User Interfaces and Data Analysis on Digital Handwriting Environment

デジタル手書き環境におけるユーザインタフェースとデータ解析手法に関する研究

December 2015

Hiroki ASAI

浅井 洋樹

User Interfaces and Data Analysis on Digital Handwriting Environment

デジタル手書き環境におけるユーザインタフェースとデータ解析手法に関する研究

December 2015

Waseda University

Graduate School of Fundamental Science and Engineering
Department of Computer Science and Engineering,
Research on Parallel and Distributed Architecture

Hiroki ASAI

浅井 洋樹

Acknowledgements

First and foremost, I would like to express the deepest appreciation to Prof. Hayato Yamana, of the Faculty of Science and Engineering at Waseda University. He is the supervisor of my master and doctor thesis. He supported all of my research life and invited me to doctor course. If it were not for the guidance of him, it should have been more difficult to complete the thesis.

I also would like to offer my special thanks to Prof. Tatsuo Nakajima and Prof. Yasuo Matsuyama, both of the Faculty of Science and Engineering at Waseda University. They gave me valuable comments and suggestions for improving my thesis.

Here, I would like to thank all of the members of Prof. Yamana's laboratory. Especially, Dr. Takanori Ueda of IBM Research Tokyo, who has graduated from the laboratory, gave me many supports after his graduation as well as when he belonged to the laboratory. Dr. Sayaka Akioka who is the associate professor of Meiji University gave me great experiences as well as helping my research. I sincerely thank Mr. Shogo Sonoda for the collaborative work about the estimation of human memory level.

I wish to thank to ACM SIG CHI community for giving many useful comments. When I presented the study about this thesis at the conference related to the community, participants gave a lot of comments. Also, I received many valuable comments from the review of the accepted and rejected papers. I could advance my thesis based on the comments.

Finally, I thank my family and friends for their kind encouragement and support for 5 years of my research life.

December 2015

Hiroki Asai

Abstract

The use of tablet devices is widely spreading due to an increasing demand for mobile computing. These devices employ a multi-touch interface whose users can navigate information by touching directly on the display; however, while the current interface is suitable for navigating information, it is unsuitable for inputting information such as diagrams and free-form note-taking. A promising technology to solve this problem is pen-based computing, which enables the use of a stylus on a device's display. Previous proposals about interaction techniques of pen-based computing enable to implement extra functions like information retrieval. However, they can reduce the learnability of the systems using the techniques because the techniques require users to remember extra gestures besides paper-based handwriting interactions. In addition, proposed recognition techniques of handwritten data mainly focused on visible handwritten data while "invisible" handwritten data such as pressure and velocity have a potential to extract effective information.

This thesis therefore proposes the new technologies in such digital handwriting environments from two points of view: (1) the user interfaces of digital handwriting that increases the learnability of the systems to detect user's intentions, and (2) the data analysis methods to extract effective information from invisible handwritten data. As the study of the user interfaces, Part I focuses on document annotation on electronic documents, such as information-appending and emphasizing contents by handwriting. Part I first proposes the recognition model for handwritten annotation of electronic documents to help users to annotate documents and to search documents towards intelligent user interfaces. The study found the proposed model can find user's content-targeting intention for 95% precision, and estimate targeted contents for 70 to 88% accuracy. Based on the recognition model, the rest of Part I proposes two applications. The first is the "Intelligent Ink Annotation Framework," supporting handwritten annotation of electronic documents. This framework enables improved availability of annotation data without users' undue overhead. User study found proposed framework was preferred to 75% of participants. The other is "EA Snippets," which refers to the thumbnails and text snippets of handwritten notebooks. The snippets are generated by summarizing handwritten documents based on emphasis annotation by users. The user study found proposed snippets improve search time 42% faster on average.

Part II focuses on the extraction of effective information from digital handwritten data. Online handwritten data, which are obtained by computers, include additional features—such as pressure and velocity—that cannot be obtained from paper-based handwriting. Such data provide the potential to extract effective information. Part II

describes two methods for extracting effective information from online handwritten data. One method involves extracting the psychological state of students. Students' psychological information is indispensable for teachers to know the understanding levels of the students and to teach them appropriately based on their understanding. This method detects the psychological state of students, such as frustration and the need for help, from online handwritten data. The experiments showed proposed method detects the frustration in 87% precision and 90% recall. The other method involves the estimation of human memory levels. Rote learning, which is a memorization technique based on repetition, such as the memorization of Chinese characters and English words by Japanese students, is required to grasp the incompletely memorized items for efficient memorization since learning completely memorized items wastes time. This method estimates people's degrees of human memory by using their online handwritten data to determine the most effective rote learning system. The experiments showed proposed model achieved the best performance with 0.69 F-value.

Finally, this thesis describes the conclusions of each part and discusses the future direction of this research area.

Contents

List of Figures

List of Tables

Introduction

1.1. Introduction

Both the mouse and keyboard have been adopted as the input devices of personal computers from the advent of the Graphical User Interface (GUI) that consists of Windows, Icons, Menus, and Pointer (WIMP). The mouse is mainly used for navigating information on personal computers, and the keyboard is used for inputting information. On the other hand, the use of tablet devices has been widely spreading in recent years since the demand of mobile computing is increasing. The investigator says the increases in worldwide tablet shipments will outgrow increases in traditional Personal Computer (PC) shipments[1].

The input device for a tablet PC adopts a multi-touch interface by which users can navigate information by touching directly on the display. In the multi-touch interface on a tablet device, users navigate information by touching displayed objects and input information by touching the keyboard displayed on the display; however, the interface is unsuitable for inputting information such as diagrams and free-form note-taking. A promising technology to solve this problem is pen-based computing, which uses a pen for inputting information to a computer.

Previous fundamental research related to pen-based computing can be classified into two types. One is the recognition of handwritten data such as characters [1] [2] [3], mathematics [4] [5] [6] and diagrams [7] [8] [9]. The other is an interaction technique like using additional sensors [10] [11] [12] [13] and gestures [14] [15] [16]. These studies enable 1) computers to understand what users write to computers, and 2) users to use handwriting on computers without impairing current GUI functions.

While the above techniques can achieve effective functions such as information retrieval and interactive document navigation, it reduces the learnability of the digital handwriting interface than paper-based handwriting because it requires users more complicate operation. For example, traditional handwriting annotation systems on electronic documents, which are one of the common use cases of digital handwriting, require users to indicate computers what users want to do like underlining content and adding comments by using some interactive techniques mentioned above in 2) to recognize annotation information. Part I of this thesis therefore proposes the interface that

[1] Gartner Says Worldwide Traditional PC, Tablet, Ultramobile and Mobile Phone Shipments to Grow 4.2 Percent in 2014, "http://www.gartner.com/newsroom/id/2791017."

helps users to perform digital handwriting to detect user's intentions. Proposed framework detects user's intention of digital handwriting annotation automatically, and then support user's input to reduce learnability of the interface.

On the other hand, most of past recognition methods of digital handwritten data, which is mentioned above in 1), are focused on visible handwriting data. In digital handwriting environment, however, computers also obtain "invisible" handwritten data like a pressure factor and velocity. These invisible handwritten data have a potential to extract effective information like [17]. Therefore, Part II of this thesis also attempt to extract effective information from invisible handwritten data in educational situations that is frequently used in handwriting. The proposed techniques detect a psychological state of students and estimate human memory level.

1.2. Contributions

This thesis proposes the technologies in a digital handwriting environment from two points of view: (1) the user interfaces of digital handwriting that increases the learnability of the systems to detect user's intentions, and (2) the data analysis methods to extract effective information from invisible handwritten data. The contributions are described as follows:

- Part I: User Interfaces for Handwriting Annotation
  1) Chapter 2: A Recognition Model for Handwritten Annotation
     - ✧ Proposing the model that detects information-appending and content-targeting annotation.
     - ✧ Helping users with annotation and improving availability on electronic documents.
  2) Chapter 3: An Intelligent Ink Annotation Framework (IIA Framework)
     - ✧ Proposing the application framework using the proposal in 1).
     - ✧ Increasing the learnability of annotation systems to detect annotation behavior automatically.
  3) Chapter 4: Snippets for Handwritten Documents
     - ✧ Proposing the application snippets using the proposal in 1).
     - ✧ Summarizing handwritten documents based on emphasis annotation.
     - ✧ Improving the search performance of handwritten notebooks.
- Part II: Data Analysis for Extractive Effective Information
  1) Chapter 2: A Psychological State Detection of Students

- ✧ Extracting the psychological state of students such as frustration and need help
- ✧ Decreasing the burden of teachers grasping the understanding of students

2) Chapter 3: An Estimation of Human Memory Level
   - ✧ Estimating the degree of human memory of the specific memorizing item.
   - ✧ Improving the efficiency of rote learning

1.2.1. User Interfaces for Handwriting Annotation (Part I)

Part I focuses on handwritten document annotation, such as information-appending and emphasizing contents of electronic documents. First the recognition model of handwritten annotation of electronic documents is proposed to help users both to annotate documents and to search documents towards intelligent user interfaces. The applications of digital handwriting document annotation require the accurate recognition of the user's selected range on documents; however, there is no research recognizing the strict user's selected range of documents, while traditional heuristic methods can recognize the rough user's selected range. Therefore, the goal of Part I is proposing the recognition model that uses the collected human annotation data.

In Part I two applications using the recognition model are also proposed. The first is the "Intelligent Ink Annotation Framework," supporting handwritten annotation of electronic documents. Traditional handwriting annotation systems require users to perform system-defined gestures and additional operations such as menu selection. The framework enables improved availability of annotation data without users' undue

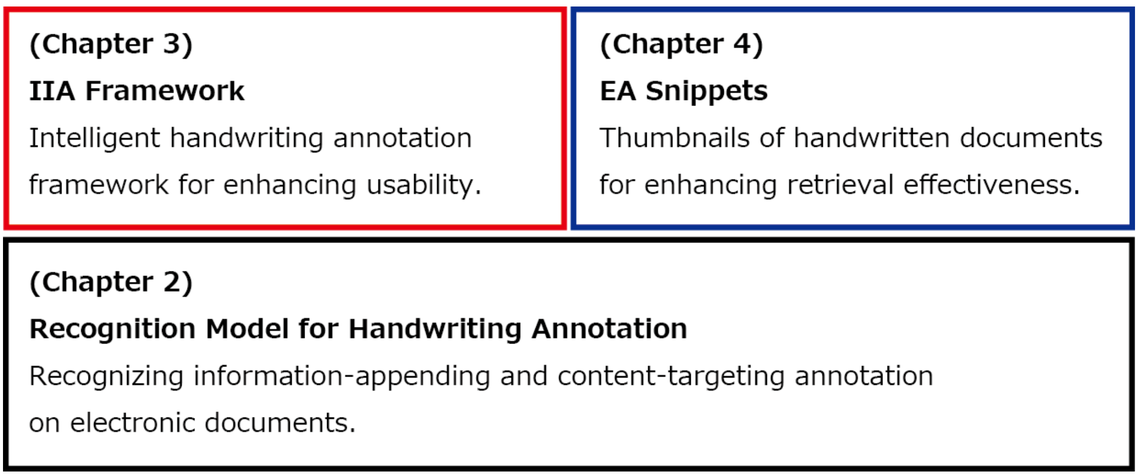


Figure 0.1 Architecture of Part I

overhead. The other is "EA Snippets," which refers to the thumbnails and text snippets of handwritten notebooks. While there is research about the snippets of images and Web pages, the proposed methods of the research cannot be applied to handwritten data. The proposed snippets are generated by summarizing handwritten documents based on emphasis annotation by users and will improve the search performance of digital handwritten notebooks.

