

ALS患者のための自身の手書きを残しコミュニケーションに利用可能とする Web サービスの実現

古賀壮^{1,a)} 中村聡史¹ 掛晃幸² 石丸築²

概要: 筋萎縮性側索硬化症 (ALS) は手足などの筋肉がだんだんやせて力がなくなっていく病気であり、その患者は徐々に手が動かなくなり自分で字を書くことが困難になっていくという問題を抱えている。我々は、ALS の当事者からのニーズをもとにして、筋肉が痩せ手書きができなくなる前に自身の手書き文字の形状を記録し、平均手書き文字生成手法により手書きのメッセージ画像を生成する Web サービスを実現した。ここでは、サービスの実現にあたり、試作したシステムを随時使ってもらったうえでまた別の試作を行うといった開発を行った。本稿ではその過程と残された課題について報告をする。

キーワード: 手書き, デジタル手書き, 加重平均化, 個性, ALS

1. はじめに

筋萎縮性側索硬化症 (ALS) とは徐々に筋肉が衰えていく病気である。この病気の患者は筋肉の衰えにより徐々に手が動かなくなるため、次第にペンなどを握って字を書くことが困難になる。しかし技術の進歩により、視線入力や脳波を読み取ることで文字入力を可能にする研究 [11][8] も推進されており、病気が進行した後も入力したテキストを用いたコミュニケーションが可能になりつつある。

ここで、コンピュータ上のフォントを利用した文字より手書きのほうが味やあたたかみがあり好印象であると考える人も多くおり [9], そのような人にとって手紙などで手書きできなくなるのは大きな問題である。実際我々はこの問題について、ALS の当事者の方から、「私らしさを表現するものとして、ペン持たなくなっても自分の文字を残したい」という直接の依頼を受けた。自身の手書きをデジタルの形で残し、利用可能とするものとして手書きの入力からフォントファイルを生成するアプリケーション [1] が存在しており、そうして生成されるフォントファイルはさまざまなアプリケーションで使えるというメリットがあるが、文章中に複数回同じ文字の並びが出てきたときに、全て同じ形状になってしまい、違和感が生じるという問題がある。

そこで本研究では、手書き文字を書くことができる進行

前の段階で手書き文字を記録しておき [13], その記録された手書き文字群をゆらぎのある加重平均化 [14] を行うことで、違和感のない手書き文字列からなる画像を生成する Web サービスの実現する。具体的には、登録される手書き文字を数式化し、その数式化された手書き文字を様々な重み付けを行い平均化して 1 つの合成文字を作ることで、同じ字であっても全て違う形にする。また、その生成される文字を適切にレイアウトすることで、ひとが書いた手書きらしさを実現する。ここでは、本サービスの実現にあたり、システムの試作と試作への ALS の当事者からのフィードバックに基づき再度システムを試作するといった過程と、その成果について報告する。

2. 関連研究

手書き文字自体を平均化することで、手書きをより良くする研究として、Zitnick ら [5] は、文字認識なしで手書きのストロークをリアルタイムで平均化することで、手書きを美化する手法を提案しており、幅広い書体で字が改善されることを示している。一方田村ら [14] は、機械が生成した手書き文字と実際の手書き文字の区別がつかないような平均手書き文字の生成手法と、物理的な紙にロボットを使って書く方法を提案し、実際に多くの人は区別ができないことを明らかにしている。Matayoshi ら [2] は平均文字を応用して自他の文字と融合してより良い字にするノートアプリケーションの実装と検証を行い、他者の字と融合することでより良い印象となることを明らかにしている。

一方、手書き文字をより良くする研究にもさまざまなも

¹ 明治大学
Meiji University

² 株式会社ワコム
Wacom Co., Ltd.

a) me@kgtkr.net

のがある。Xia ら [4] は手書き文字にお手本となるテンプレートを適用することによって、手書き文字をお手本に近づけつつ、楷書体生成手法を使い、整った手書き風の楷書体を生成する手法を実現している。Sasaki ら [3] は、手書きとフォントを融合したメッセージカードを生成可能とする仕組みを実現し、手書きとフォントの融合は恥ずかしさを軽減しつつ、もらって嬉しいものになることを明らかにしている。本研究は、手書きとフォントを融合するのではなく、そのまま活用することで、味のある手書きメッセージを送付可能とするものである。

