

特別研究報告書

ペンストロークの時間間隔を用いた
答案の解答停滞箇所の検出

指導教員 飯山 将晃 准教授

京都大学工学部情報学科

中塚 智尋

平成 28 年 1 月 29 日

ペンストロークの時間間隔を用いた答案の解答停滞箇所の検出

中塚 智尋

内容梗概

情報通信技術 (Information and Communications Technology; ICT) の発展に伴い, 学校現場での ICT 活用基盤の構築が進められている. その中で, 生徒 1 人 1 台のタブレット PC の普及を目指す動きがある. タブレット PC はその優れた携帯性から, 普通教室での授業や家庭学習など幅広い場面での利用が可能であり, さらにその間の生徒の学習ログデータを蓄積することができる. これを活用することで, 個々の興味・関心や学習の習熟度, 学習ペースなどを把握し, 個々のレベルや特性に合わせた指導を行う「テイラーメイド教育」の実現が期待されている.

一般に, 生徒の学習指導において, 答案の解答過程を分析することが重要視されている. その中で習熟度が低い問題に対してさらに詳細に, 解答中のどの箇所について習熟度が低いのか把握したいという要求がある. 解答中の習熟度が低い箇所を把握することで, 教師が問題中でも特に重点をおいて解説すべき箇所を把握でき, 指導方針の決定に有用である. ここで習熟度が低い箇所とは, 「誤りが生じる箇所」あるいは「時間がかかる箇所」のどちらかであると考えられるが, 従来の紙と鉛筆による学習では前者の分析のみ行われてきた. 一方で, タブレット PC を用いた学習では前者に加えて後者の分析も行える. タブレット PC は画面に指やタッチペンで触れることにより, 紙に鉛筆で記述する形式と似た入力を行うことが容易であり, さらに解答過程を記述する際の時間や筆跡といった経過の様子を学習ログデータとして収集することができるためである. 「時間がかかる箇所」の分析によって, 今まで看過されがちであった正解は導いたものの手を焼いた箇所, 自信がない箇所を発見できる. 以降では, 「時間がかかる箇所」を「解答停滞箇所」と呼ぶ.

生徒が解答過程を記述する様子を表す学習ログデータとして, ペンストロークデータがある. 「ペンストローク」とは記述における一筆のことであり, ペンダウンの時刻, ペン先が通った軌跡を表す点の座標, ペンアップまでの時間などのデータとして記録される. ある問題解答中のペンストロークを時系列に並べたものを「ペンストロークデータ」という. ペンストロークデータ中のあるペンストロークのペンダウン時刻とあるペンストロークのペンダウン時刻の間

の経過時間を「ペンストロークの時間間隔」という。本論文では解答過程の記述が特に求められる科目である数学の問題を対象に、ペンストロークの時間間隔を用いて解答過程の中から停滞箇所を検出する手法を提案する。

解答過程中の各ステップにおいてどの程度以上の時間を停滞とするのかの基準を得るため、同一問題に対する複数の答案間でペンストロークの時間間隔の比較を行う。複数の他者と比較することで、各ステップで平均的に要する時間を基準にすることができる。この比較は、答案間における解答過程内の同一ステップにあたる箇所（同等箇所と呼ぶ）ごとに行われるべきであり、答案間の同等箇所がペンストロークデータ上で対応している必要がある。しかし、この対応付けは、筆跡や数式の表現方法の違いが問題になり、ペンストロークデータから直接行うことはできない。筆跡の違いとは、筆順や字体の個人差やぶれを指し、表現方法の違いとは、例えば数式における積算や和算の項の順番や、変数を表す際に用いた文字の違いなどである。

そこで本研究では、筆跡の違いをペンストロークデータから文字抽出を行うことで対処する。文字を抽出することで個人差やぶれを排除し、ペンストロークデータが表す内容だけを取り出せるため、解答過程内の同一ステップにあたる箇所を対応付けるためには扱いやすい。数式から文字抽出を行うが、数式は一般的な文章に比べ一行中の文字配置が複雑であり、複数行にわたる答案全体から一度にすべての文字を抽出することは難しい。そこで事前に、ストロークの時空間的な情報に基づき、答案を一行ごとに分割する。この分割は答案 211 件中 143 件で成功した。

また、表現方法の違いには DP マッチングを用いて対処する。DP マッチングは途中で違いを含むことを許して文字列間の最適な対応付けを求める手法である。式の途中に含まれる違いに対応しながら全体の対応付けをとることができる。8 割以上の文字を対応づけることができた二組の答案間については、適合率、再現率が 94.6%、89.7%と 76.6%、73.5%という精度でマッチングを行えた。

求めたペンストロークの対応付けを用いて時間間隔の比較を行い、停滞していた箇所を検出する。相対的に平均 2 秒以上の遅れが発生した箇所を停滞とした実験では、93 個の答案から 298 個の解答停滞箇所を検出し、そのうち 81.2%は近傍に 3 秒以上の記述の停止が存在したことを確認した。また、検出結果について、教師への解答停滞箇所の効果的な提示のため、その可視化も行った。

Detecting answer stagnation point using time intervals of pen strokes

Chihiro NAKATSUKA

Abstract

Recent Information and Communications Technology (ICT) promotes the use of computer systems in education. For example, there are some attempts to provide every student his/her own tablet PC. Tablet PC's excellent portability enables us to use PCs in various situations such as in classes or home learning, and accumulates learning log data. It is expected that this data provides teachers information about individual student's interests, learning level and learning pace, and is expected that teachers provide students with the guidance that matches the individuals, so-called "tailor-made education."

It is important to analyze processes of answer in education, and teachers require to know weak points in the questions for making a teaching plan. The weak points are categorized into "time-consuming points" and "error points." In conventional learning on paper the teachers can analyze only the former weak points. On the other hand, learning on tablet PCs will enable us to analyze the latter weak points, because tablet PCs provide an easy input method like handwriting and it collects learning log data about answer process such as handwriting timing. Analysis of the "time-consuming point" leads to find ever discovered points where students struggle to answer. In the following, we call "time-consuming point" as "answer stagnation point."

Pen stroke data is one of the learning log data and records students' answer process. This data consists of the time to pen-down, the coordinates of the points representing the orbit and the time to pen-up of all pen stroke in a time series. "Time interval of the pen strokes" is elapsed time between the pen-down time of two pen strokes in one data. We propose an approach for detecting a stagnation point in answer process by using time intervals of the pen strokes. We focus on math problems, which especially require learners to describe answer process.

We compare the time interval of pen strokes across multiple answers of the same question and determine how long time we treat as stagnation point in

the answer process. This comparison should be performed for each location of the same answer steps between answers (referred to as equivalent positions). It is necessary that equivalent positions among answers are corresponded on the pen stroke data. However, this corresponding is difficult for the difference in the representation of handwriting and formulas, and it can not be performed directly on the pen stroke data. The difference in the representation of handwriting refers to distortion and the personal habit on character style or stroke order, and the difference in the representation of formulas is, for example, the order of terms in integration and summation or the differences of characters used in representing the variables.

In this study, we solve the problem about the differences in handwriting by performing text extraction from the pen stroke data. Personal characteristic in text writing and distortion in characters are ignored by applying text extracting, so we can make correspondences the equivalent positions. When we perform character extraction from formulas, characters in formula are more complicated in a row compared to the general text, therefore it is difficult to extract all of the characters at once from the whole answer. Then, we divide a pen stroke data in each row depending on temporal and spatial information in pen stroke data. This division was succeeded in 143 among 211 answers.

Moreover, we solve the problem about the differences in formulas by using DP matching. DP matching is a method of obtaining an optimum correspondence between the strings, which allows small differences. Two of the answer sets where over 80% characters show 94.6 %, 89.7 % and 76.6 %, 73.5% adaptation rate and reproduction rate for matching.