1.2.2. Data Analysis for Extracting Effective Information (Part II)

Part II focuses on the extraction of effective information from digital handwritten data. Online handwritten data, which are obtained by computers, includes additional features—such as pressure and velocity—that cannot be obtained from paper-based handwriting. Such data provide the potential to extract effective information. Part II proposes two methods for extracting effective information from online handwritten data.

One method involves extracting the psychological state of students. Students' psychological information is indispensable for teachers both to know the understanding levels of the students and to teach them suitably for their understanding; however, the traditional studies did not try to detect such information from online handwritten data although previous studies revealed the extraction of cognitive load from online handwritten data. The proposed method detects the psychological state of students, such as

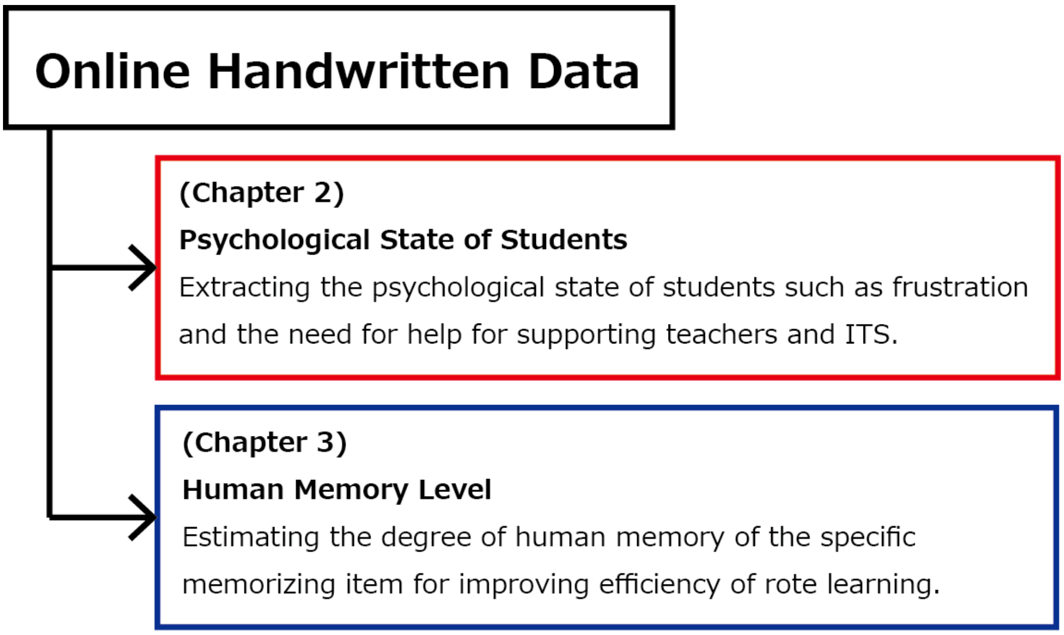


Figure 0.2 Architecture of Part II

frustration and the need for help, from online handwritten data.

The other method involves the estimation of human memory levels. Rote learning, which is a memorization technique based on repetition, such as the memorization of Kanji and English words by Japanese students, is required to grasp the incompletely memorized items for efficient memorization since learning completely memorized items wastes time. The handwriting behavior of learners has the potential to improve the estimation of human memory levels, whereas traditional research detects these levels by using the results of recall tests and subjective evaluations. This method estimates people's degrees of human memory by using their online handwritten data to determine the most effective rote learning system.

1.3. Outline of this Thesis

The remainder of this thesis is organized as follows:

- Part I
  - Chapter 1 describes the introduction of the user interfaces of digital handwriting annotation.
  - Chapter 2 describes the recognition model for handwriting annotation on electronic documents.
  - Chapter 3 describes Intelligent Ink Annotation Framework (IIA Framework) using the recognition model.
  - Chapter 4 describes the EA Snippets that show the thumbnails and text snippets summarized by emphasis annotation using the recognition model.
  - Chapter 5 describes the conclusion of the part.
- Part II
  - Chapter 1 describes the introduction of the data analysis for effective information extraction.
  - Chapter 2 describes the extraction of students' psychological state
  - Chapter 3 describes the estimation of human memory level
  - Chapter 4 describes the conclusion of the part.

Part I.

User Interfaces for Digital Handwriting Annotation

Chapter 1. Introduction

This part focuses on the handwriting document annotation on electronic documents such as information-appending and content-targeting. Due to the development of digital handwriting environment like tablet PC, we can annotate electronic documents by handwriting as well as on paper-based document. While such development has a potential to enhance the performance of information retrieval, there are two research questions: (1) how annotation system is required to achieve the enhancement without degrading usability, and (2) how information retrieval interface is required to achieve the enhancement. This part proposes two kinds of applications to answer these questions followed by proposing the recognition model of user's intention for handwriting document annotation.

At first, Chapter 2 proposes the recognition model of handwritten annotation on electronic documents to help users both to annotate documents and to search documents. These applications require the accurate recognition of the user's selected range on documents. However, traditional heuristic methods recognize the rough user's selected range. Therefore, the part proposes the recognition model that uses the collected human annotation data, and then recognizes accurate selected range on documents.

Based on the recognition model, the rest of this part describes two proposals. One is the intelligent ink annotation framework supporting handwriting annotation on electronic documents described in Chapter 3. Traditional handwriting annotation system requires users to perform system-defined gesture and additional operations such as menu-selection. The framework enables to improve availability of annotation data without user's undue overhead.

The other is the EA Snippets that show the thumbnails and text snippets of handwritten notebooks described in Chapter 4. Previous studies generating the snippets of images and web pages cannot be applied to handwritten data. The proposed snippets are generated by summarizing handwritten documents based on the emphasis annotation by users, and then improve search performance of digital handwritten notebooks.

Chapter 2. Recognition Model for Handwriting Annotation System

2.1. Introduction

Document annotation, such as appending information to books and printed documents by readers, is important interaction between human and documents [18]. Especially, handwriting annotation, which represents information-appending or emphasizing contents like underline and enclosure by handwriting, is indispensable since it is used in many situations such as document understanding, editing and proofing. The process of document annotation can be divided into two steps:

1) **Content-targeting**
   means the behavior such that a reader selects the range of contents to emphasize or append information.
2) **Information-appending**
   means the behavior such that a reader writes information to append comments and diagrams.

Content-targeting is like underlines and enclosures when we perform on paper-based notebook. Information-appending is like writing text and diagrams.

On the other hand, we commonly perform content-targeting and information-appending by using mouse and keyboard that are input devices of computer instead of using a stylus in handwriting annotation on electronic documents [19]. It is important to make a connection between appended information and selected range of contents in this kind of annotation on electronic documents. This is because the connection results in realizing the applications such as information retrieval and information navigation [20].

There are some researches about an interface for handwriting document annotation on electronic documents using the advantages of an electronic display [21] [22] [23] [24] [25] since there is a study reporting the inefficiency of document annotation using a keyboard and mouse compare with handwriting [26]. However, such researches related to handwriting annotation on electronic documents revealed the problem about the recognition of the target range on contents to emphasize and to append information to make a connection between annotating or emphasizing information and the target range of contents.

This chapter therefore proposes the recognition model of handwriting annotation on electronic documents. The proposed model recognizes the followings:

1) Divide handwritten strokes into content-targeting and information-appending.
2) Recognize the selected regions intended by handwritten annotations, i.e., the selected range intended by underline.

The recognition model enables to make a connection between appending handwritten information and indicated contents on electronic documents without additional special operations by users.

2.2. Related Work

The recognition model proposed in this chapter aims at making a connection between targeted contents by handwriting and appended handwritten comments on electronic documents. This section initially describes investigational studies of handwriting annotation on paper-based documents to define what kind of annotation the proposed model recognizes. Next section describes the studies mentioned to handwriting annotation on electronic documents to describe the position of the proposed recognition model.

2.2.1. Handwriting Annotation for Paper-based Documents

Marshall et al. reported the investigation of handwriting annotation, "telegraphic" and "explicit", on the text [18]. The investigation was undertaken on university's classes of language, history, mathematics and chemistry.

1) **Telegraphic**

   Notes like underline and enclosure on text; brackets (we call vertical), symbol and arrows on margin.
2) **Explicit**

   Notes like comments and translation on text; comments which cannot include on text.

On the other hand, Wang et al. [27] argue following two categories of annotation functions:

1) **Actionable**

   The annotation indicating editorial operation such as insert, delete, move and replace.

2) **Non-Actionable**

The annotation for information-appending such as an explanatory note, summarization, emphasis and comments.

2.2.2. Recognition of Handwriting Annotation on Electronic Documents

There are some researches related to the recognition of handwritten annotation on electronic documents [28] [29] [27] [30] [31].

Schilit et al. [28] and Olsen et al. [29] proposed the method to generate captured images around handwritten annotation on electronic documents in their system to detect handwriting annotation. Schilit et al. [28] proposed the system of active reading on electronic documents, then produced thumbnail images to extract around annotated area on the document in their system. The thumbnail is produced by expanding the bounding box of handwritten strokes simply to include their annotated contents. They also mentioned the way extracting a search keyword underlined by a stylus. However, they did not mention the detail of the method to recognize selected characters from a handwritten stroke. Olsen et al. [29] proposed the system that enables to annotate comments by handwriting on the various contents displayed by computer such as emails and web browsers, and that accumulates the data. Their proposed system includes the method extracting annotated area on the document. The method achieved extraction of annotated area according to their defined heuristic rule of searching annotated contents by image processing techniques.

On contrast, Wang et al. [27] described the method to recognize handwritten annotation on free-form notebooks. They extracted the semantic features that have relationships between handwritten annotation and elements of notebook contents, in addition to the shape features described in the research by Fonseca et al. [32], then recognized handwritten annotation using machine learning method. Wang et al. achieve the detection of free-form handwritten annotation same on paper-based documents with high accuracy by using the method.