さらに、手書きフォント生成に関する研究も様々なものがある。高村ら [10] は、手書きフォントを生成してコミュニケーションをするためのシステムを作り、手書きフォントが感性情報を有することを明らかにしている。また安本ら [6] は、手書きフォントを自動生成するアルゴリズムを提案し、限られた手書き文字から自然な字を生成できることを示している。こうした手書きフォントは、画一的な印象を与えてしまうという問題がある。しかし、こうした研究における生成手法は、限られた入力手書き文字から、登録していない手書き文字を生成可能となる可能性がある。

一方、ALS の患者のコミュニケーションを支援する研究も多数存在している。小林ら [11] は、ALS 患者が視線入力で意思伝達を行う装置を開発しており、実験により多くの人がこの装置で入力可能なことを明らかにしている。また井上ら [7] は、ALS の患者が少ない操作で文章作成を行う方法を提案し、操作回数と操作時間が減少することを明らかにしている。さらに加納 [8] は、Android スマートフォンで脳波を用いて患者の意思を特定するアプリケーションを開発しており、慣れていない患者に対しては誤差があるが意思の特定が可能であることを示している。このようなコミュニケーション支援は今後実用化される可能性がある。

3. 手法

本稿では、ユーザの手書き文字を蓄積し、入力されたテキストデータから手書き文章の画像を生成するシステムを実現する。本システムの実現にあたり、試作とその試作に対するフィードバックをもらうということを繰り返しているが、本章ではその試作の基本となる手法について述べる。

3.1 手書き文字の平均化

まず、図 1 のように手書き文字の一画ごとの点の座標データを取得し、その点を出来るだけ接続するように 3 次スプライン補間を行い、間を埋める点を生成する。次に、その補完された点からなる平面曲線である文字の数式をフーリエ級数によって求める。これにより、平面曲線において一般的な曲線を媒介変数表示で数式化し、数式の平均によって平均文字を生成可能とする。

ここでシステムの性質上、手書き文字の平均化は、少な

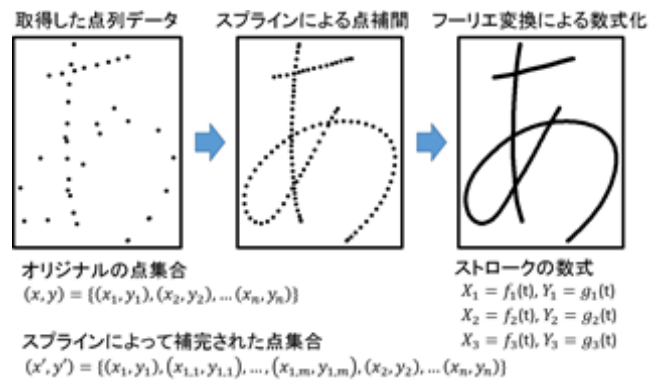


図 1 手書き文字の数式化

い入力データからユーザの特徴を残したまま多くの文字のバリエーションを生成可能とする必要がある。そこで田村ら [14] の加重平均化手法を利用し、 n 件の平均化に使う文字のうち i 番目の文字の各ストロークをフーリエ変換することで各ストロークの x, y 座標それぞれを表す t ($-\pi \leq t \leq \pi$) をパラメータとする関数 $f_{xi}(t), f_{yi}(t)$ を求め、それを各文字の重みパラメータ w_i を使って以下の式で加重平均して平均化後の $x(t), y(t)$ を求めている。

$$\begin{cases} x(t) = \sum_{i=1}^n w_i f_{xi}(t) \\ y(t) = \sum_{i=1}^n w_i f_{yi}(t) \end{cases}$$

