

平成28年度卒業論文

手書き入力システムへのストロークの
自動クラスタリング機能の実装

情報・通信工学科 コンピュータサイエンスコース

1311101 下澤一輝

指導教員 寺田 実 准教授

提出日 2017年 1月31日

概要

目的

本研究室の寺田が作成した、手書き Wiki という手書き入力システムがある。これは主に講義において電子ホワイトボードの役割で利用されている。手書き Wiki では、グループ機能を用いることでユーザーがシート内容を任意の位置でグループ化することができる。また、グループ化したものをコピー&ペーストや移動など編集をすることも可能である。

このシステムではシート描画内容をグループ化する際、ユーザーが対象をグループの枠で囲む必要がある。しかし、講義中などに編集したい場合、操作に手間取り聞き手側の集中力を欠いてしまうなどの問題点がある。

そこで、Lee らのクラスタリングアルゴリズム [1] を手書き Wiki に適用し、シート内容を自動的にクラスタリングすることで編集のしやすさを向上させることを本研究の目的とする。

方法

手書き入力されたストロークを構成する点の座標からストローク毎の距離関係を計算することで、シート内容を容易に整理、編集できるように自動クラスタリング機能を追加した。この機能では3つのクラスタリングパターンを用意し、グループの選択や移動を可能にしている。

結論

自動クラスタリングシステムの実装は、異なるクラスタリングパターンをユーザーの意図に応じて使い分けることで、シート描画内容の編集を円滑に行う手助けとなった。一方、クラスタリング結果に関しては重みの調整が最終的にうまくいかず、細かい箇所では不自然なグループに分割されてしまうなど、一部使いにくいような点を残してしまった。

目次

第 1 章	序論	6
1.1	背景	6
1.2	目的	6
1.3	本論文の構成	6
第 2 章	関連研究	7
2.1	A flexible framework for online document segmentation by pairwise stroke distance learning[1]	7
2.1.1	概要	7
2.1.2	本研究との関係	8
2.2	cLuster: Smart Clustering of Free-Hand Sketches on Large Interactive Surfaces[2]	8
2.2.1	概要	8
2.2.2	本研究との相違点	8
第 3 章	クラスタリング機能	9
3.1	概要	9
3.2	操作方法	10
3.2.1	クラスタリング	10
3.2.2	クラスタリングパターン	10
3.2.3	グループ数の増減	11
3.2.4	グループの選択	11
第 4 章	実装	12
4.1	開発環境	12
4.2	処理の流れ	12
4.3	ストロークのデータ構造	12
4.4	タイムスタンプ	13
4.5	クラスタリングアルゴリズム [4][5]	13
4.5.1	アルゴリズム	13
4.5.2	距離の定義	15
4.5.3	比較尺度の種類	15
4.6	GUI	18
4.6.1	クラスタリングパターン	18
4.6.2	ブロック数の変更	18
4.6.3	ブロックの移動	18

第 5 章	評価実験	19
5.1	評価方法	19
5.1.1	使用端末	19
5.1.2	実験の内容	19
5.2	クラスタリング結果	19
5.2.1	シート 1	19
5.2.2	シート 2	20
5.2.3	シート 3	20
5.2.4	考察	21
5.3	分割数の増減	21
5.3.1	考察	21
5.4	実行時間	22
5.4.1	考察	23
第 6 章	結論	24
6.1	結論	24
6.2	今後の課題	24
6.2.1	機能の改善	24
6.2.2	機能の追加	24

目次

2.1	クラスタリング結果 ([1] より引用)	7
2.2	cLuster の操作画面 ([2] より引用)	8
3.1	提案システムの画面	9
3.2	クラスタリング後の画面	10
4.1	クラスタリング	13
5.1	Object1	19
5.2	Horizontal1	19
5.3	Vertical1	19
5.4	Object2	20
5.5	Horizontal2	20
5.6	Vertical2	20
5.7	Object3	20
5.8	Horizontal3	20
5.9	Vertical3	20
5.10	Object(分割数 10)	21
5.11	Horizontal(分割数 10)	21
5.12	Vertical(分割数 10)	21
5.13	Object(分割数 2)	21
5.14	Horizontal(分割数 2)	21
5.15	Vertical(分割数 2)	21
5.16	Object パターンの実行時間	22
5.17	Horizontal パターンの実行時間	22
5.18	Vertical パターンの実行時間	23

表目次

4.1 ストロークのデータ構造	12
4.2 特徴ごとにわけたグループ	17

第1章 序論

1.1 背景

手書き入力では、「マウスよりもポインティングが簡単である」「図を容易に描くことができる」など、マウス操作に慣れていない人が手軽に操作できる。また、手で描くという行為は人間の自然の行為であり親しみやすい。授業や会議において電子ホワイトボードが用いられることが多くなり、タブレット型端末では手書き入力を主とするアプリケーションなども増えている。

しかし、描かれた内容の一部をグループ機能によってまとめる際、多くの場合ユーザーが必要な箇所をペンで囲う必要がある。これでは、囲う動作自体に手間がかかってしまったり、適切な箇所を囲むことができないなどの問題点がある。

1.2 目的

本研究の目的は、先行研究のクラスタリング手法を手書き Wiki に実装し、シート描画内容を自動的にグループ化することで、ユーザーの操作を補助し、容易に編集を行うことができるようにすることである。

1.3 本論文の構成

第2章では、本研究に関連する研究について述べる。

第3章では、クラスタリング手法を実装したシステムの説明や使用方法について述べる。

第4章では、クラスタリングアルゴリズムや GUI の実装について述べる。

第5章では、評価実験について述べる。

第6章では、結論と今後の課題について述べる。

第2章 関連研究

2.1 A flexible framework for online document segmentation by pairwise stroke distance learning[1]

2.1.1 概要

Lee らの研究では、手書きで描かれた文字や図形を階層化クラスタリングするための計算アルゴリズムを提案している。このアルゴリズムではストローク間距離を定める比較尺度として 30 種類以上用意し、それらに重みをかけて足し合わせることでストローク間の距離を計算している。比較尺度には空間的な位置で決まるもの以外にも時間的な位置 (2 つのストロークの描画時刻の差) なども含まれている。

そして、この重みやクラスタを併合する基準となる閾値を正確に決めるために、正解のクラスタ内に含まれるストロークに対しクラスタリングを行い、学習させている。正解のクラスタと計算結果がどれぐらい合っているかは、SR (Segmentation Recall) と TSR (Tolerant Segmentation Recall) という尺度を用いて評価している。SR とは正解クラスタと完全に一致するストローク集合の割合を、TSR は細かく分類しすぎではあるが和集合をとったときには一致している割合を示している。