In addition, there is the Reflow Problem so that the consistency of handwritten annotation positions cannot be maintained with changing layouts of document contents in handwriting annotation on electronic documents, which can be changed with layouts dynamically. Bargeron et al. [33] mentioned the necessity to recognize the relationships between handwritten annotation and selected contents of documents to propose the framework to solve such problem. In the researches dealing with the Reflow Problem, Golovchinsky et al. [30] and Shilman et al. [31] mentioned the recognition of handwritten annotation. Golovchinsky et al. [30] proposed the method to reallocate handwritten annotation on electronic documents with the change of layouts. The method makes a connection between the selected range of text on documents and handwritten annotations by the heuristic method using location relationships between handwritten stroke and content of document, according to the process defined per each type of handwritten annotation. Shilman et al. [31] proposed the method to classify handwritten annotation strokes into content-targeting, remarks and comments by using context features and features compared with ideal shapes. They described the method solving Reflow Problem by using the results of recognition.

As previously noted, there are two approaches to detect content-targeting annotations. One is heuristic approach [28] [29] [30], while the other is learning approach using collected handwritten data [27] [31]. Proposed method adopts learning approach using handwritten data to use shape features and semantic features mentioned in [27] [31]. Furthermore, there are two approaches to recognize the selected range on document contents. One is the approach of extracting the area images around handwritten annotation [28] [29], while the other is the approach of making a connection between actual selected contents and handwritten annotation [30] [31]. However, the method which extracts captured images cannot make a connection between selected contents and handwritten stroke. In addition, only [30] describes the method recognizing selected range on electronic documents in researches about making a connection between selected contents and handwritten strokes. The problem of the method is not considering human habits of handwriting annotation, and then the method cannot recognize exact selected range of annotation.

2.3. Recognition Model of Handwriting Annotation

The related research about recognizing selected range intended by handwritten annotation described in previous section has the problem not to achieve sufficient recognition accuracy because their heuristic approach cannot be adapted to slippage by human habits. This chapter therefore proposes the recognition model based on handwritten data collected by participants. The input and output data of proposed recognition model are described below:

- **Input data**
  1) **On-line handwritten stroke**
     The trajectory coordinates of pen tips from pen contacting with the screen to a stylus away from display.
  2) **Electronic document source**
     The document source which can be obtained the size and location of

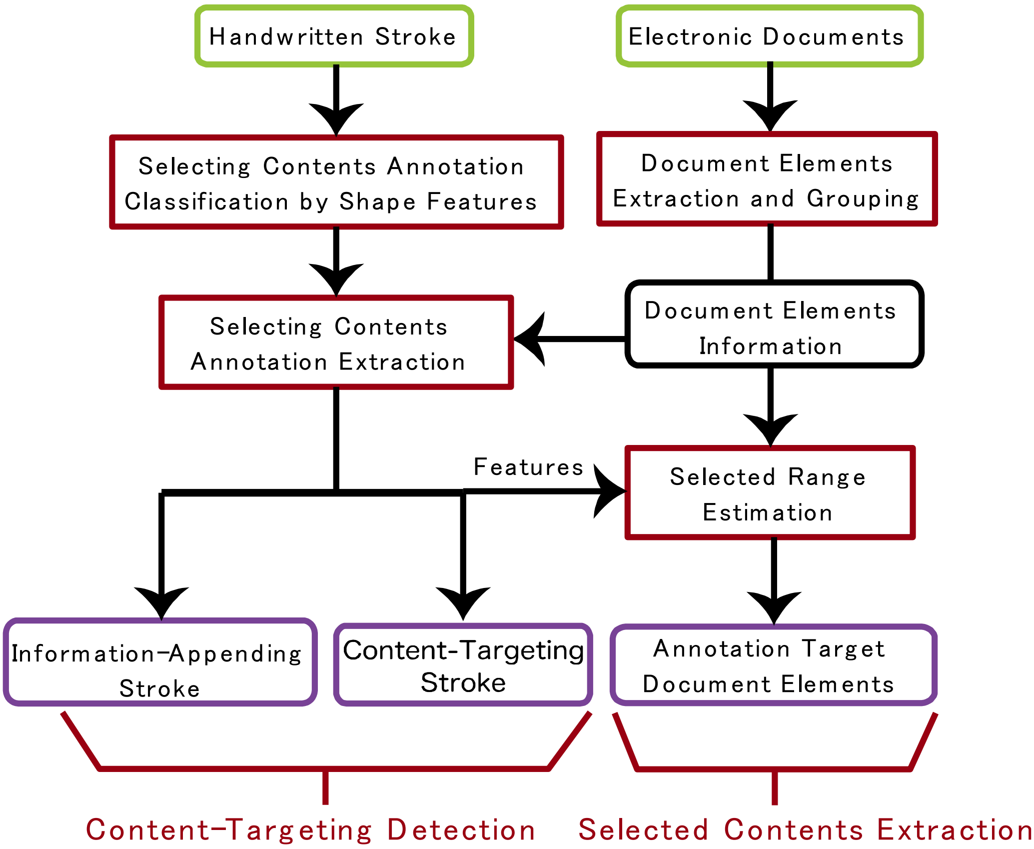


Figure 2.1 Processing flow of the proposed recognition model

characters like PDF and HTML documents.

- **Output data**
  1) **Classification result of handwritten stroke**
     The recognition result that a handwritten stroke is classified into content-targeting and information-appending.
  2) **The selected range on document contents**
     The recognition result of selected range intended by content-targeting annotation.

"Content-targeting" represents the handwritten strokes which selects the range of contents on electronic documents like underline and enclosure. "Information-appending" represents the handwritten strokes except for content-targeting like handwritten comments.

Figure 2.1 shows the process flow of proposed recognition model mentioned above. First, the proposed recognition model extracts the elements, which are the components of documents such as text and picture, from electronic documents. This chapter calls this “document element.” In addition, the text data in document elements is grouped on a row-by-row basis. Besides, all of the inputted handwritten strokes are initially assumed as content-targeting strokes, and then the strokes classified into three types of content-targeting annotation. After that, the model classifies handwritten strokes as content-targeting or information-appending by using the location information of document elements and classification information of content-targeting. Selected region of contents is estimated from the handwritten strokes classified as content-targeting. The terms and its definitions are described in Table 2.1. The definitions of symbols in this section are described in Table 2.2. The detail of the recognition model is described as below.

Table 2.1 Terms and definitions in this chapter

Terms	**Definitions**
Content-targeting	The handwritten strokes which selects the range of contents on electronic documents like underline, enclosure and vertical
Information-appending	The handwritten strokes except for content-targeting like handwritten comments
Document element	The components of documents such as text and picture

Table 2.2 Definitions of symbols

Sympols	**Definitions**
t_i	The bounding box of a document element which is occurred in the ith element in the target document data
d_i	Connection distance between the document elements t_i and t_{i+1}
S	Threshold of dividing document elements into text line
$\widetilde{D}$	The median of d
W	The average width of t
W_{stroke}, H_{stroke}	The width and height of the bounding box in the handwritten stroke
L_{stroke}	The length of the handwritten stroke
p_{rc}	The relative point from annotation start/end point to the gravity point of the bounding box
AD	Annotation distance
AD_{thres}	The threshold value of AD to detect content-targeting
t_{target}	The document element which indicates the start/end element of content-targeting

2.3.1. Recognition Target Annotations

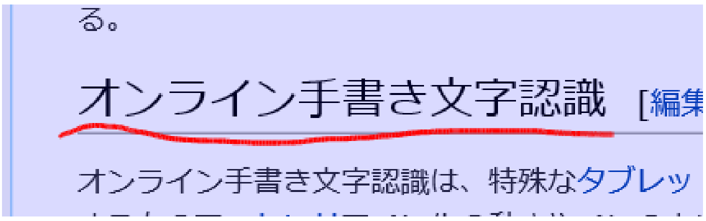
The types of annotation in proposed recognition model are information-appending and content-targeting on documents, such in Active Reading [34] [20] that is annotation to understand reading documents. The proposed model regards these annotation processes as repeating content-targeting which indicate the range of content on documents, and information-appending which appends comments on documents. The method classifies handwritten annotation as content-targeting and information-appending, and then estimates the range of selected contents by content-targeting annotation.

Proposed model recognizes following four types of content-targeting.

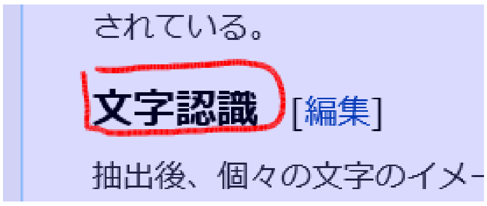
1) Underline
2) Enclosure (single-line)
3) Enclosure (multi-line)
4) Vertical

Content-targeting corresponds to Telegraphic category in Marshall's proposed classification and Non-Actionable category in Wang's proposed classification described in 2.2.1. Actionable category in Wang's proposed classification represents editing operation to documents but not information-appending in proposed model. Hence, the proposed recognition model excludes the recognition type since the category should be discussed in specific application systems like operating menu and recognizing gesture. Figure 2.2 shows the kind of detection targets of content-targeting annotation.

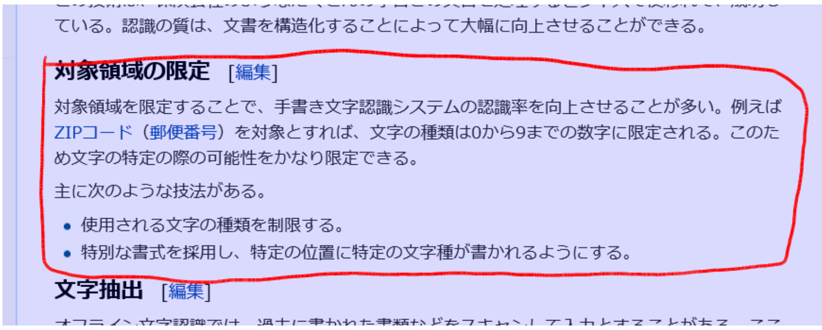
Moreover, Marshall [18], Wang et al. [35] [27] and Schilit et al. [28] treated remarks (arrows, callout) as elements of handwriting annotation. The remark, which connects information-appending annotation and the annotated content on documents, corresponds to make a connection between content-targeting and information-appending. The remark is used with content-targeting annotation represented in Figure 2.2 when we indicate the range of content-targeting behavior clearly. Consequently, proposed model recognizes the remark as information-appending behavior not as content-targeting behavior.