また、加重平均化を行っても、そのユーザ自身が登録した手書きの文字数が 1 つしかない場合や、そもそも登録されていない場合はその対象文字を表現することができない。そこで、本稿では他者が登録した字の形状も使うことを可能とする。 $n, w_i, f_{xi}(t), f_{yi}(t)$ に対応する他者の字の変数をそれぞれ $m, v_i, g_{xi}(t), g_{yi}(t)$ とした時以下の数式で平均化した。

$$\begin{cases} x(t) = \sum_{i=1}^n w_i f_{xi}(t) + \sum_{i=1}^m v_i g_{xi}(t) \\ y(t) = \sum_{i=1}^n w_i f_{yi}(t) + \sum_{i=1}^m v_i g_{yi}(t) \end{cases}$$

なお、加重平均の重み w_i, v_i を一様分布から取ってきた乱数を使うと件数が増えたときに、平均値の一定の値に収束する性質によってほとんどの文字が似たような形になってしまいバリエーションが出ない。またこのような平均文字は過度に綺麗で、不自然かつ字の個性が消えてしまう。そこで、Algorithm 1 で示す疑似コードによって w_i, v_i を求めた。ただし `std_normal_dist` は標準正規分布を生成する関数、`shuffle` は配列をシャッフルする関数である。

上記のアルゴリズムにより、各重みは $s^0, s^1, \dots, s^{(n-1)}$ に乱数を加えた値となる。ユーザが設定する s の値が大きいほど字のばらつきが大きくなるため個性的となり、 s の値が小さいほど全ての字を等しく使うようになるのでばらつきが少なくなり、個性の少ない平均的な文字となってしま

Algorithm 1 $w_i v_i$ を求める

```

Require:  $0 < s < 1$ : ユーザが設定したゆらぎパラメータ,
 $0 < q < 1$ : 他人の字をどのくらいの割合で使うか
1: for  $i = 1 \rightarrow n$  do
2:    $w[i] \leftarrow \max(0.5 \times \text{std\_normal\_dist}() + \text{pow}(s, i - 1), 0.01)$ 
3: end for
4:  $w \leftarrow \text{shuffle}(w)$ 
5: for  $i = 1 \rightarrow m$  do
6:    $v[i] \leftarrow \max(0.5 \times \text{std\_normal\_dist}() + \text{pow}(s, i - 1), 0.01)$ 
7: end for
8:  $v \leftarrow \text{shuffle}(v)$ 
   { 重みの合計を 1 に正規化する }
9:  $w \leftarrow w / \text{sum}(w) \times (1 - q)$ 
10:  $v \leftarrow v / \text{sum}(v) \times q$ 
    
```

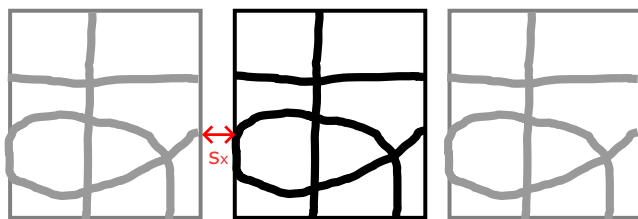


図 2 横書きの x 座標

3.2 文字の配置

手書き文字のレイアウトについては、x 座標は図 2 のように、実際にユーザが書いた字の最小矩形を基準に、字間のパラメータ s_x を使って等間隔に並べる。ここで文字を配置する y 座標は下揃えにしていると違和感が生じる。そこで最小矩形が横に長い場合は図 3 のように固定長の字のサイズ (letter size) の中で、 $r = w / (w + h)$ (ただし w, h は最小矩形の幅と高さ) とした時、最小矩形が中央揃えになる位置と下揃えになる位置が $r : 1 - r$ となる位置に配置した。行同士は行間を設定するパラメータ s_y を使って等間隔に配置した。