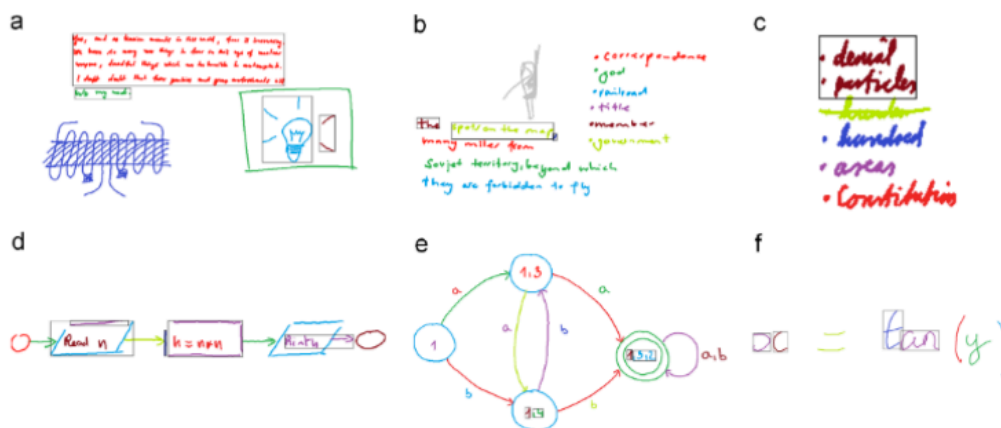


図 2.1: クラスタリング結果 ([1] より引用)

図 2.1 は Lee らのクラスタリングアルゴリズムをデータセットに適用したあとの結果である。a から順に (a) 「図」 (b)(c) 「テキストライン」 (d) 「フローチャート」 (e) 「図」 (f) 「数式」 これらのスケッチはフローチャートデータベース、オンライン文書データベースなど様々な種類のデータベースから用いられている。図の黒い四角で囲まれている箇所は間違った結果を表している。分割のしすぎや、分割されるべき箇所で分割されていないなどの誤りがみられる。

2.1.2 本研究との関係

本研究ではこの Lee らのアルゴリズムを手書き Wiki に適用している. Lee らは 31 の比較尺度を用いているが, 本研究ではこのうち 18 個のみを引用した.

2.2 cLuster: Smart Clustering of Free-Hand Sketches on Large Interactive Surfaces[2]

2.2.1 概要

Florian らの研究では, 板書内容を容易に整理することを目的に, 手書きで描かれた文字や図形を自動的にクラスタリングする手法 cLuster を開発, 提案している. cLuster では, ユーザーが手書きスケッチの一部を選択することで, 残りの部分を自動的にクラスタリングすることができる. また, 上下左右にペンをスライドすることで選択範囲を広げたり, クラスタリングされたもの(グループ)同士の位置を入れ替えるなど, インタラクティブな操作が可能になっている.

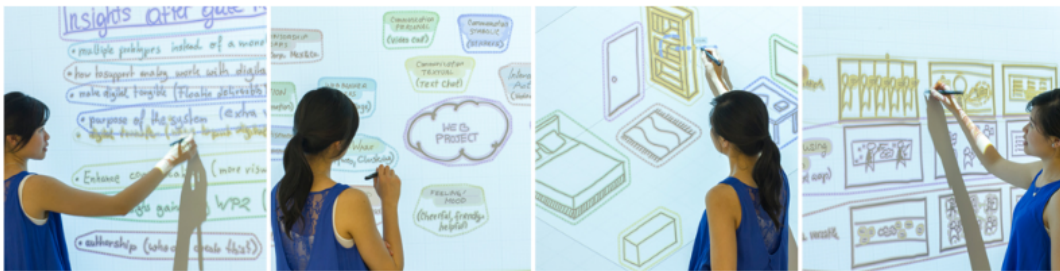


図 2.2: cLuster の操作画面 ([2] より引用)

2.2.2 本研究との相違点

Florian らの提案システム cLuster では, 前節で紹介した Lee らのクラスタリングアルゴリズムを用いて, 最初に選択したストローク集合を元にスケッチ内容をクラスタリングしている. また, ユーザーが任意に指定した位置でブロックを分割することができる. 一方, 本研究ではシステムが一方的にクラスタリング結果を提示するのみであり, ユーザーが変更できるのはクラスタリングパターンと分割数のみである. さらに cLuster で用意されているクラスタリングパターンは学習データによって求めた 1 つのみであるが, 本研究では 3 つの異なるパターンを用意した. その点でも本研究とは異なる.

第3章 クラスタリング機能

3.1 概要

本研究で実装した機能は手書き入力されたストロークを自動的にクラスタリングすることで、シート内容をいくつかのまとまり (グループ) に分割することができる。既存のグループ機能では選択したい箇所をユーザーが四角で囲う必要があったが、この機能によりその操作が不要になり提示されたグループを容易に編集することが可能である。