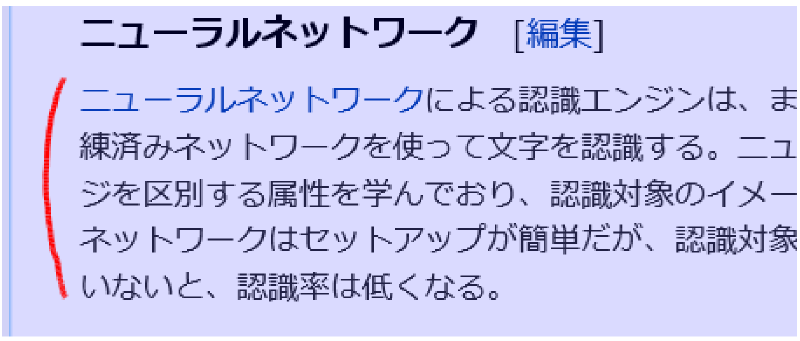
(a) Underline



(b) Enclosure

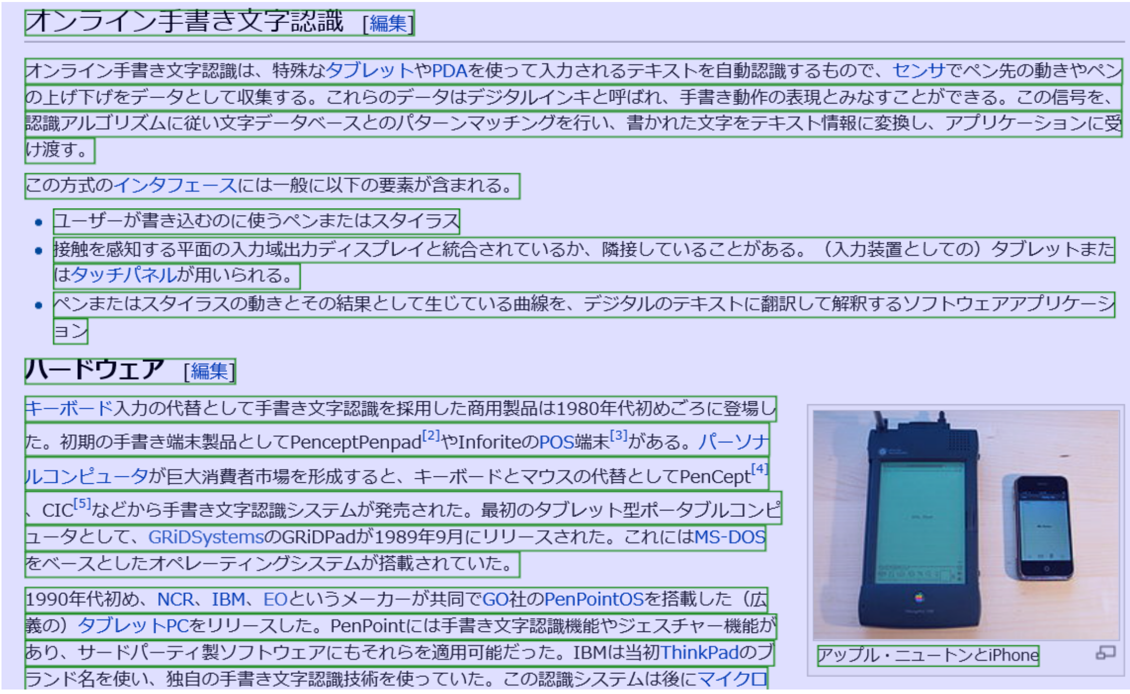


(c) Multiline-Enclosure



(d) Vertical

Figure 2.2 Detection targets of content-targeting annotation



オンライン手書き文字認識 [編集]

オンライン手書き文字認識は、特殊なタブレットやPDAを使って入力されるテキストを自動認識するもので、センサでペン先の動きやペンの上げ下げをデータとして収集する。これらのデータはデジタルインキと呼ばれ、手書き動作の表現とみなすことができる。この信号を、認識アルゴリズムに従い文字データベースとのパターンマッチングを行い、書かれた文字をテキスト情報に変換し、アプリケーションに受け渡す。

この方式のインタフェースには一般に以下の要素が含まれる。

- ユーザーが書き込むのに使うペンまたはスタイラス
- 接触を感知する平面の入力域出力ディスプレイと統合されているか、隣接していることがある。（入力装置としての）タブレットまたはタッチパネルが用いられる。
- ペンまたはスタイラスの動きとその結果として生じている曲線を、デジタルのテキストに翻訳して解釈するソフトウェアアプリケーション

ハードウェア [編集]

キーボード入力の代替として手書き文字認識を採用した商用製品は1980年代初めごろに登場した。初期の手書き端末製品としてPenceptPenpad[2]やInforiteのPOS端末[3]がある。パーソナルコンピュータが巨大消費者市場を形成すると、キーボードとマウスの代替としてPenCept[4]、CIC[5]などから手書き文字認識システムが発売された。最初のタブレット型ポータブルコンピュータとして、GRiDSystemsのGRiDPadが1989年9月にリリースされた。これにはMS-DOSをベースとしたオペレーティングシステムが搭載されていた。

1990年代初め、NCR、IBM、EOというメーカーが共同でGO社のPenPointOSを搭載した（広義の）タブレットPCをリリースした。PenPointには手書き文字認識機能やジェスチャー機能があり、サードパーティ製ソフトウェアにもそれらを適用可能だった。IBMは当初ThinkPadのブランド名を使い、独自の手書き文字認識技術を使っていた。この認識システムは後にマイクロ

アップル・ニュートンとiPhone

Figure 2.3 Grouped document objects (Green line)

The recognition model further classifies content-targeting stroke into following two types of content-targeting

1) Per document element
   - Underline
   - Single-line enclosure
2) Per text line
   - Multi-line enclosure
   - Vertical

The proposed model is designed to enable to append recognizable annotation types out of these annotations since there are other types of annotations in some cases.

2.3.2. Document Elements Extraction and Grouping

The proposed model initially obtains document elements from electronic documents. The model uses the size and location of bounding boxes enclosing document elements as input data. The bounding box is extracted to recognize the bounding box of the character's font. The bounding box of a document element which is occurred in the ith element in the target document data, is represented as t_i.

On the other hand, the document elements are grouped into lines after obtaining the bounding box of a document element to handle the recognition of content-targeting per line. Specifically, the model organizes document elements to calculate the connection distance d between the document elements that is adjacent to each other in order of appearance in the document file sequence. d_i represents the distance between the center of the right side in t_{i-1} and the center of the left side in t_i (see Figure 2.4).

The document elements are organized into lines to divide elements if the d is over than the following threshold S

$$S = \widetilde{D} + 2W \tag{1}$$

where $\widetilde{D}$ represents the median of d in $t \in T$, and W represents the average width of $t \in T$. Figure 2.3 shows the example result of line grouping.

2.3.3. Content-targeting Annotation Classification by Shape Features

Content-targeting strokes are classified into three kinds of categories by using features per stroke and decision tree. Following two features are used as shape features.

1) Aspect ratio

$$\frac{H_{stroke}}{W_{stroke}} \tag{2}$$

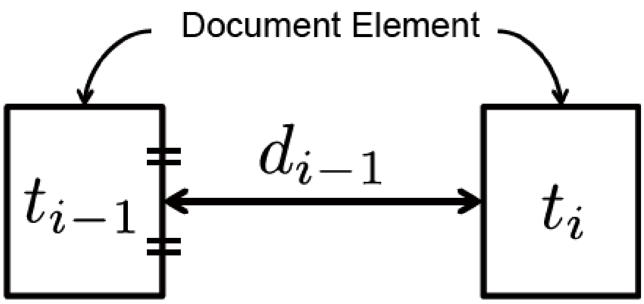


Figure 2.4 Document elements and connection distance

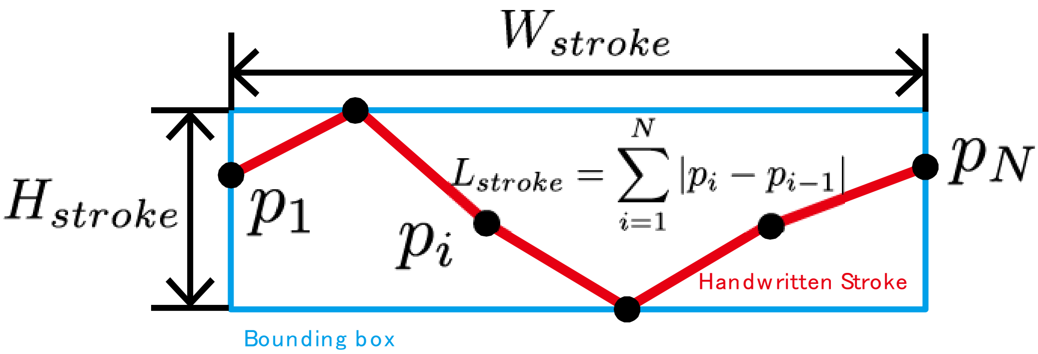


Figure 2.5 Features of a handwriting stroke

2) Density

$$\frac{L_{stroke}}{W_{stroke} + H_{stroke}} \tag{3}$$

W_{stroke} and H_{stroke} represent the width and the height of the bounding box in a handwritten stroke. L_{stroke} represents the length of a handwritten stroke (see Figure 2.5). p_i in Figure 2.5 represents coordinates which is occurred ith in a set $P \ni p_i$ $(1 \leq i \leq N)$. L_{stroke} represents the length of the handwritten strokes calculated by the sum of the distance between p as follows.

$$\sum_{i=2}^{N} |p_i - p_{i-1}| \tag{4}$$

Note that the proposed method assumes all the handwritten strokes as content-targeting in this step although the final output of the proposed method classify annotation strokes into content-targeting and information-appending. The classification of content-targeting and information-appending is described in the next section.

The model classifies handwritten strokes into underline, enclosure and vertical using these features. This step regards single-line enclosure and multi-line enclosure as same group. This is due to difficulty of discrimination among them when using shape features. Thus, these two types of annotation are classified in the next step. The method in this step uses decision tree in "mvpart"[1] package of R to classify handwritten strokes into the kind of content-targeting.

