の半径は小さすぎると中にある文字のかどを削り取ってしまうことになり、また、大きすぎると白い部分が目だって間延びした印象になる。円は必ずしも厳密に文字集合を包含している必要は無く、多少接触して文字が削られても印鑑らしく見える場合が多い。円の半径を ρ とすると、文字集合を囲む矩形（正方形）の1辺 r との関係は単純に外接円と考えると、

$$\rho = (\sqrt{2}/2) r \div 0.707 r \quad (8)$$

となるが、実験的に検討した結果、これではやや大きすぎる印象の場合が多く、ほぼ次式の範囲が実用的な許容範囲となるようである。

$$0.55 r \leq \rho \leq 0.7 r \quad (9)$$

図1に円の大きさを種々変えた場合の印影の例を示す。以後は、 $\rho = 0.6 r$ を標準とする。

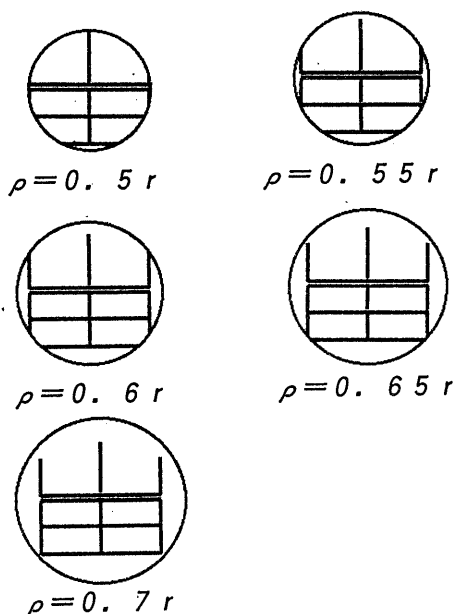


図1 印鑑の円の半径の制御

5. 自動生成システム

以上に述べた方法により、印鑑画像の自動生成システムを構築した。ハードウェアはパソコンPC98系で、言語はBASICコンパイラである。ディスプレイ画面の例を図2に示す。画面からマウスで制御可能なパラメータは以下の10種類である。

- (1) 縦方向傾斜ひずみのパラメータ s_1
- (2) 横方向傾斜ひずみのパラメータ s_2
- (3) 台形ひずみのパラメータ d
- (4) 縦方向たる型（糸巻き）ひずみのパラメータ β_1
- (5) 横方向たる型（糸巻き）ひずみのパラメータ β_2
- (6) 縦方向の偏心パラメータ ξ
- (7) 横方向の偏心パラメータ η
- (8) 印鑑画像表示画面の背景色 σ (0 = 黒、1 = 白)
- (9) 印鑑の文字と円の線の太さ τ
- (10) 円の半径 ρ

これらのパラメータは任意の値を選択できるようなもできるが、ユーザが使いやすくするために決まった段階数に量子化してある。 s_1 は

0の場合を含めて合計11段階、 s_2 も同様に11段階、 d も11段階、 β_1 、 β_2 もそれぞれ11段階、 ξ 、 η はそれぞれ6段階、 τ は3段階、 ρ は5段階である。従ってこの場合に1個の印鑑画像について生成可能な種類は、 $11 \times 6^2 \times 3 \times 5 = 86,967,540$ 通り、則ち、約8700万通りである。更にこれに多様性を付加するために、次に述べる乱数による処理を加えている。

6. 乱数処理

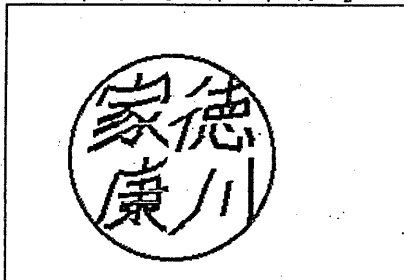
ある幾何学的変換のパラメータを z 、その乱数処理後のパラメータを z' 、値域 $[0, 1]$ なる一様乱数を ϕ とすると、

$$z' = z + k(\phi - 0.5)z / 10 \quad (10)$$

なる処理を行う。これは乱数によってパラメータ z を増減させる処理である。 k はその程度を制御する係数で、 $k=1$ のときは、 z を $\pm 5\%$ の範囲で増減させる。同様に $k=2$ のときは $\pm 10\%$ 、 $k=3$ のときは $\pm 15\%$ 、 $k=4$ のときは $\pm 20\%$ 、 $\dots$ 、 $k=10$ のときは $\pm 50\%$ となる。増減があまり小さいと殆ど原画像