Finally, we detected stagnation points by comparing the time interval with the correspondence of the pen strokes. In the experiments where we regard over 2 seconds delay as stagnation points, our approach detect 298 stagnation points from 93 answers, and 81.2 % of them have 3 seconds or more stop time in the vicinity. Moreover, we visualized stagnation points for effective presentation to teachers.

ペンストロークの時間間隔を用いた答案の解答停滞箇所の検出

目次

第1章	序論	1
第2章	学習ログとしてのペンストロークデータ	3
2.1	学習ログデータ	3
2.2	e-learningにおける学習ログデータ	3
2.3	入力ログデータ	4
第3章	ペンストロークの時間間隔を用いた答案の解答停滞箇所検出手法	4
3.1	定式化	4
3.2	ペンストロークデータの対応付け	6
3.3	ペンストロークデータからの文字抽出	8
3.4	答案間の同等箇所の対応付けと解答停滞箇所の検出	9
3.5	解答停滞箇所の提示	11
第4章	実験	13
4.1	ペンストロークデータの収集	13
4.2	ペンストロークデータからの文字抽出結果	14
4.3	答案間の同等箇所の対応付け結果	16
4.4	解答停滞箇所の検出結果	21
4.5	考察	22
第5章	結論	23
	謝辞	23
	参考文献	24

第1章 序論

情報通信技術 (Information and Communications Technology; ICT) の発展に伴い、学校現場での ICT 活用基盤の構築が進められている。文部科学省は 2011 年 4 月に、21 世紀にふさわしい学校教育の実現を目指す推進方策として「教育の情報化ビジョン [1]」を取りまとめ、これに基づいて、総務省の「フューチャースクール推進事業 [2]」と連携の下、「学びのイノベーション事業 [3]」を実施した。これらの事業では全国の小中学校及び特別支援学校から 20 校を対象として、生徒 1 人 1 台のタブレット PC、全ての普通教室への電子黒板、無線 LAN などの環境を導入し、ICT を活用した教育モデルの構築に向けた実証研究が行われた。この成果をまとめ、教育現場での ICT の普及展開を推進すると共に、今後取り組むべき課題を提示している。

課題の一つに、生徒 1 人 1 台のタブレット PC の活用方法の充実がある。タブレット PC はその優れた携帯性から、普通教室での授業や家庭学習など幅広い場面での利用が可能であり、その間の生徒の学習ログデータを蓄積することができる。これを上手く活用することで、個々の興味・関心や学習の習熟度、学習ペースなどを把握し、個々のレベルや特性に合わせた指導を行う「テイラーメイド教育」の実現が期待されている。これを受けて、学習ログデータを利用する研究が盛んに行われており、例えば e ラーニングに代表される教育システムの学習ログデータから生徒の主観的な難易度を予測する研究 [4][5] などがある。

一般に、生徒の学習指導において、答案の解答過程を分析することが重要視されている。習熟度が低い問題に対してさらに詳細に、解答中のどの箇所について習熟度が低いのか把握したいという要求がある。解答中の習熟度が低い箇所を把握することで、教師が問題中에서도特に重点をおいて解説すべき箇所を把握でき、指導方針の決定に有用である。習熟度が低い箇所とは、「誤りが生じる箇所」あるいは「時間がかかる箇所」のどちらかであると考えられるが、従来の紙と鉛筆による学習では前者の分析のみ行われてきた。一方で、タブレット PC を用いた学習では前者に加えて後者の分析も行える。タブレット PC は画面に指やタッチペンで触れることにより、紙に鉛筆で記述する形式と似た入力を行うことが容易であり、さらに解答過程を記述する際の時間や筆跡といった経過の様子を学習ログデータとして収集することができるためである。「時間がかかる箇所」の分析によって、今まで看過されがちであった正解は導いたものの

手を焼いた箇所，自信がない箇所を発見できる．以降では「時間がかかる箇所」を「解答停滞箇所」と呼ぶ．

生徒が解答過程を記述する様子を表す学習ログデータとして，ペンストロークデータがある！「ペンストローク」とは記述における一筆のことであり，ペンダウンの時刻，ペン先が通った軌跡を表す点の座標，ペンアップまでの時間などのデータとして記録される．ある問題解答中のペンストロークを時系列に並べたものを「ペンストロークデータ」という．ペンストロークデータ中のあるペンストロークのペンダウン時刻とあるペンストロークのペンダウン時刻の間の経過時間を「ペンストロークの時間間隔」という．本稿では解答過程の記述が特に求められる科目である数学の問題を対象に，ペンストロークの時間間隔を用いて解答過程の中から停滞箇所を検出する手法を提案する．

解答過程中の各ステップにおいてどの程度以上の時間を停滞とするのか，基準を得るために同一問題に対する複数の答案間でペンストロークの時間間隔の比較を行う．複数の他者と比較することで，各ステップに通常要する時間を基準にすることができる．この比較は，答案間における解答過程内の同一ステップにあたる箇所（同等箇所と呼ぶ）ごとに行われるべきであり，答案間の同等箇所がペンストロークデータ上で対応付いている必要がある．しかし，この対応付けは，筆跡や数式の表現方法の違いが問題になり，ペンストロークデータから直接行うことはできない．筆跡の違いとは，筆順や字体の個人差やぶれを指し，表現方法の違いとは，例えば数式における積算や和算の項の順番や，変数を表す際に用いた文字の違いなどである．

そこで本研究では，筆跡の違いには，ペンストロークデータから文字抽出を行うことで対処する．文字を抽出することで個人差やぶれを排除し，ペンストロークデータが表す内容だけを取り出せるため，解答過程内の同一ステップにあたる箇所を対応付けるためには扱いやすい．各文字と元のペンストロークの対応関係を保存しておけば，文字列の対応付けをペンストロークデータ上の対応付けに反映することは容易にできる．また，表現方法の違いにはDPマッチングを用いて対処する．DPマッチングは途中に違いを含むことを許して文字列間の最適な対応付けを求める手法である．式の途中に含まれる違いに対応しながら全体の対応付けをとることができる．今回の場合は，例えば変数を表す際に用いた文字の違いから生じる文字列上の違いは意味的に同じものであるため，この違いによる影響を小さいものにするような重み付けを与える．これら

を利用してペンストロック対応付けを求め、時間間隔の比較を行い、大きく停滞していた箇所を検出する。さらに、教師への解答停滞箇所の効果的な提示のため、可視化する。

以降、第2章で関連研究に触れてから本研究の問題設定を定義し、第3章で提案手法を説明する。第4章で提案手法の実験と考察を行い、第5章で結論を述べる。

第2章 学習ログとしてのペンストロックデータ

2.1 学習ログデータ

本研究では学習ログデータとしてのペンストロックデータを研究対象として用いる。「ペンストロック」とは記述における一筆のことであり、ペンダウンの時刻、ペンアップまでの経過時間、ペン先が通った軌跡を表す点の座標などのデータとして記録される。本稿では、生徒に出題される問題に解答過程を記述するための空白の欄が設けられていることを想定し、「ペンストロックデータ」はある生徒がその範囲内に記述したすべてのペンストロックを時系列順に記録したデータとする。紙の答案から得られる情報に加え、ペンストロックデータはその解答の過程も得ることができる。

2.2 e-learning における学習ログデータ

e-learning において得られる学習ログデータを用いた研究は盛んに行われている。e-learning とは Web 上の教材をユーザが自学自習するという形の学習形態である。この学習ログデータの例として、アクセス履歴や解答履歴、確認テストの結果、所要時間が挙げられる。

向後ら [6][7] は、e-learning の確認テストの合格数を生徒ごとに分析することで、ドロップアウトの兆候を発見している。ドロップアウト兆候があればなんらかの介入援助を行うことが、ドロップアウトの減少に効果的であることが示唆されている。植野 [8] は e-learning のコンテンツに対する学習所要時間データからオンラインで学習者の異常学習プロセスを検知する手法を提案した。過去の学習所要時間データの系列から、今回の学習所要時間データが出現する確率を導き、その特異性を検定する。この手法によって検出された異常学習プロセスは、学習者のコンテンツへの不理解の兆候を示していたとしている。また、中

村ら [4][5] は, e-learning で得られる学習ログデータに加えて, 学習者の顔画像を収集し, 観測情報と呼ばれるそれらの情報を用いて学習者の主観的難易度の推定を試みている. 観測情報から特徴抽出を行い, 学習者ごとにあらかじめ機械学習で獲得した識別機を用いて主観的難易度を推定したところ, 有意な識別率での結果を得ている.