[1] CRAN – Package mvpart, http://cran.r-project.org/web/packages/mvpart/index.html

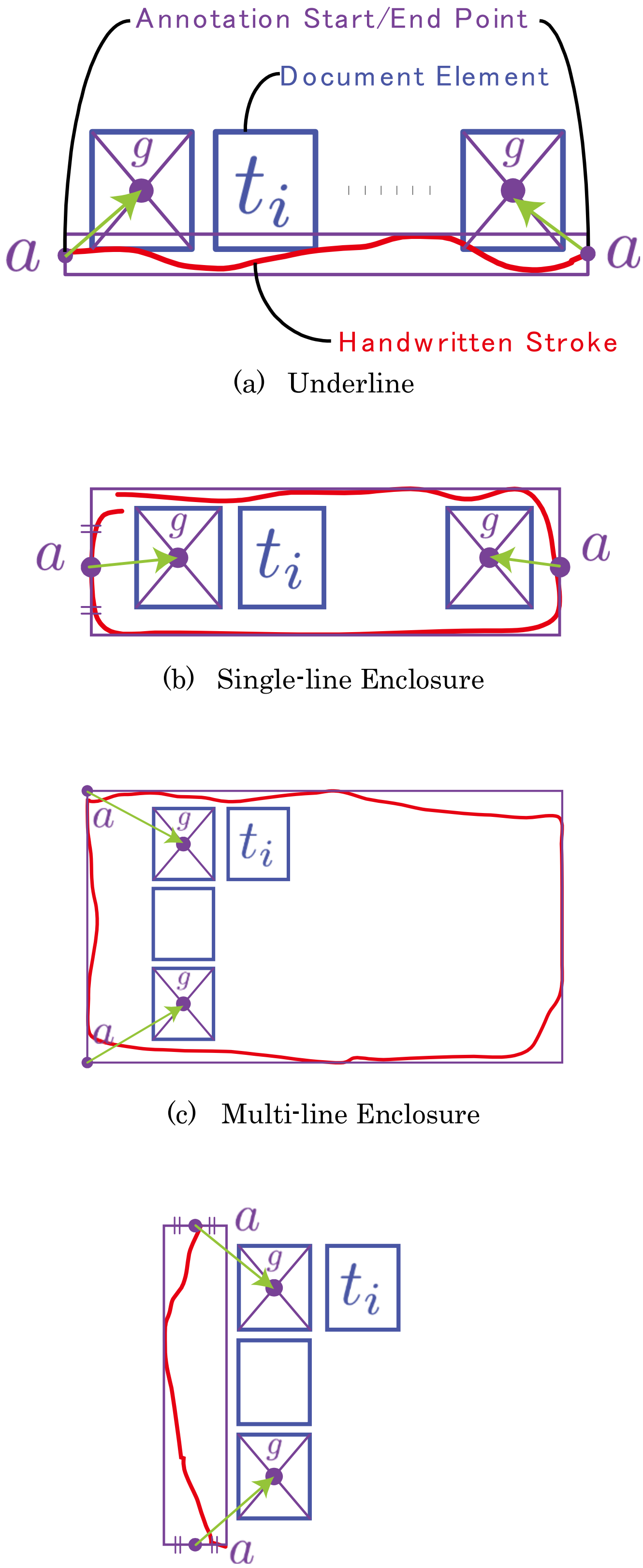
2.3.4. Classification of Content-Targeting and Information-Appending

This step detects content-targeting annotation from all input handwritten strokes using both the assumed types of contents selection obtained in 2.3.3 and the document element information obtained in 0. Proposed model identifies content-targeting stroke using the annotation distance defined in this section. The distance is defined using collected patterns of human annotation based on the assumption that human error follows a normal distribution. The model can accept to the type of content-targeting out of defining in this chapter by defining annotation start/end point described in this section.

The proposed recognition model first estimates the range of selected region by detecting start/end document elements on electronic documents. The method estimates each start and end elements using the relationships between handwritten stroke and document elements to assume all input stroke as content-targeting. This section defines start and end corresponding point to each type of content-targeting annotation described in 2.3.3, then uses this and the relative coordinates calculated by the location to document elements for identifying features. Figure 2.6 describes annotation start and end point, and Table 2.3 represents the definition of annotation start and end point. Note that the bounding box mentioned in Table 2.3 represents the bounding box enclosing a handwriting stroke.

Table 2.3 Definitions of annotation start/end point

Types of Content-targeting	Start Point a	End Point a
Underline	The center point on left side of the bounding box	The center point on right side of the bounding box
Single-line Enclosure	The center point on left side of the bounding box	The center point on right side of the bounding box
Vertical	The center point on upper side of the bounding box	The center point on lower side of the bounding box
Multi-line Enclosure	The top-left point of the bounding box	The bottom-right point of the bounding box



(a) Underline

(b) Single-line Enclosure

(c) Multi-line Enclosure

(d) Vertical

Figure 2.6 Document elements and annotation start/end point

In the beginning, the proposed model calculates the relative point from annotation start/end point to the gravity point of the bounding box in document elements according to the assumption of content-targeting categories. Note that the width and the height of the bounding box in document elements are normalized to 1 to prevent from varying the relative point by changing the size of the document elements. This can be represented as the following equation.

$$p_{rc} = \left(\frac{g_x - a_x}{W_{obj}}, \frac{g_y - a_y}{H_{obj}}\right) \tag{5}$$

g represents the gravity point in t. a represents the annotation start/end point defined in Table 2.3. W_{obj} and H_{obj} represents the width and the height of t.

Next, the model identifies content-targeting strokes by using the normal distribution of the relative point. Specifically, the method determines whether the handwritten stroke represents content-targeting or not by using defined distance from the center of normal distribution in the relative points to annotation start/end point of handwritten stroke. This section calls the distance as Annotation Distance (AD). AD is defined based on Mahalanobis distance as follows:

$$AD(a,t) = \frac{1}{1-\rho^2}\left(\frac{\left(p_{rc_x} - \mu_x\right)^2}{\sigma_x^2} + \frac{\left(p_{rc_y} - \mu_y\right)^2}{\sigma_y^2} - \frac{2\rho\left(\left(p_{rc_x} - \mu_x\right)\left(p_{rc_y} - \mu_y\right)\right)}{\sigma_x \sigma_y}\right) \tag{6}$$

ρ, σ_x^2, σ_y^2, μ_x, μ_y represent a correlation coefficient, variances and averages in bivariate normal distribution model.

The method calculates AD in annotation start/end points each document elements, and then regard input stroke as content-targeting if the both minimum AD values are within 99% of confidence interval in the normal distribution. The threshold value of AD (called AD_{thres}) is calculated as follows since the square value of the Mahalanobis distance in the bivariate normal distribution follows the chi-square distribution with degrees of freedom 2.

$$AD_{thres} = \chi^2(2{,}0.01) \sim 9.2103 \tag{7}$$

The method detects content-targeting annotation by the above process. Furthermore, the handwritten stroke which is classified as enclosure in 2.3.3 calculates two types of AD; single-line enclosure and multi-line enclosure, then decides the category of content-targeting based on the following conditions in Table 2.4.

Table 2.4 Conditions and results if strokes are classified as enclosure

Conditions	Results
Both single-line and multi-line enclosure assumption are detected as content-targeting	The stroke is recognized as multi-line enclosure, and then the method finally decides whether the stroke represents single line or not (see 2.3.5).
Either single-line or multi-line enclosure assumption are detected as content-targeting	The stroke is recognized as the detected type.
Neither single-line and multi-line enclosure assumption are detected as content-targeting	The stroke is recognized as information-appending.

2.3.5. Selected Range Estimation by Content-Targeting

The selected range of contents is estimated from the handwritten strokes that are recognized as content-targeting. The document element t_{target} which indicates the minimum AD value is calculated as follows.

$$t_{target} = \arg\min_{t} AD(a, t) \qquad (8)$$

The start/end document elements of selected by content-targeting annotation are calculated from the equation (8). Incidentally, the single-line enclosure was recognized as multi-line enclosure at 2.3.3. It is recognized as actual single-line enclosure if the recognized start and end document elements exist in same line in this step, and then the selected range of the documents is estimated again.

According to the above methods, computers can recognize handwritten annotation information.

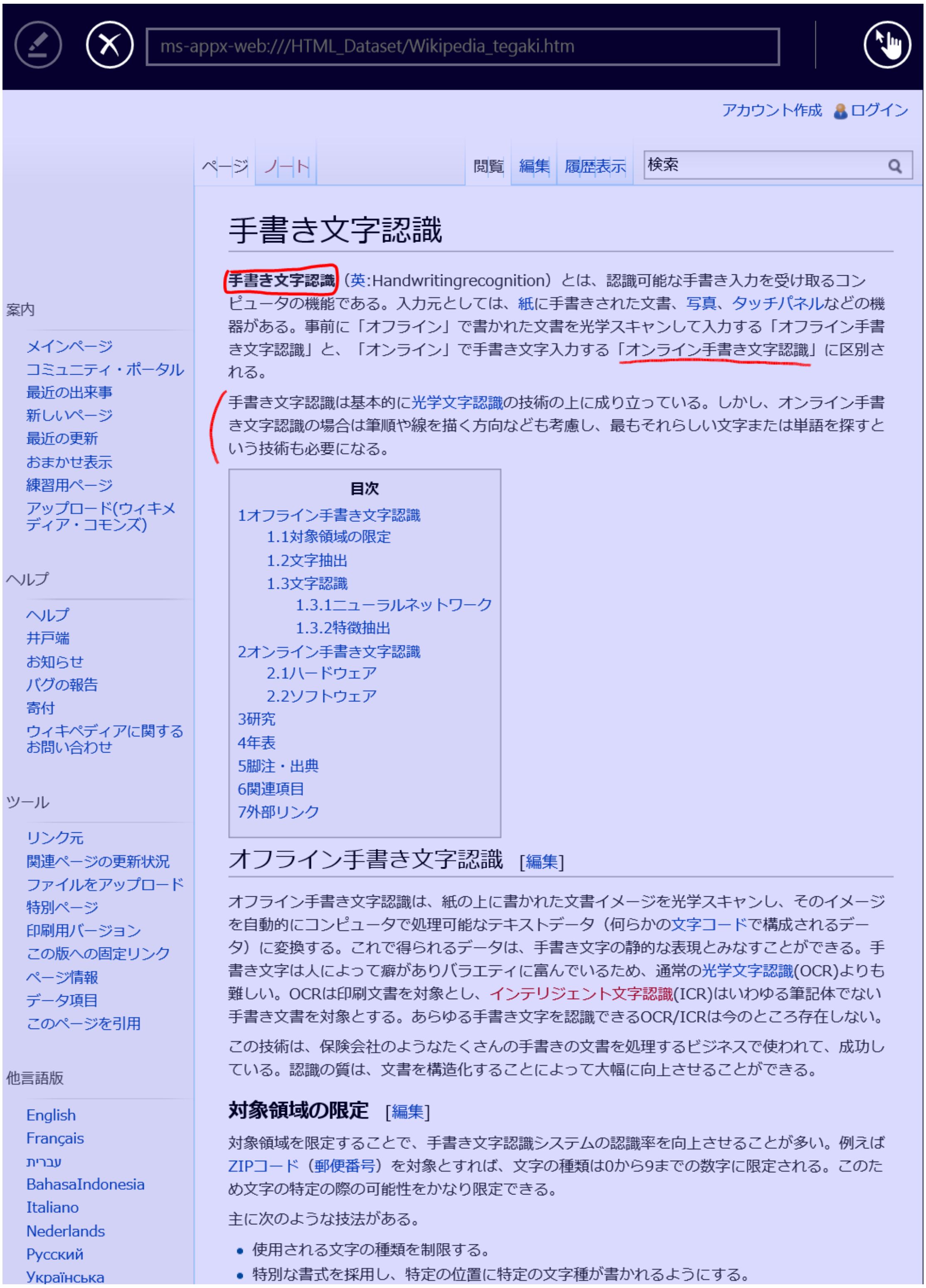
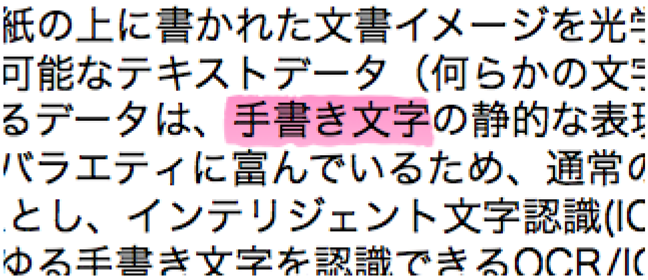
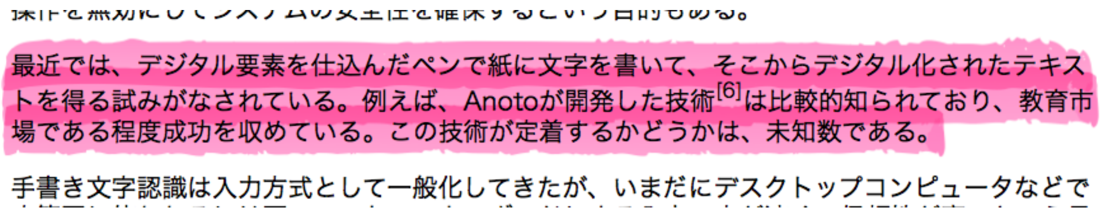