縦書きについても、x 座標は横書きの y 座標と、y 座標は x 座標と同じアルゴリズムを縦横を入れ替えて実装した。

なお、一般的に手書き文字では漢字は大きく、ひらがなは小さく手書きされやすい。また、画数の多い手書き文字は大きく、また少ない手書き文字は小さく手書きされやすい。そうした点を考慮して、ひらがなやカタカナ、漢字などの文字のデフォルト値を設定するとともに、前の文字の大きさに基づき手書き文字の大きさを設定するものとした。さらに、生成される手書き文字の羅列はアルゴリズムで生成するだけでは違和感が残る。そこで、生成された文字をクリック (タッチ) することで、大きさを変更可能とする。図 4 に、文字をレイアウトしている様子を示す。

4. システムの試作

最終的なサービスの実現にあたり、複数の試作を行った。

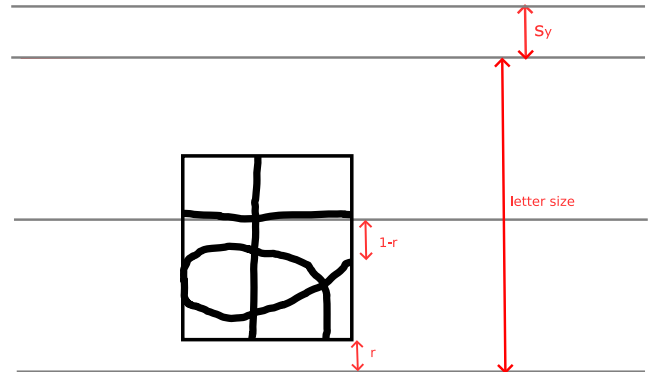


図 3 横書きの y 座標

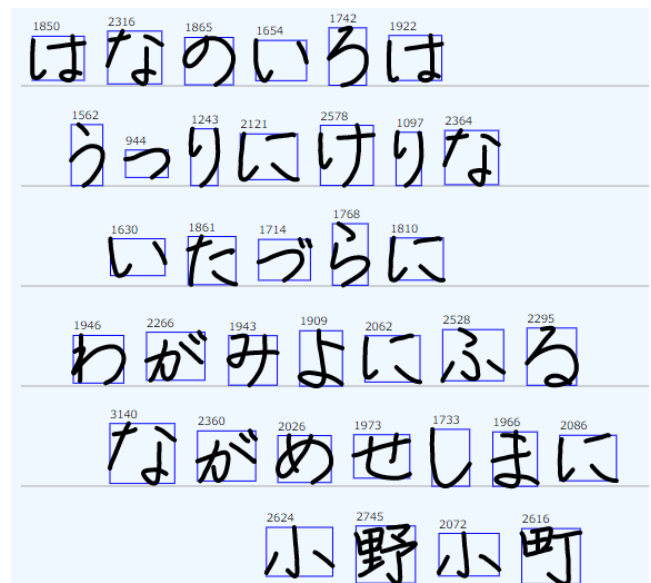


図 4 手書き文字のレイアウト

ここでは、その試作とその狙い、またフィードバックから明らかになった問題点について整理する。

4.1 試作 1

ALS の当事者のための手書き入力および再現サービスを実現するにあたり、日常的に気軽に手書き文字を登録してもらう必要がある。そこで、今回は日常的に持ち歩き、空き時間で利用することが可能なスマートフォンアプリケーションとして実現することとした。

ここで実装では、事前に手書き文字を登録してから利用するのではなく、手書き文字で表現したいメッセージを作成してから足りない手書き文字を登録するよう要求するものとした。また、依頼を頂いた方が iPhone を利用してい



図5 試作1

ため、iPhoneで動作するアプリケーションとしてUnityを用いて実装した。この試作の第1バージョンとして平均手書き文字手法をベースとしたプロトタイプシステムを図5に示す。

試作1では、ユーザは入力する文字とその画数を指定して登録し、その登録した文字について複数回の手書きを行うことで順次登録可能となっている。また、テキストボックスに表現したい文章を入力し、1行あたりの文字数や文字の太さ、平均化に使用する文字データの最大数などを指定し「平均文字で表示」ボタンをおすことで、平均文字を用いた文章が生成可能となっている(図5右)。