これらの研究は, PC を通してキーボード入力やマウス入力を行う解答形態を想定しており, タブレット PC で容易に扱えるようになったペンを用いた自然な入力方法による解答形態については未検討である.

2.3 入力ログデータ

学習ログデータを入力ログデータの種類として着目したとき, これに関連した研究として Epp らによるキーストロークデータを用いた感情推定がある [9]. キーストロークデータとは, キーボードからの入力においてキーを押した時刻, 離れた時刻, 押したキーを特定するための番号を記録したものである. Epp らは, このデータのうち時間情報を用いて, いくつかの感情について高い精度での推定に成功している. 本研究では入力ログデータのうち, ペンstroークデータに着目する.

第3章 ペンstroークの時間間隔を用いた答案の解答停滞箇所検出手法

3.1 定式化

本研究では, ペンstroークデータから得られる生徒の解答過程を表す情報に着目し, 生徒の習熟度が低い箇所を検出することを目的としている. そのために, 解答の停滞が見られた箇所を検出する手法を提案する. ここで, 手法を定式化する.

ある問題に対するある答案 A のペンstroークデータを S_A , その i 番目のペンstroークを s_{iA} とすると,

$$S_A = \{s_{1A}, \dots, s_{N_A A}\}$$

と表すことができる. ただし, N_A はペンstroークデータ S_A に含まれるペンstroークの総数である. ペンstroーク s_{iA} の軌跡を d_{iA} , ペンダウン時刻を

t_{iA} で表す．軌跡は点座標の順序付き集合として，

$$d_{iA} = \{(x_1^{iA}, y_1^{iA}), \dots, (x_{n_{iA}}^{iA}, y_{n_{iA}}^{iA})\}$$

と表すことができる．ただし， n_{iA} は軌跡 d_{iA} を構成する点の総数である．ペンストロークデータとペンストローク，そしてそれを構成する点の関係の一例を図1に示す．また，ペンストローク s_{iA} と s_{jA} の時間間隔 $I(s_{iA}, s_{jA})$ はペンダウ

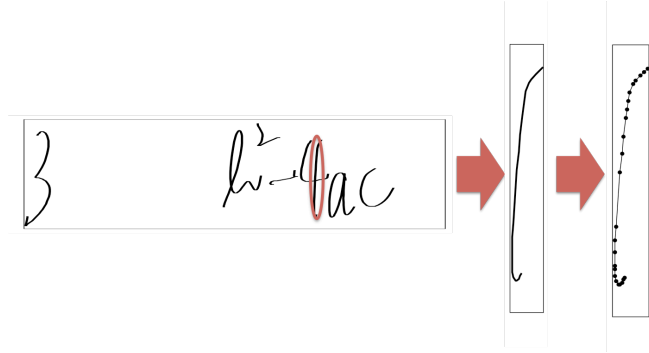


図1: ペンストロークデータ, ペンストローク, 点

ン時刻 t_{iA}, t_{jA} を用いて，

$$I(s_{iA}, s_{jA}) = t_{jA} - t_{iA}$$

であると定義する．

答案 A から解答停滞箇所の検出をするために，同一問題に対する複数の答案と解答過程の同等箇所について時間間隔を比較する．このため，答案 A と比較対象の答案間で同等箇所を対応づける必要がある．比較対象の答案のひとつに答案 B があるとする．答案 A と答案 B の間で対応付いた箇所がいくつか存在したとすると，ペンストロークデータ上では k 番目の対応付けを c_k^{AB} とおき，

$$c_k^{AB} = ((s_{(m_{k-1}+1)A}, s_{m_kA}), (s_{(m'_{k-1}+1)B}, s_{m'_kB}))$$

と表す．これは答案 A の $m_{k-1} + 1$ 番目から m_k 番目までのペンストロークが答案 B の $m'_{k-1} + 1$ 番目から m'_k 番目までのペンストロークと対応付いていることを意味している．ただし， m_k, m'_k は二つの答案の k 番目の対応付けにおける最後尾のペンストロークの番号である．また，答案 A, B のペンストロークデータ上の全対応付け C_{AB} を，

$$C_{AB} = \{c_1^{AB}, \dots, c_{M_{AB}}^{AB}\}$$

と表す．ここで， M_{AB} は答案 A，B 間の対応付けの総数である．ペンストロークデータ S_A, S_B から全対応付け C_{AB} を求める手続きをアルゴリズム M とする．アルゴリズム M の手続きは 3.2，3.3，3.4 節に記す．

時間間隔の比較に用いる答案として， n_B 個の答案 B_1, \dots, B_{n_B} があるとする．アルゴリズム M により，答案 A と答案 B_1, \dots, B_{n_B} 間の対応付け $C_{AB_1}, \dots, C_{AB_{n_B}}$ を求め，それらを用いて時間間隔の比較を行うことで答案 A の解答停滞箇所を検出する．ここで，時間間隔の比較から得られる，遅れの程度を表す指標として *timescore* を導入する．すべての連続する二つのペンストローク間においてこの指標を求める．答案 A のある連続する二つのペンストローク $s_{(i-1)A}$ と s_{iA} の間の *timescore* を特に， $timescore_A(i)$ とする． $timescore_A(i)$ が閾値以上のとき， s_{iA} が解答停滞箇所として検出される．*timescore* を算出し，解答停滞箇所を検出する手続きをアルゴリズム F と定義し，Algorithm1 に示す．

以上の操作により，解答停滞箇所が検出できる．ただし，対応付けを求めるアルゴリズム M において，筆跡や数式の表現方法の違いに対処するための工夫が必要である．

3.2 ペンストロークデータの対応付け

筆跡の違いには，ペンストロークデータから文字抽出を行うことで対処する．文字として扱うことで，ペンストロークデータが表す内容だけを取り出せるため，解答過程内の同一ステップにあたる箇所の対応付けが容易になる．また，各文字と元のペンストロークの対応関係を保存しておけば，文字列の対応付けをペンストロークデータ上の対応付けに反映することは容易にできる．しかし，ペンストロークデータ全体から一度にすべての文字を抽出することは困難である．本研究で扱うペンストロークデータは数学の答案を表すものであるが，数式は一般的な文章と異なり，文字が左から右に整列しているだけでなく，文字が上下に配置されている場合もある．その一方で，数学の答案は式が上から下に行ごとに展開されていく．そのため，文字抽出の前にまずペンストロークデータを一行分ずつに分割する必要がある．

数式の表現方法の違いには，DP マッチングを用いることで対処する．DP マッチングは文字列中の細かな違いを吸収しつつ全体を通して対応付けをとることができる．

Algorithm 1 解答停滞箇所検出アルゴリズム F

Require: 答案 A から解答停滞箇所を求める . 比較対象は答案 B_1, \dots, B_{n_B} .