図 3.1 は機能を追加後の手書き Wiki の画面の一部である。

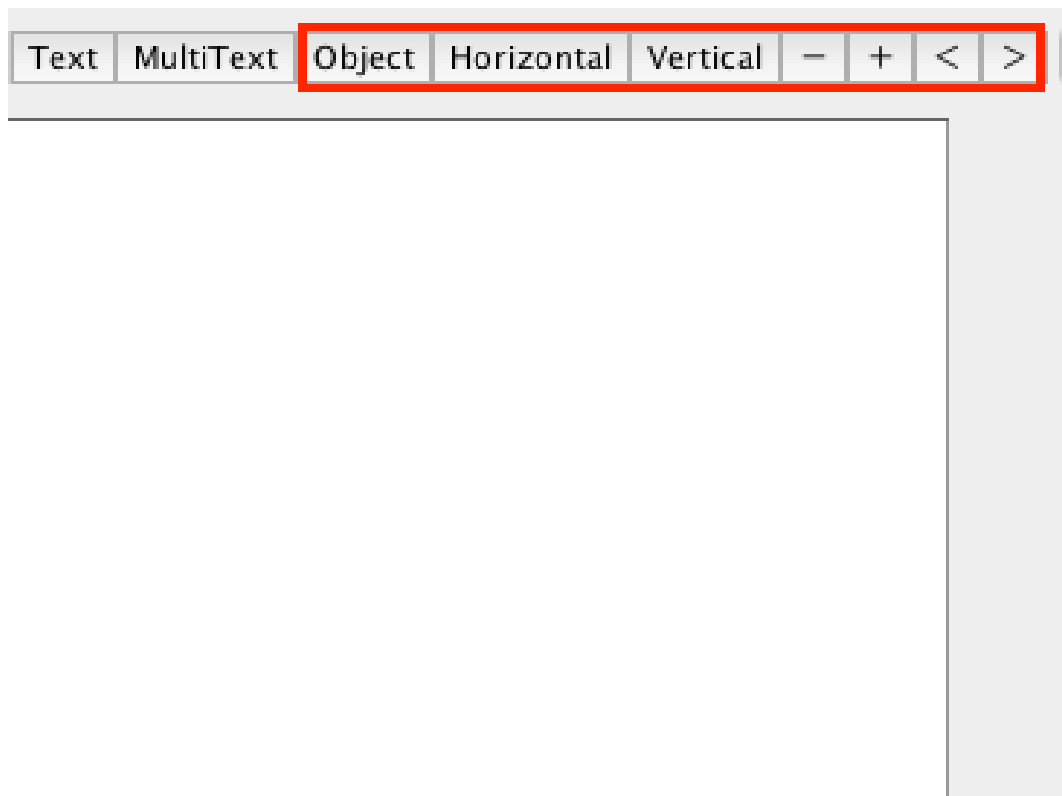


図 3.1: 提案システムの画面

赤い枠で囲まれた箇所が本研究で追加した機能である。「Object」「Horizontal」「Vertical」ボタンはクラスタリングを実行するためのボタン,「-」「+」はグループ数を増やすためのボタン,「<」「>」はグループの選択に用いるボタンである。詳細は以下で述べる。

3.2 操作方法

3.2.1 クラスタリング

提案する3つのクラスタリング手法から1つを選択することで、シート内に描かれた手書き文字を自動的にいくつかのグループに分類することができる。具体的な操作方法としては、ツールバーの「Object」「Horizontal」「Vertical」(以下「クラスタリングボタン」)のいずれかのボタンを押すと、初期状態として4つのブロックに分類される。また、ブロックはそれぞれ緑の枠で囲まれるが、これは「Group」機能を使用した時に表示されるものと同一のものである。

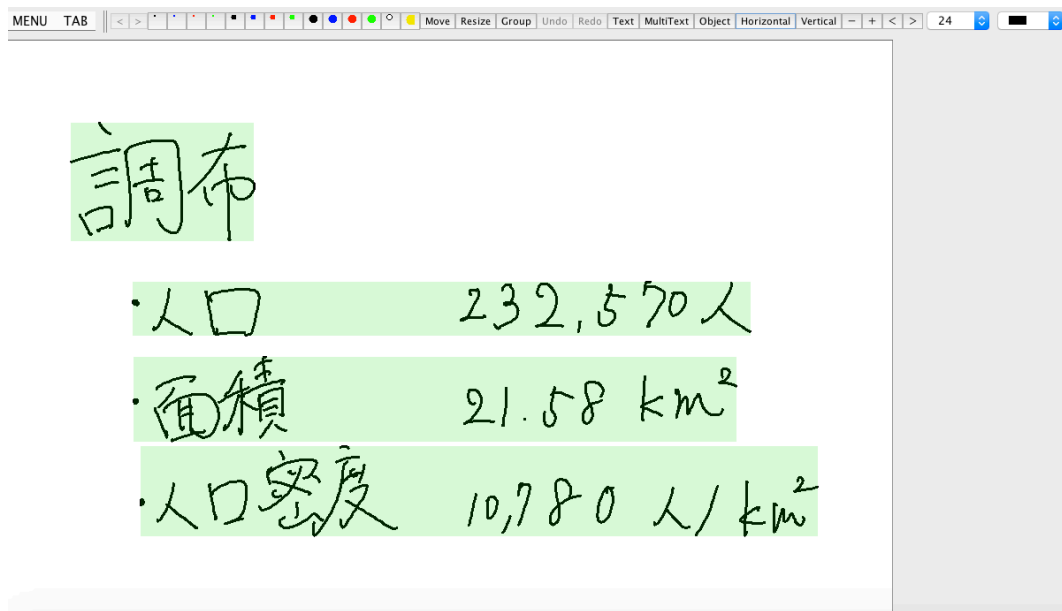


図 3.2: クラスタリング後の画面

3.2.2 クラスタリングパターン

本研究ではユーザーがクラスタリングの方法を切り替えられるように3つのパラメータセットを用意した。実装した3つのパターンは以下の通りである。

Lee らの論文を参考に、使用する距離の尺度を組み合わせ、実験を行いながら重みを調整した。

- パターン「Object」：実際の距離が近いものを優先するクラスタリング
- パターン「Horizontal」：横方向優先のクラスタリング
- パターン「Vertical」：縦方向優先のクラスタリング

3.2.3 グループ数の増減

グループ数を増やしたい場合はツールバーの「-」「+」ボタンを押すことで増減させることができる。また、この時グループ数を記録しておき、次にクラスタリングボタンを押した時、そのグループ数でクラスタリングされるようにした。

3.2.4 グループの選択

クラスタリング後のグループは「<」「>」ボタンを押すことで選択することができる。グループを移動させる際にはこの機能は不要であるが、コピー&ペーストをしたい場合には最初にこの機能で選択してから行う必要がある。

第4章 実装

4.1 開発環境

本システムは、手書き Wiki と同様に Java で記述し、GUI 部分には Swing[3] を用いている。また、統合開発環境として Eclipse を用いた。

4.2 処理の流れ

クラスタリング結果を表示するまでの流れは以下のとおりである。

1. シートのデータからストローク情報を読み込む。
2. 押されたボタンに対応したパターンの距離を計算。
3. 指定されたグループ数になるまでクラスタリング。
4. グループ構造の中にストロークのクラスタを入れ、画面に表示。
5. 増減ボタンが押された場合はその都度クラスタリングし表示。

4.3 ストロークのデータ構造

表 4.1 に手書き Wiki におけるストロークのデータ構造の例を示す。

表 4.1: ストロークのデータ構造

ファイルの中身	意味
STROKE	レコードの種類
52	レコードの行数
1	ペンの色
3	ペンの太さ
1484565319581	書き始めのタイムスタンプ
1484565320205	書き終わりのタイムスタンプ
165	ストローク上の点 1 の x 座標
120	ストローク上の点 1 の y 座標
164	ストローク上の点 2 の x 座標
121	ストローク上の点 2 の y 座標
⋮	⋮

表 4.1 の様に、ストロークは色や太さ、タイムスタンプ情報、ストロークを構成する点の座標によって成り立っている。本研究では、このうちタイムスタンプと点の座標を用いてストローク間の距離を定義する。

4.4 タイムスタンプ

タイムスタンプとはある時刻を記録しておくためのものである。既存の手書き Wiki にはストローク毎にタイムスタンプは実装されていないが、本研究ではストローク同士の時間的な距離を計算するため、ストロークの書き始めと書き終わりにタイムスタンプを実装した。