実現したシステムについて、手書きを登録してもらい、また手書き文字を作成してもらった。好意的なフィードバックを多く頂いたが、長時間ペンをもって手書きをすることには抵抗があるとのコメントを頂いた。また、手書きを登録する前に画数を指定することへの手間に関するコメントも頂いた。さらに、ネイティブアプリケーションとして実現していたため、スマートフォンにトラブルが合った場合や、スマートフォンを乗り換えた場合、また間違っただけでアプリケーションを削除したときにこれまでに入力した手書きデータが失われてしまうという問題もあった。Dropboxなどにバックアップする機能を追加で実現していたが、そのバックアップに時間が掛かるなどの問題もあった。一方、一つの閉じた環境でしか手書きを行うことができず、自身の手書きを利用してパソコンで手紙などの作成を行うことができなかった。

4.2 試作2

試作1における当事者の方からの、ペン入力を長時間行うことに対する抵抗があったこと、画数を指定することの手間などの問題、また様々な環境で利用可能とするため、手書きを指で入力可能とするWebベースのアプリケーションとして実装した。また、画数を指定せずとも登録可能とした。実装したアプリケーションの画面を図6に示す。

ここでは、手書きにおいて「中」を4画で書いたり、続け字で3画で書いたりといったように、手書きによるブレ



図6 試作2. 入力画面

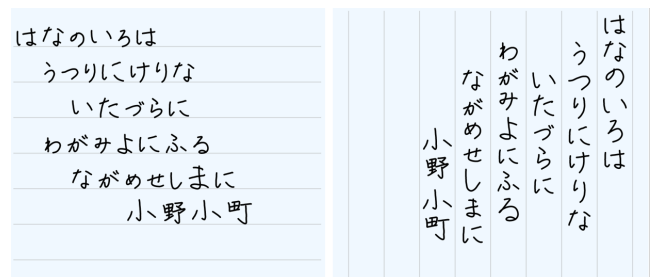


図7 試作2. 縦書き横書き

を考慮し、画数を指定せずに入力可能とし、利用時に自動で使い分けることとした。また、どの程度手書き文字を登録したかを確認できるようにしつつ、これからどのような文字を登録しなければならないかを示すため、図6右のように入力文字列を登録できるようにした。

また、テキストボックスへの入力から、図7のように縦書き・横書きのどちらについてもレイアウト可能とした。ここで、手書き文字をタップすることで、平均化のやり直しなどを行い、より自然な手書き表現が可能となった。さらに画像として手書き文字メッセージをダウンロードし、他者へ送ることを可能とした。

実際に利用してもらい、縦書きや横書きのレイアウトができること、スマートフォンで指でなぞることで手書きができることの良さに関するフィードバックを得た。一方で、指でしっかりとスマートフォンの面をなぞらなければならず、指の力が弱い場合にうまく入力できないなどの問題があった。また、先にテキスト入力したものを表示し、足りない文字の登録を要求する仕組み自体は評価されていたものの、登録しなければならない文字のリストを図6右のように多数表示するのはプレッシャーになるという問題があった。さらに、手紙として利用するためには背景画像を設定したり、レイアウトを自由にする必要があった。

5. tegaki.fun

5.1 実装

試作におけるフィードバックをもとに、Webサービスtegaki.funを実現した。ここでは筆圧が弱めでも登録できるようにするとともに、入力リストは表示しないようにした。また他者へ手書きを共有する仕組みを実現し、他社の

文字を利用可能とした。さらに、背景画像を設定可能とするとともに、行間や文字サイズなど様々な設定を可能とした。なお、これまでの試作からデータをインポートし、書き溜めた手書きを流用可能とした。

実装においては、Web システム*1としてフロントエンド*2は TypeScript と React、バックエンド*3は Rust を用いた。また、手書き文字の平均化処理を行うライブラリとして average-figure-drawer*4を、手書き入力部用のライブラリとして WILL SDK for ink*5を利用して実装した。