Ensure: ペンストロークデータ $S_A = \{s_{1A}, \dots, s_{N_AA}\}$

****timescore* の初期化***

for all i such that $2 \leq i \leq N_A$ **do**

$timescore_A(i) \leftarrow 0$

end for

****timescore* の算出***

for $i = 1$ to n_B **do**

for $j = 1$ to M_{AB} **do**

if $I(s_{(m_{j-1}+1)A}, s_{m_jA}) - I(s_{(m'_{j-1}+1)B}, s_{m'_jB}) \geq 0$ **then**

for all k such that $m_{j-1} + 1 \leq k \leq m_j$ **do**

$timescore_A(k) \leftarrow \frac{I(s_{(m_{j-1}+1)A}, s_{m_jA}) - I(s_{(m'_{j-1}+1)B}, s_{m'_jB})}{m_j - m_{j-1}}$

end for

end if

end for

end for

比較した答案数で *timescore* を正規化

for all i such that $2 \leq i \leq N_A$ **do**

$timescore_A(i) \leftarrow \frac{timescore_A(i)}{n_B}$

end for

解答停滞箇所として検出されたペンストロークの集合

return $\{s_{iA} | \forall i \text{ such that } timescore_A(i) \geq \alpha\}$ (α は定数)

3.3 ペンストロークデータからの文字抽出

ペンストロークデータの分割

ペンストロークデータを答案上で一行分にあたる区間に分割する．ペンストロークデータは時系列データであり，ペンストロークが書かれた順に記録されている．前から順にペンストロークを座標について見ていき，改行が起こったと考えられる座標の変化が生じたペンストロークの直前で分割することを繰り返す．

改行には図2の2種類がある．図中の①では改行前後で x 座標の値が改行前の行の幅だけ減少し， y 座標の値が文字の縦幅だけ増加する．図中の②では改行前後で x 座標の値が段落間の幅だけ増加し， y 座標の値が何行かの高さの分だけ減少する．つまり改行時には，毎回多少の違いはあるが， x 座標 y 座標共に同じ行を書いているうちには見られないような，一定以上の変化が生じることがわかる．この変化量は答案中の文字の大きさに依存している．文字の大きさとして，ペンストロークデータから各ペンストロークの x 座標幅， y 座標幅を用いて中央値を算出する．ペンストロークには点やマーカなどの線，または誤入力といった文字以外を表すものも存在するが，それらに影響されず文字の大きさを求めるためである．

答案 A のある行が， s_{iA} から始まり s_{jA} までを含むことがわかっているとすると，このとき， $s_{(j+1)A}$ もこの一行に含まれているか，あるいは改行を挟んでいるのかどうかを以下の条件で判定する．

$$\begin{aligned} X_{A,max} &= \max(\max(x^{iA}), \dots, \max(x^{jA})) \\ Y_{A,min} &= \min(\min(y^{iA}), \dots, \min(y^{jA})) \\ x_{A,min} &= \min(x(j+1)) \\ y_{A,min} &= \min(y(j+1)) \\ width_i &= \max(x^{iA}) - \min(x^{iA}) \\ height_i &= \max(y^{iA}) - \min(y^{iA}) \\ fontsize &= \text{median}(width_i, height_i, \dots, width_j, height_j) \end{aligned}$$

とおく．

$$|X_{max} - x_{min}| \quad \text{and} \quad |Y_{min} - y_{min}| > fontsize$$

これを満たすとき， s_{jA} と $s_{(j+1)A}$ の間には改行があると判定する．

$$(M+3)(M-5) - 9$$

$$M^2 - 2M - 15 - 9$$

$$M^2 - 2M - 24$$

$$(M-6)(M+4)$$

図2: 改行

分割されたペンストロークデータからの文字抽出

分割されたペンストロークデータから文字抽出を行う．文字抽出には Alvaro らが提案した手書き数式認識手法”SESHAT[10][11]”を用いる．数式の認識には大きくわけて，記号の認識と数式の構造解析の二つの行程が必要になる．数式の構造は使われている記号に依存するので，”SESHAT”では記号の認識結果と数式の構造解析結果を組み合わせたとき最適となる結果を選択して出力している．ペンストロークデータを入力に与えると，その認識結果の文字列と各文字を構成しているペンストロークの座標情報が得られる．分割されたペンストロークデータから得られた文字抽出結果は連結させて，ひとつの文字列にする．

3.4 答案間の同等箇所に対応付けと解答停滞箇所の検出

比較するふたつの答案から得た文字列を，DP マッチングにより対応付ける．本研究では一般的な DP マッチング手法 [12] に，数式のマッチングに適した重み付けを加えて用いる．

	a	b	4	c
b	5	7	19	26
4	17	16	8	20
c	24	23	20	9

図3: ペナルティ表

与えられた二つの文字列 $text1$, $text2$ から図3のようにペナルティ表を作成

する．ペナルティ表の最下段のうち最小値のマス（最小値が複数ある場合，その最右マス）から，左，左上，上のうち最小値を辿って最上段の最左マスに至る経路を得る．経路が (i, j) のマスを通り $text1(i) = text2(j)$ であるとき，この両文字は対応付いていると考える．この文字間の対応関係から答案間の対応付け C_{AB} がわかる．ペナルティは左から右，上から下の順に決定され，文字列 $text$ の i 番目の文字を $text(i)$ と表すと， i 行 j 列のペナルティ $P(i, j)$ は，

$$addP(i, j) = \begin{cases} 0 & text1(i) = text2(j) \\ 5 & text1(i) \neq text2(j) \text{ (共に文字)} \\ 10 & text1(i) \neq text2(j) \text{ (それ以外)} \end{cases} \quad (1)$$

$$\begin{aligned} l(i, j) &= P(i-1, j) + 2 \\ d(i, j) &= P(i-1, j-1) + 1 \\ u(i, j) &= P(i, j-1) + 2 \end{aligned} \quad (2)$$

とおいたとき，

$$P(i, j) = \begin{cases} \min\{l(i, j), d(i, j), u(i, j)\} + addP(i, j) & i, j > 1 \\ addP(i, j) & i, j = 1 \\ u(i, j) + addP(i, j) & i = 1, j > 1 \\ l(i, j) + addP(i, j) & i > 1, j = 1 \end{cases} \quad (3)$$

に従って算出される．

式1では，ふたつの文字が異なっていた場合のペナルティを設定している．ふたつの文字が 'x' や 'a' など，共に数字ではなかった場合，両者は表現方法が異なるだけで意味的には同じという可能性がある（例えば， $x^2+1=0$ と $a^2+1=0$ は同じ方程式である．）このため，ペナルティを比較的小さくした．また，式2では上下左右の移動経路はどちらかの文字列の伸縮を表しており，文字列を伸縮させないななめ移動の経路よりも大きいペナルティを設定している．式3ではペナルティ表の最上段，最左列といった l, d, u のすべてが求められないマスについて場合分けをしている．

表1の例のように答案 A, B の間で文字 'b', '4', 'c' の対応付けがとれた場合，'b' は，答案 A ではペンストローク s_{2A} ，答案 B ではペンストローク s_{1B} で表され，'4' は，答案 A では s_{3A}, s_{4A} ，答案 B では s_{2B}, s_{3B} で表される．このとき， $c_1^{AB} = ((s_{1A}, s_{2A}), (s_{1B}, s_{1B}))$ の対応付けが得られる．このようにして，す

すべての対応文字からペンストロークの対応付けを行うことができ、 C_{AB} が得られる。

対応付いた文字	答案 A	答案 B
	s_{1A}	
'b'	s_{2A}	s_{1B}
'4'	s_{3A}	s_{2B}
	s_{4A}	s_{3B}
'c'	s_{5A}	s_{4B}