4.5 クラスタリングアルゴリズム [4][5]

4.5.1 アルゴリズム

クラスタリングとはデータを「クラスタ」という部分集合に分けることである。同じクラスタに分類されたもの同士は似ている (内的結合) が、別のクラスタに分類されたもの同士は似ていない (外的分離)。図 4.1 の例では、クラスタ A に入っているもの同士、クラスタ B に入っているもの同士は似ているが、クラスタ A とクラスタ B に入っているもの同士は似ていない、ということになる。

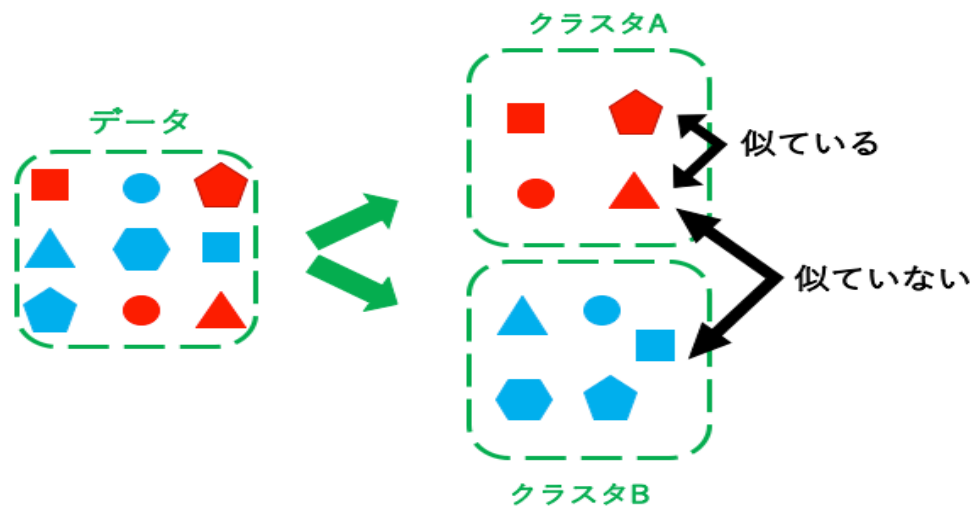


図 4.1: クラスタリング

クラスタリングには特定のクラスタに分類される非階層化クラスタリングと、階層的に分類される階層化クラスタリングというものがあり、本研究では階層化クラスタリングを用いた。以下では、それぞれのクラスタリング手法のうち代表的なもののアルゴリズムについて説明する。

非階層化クラスタリング (分割最適化手法)

非階層化クラスタリングは分割最適化手法とも言われ、分割の状態を表す評価関数を最適にするクラスタリング手法である。非階層化クラスタリングの代表的なものとして、k-平均法というものがある。クラスタ集合を $X = X_1, X_2, X_3, \dots, X_i, \dots, X_k$ 、クラスタ X_i のセントロイド (重心) を \bar{X}_i 、データ集合を $x = x_1, x_2, x_3, \dots, x_n$ とする時、評価関数

$$E = \sum_i^k \sum_{x \in X_i} \|x - \bar{X}_i\|^2 \quad (4.1)$$

を最小にするようにクラスタを分割する. 具体的なステップは以下の通りである.

1. データ集合 x を k 個のランダムなクラスタに分割する.
2. 各クラスタについてセントロイド $\bar{X}_i = \frac{1}{|x_i|} \sum_{x \in x_i} x$ を求める.
3. 全てのデータ x を 2. で求めた各クラスタのセントロイドとのユークリッド距離 $\|x - \bar{X}_i\|$ が最小となるクラスタへ分類する.
4. 2. および 3. の操作を, 前のクラスタ集合と変化がなくなるか, 初期値として定めた最大反復回数を超えるまで繰り返す.

階層化クラスタリング

一方, 階層化クラスタリングとは, 初期状態として全てのクラスタにそれぞれ1つのデータが入っており, クラスタ間距離を元に再帰的に併合することで階層構造のようにクラスタリングする手法である. 階層化クラスタリングの場合, クラスタ間距離の定義の仕方により様々な手法が存在している. その例をいくつか説明する. ただし, C_1, C_2 はそれぞれクラスタを, $d(x_1, x_2)$ は x_1 と x_2 の距離を表す.

- ・最短距離法 2つのクラスタの要素間の距離の最小値をクラスタ間距離とする手法

$$d(C1, C2) = \min_{x_1 \in C_1, x_2 \in C_2} d(x_1, x_2) \quad (4.2)$$

- ・最長距離法 2つのクラスタの要素間の距離の最大値をクラスタ間距離とする手法

$$d(C1, C2) = \max_{x_1 \in C_1, x_2 \in C_2} d(x_1, x_2) \quad (4.3)$$

- ・群平均法 2つのクラスタの要素間の距離の平均値をクラスタ間距離とする手法

$$d(C1, C2) = \frac{1}{|C_1||C_2|} \sum_{x_1 \in C_1} \sum_{x_2 \in C_2} d(x_1, x_2) \quad (4.4)$$

- ・ワード法 クラスタを合成した際のセントロイドと各点までの距離の二乗の総和を最小にするように併合

$$d(C1, C2) = E(C_1 \cup C_2) - E(C_1) - E(C_2) \quad (4.5)$$

ただし, クラスタ C_i のセントロイドを \bar{C}_i とすると

$$E(C_i) = \sum_{x \in C_i} (x - \bar{C}_i)^2 \quad (4.6)$$

階層化クラスタリングの具体的なステップは以下のとおりである.

1. データ $s_1, s_2, s_3, \dots, s_n$ をそれぞれクラスタとして設定し, n 個のクラスタを作る.
2. クラスタ間の距離を計算し, 最も距離が近いクラスタ同士を併合する.
3. 2. の処理をクラスタが1つになるか, 閾値を超えるまで繰り返す.