5.2 利用方法

ユーザが Google アカウントでログインすると 図 8 で示す文章生成画面が表示される。背景画像を選択し、生成したい文章を入力すると 図 9 のようにプレビューが表示され、形状が登録されていない文字（ここでは「ど」）を登録するように促される。登録されていない文字である「!」をクリックすると 図 10 のような字を書くフォームが表示され、手書き文字を登録することができる。

背景画像と文章の位置は、図 11 のように左と上の余白を調整が設定可能である。他にも配置に関するパラメータとして書字方向、文字間と行間を、字の生成に関するパラメータとして文字色、文字の大きさと太さ、ゆらぎパラメータ s 、他人が書いた字をどのくらいの割合で混ぜるかというパラメータを指定可能とした。またプレビューで表示されている各文字をクリックすることで乱数のシード値を変更することができ、気に入らない文字がある時に調整ができるようになっている。最後にダウンロードボタンをクリックして png 画像としてダウンロードした結果が 図 12 である。

生成画面の下には 図 13 のように使われた文字の一覧と件数（最大 10 件）が表示され、ここから文字を登録することもできる。ここでは自分で 1 件も書いていない文字は赤く強調表示されるため、必要な文字をまとめて入力することで順番に登録可能となっている。

他の画面としては 図 14 で示す文字設定一覧画面と、図 15 で示す形状の一覧ページとユーザ設定画面がある。文字設定では文字の画数の設定が可能である。形状の一覧ページでは、登録した形状の一覧とその平均の確認、形状の削除ができる。ユーザ設定画面では、書いた字を他人も使えるようにするかを設定できる。デフォルトでは自分しか使えない設定になっているが、他人も使えるように設定してくれるユーザが増えればサービスを使い始めたばかり

*1 <https://tegaki.fun>

*2 <https://github.com/nkmr-lab/average-character-cloud-frontend>

*3 <https://github.com/nkmr-lab/average-character-cloud-backend>

*4 <https://github.com/nkmr-lab/average-figure-drawer>

*5 <https://developer.wacom.com/ja-jp/products/will-sdk-for-ink>

のユーザでも試しやすくなる。



図 8 ログイン後の画面



図 9 背景画像と文章の入力

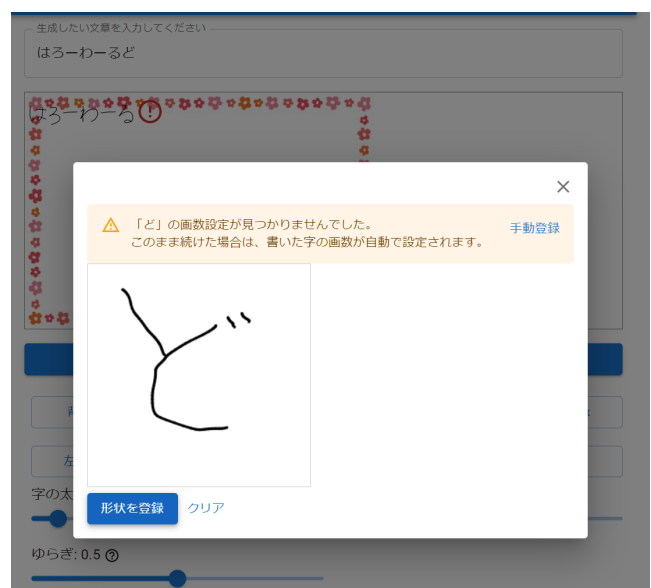


図 10 文字形状の登録

5.3 生成されたメッセージカード

実現したサービスを Web で公開し、ALS の当事者に利用してもらった。利用してもらったうえで頂いたメッセージカードを図 16 に示す。ここで生成されている手書きは、まさにその当事者の方が登録した手書き文字をベースとして生成されており、味わいがある。また、このメッセージ

カードにおいて「と」と「き」は複数回登場しているが、それぞれ形が異なるものであるため、違和感がない。

6. 考察

試作と試作に対するフィードバックから、手書きを記録し、手書きメッセージ画像を生成可能とする仕組みを実現できた。また、もともとの依頼主である ALS の当事者の方からクリスマスカードを頂くとともに、様々な人に対してクリスマスカードを送るために利用させて頂くとコメントを頂いており、利用可能性も高いものであると考えられる。