表 1: ペンストロークデータ上の対応付け

答案間のペンストロークデータ上の対応付けが得られれば、3.1 の手順により、解答停滞箇所の検出を行うことができる。

3.5 解答停滞箇所の提示

教師に解答停滞箇所を提示する手段も考案した。提示にあたっては、答案から解答停滞箇所が視覚的に把握できることが重要である。ペンストロークデータから答案を再現することができるが、このときペンストロークを直前の *timescore* に応じて着色する。解答停滞箇所が色によって一目瞭然に提示できる上に、それ以外の部分、例えば人よりも素早く進めた箇所などを見ることもできる。図 4 は答案を 15 人と比較した結果、一例として直前の *timescore* が 2 秒以上であるペンストロークを解答停滞箇所として赤く、*timescore* が 0 に近づくほど青く着色した例である。

また、ペンストロークデータには解答過程の時間情報も含まれているため、その解答過程を動画再生することも可能である。ふたつの答案を、対応付いている箇所ごとにタイミングを同期させながら再生することで、各箇所ですれどちらがどれだけ停滞しているかがわかる。各箇所にかかった時間をそれぞれの答案ごとに棒グラフで表示する。棒グラフを見ながら、解答停滞箇所として検出されたペンストローク付近の箇所には実際どれくらいの時間がかかっているのか、またその時間は他者の答案と比べてどうなのかをより詳細に見ることができる。図 5 に提案する提示法による動画の 1 フレームを載せる。

Compared to the 15 peoples

$$\begin{aligned}
 & \frac{A}{\quad} - \frac{A}{\quad} \\
 &= A^2 - 2A - 15 - 9 \\
 &= A^2 - 2A - 24 \\
 &= (A - 6)(A + 4) \\
 &= (\lambda^2 - 5\lambda - 6)(\lambda^2 - 5\lambda + 4) \\
 &= (\lambda - 6)(\lambda + 1)(\lambda - 4)(\lambda - 1) \\
 & (\lambda - 6)(\lambda + 1)(\lambda - 4)(\lambda - 1)
 \end{aligned}$$



図 4: 解答停滞箇所

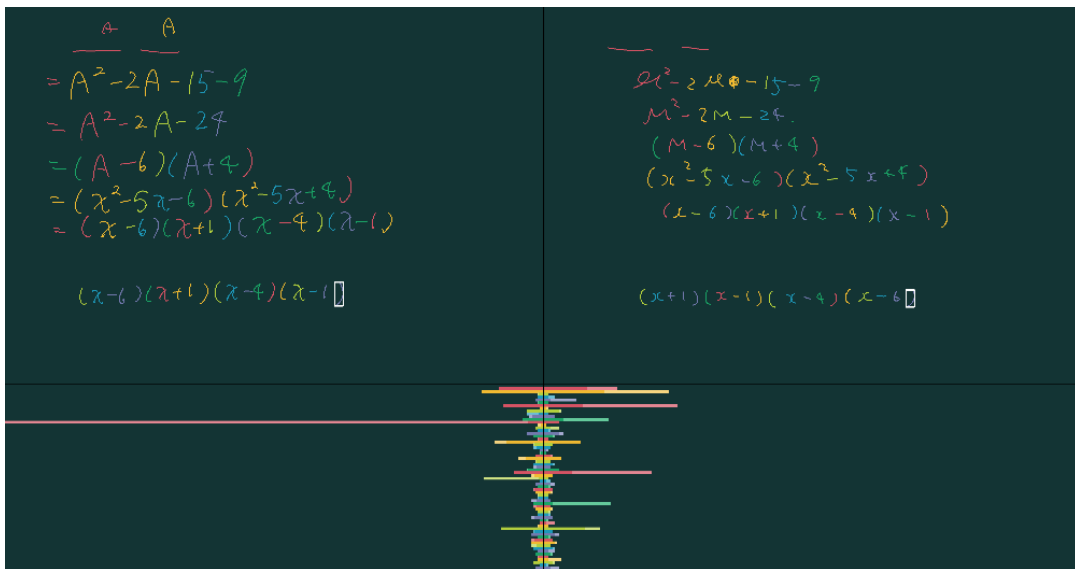


図 5: 答案の動画再生

第4章 実験

4.1 ペンストロークデータの収集

これまでに提案した検出手法を実際の教育現場で得られたペンストロークデータに適用する実験を行った。本実験で用いたペンストロークデータは、京都市立西京高校附属中学校3年生がタブレットPCの持ち帰り学習において、数学の因数分解を中心とした課題に解答した際に得られたものである。課題全7題を図8に示す。これらの課題は”Answer Box Creator (以下ABCと呼ぶ。)”¹⁾ というデジタルテストシステムを用いてWordファイルとして作成され、図6のような画面でタブレット上に手書き入力することで解答できる。解答中、ABCによって筆跡や時間情報がペンストロークデータとして記録される。図7はABCによって記録されたペンストロークデータの一部である。ペンストロークのペンダウン時刻、座標、ペンダウンからペンアップまでの経過時間、その他記述が行われたページ番号、使用ペンの種類などの情報が記録されている。使用ペンの種類にはデジタルペン、マウス、スタイラスペン、イレースペンがある。デジタルペン、マウス、スタイラスペンは記述を行うために用いられ、イレースペンは記述を削除するために用いられる。



図6: ABCの解答画面

本実験では生徒86名が上記の課題7題に取り組んだ際に得られた86個のペンストロークデータを使用する。すべての課題に対するペンストロークデー

¹⁾ ゼッターリンクス株式会社 — Answer Box Creator
<http://answerbox.jp> (2016年1月18日閲覧)

```

2015/11/21-13:55:19.896 MODE_ANSWER STYLUS;FF000000;1,1 2 292,3B0;293,3AE;294,3AC;294,3AB;
296,3A9;299,3A6;29D,3A4;2A0,3A2;2A3,3A0;2A7,39F;2AA,39E;2AD,39E;2B0,39E;2B3,39F;
2B6,3A1;2B8,3A3;2BA,3A5;2BB,3A8;2BC,3AB;2BC,3AE;2BB,3B1;2BA,3B5;2B8,3B8;2B7,3BB;2AD,3CC;
2A9,3D2;2A5,3D8;2A1,3DE;294,3ED;2B8,3FA;2B0,401;27F,402;27D,404;27C,405;27C,406;27E,
406;2B1,405;2B3,404;29B,3FE;2B4,3FA;2C9,3F7;2D6,3F5;2D8,3F4;2DB,3F3;2DD,3F2;2DF,
3F1;2E0,3F0;2E1,3EE;2E2,3EC 242
2015/11/21-13:55:22.777 MODE_ANSWER STYLUS;FF000000;1,1 2 555,390;554,394;553,398;552,39C;
551,3A5;550,3AD;54F,3B6;54C,3D2;54A,
3E9;549,3F6;549,3F7;549,3F8;549,3F9;549,3F9;549,3F8;54B,3EA;54C,3E4;54F,3DD;
552,3D7;554,3D4;556,3D2;559,3D0;55C,3CF;55F,3CE;562,3CF;
564,3D0;566,3D2;567,3D6;566,3D9;565,3DC;563,3DF;55F,3E3;55B,3E6;557,3E9;552,3EC;54D,
3F0;546,3F3;540,3F6;53C,3F7;53B,3F7;535,3F8;534,3F6;535,3F4;537,3F3;53B,3F2;53B,3F0;53D,
3EE;53E,3ED;540,3EB 1D1