4.5.2 距離の定義

本研究では階層化クラスタリングを用い、クラスタ間距離 $d(s, t)$ を、空間的な距離、時間的な距離等様々な尺度に重みをかけ足し合わせることで求めている。そのいくつかある尺度を $f_i(s, t)$ 、それにかける重みを w_i とすると、

$$d_i(s, t) = 1 - \frac{1}{1 + f_i(s, t)} \quad (4.7)$$

$$d(s, t) = \sum_{i=1}^k w_i d_i(s, t) \quad (4.8)$$

(4.7) でそれぞれの尺度 $f_i(s, t)$ が 0 から 1 の値になるように標準化して $d_i(s, t)$ とし、(4.8) で重みをかけたものの総和をとっている。本来、重み w_i は学習データによって算出するものであるが、本研究では比較尺度の意味を考慮し、実験を行うことで調整した。

4.5.3 比較尺度の種類

ここではクラスタ間距離を定義するのに使用した比較尺度について説明する。尚、比較尺度は Lee らの提案手法で使用されたものを引用した。まず、数式の例とその意味を挙げておく。

- $P_{s,a} = (x_{s,a}, y_{s,a})$: ストローク s 上の a 番目の点 P の x 座標は $x_{s,a}$, y 座標は $y_{s,a}$
本研究では表 4.3 で示したようにタイムスタンプは書き始めと書き終わりのみに記録している。
- $\|P_{s,a} - P_{t,b}\|$: ストローク s 上の点 $P_{s,a}$ とストローク t 上の $P_{t,b}$ のユークリッド距離
- n_s : ストローク s 上の点の数
- $P_s \cdot = \frac{1}{n} \sum_a P_{s,a}$: ストローク s の重心
- l_s : ストローク s の長さ
- x_s^m : ストローク s 上で最小の x 座標 x_s^M : ストローク s 上で最大の x 座標
- y_s^m : ストローク s 上で最小の y 座標, y_s^M : ストローク s 上で最大の y 座標
- $w_s = x_s^M - x_s^m$: ストローク s の横幅
- $h_s = y_s^M - y_s^m$: ストローク s の高さ
- $w_{st} = \max(x_s^M, x_t^M) - \min(x_s^m, x_t^m)$: ストローク s とストローク t 全体の横幅
- $h_{st} = \max(y_s^M, y_t^M) - \min(y_s^m, y_t^m)$: ストローク s とストローク t 全体の縦幅

以下に、本研究で使用した距離の定義を述べる。

- (1) $f_{min}(s, t) = \min_{a,b} |P_{s,a} - P_{t,b}|$:
ストローク s 上の点とストローク t 上の点を結んだ線の距離のうち最短の距離
- (2) $f_{max}(s, t) = \max_{a,b} |P_{s,a} - P_{t,b}|$:
ストローク s 上の点とストローク t 上の点を結んだ線の距離のうち最長の距離
- (3) $f_{avg}(s, t) = \frac{1}{n_s n_t} \sum_{a,b} |P_{s,a} - P_{t,b}|$:
ストローク s 上の点とストローク t 上の点を結んだ線の距離の平均
- (4) $g_{minXC}(s, t) = \min_a |P_{s,a} - P_t \cdot|$:
ストローク s 上の点からストローク t の重心までの距離のうち最短の距離
- (5) $g_{minCX}(s, t) = \min_a |P_s \cdot - P_{t,b}|$:
ストローク t 上の点からストローク s の重心までの距離のうち最短の距離
- (6) $f_{minXC}(s, t) = \min(g_{minXC}(s, t), g_{minCX}(s, t))$: (4) と (5) のうち短い方
- (7) $f_{maxXC}(s, t) = \max(g_{minXC}(s, t), g_{minCX}(s, t))$: (4) と (5) のうち長い方
- (8) $g_{minXL}(s, t) = \min_a (\|P_{s,a} - P_{t,1}\|, \|P_{s,a} - P_{t,m_t}\|)$:
ストローク s 上の任意の点からストローク t の始点、終点までの距離の短い方を集めた距離のうち、最短の距離
- (9) $g_{minLX}(s, t) = \min_b (\|P_{s,1} - P_{t,b}\|, \|P_{s,n_s} - P_{t,b}\|)$:
ストローク t 上の任意の点からストローク s の始点、終点までの距離の短い方を集めた距離のうち、最短の距離
- (10) $f_{minXL}(s, t) = \min(g_{minXL}(s, t), g_{minLX}(s, t))$: (8) と (9) のうち短い方
- (11) $f_{maxXL}(s, t) = \max(g_{minXL}(s, t), g_{minLX}(s, t))$: (8) と (9) のうち長い方
- (12) $f_{left}(s, t) = |x_s^m - x_t^m|$:
ストローク s の最小の x 座標とストローク t の最小の x 座標との差の絶対値
- (13) $f_{top}(s, t) = |y_s^m - y_t^m|$:
ストローク s の最小の y 座標とストローク t の最小の y 座標との差の絶対値
- (14) $f_{right}(s, t) = |x_s^M - x_t^M|$:
ストローク s の最大の x 座標とストローク t の最大の x 座標との差の絶対値
- (15) $f_{bottom}(s, t) = |y_s^M - y_t^M|$:
ストローク s の最大の y 座標とストローク t の最大の y 座標との差の絶対値
- (15) $f_{bovlpX} = \max(\min(x_s^M, x_t^M) - \max(x_s^m, x_t^m), 0)$:
ストローク s とストローク t が x 軸方向に重なっている分の幅 (重なっていなければ 0)
- (16) $f_{bovlpY} = \max(\min(y_s^M, y_t^M) - \max(y_s^m, y_t^m), 0)$:
ストローク s とストローク t が y 軸方向に重なっている分の高さ (重なっていなければ 0)
- (17) $f_{novlpX}(s, t) = \frac{f_{bovlpX}(s, t)}{2} \left(\frac{1}{w_s} + \frac{1}{w_t} \right) - \frac{|(x_s^m + x_s^M) - (x_t^m + x_t^M)|}{2 \cdot w_{st}}$: (15) の値を標準化

$$(18) f_{novlpY}(s, t) = \frac{f_{novlpY}(s, t)}{2} \left(\frac{1}{h_s} + \frac{1}{h_t} \right) - \frac{|(y_s^m + y_s^M) - (y_t^m + y_t^M)|}{2 \cdot h_{st}}: (16) \text{の値を標準化}$$

(17),(18) は負の値を取りうるので (4.1) 式の標準化の代わりに,

$$d_i(s, t) = \frac{1}{1 + \exp(-f_i(s, t))} \quad (4.9)$$

を適用する.

$$(19) d_{wratio} = f_{wratio}(s, t) = \frac{\max(w_s, w_t)}{w_{st}}:$$

ストローク s とストローク t 全体の幅に対して, 幅の大きい方が占める割合

$$(20) d_{hratio} = f_{hratio}(s, t) = \frac{\max(h_s, h_t)}{h_{st}}:$$

ストローク s とストローク t 全体の高さに対して, 高さの大きい方が占める割合

$$(21) f_{timeb} = \max(u_{s,1} - u_{t,n_t}, u_{t,1} - u_{s,n_s}):$$

前のストロークを書き終えてから次のストロークを書き始めるまでの時間的距離

$$(22) f_{velocity1}(s, t) = \left| \frac{\|P_{s,n_s} - P_{s,1}\|}{u_{s,n_s} - u_{s,1}} - \frac{\|P_{t,n_t} - P_{t,1}\|}{u_{t,n_t} - u_{t,1}} \right|:$$

ストロークを描く速度の差 (始点と終点との距離/描くのにかかった時間)

$$(23) f_{velocity2}(s, t) = \left| \frac{l_s}{u_{s,n_s} - u_{s,1}} - \frac{l_t}{u_{t,n_t} - u_{t,1}} \right|:$$

ストロークを描く速度の差 (ストロークの長さ/描くのにかかった時間)

これらの比較尺度を特徴ごとにいくつかのグループに分ける. それを表 4.2 に示す (括弧内の番号は上記の箇条書き番号と対応).