印鑑自動デザインシステム (Ver.17)

「デザ印くん」



日本語入力モードにしてから、姓と名を空白1つで区切って入力して下さい。
徳川 家康

姓/名の文字数=2/2
黄色の枠内をダブルクリックするとその設定値で実行します

1. 右上がり/左上がりの度合い

右上がり	0	1	2	3	4	5	3
左上がり	-1	-2	-3	-4	-5		

2. 右倒れ/左倒れの度合い

右倒れ	0	1	2	3	4	5	2
左倒れ	-1	-2	-3	-4	-5		

3. 上すぼみ/下すぼみの度合い

下すぼみ	0	1	2	3	4	5	-3
上すぼみ	-1	-2	-3	-4	-5		

4. 縦方向の膨らみ/へこみの度合い

へこみ	0	1	2	3	4	5	1
膨らみ	-1	-2	-3	-4	-5		

5. 横方向の膨らみ/へこみの度合い

へこみ	0	1	2	3	4	5	-1
膨らみ	-1	-2	-3	-4	-5		

6. 文字中心の偏りの度合い

縦方向	0	1	2	3	4	5	2
横方向	0	1	2	3	4	5	3

印鑑背景色	黒	白					白
文字の太さ	細	中	太				太
円の半径	1	2	3	4	5		3

氏名の入力 終了 乱数処理

図2 画面表示の例

と変わらなくなり、乱数処理の意味がなくなる。また、増減があまり大きいと、乱数処理の結果、折角ユーザが作成した原画像から大幅に変化してしまった画像になり、ユーザの意図にそぐわないものとなる。式(10)の乱数処理を s_1 、 s_2 、 d 、 β_1 、 β_2 、 ξ 、 η の7つのパラメータに対して施す。ここで乱数処理を行う目的は以下のとおりである。ユーザが各パラメータをマウスでいろいろと操作して、自分の気に入った印鑑画像が出来上がったときに、そのままでは上述したように同姓同名の中の8700万通りの一つとなる。しかし、人間の好みというものはさほど大きな差異がなく、あまり極端な変形が好んで選択される可能性は低く、この8700万通りが等確率で選択されるとは考えられない。従って、實際上、選択される可能

性のある組合せを仮にこの $1/100$ としても87万通りある。従ってこのままでも十分に唯一性(ユニークネス)は高いが、更に乱数処理を施すことにより、唯一性をほぼ無限にまで高めることが可能となる。用いる乱数のシーズ(種)はいつも同じでは、いつも同じ乱数列が発生されて意味がないので、ユーザの生年月日と、使用日時、並びにその秒を含む時刻の数字を元にしてシーズを生成している。以上の乱数処理の結果、厳密に同一の印鑑画像はほぼ、二度と出現しないといってよいであろう。このような工夫によって、生成する印鑑画像の唯一性を極めて高めることが可能となる。図3に係数 k を種々変えた場合の印鑑画像の変化の様子を示す。表示している類似度は単純類似度で、画像としての類似性を示す。 k が大きくなるに連れ