```

図 7: ペンストロークデータ

タがひとつにまとまってしまっているため、各問題に対してページ数や座標によって解答領域を定め、各ペンストロークデータを7個の課題ごとに分割した。86 × 7 = 602 個のペンストロークデータが得られたが、このうち4個は白紙であったため取り除く。また、このペンストロークデータにはイレースペンによる入力（イレースペンストロークと呼ぶことにする。）が含まれる。生徒は誤って書き込んだ記述をイレースペンによって削除し、その上に新たに記述を行うことができる。このとき、ペンストロークデータでは書き損じによるペンストローク、イレースペンストローク、その上に重ねて入力されたペンストロークのすべてが記録されている。イレースペンストロークは簡単に取り除くことができるが、生徒が消した記述にあたるペンストロークを正確に特定し、ペンストロークデータから除くことは困難である。ペンストロークデータ上では重ね書きが発生することになる。書き損じによるペンストロークの重なりは、ペンストロークデータからの文字抽出において大きな妨げになり、失敗の主な原因となっていた。また、書き損じは、本研究で着目している解答過程の時間的な停滞を検出する妨げにもなる。そこで、今回はペンストロークの重なりが生じていると考えられるイレースペンストロークを含むペンストロークデータ387個は使用しないことにした。以上の操作を経て得られた $602 - 4 - 387 = 211$ 個のペンストロークデータを用いて実験を行った。

4.2 ペンストロークデータからの文字抽出結果

211 個のペンストロークデータを本手法で自動分割した結果を得た。人手で答案を1行ごとに分割したときの行数と誤差1行以内であれば正解、それよりも多いものを過分割、少ないものを寡分割とする。表2に課題ごとの正解、過分割、寡分割の数を示す。

正解と過分割のデータについて答案とその分割結果、それらから得られる文

2. 次の式を展開せよ.

(1) $(a^2 - b^2)(a^2 - ab + b^2)(a^2 + ab + b^2)$

1. 次の各問いに答えなさい.

(1) x に着目したとき, 次の多項式の次数と定数項を答えよ.

$$3a^2x^3 - (2a - b^2)x^2 + 3x + b^2 - 4ac$$

次数 … , 定数項 …

(2) 次の式を展開したとき, x^4 の係数を求めよ.

$$(x+1)(x^2+x+1)(x^3+x^2+x+1)$$

係数 …

【次のページへ】

【次のページへ】

3. 次の式を因数分解せよ.

(1) $3x^2y(a-b) + 12xy^2(b-a)$

(2) $x^3 - x^2 - 2x + 8$

4. 次の式を因数分解せよ.

(1) $3x^2 - 5xy - 2y^2 + 7x + 7y - 6$

(2) $a^2(b+c) + b^2(c+a) + c^2(a+b) + 3abc$

【次のページへ】

【これで終了です】

図 8: 数学課題

	正解	過分割	寡分割	計
課題 1	51	0	0	51
課題 2	29	23	0	53
課題 3	20	17	0	37
課題 4	23	1	0	24
課題 5	11	12	0	23
課題 6	6	7	0	13
課題 7	3	8	0	11
計	143	68	0	211

表 2: ペンストロークデータの分割結果

字抽出結果の例を挙げる．図 9 は正解データの例である．一行ごとに分割できていることが確認できる．図 10, 11 は過分割データの例である．本手法では答案は左から右，上から下に記述が行われていることを想定しているが，(a) では途中で記述の手戻りが起きている．これが原因となって，行途中の分割が起きている．(b) では思考の補助に使われた文字以外の記述が多い．これらの前後でしばしば行途中の分割が起きている．このような補助線は改行を決定する閾値として用いている *fontsize* に悪影響を与える．過分割と併発していたため表 2 にはあらわれなかったが，補助線などの影響で *fontsize* が大きくなりすぎたために，改行を見過ごしている例もあった．

文字抽出結果は，(a)，(b) 共にもともと文字でない，あるいは過分割により文字でなくなってしまったペンストロークから抽出された文字について誤りがある．また，正解データ，過分割データに関わらず，先行研究の手法が文字の個人差に対応できず文字抽出を誤っている例も散見された．

4.3 答案間の同等箇所に対応付け結果

同じ課題に対する全答案のペンストロークデータから答案対の全ての組み合わせについて文字抽出結果の対応付けを行った．ペンストロークデータ S_A から得られた文字列の長さを $length_A$ とおく．そのうちペンストロークデータ S_B から得られた文字列との間で文字列上の対応付けが取れた文字数を $num_{M,AB}$ と

$$\begin{aligned} &= 3x^3y(a-b) - 12xy^3(a-b) \\ &= 3xy(x^2 - 4y^2)(a-b) \\ &= 3xy(x+2y)(x-2y)(a-b) \end{aligned}$$

$$3xy(x+2y)(x-2y)(a-b)$$

$$= 3x^3y(a-b) - 12xy^3(a-b)$$

$$= 3xy(x^2 - 4y^2)(a-b)$$

$$= 3xy(x+2y)(x-2y)(a-b)$$

$$3xy(x+2y)(x-2y)(a-b)$$

【手動文字抽出結果】

【文字抽出結果】

$$\begin{aligned} &= 3x^3y(a-b) - 12xy^3(a-b) \\ &= 3xy(x^2 - 4y^2)(a-b) \\ &= 3xy(x+2y)(x-2y)(a-b) \\ &= 3xy(x+2y)(x-2y)(a-b) \end{aligned}$$

$$\begin{aligned} &- - 3x^3y(a-b) - 12xy^3(a-b) \\ &- - 3xxx^2 - 4y^2(a-b) \\ &- 3xy(x+2y)(x-2y)(a-b) \\ &= 3^xy(x+2y)(x-2y)(a-b) \end{aligned}$$

図 9: 正解分割データ

$$\begin{aligned} &(a^2 - b^2)(a^2 + b^2)^2 - a^2b^2 \\ &= (a^2 - b^2)(a^4 + 2a^2b^2 + b^4 - a^2b^2) \\ &= (a^2 - b^2)(a^4 + a^2b^2 + b^4) \\ &= a^6 - b^6 \end{aligned}$$

$$a^6 - b^6$$

$$\begin{aligned} &(a^2 - b^2)(a^2 + h^2)^2 \\ &= (a^2 - b^2)(a^4 + 2a^2h^2 + h^4) \\ &= (a^2 - b^2)(a^4 + a^2h^2 + h^4) \\ &= a^6 - b^6 \end{aligned}$$

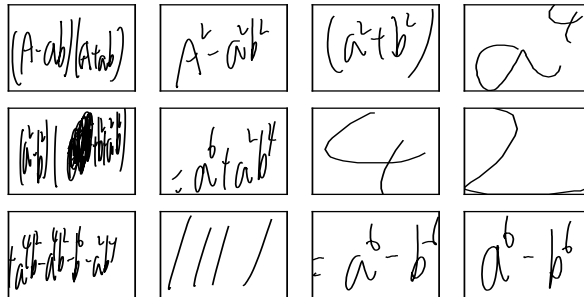
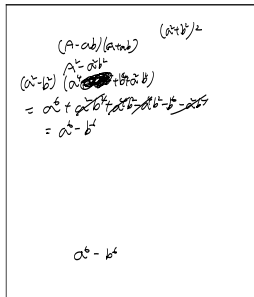
【手動文字抽出結果】

【文字抽出結果】

$$\begin{aligned} &(a^2 - b^2)(a^2 + b^2)^2 - a^2b^2 \\ &= (a^2 - b^2)(a^4 + 2a^2b^2 + b^4 - a^2b^2) \\ &= (a^2 - b^2)(a^4 + a^2b^2 + b^4) \\ &= a^6 - b^6 \\ &= a^6 - b^6 \end{aligned}$$

$$\begin{aligned} &(a^2 - b^2)(a^2 + h^2)^2 \\ &= (a^2 - b^2)(a^4 + 2a^2h^2 + h^4) \\ &= (a^2 - b^2)(a^4 + a^2h^2 + h^4) \\ &= a^6 - b^6 \end{aligned}$$