表 4.2: 特徴ごとにわけたグループ

グループ名	特徴
F1	$f_{min}(1), f_{max}(2), f_{avg}(3)$
F2	$f_{minXC}(6), f_{maxXC}(7)$
F3	$f_{minXL}(10), f_{maxXL}(11)$
F5	$f_{left}(12), f_{top}(13), f_{right}(14), f_{bottom}(15)$
F8	$f_{novlpX}(17), f_{novlpY}(18)$
F10	$f_{wratio}(19), f_{hratio}(20)$
F11	$f_{timeb}(21)$
F13	$f_{velocity1}(22), f_{velocity2}(23)$

本研究では3つの距離の定義 (クラスタリングパターン) を作成したが, その3つの組み合わせは以下の通りである. 比較尺度それぞれに重みをかけて足し合わせることで, 3通りそれぞれが異なるクラスタリングをするようにした.

$$\text{Object } f_{min}(i, j) + f_{max}(i, j) + f_{avg}(i, j) \cdots \text{F1}$$

$$+ f_{timeb}(i, j) \cdots \text{F11}$$

$$+ f_{velocity1}(i, j) + f_{velocity2}(i, j) \cdots \text{F13}$$

Horizontal $0.1 * (f_{minXC}(i, j) + f_{maxXC}(i, j)) \cdots F2$
 $+0.1 * (f_{minXL}(i, j) + f_{maxXL}(i, j)) \cdots F3$
 $+0.1 * (f_{left}(i, j) + f_{right}(i, j) + f_{top}(i, j) + f_{bottom}(i, j)) \cdots F5$
 $+f_{novlpX}(i, j) + f_{novlpY}(i, j) \cdots F8$
 $+0.1 * f_{wratio}(i, j) + f_{hratio}(i, j) \cdots F10$
 $+f_{imeb}(i, j) \cdots F11$

Vertical $0.1 * (f_{minXC}(i, j) + f_{maxXC}(i, j)) \cdots F2$
 $+0.1 * (f_{minXL}(i, j) + f_{maxXL}(i, j)) \cdots F3$
 $+0.1 * (f_{left}(i, j) + f_{right}(i, j) + f_{top}(i, j) + f_{bottom}(i, j)) \cdots F5$
 $+f_{novlpX}(i, j) + f_{novlpY}(i, j) \cdots F8$
 $+f_{wratio}(i, j) + 0.1 * f_{hratio}(i, j) \cdots F10$
 $+f_{imeb}(i, j) \cdots F11$

以上の Object, Horizontal パターンの組み合わせは, Lee らの提案手法により選択されたものであり, Vertical は Horizontal において縦方向にクラスタリングされるように重みを変えたものである. 重みは実際のシートを用いて実験を行い, それぞれのパターンが目的のクラスタリングをするように調整をし, 求めた. Horizontal から Vertical へは, $f_{wratio}(i, j)$ に 0.1 の重みをかけた場合に横方向にクラスタリングされたことを考慮し, 縦方向にクラスタリングされるようにするに今度は $f_{hratio}(i, j)$ に 0.1 の重みをかけることで応用した.

4.6 GUI

4.6.1 クラスタリングパターン

クラスタリングパターンを選択するために3つの「Object」「Horizontal」「Vertical」ボタンを用意し, 押されたボタンに対応した番号をクラスタリングを行う関数に渡すことで距離の定義を変更している.

4.6.2 ブロック数の変更

ブロックの数を変更するために, 「+」ボタンと「-」ボタンを用意した. 「+」ボタンはシート内に含まれるストローク数を上限としてブロック数を増やし, 「-」ボタンはブロック数が1つになるまでブロック数を減らすように実装した.

4.6.3 ブロックの移動

ブロックの移動には, 「>」ボタンと「<」ボタンを用意した. 「>」ボタンは次のインデックスのブロックに移動, 「<」ボタンは前のインデックスのブロックに移動するように実装した. 尚, リング構造のように「>」か「<」の一方を押し続けると再び同じ場所に戻ってくるようにしている.

第5章 評価実験

実装したクラスタリング機能について評価実験を行った。

5.1 評価方法

手書き Wiki のスケッチ内容に対し、提案手法を適用し、3つのクラスタリングパターンがそれぞれ異なるクラスタリングを行えているか評価を行った。また、クラスタリングアルゴリズムの実行時間を計測し、ストローク内に含まれる点の数との関係性についても評価を行った。

5.1.1 使用端末

コンバーチブル型タブレット PC(ThinkPad X220), MacBook Pro を使用した。

5.1.2 実験の内容

5.2 クラスタリング結果

同じシートに対し、左から Object, Horizontal, Vertical パターンを適用した図を並べた。

5.2.1 シート 1

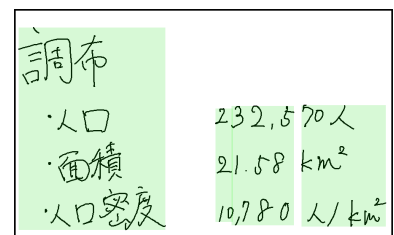
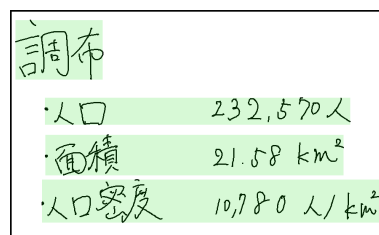
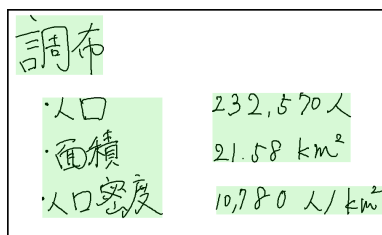


図 5.1: Object1

図 5.2: Horizontal1

図 5.3: Vertical1

3つの結果を比較してみると、3通りとも異なるグループに分かれていることがわかる。Object では実際のストロークの距離に近いもの同士がグループになり、Horizontal では横方向で同じラインにあるものがグループになっている。また、同様に Vertical では同じ縦のラインにあるものがグループになっている。しかし、Vertical では数字が途中で切れてしまい、ユーザーが数字をつかもうとしても一部しか移動させることができない。