Figure 2.7 A screen capture of developed experiment system



(a) Single line



(b) Multi line

Figure 2.8 Example of indicated annotation range

2.4. Evaluation

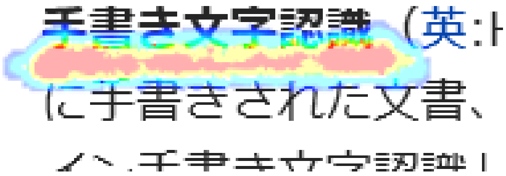
This section describes the evaluation experiment on the following point:

- accuracy of selected range of contents by leave-one -out cross validation in a participant unit,
- accuracy comparison with traditional heuristic method [30]
- recognition accuracy of the data that information-appending and content-targeting stroke are mixed.

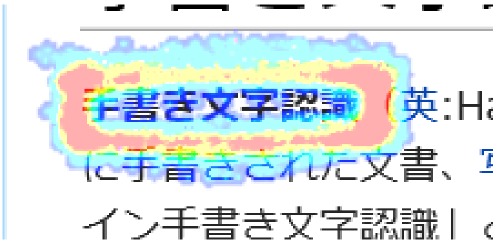
2.4.1. Evaluation Environment

The system collecting handwritten annotation is developed to conduct the evaluation. The device using in the evaluation is Sony VAIO Duo 11[1]. The software developing environment is Windows Store App in Windows 8.1. Figure 2.7 shows the screen capture of the developed collecting system. Participants can perform handwriting annotation on the HTML documents by using the system. Navigation mode and writing mode are implemented in the system. The navigation mode accepts scrolling documents by touching the display. On the other hand, the writing mode accepts handwriting annotation on contents with a non-scrollable. This mode only accepts handwriting and erasing.

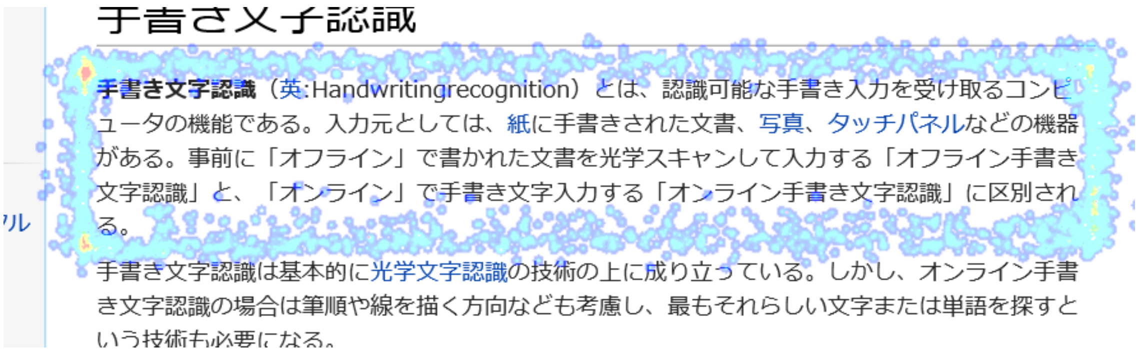
[1] Sony VAIO Duo 11, "http://www.sony.jp/vaio/products/VD21/."



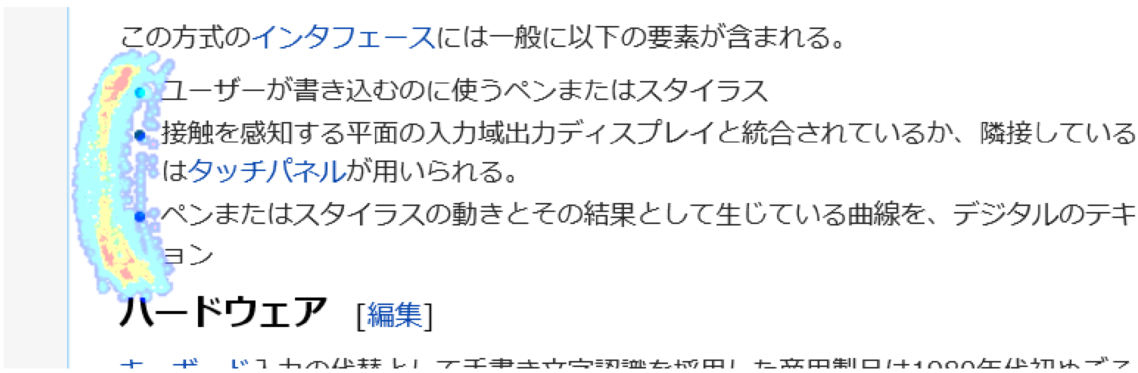
(a) Underline



(b) Single-line enclosure



(c) Multi-line enclosure



(d) Vertical

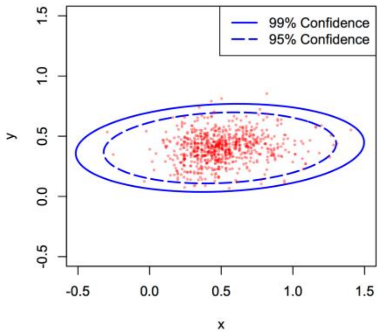
Figure 2.9 Heatmaps of handwritten annotation coordinates

The experiment provided the collecting system and the HTML documents printed on paper to participants. The collecting system used the fonts consisted by a proportional font in alphabets and a moonscape font in Japanese characters. The HTML documents printed on paper included some highlighted range of text (see Figure 2.8), and then participants selected the text highlighted on the paper-based documents by using a stylus on the collecting system. The size of the font highlighted on the paper was either 15.06 pixels or 13.26 pixels. Moreover, participants were orally instructed to the kinds of annotation such as underline, enclosure and vertical, and confirmed the kinds of annotation by reading the documents presenting the examples of annotation as well as instructed the method using the collecting system. Participants are also instructed to write content-targeting annotation by single stroke.

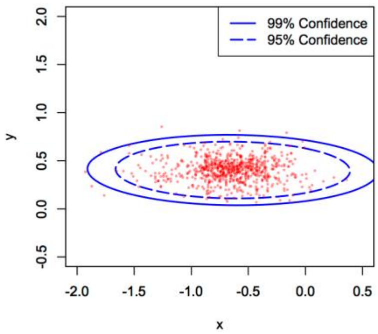
2.4.2. Parameter Estimation

This section describes the estimation of the proposed model parameters. In the beginning, the data of content-targeting annotation using the parameter estimation was collected from the participants. 26 students (men: 19, women: 7) belonging to the Waseda university are invited to the experiment. The collecting experiment distributed the paper-based documents highlighted the parts of the text to participants, and instructed them to perform specific content-targeting annotation on the electronic documents indicated by the text highlighted on the paper-based documents. The number of the places of content-targeting annotation is 10. Participants performed content-targeting annotation 3 times per the one place. The kinds of annotation are underline, single-line enclosure, multi-line enclosure and vertical. As a consequence, the experiment collected 10 * 3 * 4 = 120 content-targeting annotation per a participant.

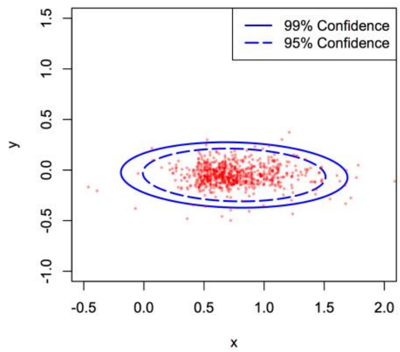
The damaged data due to the defect of the collecting system was excluded in the parameter estimation. In addition, the annotation data which was annotated on the clearly different position from the instruction was also excluded. In other words, the collected annotation data was excluded if the AD from the annotation start and end point is over 100. The excluded data is 2.2 % of the total collected data (the defect of the collecting system: 1.3%, the incorrect annotation: 0.9%). The number of annotations using the experiment is 775 strokes in underline, 772 strokes in single-line enclosure, 732 strokes in multi-line enclosure and 772 strokes in vertical. Table 2.5 shows the summary of collected data.