図 10: 過分割データ (a)



【手動文字抽出結果】

$$\begin{aligned}
 & (A - ab)(A + ab) \\
 & \quad A^2 - a^2b^2 \\
 & (a^2 - b^2)(a^4 + b^4 + a^2b^2) \\
 = & a^6 + a^2b^4 + a^4b^2 - a^4b^2 - b^6 - a^2b^4 \\
 = & a^6 - b^6 \\
 & \quad a^6 - b^6
 \end{aligned}$$

《ただし黒塗り，斜線，飛び領域は対象外とする》

【文字抽出結果】

$$\begin{aligned}
 & (n - -ab)(0 - 1 + ab) \\
 & \quad A^2 - a^2b^2 \\
 & \quad (a^2 + b^2) \\
 & \quad \quad a^4 \\
 & (a^2 - b^2)(2 + b^2 + a^2b^2) \\
 & \quad \quad \quad xk + a^2b^4 \\
 & \quad \quad \quad \quad 4 \\
 & \quad \quad \quad \quad 2 \\
 & + a^4b^2 - a^4b^2 - b^6 - a^2b^4 \\
 & \quad \quad \quad \quad 1 \cdot -1 \\
 & \quad \quad \quad \quad xa^b - b^1 \\
 & \quad \quad \quad \quad a^6 - b^6
 \end{aligned}$$

図 11: 過分割データ (b)

おくと，組み合わせ A, B の文字列上のマッチング割合 M_{AB} は

$$M_{AB} = 2num_{M,AB}/(length_A + length_B) \times 100$$

で算出される．これを課題ごとにヒストグラムにした結果を図 12 に示す．横軸がマッチング割合，縦軸が該当する組み合わせ数である．総組み合わせ数は各課題それぞれ 1275, 1225, 561, 253, 231, 78, 55 個である．文字列の長さの平均は，6.6, 28.6, 53.5, 62.5, 57.3, 56.5, 90.7 である．

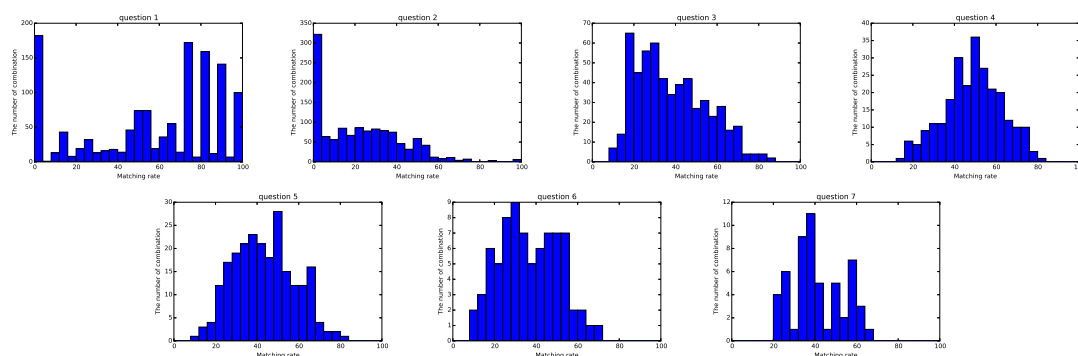


図 12: マッチング割合のヒストグラム

課題 1 は全体的に抽出された文字が少ないために，ほとんどが対応付く組とほとんど対応付かない組に二極化した．また，課題 2 についてはペンストロークデータから 1 文字も抽出されない例があったために，マッチング割合が 0 の組が多い結果となった．課題 3 は必要な知識がない人には難しいが，ある人には容易にとける問題であったために，答案中の式の量に個人差が大きかった．そのためマッチング割合の小さい組が多い結果となった．そのほかの課題についてはおおよそ山形の分布をしてることがわかった．

平均的に文字列の長さが十分にある課題 3,4,5,6,7 のうち，よく対応が取れている場合の代表例としてマッチング割合が 80% を超える組からふたつ選んでマッチング精度を調べる．手動で両答案から文字抽出を行い，その式や式中の項の意味を考慮しながら対応付けを行った結果を正解として用意する．実験結果をこれと比較し，次のふたつの尺度により評価する．

$$\text{適合率} = \frac{tp}{tp + fp}$$

$$\text{再現率} = \frac{tp}{tp + fn}$$

ただし, tp は実験結果で対応付いた文字の組のうち正解の組数, fp は実験結果で対応付いた文字の組のうち不正解の組数, fn は正解で対応付いている文字の組のうち実験結果では対応付かなかった文字の組数を表す.

まず一組目の精度を示す.

- 対応付けの正解

$$\begin{array}{ll} \frac{(x+2)(x^2+2x+4)-x(x+2)}{(x+2)(x^2+x+4)} & \frac{(x+2)(x^2-2x+4)-x(x+2)}{(x+2)(x^2-3x+4)} \\ \frac{(x+2)(x^2+x+4)}{(x+2)(x^2+x+4)} & \frac{(x+2)(x^2-3x+4)}{(x+2)(x^2-3x+4)} \end{array}$$

- 対応付けの実験結果

$$\begin{array}{ll} \frac{(x+2)(x^2+2x+4)-x(x+2)}{(xtL)(x^2+xx4)} & \frac{(x+2)(x^2-2x+4)-x(x+2)}{(xx^2)(x^2-3x+4)} \\ \frac{(xt2)(x^2+x+4)}{(xt2)(x^2+x+4)} & \frac{(x+2)(x^2-3x+4)}{(x+2)(x^2-3x+4)} \end{array}$$

- 適合率/再現率

94.6%/89.7%

内容がよく似た答案であるため高い精度でマッチングが行えた. 答案の文字列には少数, 符号や数字の違い, 文字抽出の段階での誤りが含まれていたが, これによる文字列間の差異にはあまり影響されず, 全体をうまく対応付けられた.

続いて二組目の精度を示す.

- 対応付けの正解

$$\begin{array}{ll} = \frac{(a-b)(3x^3y-12xy^3)}{(a-b)3xy(x^2-4y^2)} & \frac{(3x^3y-12xy^3)(a-b)}{3xy(x^2-4y^2)(a-b)} \\ = 3xy(x+2y)(x-2y)(a-b) & \frac{3xy(x+2y)(x-2y)(a-b)}{3xy(x+2y)(x-2y)(a-b)} \end{array}$$

- 対応付けの実験結果

$$\frac{c(a-b)(3x^3y-12xy^3)}{\pi(a-b)3xy(x^2-4y^2)} \quad \frac{(3+3y-12xy^3)(a-b)}{3xy(x^2-4y^2)(a-b)}$$

$$\frac{z3xy(x+2y)(x-2y)(a-b)}{3xy(x+2y)(x-2y)(a-b)} \quad \frac{3xy(x+2y)(x-2y)(a-b)}{3xy(x+2y)(x-2y)(a-b)}$$

- 適合率/再現率

76.6%/73.5%

この例でも文字抽出の誤りが含まれていたが、これによるマッチング精度への影響は小さい。一方で、式内部の項の順序が異なり、これがマッチング精度に悪い影響を与えていた。その結果、マッチング精度は一組目に比べて劣る。しかし、この二つの答案間の各対応付けにおいて時間間隔を比較したとき、解答停滞箇所が検出できるか調べたところ、一方の答案から二箇所が検出され、その近傍で記述の停止を確認した。よってこの例は、解答停滞箇所の検出を目的としたとき、低くはないマッチング精度であると言える。