5.2.2 シート2

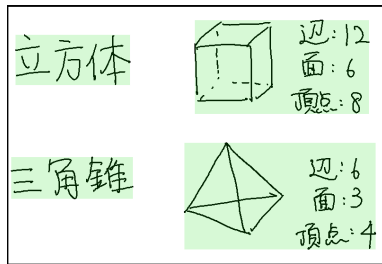


図 5.4: Object2

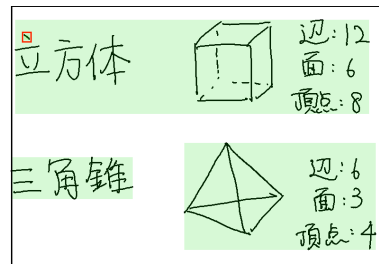


図 5.5: Horizontal2

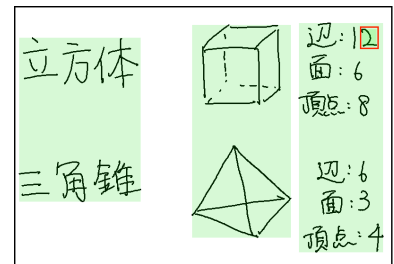


図 5.6: Vertical2

まず、3つのパターンで異なる分類がされており、Object は距離が近いまとまりで分類されていることがわかる。一方 Horizontal と Vertical では横縦にそれぞれ分類されているように見えるが、一部赤枠で囲った箇所がただしく分類されていなかった。

5.2.3 シート3

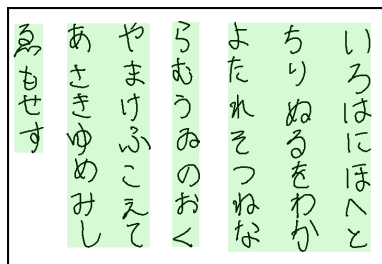


図 5.7: Object3

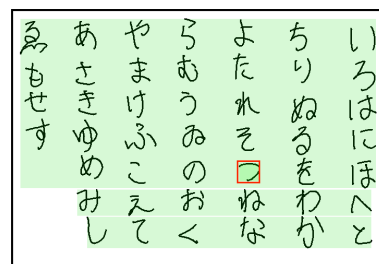


図 5.8: Horizontal3

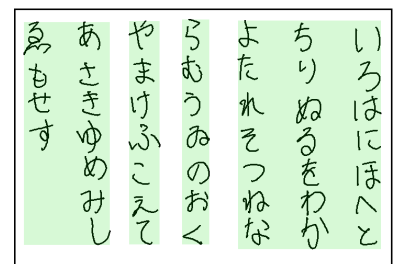


図 5.9: Vertical3

シート3 に関して3種類とも異なるグループに分割されている。しかし Object と Vertical では分類されている箇所は違うものの、分割数をふやしていくと同じような結果になってしまった。また、Horizontal では赤枠で囲った一部が別に分類されてしまっている。これは分割数を減らすことで上と同じグループに入れることができた。

5.2.4 考察

以上の結果から、図などのまとまり、縦方向、横方向といったクラスタリングはできているが、不自然な箇所で切れてしまったりしている。やはり、これは学習アルゴリズムを使わず、重みに正確な値を与えていないことが原因だと考えられる。しかし、それぞれのシート内容(図、横書き、縦書き)に適したクラスタリングパターンを用意しているのでもし思ったようにクラスタリングされなければ違うパターンを適用することでそれを解決することができると思う。

5.3 分割数の増減

それぞれのクラスタリングパターンに対し、シート1における分割数を変更しどのように分割されるのかを実験した。

図 5.10, 図 5.11, 図 5.11 は分割数を 10 にしたときのクラスタリング結果である。

調布	
・人口	232,570人
・面積	21.58 km ²
・人口密度	10,780 人/km ²

図 5.10: Object(分割数 10)

調布	
・人口	232,570人
・面積	21.58 km ²
・人口密度	10,780 人/km ²

図 5.11: Horizontal(分割数 10)

調布	
・人口	232,570人
・面積	21.58 km ²
・人口密度	10,780 人/km ²

図 5.12: Vertical(分割数 10)

図 5.12, 図 5.13, 図 5.14 は分割数を 2 にしたときのクラスタリング結果である。

調布	
・人口	232,570人
・面積	21.58 km ²
・人口密度	10,780 人/km ²

図 5.13: Object(分割数 2)

調布	
・人口	232,570人
・面積	21.58 km ²
・人口密度	10,780 人/km ²

図 5.14: Horizontal(分割数 2)

調布	
・人口	232,570人
・面積	21.58 km ²
・人口密度	10,780 人/km ²

図 5.15: Vertical(分割数 2)

5.3.1 考察

分割数を変更した場合でも、それぞれの特徴が結果に現れていることがわかった。しかし、分割数を増やしている最中に意図しない箇所で分割されてしまうことがあった。また、分割数が少ない場合には図 5.13, 図 5.15 の様に同じ結果になってしまうこともあるが、少なくとも 3 通り全てが同じわけではないので、目的に応じて使い分けることができると考える。

5.4 実行時間

それぞれのクラスタリングパターンについて、シート内に含まれるストローク数(ストロークを構成する点)を変更して実行時間を計測した。1回のシート内容に対し5回計測しその平均をとってグラフにした。横軸がストロークに含まれる点の総数であり、縦軸が実行時間である。