最終的に実現したサービスにより、スマートフォンで登録した手書き文字を利用して PC で手書きメッセージ画像を生成可能となるなど、その利用の幅は広がったと考えられる。しかし、現時点のシステムは、設定項目が多いため、複雑な操作が要求されるといった問題がある。そこで今後はテンプレートなどを用意し、複雑な操作を必要としない



図 11 位置の調整



図 12 結果画像

| 使っている文字 | |
|-------------|-----------------|
| 「た」の形状を追加する | (平均化に使われた数: 2件) |
| 「ち」の形状を追加する | (平均化に使われた数: 2件) |
| 「つ」の形状を追加する | (平均化に使われた数: 0件) |
| 「て」の形状を追加する | (平均化に使われた数: 4件) |
| 「と」の形状を追加する | (平均化に使われた数: 3件) |
| 「な」の形状を追加する | (平均化に使われた数: 2件) |

図 13 使われた文字の一覧

| 文字設定一覧 | |
|-----------|------|
| 文字設定を新規作成 | |
| あ | (1画) |
| い | (2画) |
| あ | (3画) |
| い | (2画) |
| う | (2画) |
| え | (2画) |
| お | (3画) |
| か | (3画) |
| が | (5画) |
| き | (4画) |

図 14 文字設定の一覧



図 15 登録済み形状の一覧



図 16 生成されたメッセージカード

ような仕組みを実現予定である。また、将来的には手書きのデータベースに対する API アクセスを可能とすることによって、自身の手書きを活用可能なサービスを他者が開発可能とし、様々な環境で自身の登録した手書きを利用できるような仕組みを実現していく予定である。

ALS の患者に対し、自身の声を残せるうちに残しておくことの重要性が医者などから語られることがある。声を音として録音しておけば、将来的にその録音データから学習し、その患者の声を個性を保ったまま再現することができるようになると期待される。それと同じように、手書きを残しておくことも重要であると考えられる。本稿では不完全ではあるが、手書きを残したいと思わせる仕組みを実現するものであり、こうした手書きが将来的に利活用できると期待するものである。

本サービスはユーザに手書きを 1 文字ずつ登録してもらう必要があり、登録されていない手書き文字については追加で手書き文字を登録するか、他者の手書き文字を使って表現するしか方法がない。ここで、日本語で用いられる漢字は偏（へん）と旁（つくり）からなり、ある漢字のパーツを他の漢字に流用可能である。そこで今後は、登録されている漢字の手書き文字を偏と旁などの構成要素に分解し、再構成することで手書き文字として利用可能とする予定である。また、膨大な手書き文字が集まれば、類似した手書きを書くユーザを求め、その手書きを部分的に借りることによって自身の手書きらしさのある手書き文書を作成可能とすることも考えられる。

また、今回実装したサービスでは手書き文字を登録してもらう必要があるが、将来的にはこれまで書き溜めたノートなどをカメラで撮影することで、手書き文字として利用可能にすることも検討している。オフライン手書き文字の数式化についてはこれまでも検討を進めてきている [12] が、現時点では精度は良くない。そこで今後は、アルゴリズムを改良することによりカメラで撮影した手書き文字の数式化を実現していく予定である。

7. まとめ

本稿では、ALS の当事者の方から受けた「私らしさを表現するものとして、ペン持てなくなっても自分の文字を残したい」という依頼をもとに、ユーザが自身の手書き文字を残し、手書き文字の平均化手法を活用することで手書きメッセージ画像を生成可能とするサービスを、様々な試作を経て実現した。また、サーバにデータが保存される Web システムであることを活かして他人の字の形状を使えるようにしたり、ブラウザだけあればどこでも使える手軽なものとした。その結果、手書き文章画像を生成することによるコミュニケーションの可能性とその課題が明らかになった。