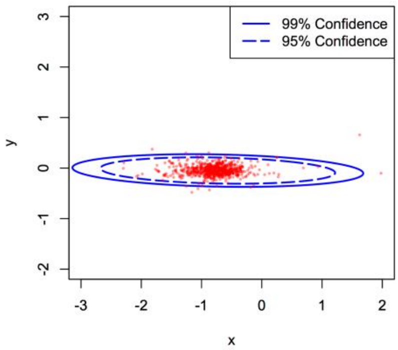
(a) Undeline (start)



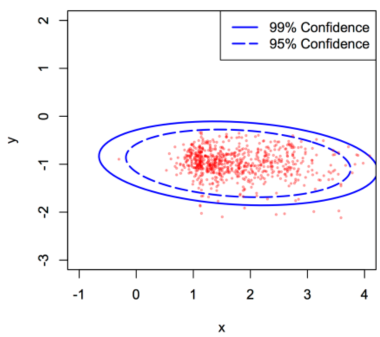
(b) Underline (end)



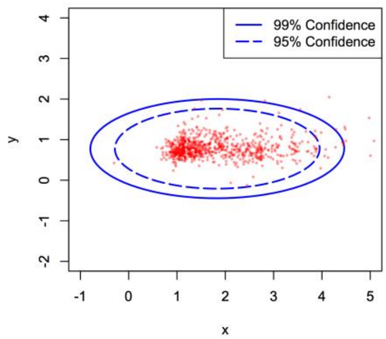
(c) Single-line enclosure (start)



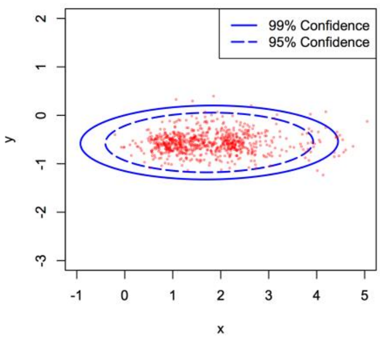
(d) Single-line enclosure (end)



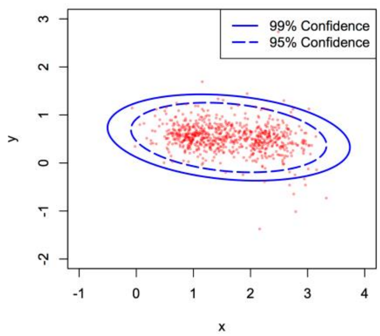
(e) Multi-line enclosure (start)



(f) Multi-line enclosure (end)



(g) Vertical (start)



(h) Vertical (end)

Figure 2.10 Distributions of annotation relative coordinates

Table 2.5 Number of collected data in parameter estimation

Descriptions	Numbers
Annotation places	10
Annotation types	4
Repeat	3
Participants	26
Removed	69
Total	3051

The parameter estimation of proposed model and the evaluation of the recognition accuracy were conducted using the collected annotation data mentioned above. Figure 2.9 shows heat maps of content-targeting annotation coordinates as one of the collected example. The parameters of proposed model were calculated by fitting to the normal distributions using these collected data. Table 2.6 shows the parameters of the model calculated by collected data. In addition, Figure 2.10 shows the relative coordinates of collected data and its normal distributions. Note that these parameters are normalized by the size of the document element in 2.3.4 not to depend on the size of characters.

Table 2.6 Model parameters calculated by collected data

		ρ	σ_x^2	σ_y^2	μ_x	μ_y
Underline	Start	−0.124	0.110	0.015	0.491	−0.403
	End	0.036	0.176	0.015	−0.637	−0.403
Single-line enclosure	Start	0.077	0.097	0.011	0.751	0.049
	End	0.173	0.633	0.011	−0.729	0.049
Vertical	Start	−0.028	0.783	0.063	1.761	0.561
	End	0.227	0.489	0.088	1.619	−0.530
Multi-line enclosure	Start	0.171	0.646	0.083	1.787	0.982
	End	−0.004	0.750	0.162	1.834	−0.778

2.4.3. Recognition Accuracy of Content-targeting

In addition to the parameter estimation, this section describes the evaluation of the recognition accuracy of proposed model using the collected data in 2.4.2. In the beginning, this section describes the classification accuracy of content-targeting annotation using shape features described in 2.3.3. Figure 2.11 shows the visualization of

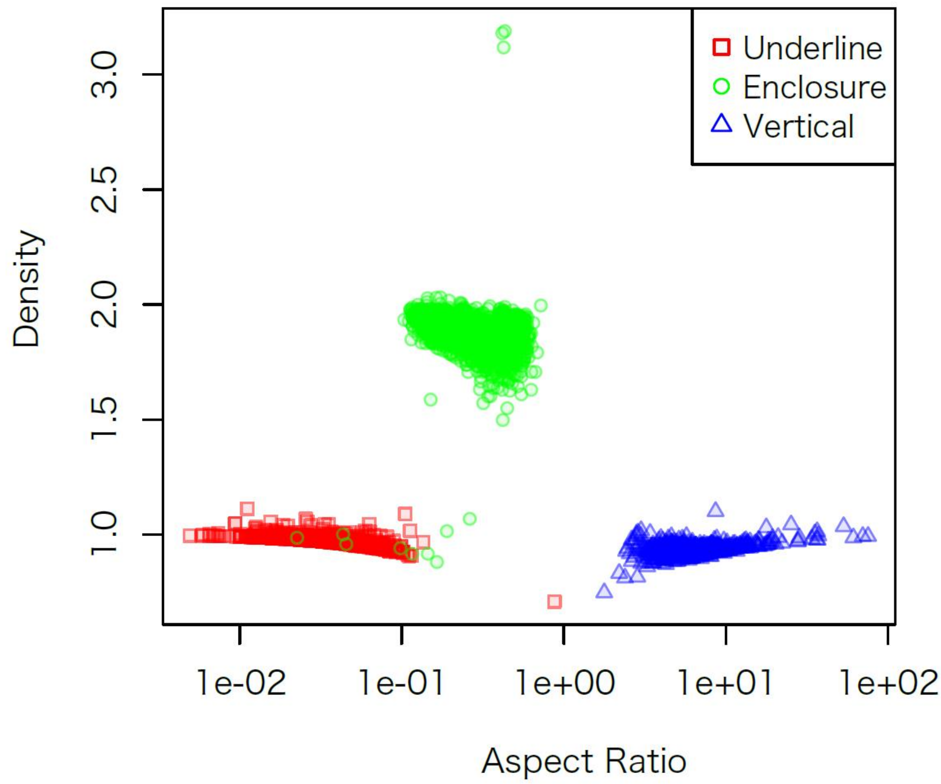


Figure 2.11 Distribution of features used in annotation classification

shape features collected from participants. The figure represents that these shape features, which are the aspect ratio and density, are effective for the classification. The estimated threshold values of the decision tree are as follows:

- Threshold of density: 1.3050
- Threshold of aspect ratio: 1.3263

Table 2.7 shows the results of the classification. The table represents that proposed model can classify annotation strokes into three types of content-targeting annotations: underline, vertical and enclosure.

Table 2.7 Annotation classification result by decision tree

		Collected data		
		Underline	**Vertical**	**Enclosure**
Classification results	**Underline**	775	0	9
	Vertical	0	772	0
	Enclosure	0	0	1495

In the next, the experiments evaluated the recognition accuracy of the selected range on contents described in 2.3.5. The evaluation applied leave-one-subject-out (LOSO) cross validation to calculate recognition accuracy without personalization. Table 2.8 shows the recognition accuracy of selected range calculated by LOSO cross validation where the accuracy is calculated by using the model parameters based on other user's annotation data.

Table 2.8 Recognition accuracy of selected range (selecting region annotation only)

		No deviation	Less than one letter deviation	No deviation (both)
Underline	**Start**	95.38%	98.72%	85.37% (STD: 0.1027)
	End	88.45%	97.95%	
Single-line enclosure	**Start**	88.56%	93.57%	77.25% (STD: 0.1978)
	End	80.46%	92.42%	
Vertical	**Start**	94.48%	96.41%	91.14% (STD: 0.0947)
	End	93.20%	96.41%	
Multi-line enclosure	**Start**	92.46%	95.71%	91.29% (STD: 0.1454)
	End	94.02%	95.58%	

The table represents that the proposed recognition model can recognize the range of contents indicated by multi-line enclosure and vertical, which select the range of the contents on a line-by-line basis, with no deviation in over 91% of accuracy. In contrast, the recognition accuracy of the selected range of contents indicated by underline and single-line enclosure, which select the range on a character-by-character basis, decreases the recognition accuracy by 10-15% compare with that of the line-by-line basis. In addition, there is the difference of the recognition accuracy in over 10% between the selecting the range of the contents on a line-by-line basis and a character-by-character basis. In other words, the results indicate that the recognition accuracy of the proposed model enables to improve the accuracy if the one letter deviation is modified.

In this manner, the failures of the recognition are mainly caused by the one letter deviation. The other failures of the recognition are the content-targeting annotations that include much margin. The examples are the underline which the handwritten stroke is written to margin beyond the end of the text line, the underline which is far from the target text, the enclosure which includes much margin and also the vertical which is far from the target text. Such annotations including much margin are caused if there is much margin around the target contents.

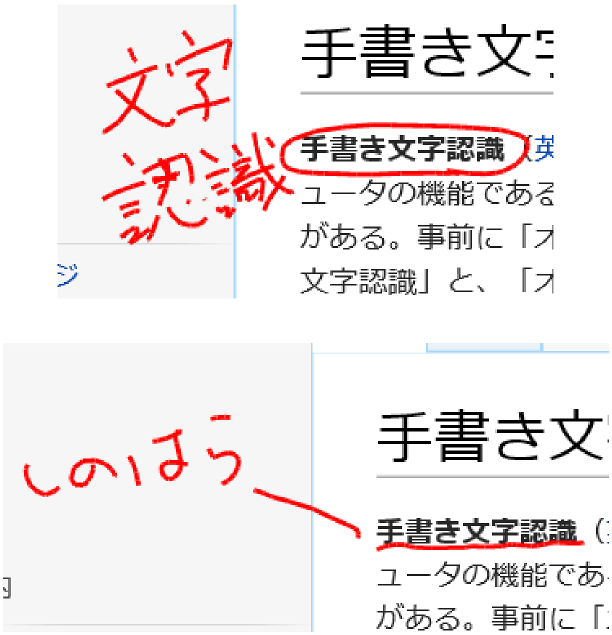


Figure 2.12 Examples of content-targeting and information-appending strokes written by participants.

2.4.4. Recognition Accuracy of Classification

In addition to the evaluation of the recognition accuracy using the collected data including content-targeting annotation only, this section evaluated the recognition accuracy using the data including both content-targeting and information-appending annotation. The number of participants in this data collection was ten students in Waseda University. The six students in the participants were also joining the data collection described in 2.4.2. Participants were instructed to append information on each selected annotation. There was no limitation of writing comments. The evaluation collected the ten locations of the handwritten annotation in each four kinds of content-targeting annotation, that is, 4 * 10 = 40 content-targeting annotation and its appended information per a participant were collected. Figure 2.12 shows the examples of information-appending annotation collected in the experiments.

Table 2.9 shows the results of classification between content-targeting and information-appending. The misclassified data recognized as content-targeting tends to close to document elements and intersect a bounding box of a document element.

Table 2.9 Classification result between content-targeting and information-appending

		Collected data	
		Content-targeting	Information-appending
Classification results	Content-targeting	383	752
	Information-appending	17	5115

Furthermore, Table 2.10 shows the recognition accuracy of selected range using the collected comment-mixed data by LOSO cross validation. Although the results show that all of the recognition accuracy except multi-line enclosure tends to decrease a little in comparison with Table 2.8, the result is almost same in Table 2.8.

Table 2.10 Recognition accuracy of selected range (comment-mixed data)

		No deviation	Less than one letter deviation	No deviation (both)
Underline	Start	91.49%	93.62%	72.34% (STD: 0.1852)
	End	75.53%	91.49%	
Single-line enclosure	Start	92.00%	96.00%	68.00% (STD: 0.2402)
	End	70.67%	90.67%	
Vertical	Start	90.63%	93.75%	82.29% (STD: 0.2008)
	End	85.42%	93.75%	
Multi-line enclosure	Start	95.96%	100.0%	92.93% (STD: 0.0781)
	end	95.96%	95.96%	

2.5. Conclusion

This chapter described the recognition of handwriting annotation on electronic documents. The recognition is important to improve the availability of annotated documents such as information retrieval. However, traditional heuristic recognition model is unsuitable for the accurate recognition required to such applications.