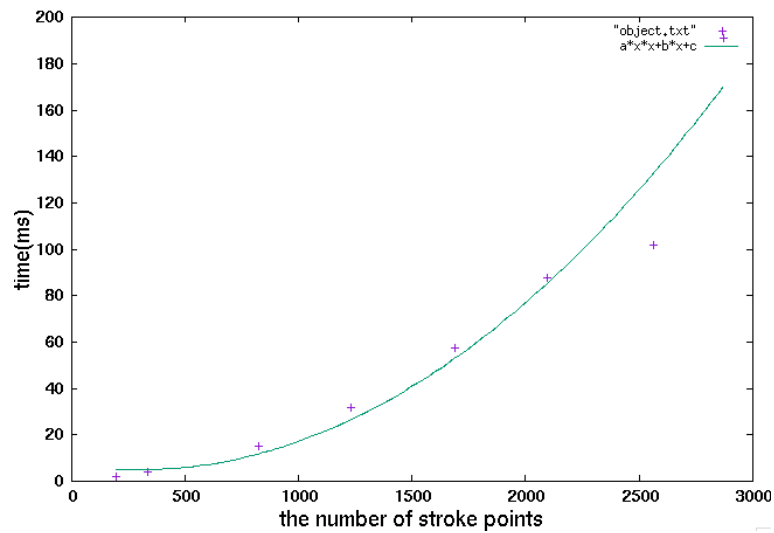


図 5.16: Object パターンの実行時間

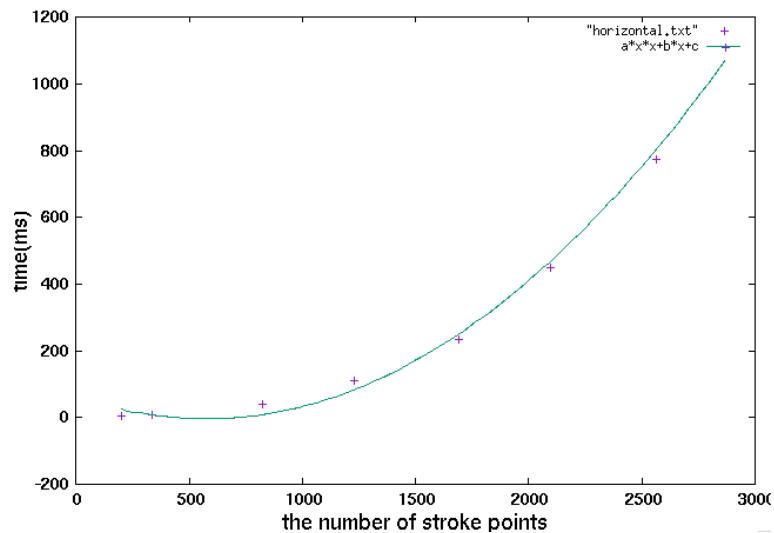


図 5.17: Horizontal パターンの実行時間

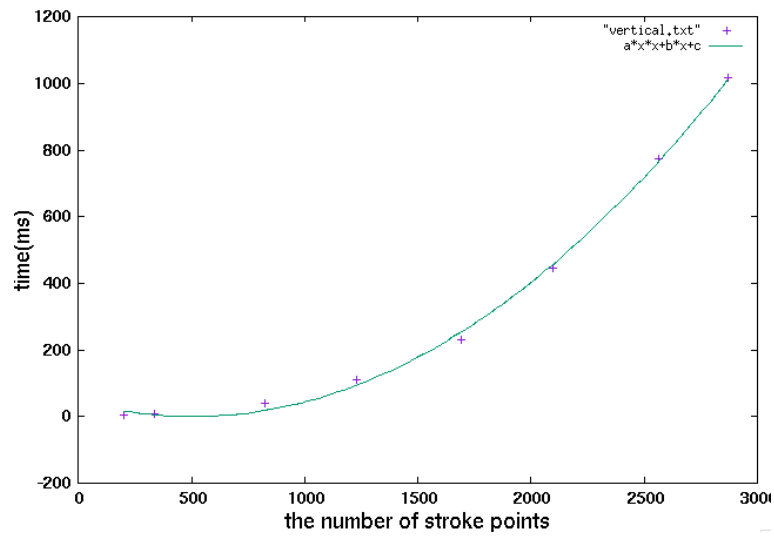


図 5.18: Vertical パターンの実行時間

3つのグラフから、いずれのパターンの場合でもストロークに含まれる点の数を増やすと $O(n^2)$ のオーダーで実行時間が増加している事がわかった。

5.4.1 考察

ストロークに含まれる点、すなわちストローク数が増えると実行時間が増えてしまうので、シート内に描かれた内容が多いとその分クラスタリング結果を表示するまでにいくらか時間がかかってしまい、ユーザーの操作を妨げてしまう。

第6章 結論

6.1 結論

本研究では、シート内容の整理、編集のしやすさを向上させるため、手書き Wiki へ自動クラスタリング機能を実装した。3つの異なるクラスタリングパターンを用意することで、意図に応じたクラスタリングを適用することができ、Group 機能の向上につながった。しかし、クラスタリング結果については学習アルゴリズムを適用するなど改善の余地はある。

6.2 今後の課題

6.2.1 機能の改善

手書き Wiki にはストロークのほかに文字列、画像、他シートへのリンクなど多様なオブジェクトが含まれているが、本研究ではそれらはクラスタリングの対象となっていないので拡張の必要がある。例えば、リンクがグループの枠内に含まれていても移動することができない。この問題点の改善方法として、リンクがグループと重なっていれば、リンクをグループの一部となるようにする方法が考えられる。

6.2.2 機能の追加

手書き Wiki にはリンクシステムというものがあり、これは現在開いているシートの一部を、グループ機能と同様に四角で囲うことでリンクを作成し、新規のシートを作成することができるというものである。本研究では手書き内容をクラスタリングし編集するのみであったが、グループを選択した状態で「Link」ボタンを押し、そのグループに対しリンクをつくるといった機能を追加することで、手書き Wiki のリンクシステムを活かすことができると考える。

謝辞

本研究は、電気通信大学情報理工学部情報・通信工学科コンピュータサイエンスコース寺田研究室において、寺田実准教授の指導の下で卒業研究として行われました。寺田実准教授には、本研究を進めるにあたり適切な御助言とご指導を頂きました。心より御礼を申し上げます。

また、寺田研究室の修士課程2年の阿部真之さん、鈴木佑樹さん、平田吉久さん、本田裕人さん、修士課程1年の安部文紀さん、山本愛美さん、渡邊裕貴さん、学部4年の岡川翔子さん、佐々木透さん、肥後亮佑さん、藤本明優さん、村松啓寛さんには研究についての助言や協力、また研究室生活に関わる様々なことでお世話になりました。心からの感謝を申し上げます。

参考文献

- [1] Adrien Delayen, Kibok Lee, “Aflexible framework for online document segmentation by pairwise stroke distance learning”, Pattern Recognition 48 1197-1210.(2015).
- [2] Florian Perteneder, Martin Bresler, Eva-Maria Grossauer, Joanne Leong, Michael Haller, “cLuster: Smart Clustering of Free-Hand Sketches on Large Interactive Surfaces”, UIST '15
- [3] Creating a GUI With JFC/Swing : <http://docs.oracle.com/javase/tutorial/uiswing/>. (最終閲覧日:2017/1/28)
- [4] 朱鷺の杜 Wiki <http://ibisforest.org/index.php>. (最終閲覧日:2017/1/28)
- [5] 神鷲敏弘 HP 「クラスタリング(クラスター分析)」
<http://www.kamishima.net/jp/clustering/>. (最終閲覧日:2017/1/28)