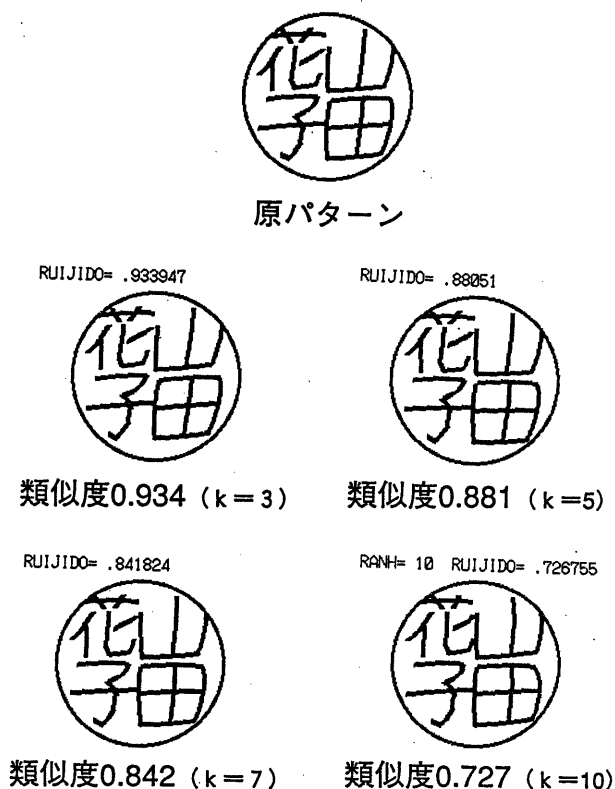


図3 乱数処理の強さと印影の変化の程度

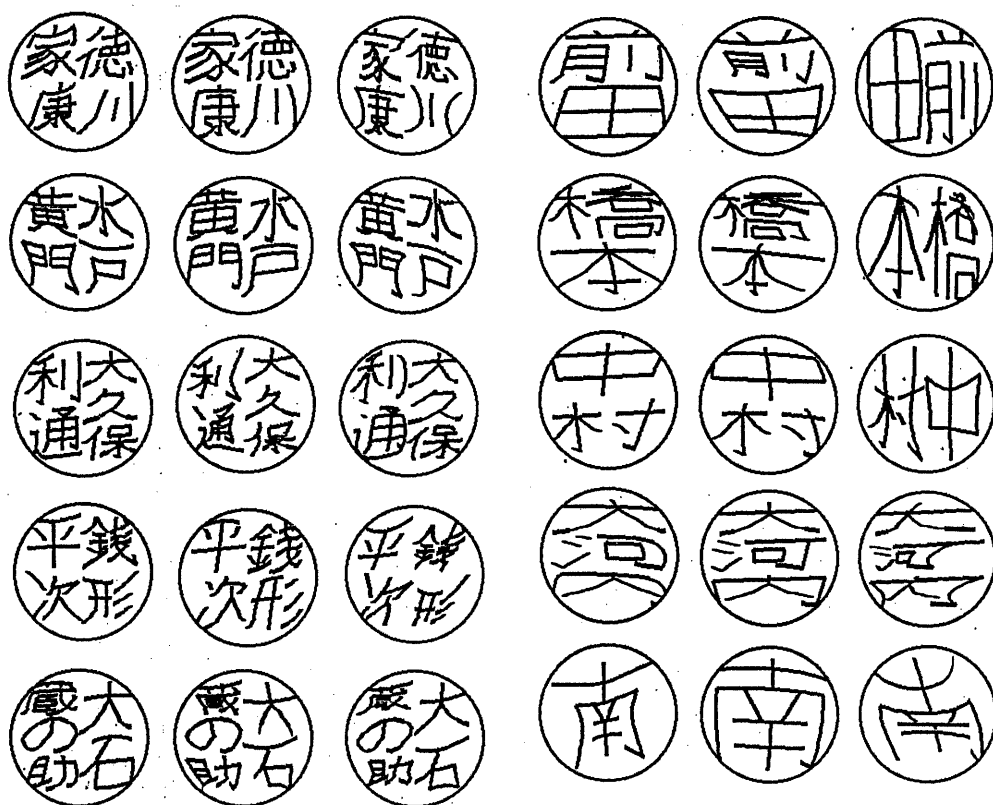


図4 出力結果例

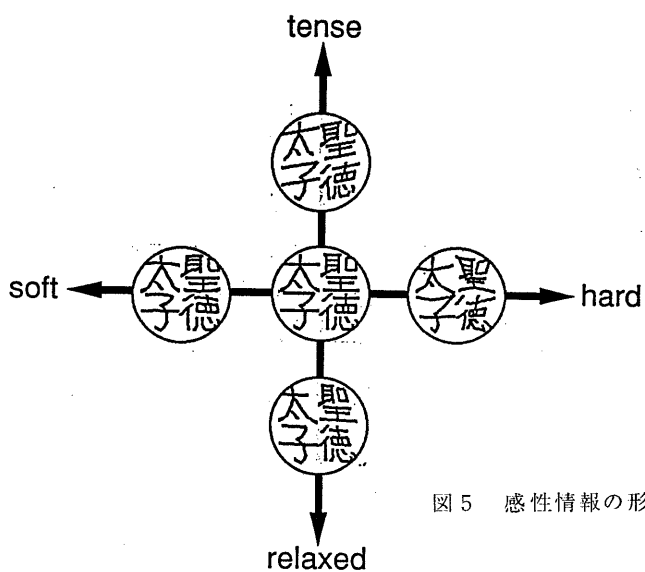


図5 感性情報の形容詞空間での配置の一例

て類似度が下がってゆくのが分かる。 $k=3$ や $k=5$ では原パターンとの違いは殆ど分からないが、 $k=10$ になると、よく見るとかなりの部分で違いが生じている。しかし、印影の全体的な雰囲気、印象は余り変わっていないことが分かるであろう。なお、この図は k を変化させた場合の一例に過ぎない。同じ k の値を用いても使用する乱数は時々刻々変化しているので、これと微視的に厳密に同一の画像を再度生成することはまず困難である。

本システムにおいては、各パラメータの値は画面上で全てマウスでクリックして選択できるようになっており、ダブルクリックすると、その時に選択してある係数で生成された印鑑画像が表示されるようになっていく。従って、キーボードを操作する必要があるのは氏名の入力の時だけである。印鑑画像の生成と表示に要する処理時間は0.1秒以下であり、実用的に問題はない。図4に、本システムで生成した印鑑画像の例を示す。これらの印鑑画像が持主の好みに合うかどうかは各個人の感性の問題であり、これらの生成結果を一律に客観的に評価することは困難である。

7. 感性情報処理の面からの検討

手書き風の生成文字が、それを見る人にどのような印象を与えるかと言うことを検討する事は感性情報処理⁽⁴⁾の一つのテーマであろう。本稿の残りの紙数でそれを詳しく検討することはできないが、一つの取り掛かりとして、手書き文字が与える印象を表す形容詞を考えてみた。手書き文字が読む人に与える印象と言うのは、文字自身の印象を通して、それを書いた人の人柄、性格、受取人に対する意識、書いたときの状況その他の属性を読み手に推測させる作用がある。それらは例えば、「上手な字」⇔「下手な字」、「丁寧な字」⇔「乱雑な字」、「柔らかな字」⇔「堅い字」、「几帳面な字」⇔「自堕落な字」、「四角い字」⇔「丸い字」、「しっかりした字」⇔「子供っぽい字」、等々が考えられる。ここでは、一つの例として、4つの形容詞（意味に幅を持たせるために英単語とした）を取り上げて検討する。それは2つずつ対比する形容詞であり、soft⇔hard、relaxed⇔tenseとする。これを2