4.4 解答停滞箇所の検出結果

ここまででペンストロークデータ上の対応付けが得られたので、3.1 節に従って解答停滞箇所を検出する。ただし、マッチング割合が低いとペンストロークのレベルでの対応付けの不安定さが大きくなり、安定して解答停滞箇所を検出することが難しくなると考えられる。そこで、この実験ではマッチング割合が 70 以上の組を用いて比較を行うこととする。この場合、各答案は平均 14.2 件の答案との比較を行う。

解答停滞箇所の *timescore* に対する検出の閾値は、人目で停滞箇所と感じられる長さを元に 2 秒と定めた。つまり平均 2 秒以上遅れている箇所を解答停滞箇所として検出する。この結果、93 個の答案のうち 84.9%からは解答停滞箇所は検出されなかった。残りの 15.1%の答案のほとんどに 5 個以上の解答停滞箇所が存在し、最大で 93 個、合計 298 個が検出された。

マッチング精度が 100%でないことにより、本来停滞していない箇所が本手法で検出されている可能性がある。そこで、検出した解答停滞箇所が確からしさを調べるために、検出箇所の前後 3 個のペンストロークの直前に記述の停止が

存在するか確認したところ、3秒以上の停止が検出箇所の81.2%に存在した。

4.5 考察

ペンストロークデータからの文字抽出

図9のように、答案の記述が各行で左から右、上から下に順序良く重ならないよう行われ、文字の大きさが一定の場合、ペンストロークデータの分割はうまくいった。本手法における分割アルゴリズムでは、主にペンストロークの時系列情報と答案中の文字の大きさを利用しているため、この二点について要求を満たしていれば、高い精度での分割が可能であると考えられる。分割が成功し、答案中に補助線などの文字以外の記述が少なければ、先行研究の手法が対応できる範囲で精度良く文字抽出を行うことができる。

答案間の同等箇所の対応付け

3.4節でマッチング精度に言及した二つの例は、解答停滞箇所の検出を目的としたとき十分な精度でのマッチングが行えていた。両者とも答案の内容がよく似ていて、文字抽出の精度も良かったために、対応付けに用いられた文字列同士が似ていたからである。これらの答案は、解法が限られている上に少ない手順で答えに至る、答案の内容が似やすい問題に対するものであった。

また、二つの例のうち後者は前者に比べてマッチング精度が低かった。後者では式中の項の順序が異なっていたためである。本手法の対応付けは、このような文字列の順序の入れ替えを苦手とするDPマッチングのみを用いている点で、改善の余地がある。

解答停滞箇所の検出

検出された解答停滞箇所は、他の答案と比べて遅れていた箇所である。この停滞箇所のうち81.2%が近傍に3秒以上の記述の停止が存在したことが確認された。これは、相対的な遅れはそもそも、一定以上の記述の停止が存在する箇所に現れるはずだからと考えられる。一方、近傍に停滞を発見できなかった18.8%については、答案間に対応付けを求める際に、多数のペンストロークがひとつの対応付けに含まれ、その中に大きく停滞している箇所が存在していたために、その他の停滞のないペンストロークにも大きい *timescore* が与えられてしまったからだと考えられる。

また、解答停滞箇所が93個検出された例があった。この答案は常に解答の記述が遅れていたということである。教師による指導が必要と考えられる一方で、

本手法ではこのような答案の中で特に停滞している箇所を発見することができないことから，発展的な課題として，本手法で用いた個人間の遅れの指標に加え，個人内の遅れの指標も統合することが挙げられる．

第5章 結論

本研究では，教育の情報化に伴って蓄積できるようになった学習ログデータを活用し，教師が生徒の習熟度を把握するための支援を行う手法として，数学答案を対象としたペンストロークの時間間隔を用いた答案の解答停滞箇所の検出手法を提案した．どの程度以上の時間を停滞とするのか，基準を得るために同一問題に対する複数の答案間でペンストロークの時間間隔の比較を行った．この比較では，答案ごとに解答過程内の同等箇所がペンストロークデータ上で対応している必要があるが，それに対して答案ごとの筆跡の違いや数式の表現方法の違いなどが問題となっていた．筆跡の違いには文字抽出によって対処し，数式の表現方法の違いには DP マッチングで抽出した文字列間の対応付けを行うことによって対処した．こうして求めたペンストロークの対応している箇所について時間間隔の比較を行い，大きく停滞していた箇所を検出した．さらに，教師への解答停滞箇所の効果的な提示のために可視化も行った．今後は，この提示方法の有用性や，そもそも解答停滞箇所自体の有用性について教師による評価を行う必要がある．また，記述のペースが常に遅いために答案全体が解答停滞箇所として検出された例に対して，答案中の特に停滞していた箇所を明らかにする指標も追加することが，発展的な課題として挙げられる．

謝辞

本研究を進めるにあたり，多くの御教示を賜りました美濃導彦教授，飯山将晃准教授，森村吉貴助教に深く感謝致します．また，本研究について多くの助言を賜りました村上正行教授，橋本敦史助教に厚く御礼申し上げます．最後に，日頃より多くのご意見をくださいました観光講義グループの皆様，ならびに美濃研究室の皆様に深く感謝いたします．

参考文献

- [1] 文部科学省: 教育の情報化ビジョン, http://www.mext.go.jp/b_menu/houdou/23/04/_icsFiles/afieldfile/2011/04/28/1305484_01_1.pdf.
- [2] 総務省: フューチャースクール推進事業, http://www.soumu.go.jp/main_sosiki/joho_tsusin/kyouiku_joho-ka/future_school.html.
- [3] 文部科学省: 学びのイノベーション事業実証研究報告書, http://www.mext.go.jp/b_menu/shingi/chousa/shougai/030/toushin/1346504.htm.
- [4] 中村和晃, 角所考, 村上正行, 美濃導彦: e-learning における学習者の顔動作観測に基づく主観的難易度の推定, 電子情報通信学会論文誌 D, Vol. 93, No. 5, pp. 568–578 (2010).
- [5] 中村和晃, 角所考, 美濃導彦: e-learning 環境における学習者の観測に基づく主観的難易度の推定, 画像の認識・理解シンポジウム (MIRU2007), Vol. 6 (2007).
- [6] 向後千春, 中井あづみ, 野嶋栄一郎: e ラーニングにおける先延ばし傾向とドロップアウトの関係, 日本教育工学会研究報告集, JSET04-5, pp. 9–44 (2004).
- [7] 向後千春, 野嶋栄一郎: e ラーニングにおけるドロップアウトとその兆候, 日本教育工学会第 20 回全国大会講演論文集, pp. 997–998 (2004).
- [8] 植野真臣: e ラーニングにおける所要時間データの異常値オンライン検出, 電子情報通信学会論文誌 D, Vol. 90, No. 1, pp. 40–51 (2007).
- [9] Epp, C., Lippold, M. and Mandryk, R. L.: Identifying emotional states using keystroke dynamics, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM, pp. 715–724 (2011).
- [10] Alvaro, F., Sanchez, J.-A. and Benedi, J.-M.: Offline Features for Classifying Handwritten Math Symbols with Recurrent Neural Networks, *Pattern Recognition (ICPR), 2014 22nd International Conference on*, IEEE, pp. 2944–2949 (2014).
- [11] Álvaro, F., Sánchez, J.-A. and Benedí, J.-M.: An integrated grammar-based approach for mathematical expression recognition, *Pattern Recognition*, Vol. 51, pp. 135–147 (2016).
- [12] Baeza-Yates, R., Ribeiro-Neto, B. et al.: *Modern information retrieval*,

Vol. 463, ACM press New York (1999).