今後は、ユーザインタフェースを整理するとともにテン

プレート機能を充実させていく予定である。また、各々のメッセージを独自に設定可能とする仕組みも実現予定である。さらに、カメラなどでこれまでに手書きしてきたノートを撮影し、その手書き文字を活用可能とする仕組みを実現予定である。

謝辞 ALS の当事者として「手書き文字の個性を記憶し利用可能とするサービスの実現」を依頼いただき、随時試作したシステムに対してフィードバックを頂いた牛上彩さまに深く感謝します。また、試作 1 の開発に寄与してくれた明治大学総合数理学部先端メディアサイエンス学科の齋藤大慶さんに深く感謝します。

参考文献

- [1] ifontmaker - create your font in 5 minutes on your ipad. <https://2ttf.com>. アクセス日: 2022-12-20.
- [2] Yasutsuma Matayoshi, Satoshi Nakamura, and Ryo Oshima. Mojirage: average handwritten note. *Proceedings of the 2018 International Conference on Advanced Visual Interfaces (AVI)*, No. 74, pp. 1–3, 2018.
- [3] Mikako Sasaki, Junki Saito, and Satoshi Nakamura. Analysis of utilization in the message card production by use of fusion character of handwriting and typeface. Vol. 11749, pp. 25–33, 2019.
- [4] Weiping Xia and Lianwen Jin. A kai style calligraphic beautification method for handwriting chinese character. In *2009 10th International Conference on Document Analysis and Recognition*, pp. 798–802. IEEE, 2009.
- [5] C Lawrence Zitnick. Handwriting beautification using token means. *ACM Transactions on Graphics (TOG)*, Vol. 32, No. 4, pp. 1–8, 2013.
- [6] 安本護, 池田尚志, 豊倉完治ほか. 手書き風フォントの自動生成について. 全国大会講演論文集, データ処理, pp. 427–428, 1994.
- [7] 井上倫夫, 小林康浩, 加納尚之, 井上公明ほか. コミュニケーション・エイドのためのかな-漢字変換法について. 情報処理学会研究報告ヒューマンコンピュータインタラクション (HCI), Vol. 1989, No. 72 (1989-HI-026), pp. 1–8, 1989.
- [8] 加納尚之. 脳波 (erp) による android スマートフォンアプリを用いた als 患者の yes/no の特定. In *IEICE Conferences Archives*. The Institute of Electronics, Information and Communication Engineers, 2019.
- [9] 寺田恵理, 保崎則雄. 手で書くこと, 手書き文字に対する認識に関する一調査. 日本教育工学会論文誌, Vol. 42, pp. 145–148, 2018.
- [10] 高村将大, 小川剛史ほか. おてがみ: 個人の手書き文字フォントを用いたコミュニケーションシステム. 研究報告グループウェアとネットワークサービス (GN), Vol. 2013, No. 18, pp. 1–7, 2013.
- [11] 小林量作, 大竹朗, 中島孝, 福原信義, 竹田恵, 服部千秋, 熊沢良彦. 590 視線入力方式による意思伝達装置の紹介: Als 患者における臨床的有用性. 理学療法学 Supplement Vol. 28 Suppl. No. 2 (第 36 回日本理学療法学会大会 第 28 巻大会特別号 No. 2: 演題抄録集), p. 295. 公益社団法人日本理学療法士協会, 2001.
- [12] 新納真次郎, 齊藤絢基, 久保田夏美, 中村聡史, 鈴木正明. オフライン手書き文字数式化手法の提案と大規模平均文字の比較. グループウェアとネットワークサービス (GN), No. 7, pp. 1–8, 2017.
- [13] 中村聡史, 鈴木正明, 小松孝徳. ひらがなの平均手書き

文字は綺麗. 情報処理学会論文誌, Vol. 57, No. 12, pp. 2599–2609, 2016.

- [14] 田村洗希, 中村聡史. 人を騙す手書き自動生成手法の提案と実装. 情報処理学会 研究会報告ヒューマンコンピュータインタラクション (HCI) , Vol. 2020-HCI-186, pp. 1–8, 2020.