次元の座標平面に見立てて、対応するであろうと思われる字体（この場合は印影）を並べてみたのが図5である。これはあくまで筆者の個人的な主観で並べたものに過ぎず、見る人によっては異論もあろう。これは手書き文字に関する感性情報処理の一つの入口として呈示したにすぎないが、今後、この方面のより本格的な研究も必要であろう。

8. むすび

非線型な幾何学的変換などを用いて手書き風の文字を生成する手法の応用の一つとして、印鑑画像の自動生成システムを構築し、実験を行った。本システムを用いれば、極めて唯一性の高い自分だけの印鑑が簡単にデザインできる。実用的には、このシステムを市販の自動刻印装置に接続し、データを転送することによって本システムで作成した印鑑画像を実際の印鑑として製作することが可能である。本システムの特徴は、姓だけ、名だけ、姓と名（フルネーム）など、所望の印鑑が、自分の好み、感性に合った字体で簡単にデザインできることである。又、その結果が乱数処理などによって、極めて高い唯一性をもつので、印鑑の本来の目的である高いセキュリティを保つことが可能である。今後の課題としては、さらに多様な幾何学的変換の数式の検討と、それによって係数が増加した場合に、ユーザが係数を制御しなくても、感性情報処理の技術を用いて、形容詞の張る多次元空間内で自由に字体を連続変化できるような仕組みを実現できないか検討したいと考える。

[文 献]

- (1) 塩野充：“非線型な幾何学的ひずみを用いた手書き風文字パターン生成の一手法”，信学論D-II，J74-D-II，2，pp. 209-219（1991-2）。
- (2) 塩野充：“持主の感性に応じた字体を生成する印鑑自動デザインシステム”，画像電子学会第143回研究会，94-04-04，（1994-11）。
- (3) 長尾真：“画像認識論”，コロナ社（1983）。
- (4) 井口征士ほか：“感性情報処理”，オーム社（1994）。