The chapter therefore proposed the handwriting annotation recognition model that can recognize the exact selected range on the contents by learning collected handwritten annotation data. The experiments in which four typical types of content-targeting are applied to proposed model showed that the model can robustly recognize the annotation not depending on users. The proposed model can estimate selected region for 70% on average in selection of characters and for 88% in the selection of text lines. The rest of this part describes the applications using the proposed models.

Chapter 3. Intelligent Ink Annotation Framework

3.1. Introduction

Annotation when reading a document significantly helps readers to understand its contents and enhance accessibility [36]. Users annotate paper-based documents by highlighting content and proving comments. However, over the past several years, with the development of hardware that accepts stylus and touch input, researchers have studied active reading systems that enable performing such annotation on electronic documents as well [37] [25].

In comparison to paper-based annotation, softcopy annotation systems for electronic documents allow for efficient functions like information retrieval and interactive navigation [37] [20]. They require users to perform gestures defined by themselves to obtain annotation information in addition to common behaviors of paper-based documents, e.g., a non-dominant-hand posture [38] and Pen + Touch interaction [14]. However, this reduces the learnability of such a system.

Thus, recognizing the need to support such systems, this chapter propose an intelligent ink annotation framework that increases the learnability of an annotation system by detecting users' intentions from natural annotation behavior for paper-based documents based on the model proposed in Chapter 2. This chapter focuses on the fundamental tasks of annotation, which are reading, targeting content and commenting.

This study used our framework to implement a prototype annotation system. After that, this chapter describes user study wherein participants annotated electronic documents using our prototype. Finally, the results of the study are analyzed to identify future direction.

3.2. Implementation of Intelligent Ink Annotation Framework

Figure 3.1 represents an overview of our proposed framework. Our framework recognizes user's intentions from natural annotation behavior. By using detected user's intention, annotation systems can support user's annotation depending on the detected intention to use the annotation information for information retrieval or annotation-based navigation.

In proposed framework, both an original document source and a user's handwritten stroke are required to detect user's intention of annotation. Users can get a feedback from the framework when a handwriting stroke has been finished, i.e., pen away from a display. The output of the framework has three types of user's intentions

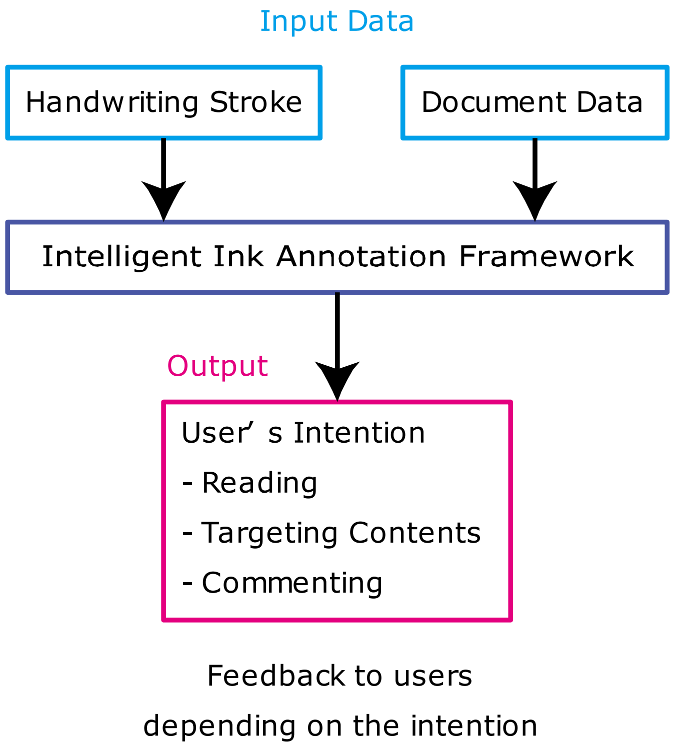


Figure 3.1 Intelligent ink annotation framework

required to annotation systems.

1) **Reading**
   This state includes reading or navigating a document not using a pen.
2) **Targeting Content**
   This task includes highlighting a range of content in a document, content-targeting to provide comments on specific content and extracting search terms.
3) **Commenting**
   This involves providing comments on specific content or in blank spaces of documents.

To detect these types of user's intentions and recognize selected content in "Targeting Content" intention, our framework uses the recognition model of handwriting annotation proposed in Chapter 2. The input strokes are classified into four types of "Targeting Content" annotation. Then the framework determines user's intention by analyzing positional relationship between the handwriting stroke and document content objects. In addition, the range of the targeted contents in a document is recognized when the framework detects user's intention as "Targeting Content." The detail of each function is described in the following sections.

3.2.1. User's Intentions based on the Annotation Lifecycle

Our framework detects user's intentions of annotation based on the Annotation Lifecycle which is defined based on natural annotation behaviors on paper-based documents. Marshall investigates annotation on paper-based university-level textbooks by students [18]. He reported telegraphic annotations such as underline and explicit annotations, i.e., brief notes written between lines, were found within text field. On the other hand, brackets and extended notes were found in marginal or blank space. Proposed framework classifies these reported annotations into two major classes based on the aspect of user's intention required to extract annotation information. One is tar-

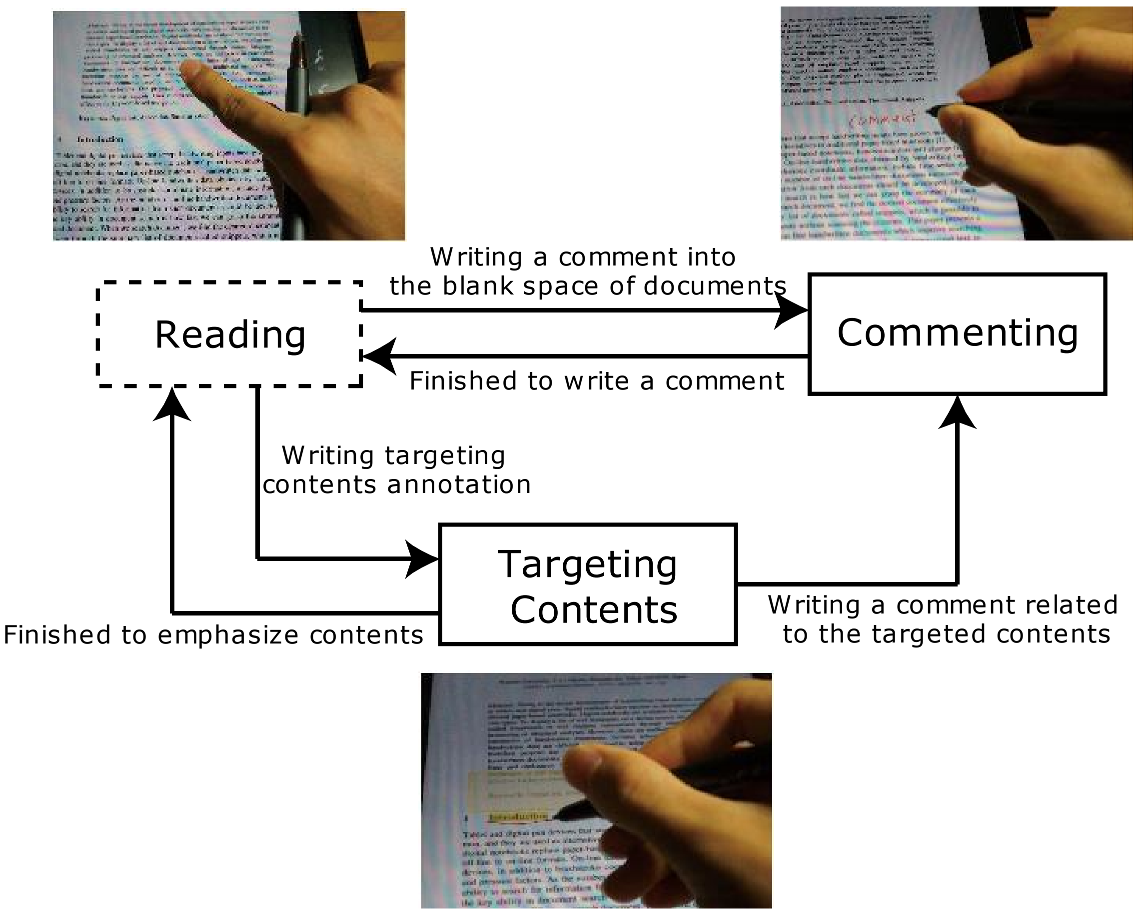


Figure 3.2 Annotation Lifecycle for detecting user's intentions of annotation

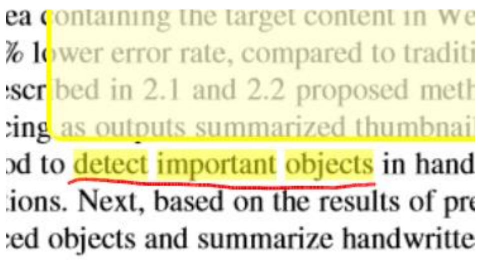
geting content, and the other is commenting.

Figure 3.2 shows the annotation lifecycle defined for detecting user's intentions of annotation. When users read or navigate documents, our framework recognizes the state as "Reading." When users write targeting annotation like underline in this state, the state is changed to "Targeting Content." On the other hand, the state is changed to "Commenting" when users write a comment into the blank space of documents. In "Targeting Content" state, the state is moved to "Reading" when users just select contents and write no comments, e.g., emphasizing contents. In contrast, the state is moved to "Commenting" when users intend to add a comment involved in the selected contents.

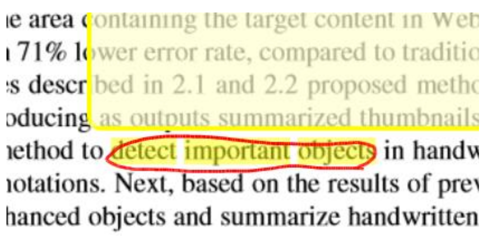
Coordinates of a handwriting stroke and bounding boxes of characters in documents are used to detect each state of the Annotation Lifecycle. Our framework recognizes the state as "Reading" when handwriting stroke is not entered. On the other hand, a handwriting stroke is assumed as "Targeting Content," and classified into the kind of targeting content annotation (see Figure 3.3) by decision tree to use aspect ratio of the bounding box and stroke density features when handwriting stroke is entered. Then, the framework determines the assumption is correct or not, e.g., "Targeting Content" or "Commenting," by comparing the position of handwriting stroke with the bounding boxes of characters in documents. Our collected handwriting annotation data, which is collected from several participants before implementing our system, is used for the each process.

3.2.2. Targeting Content Recognition

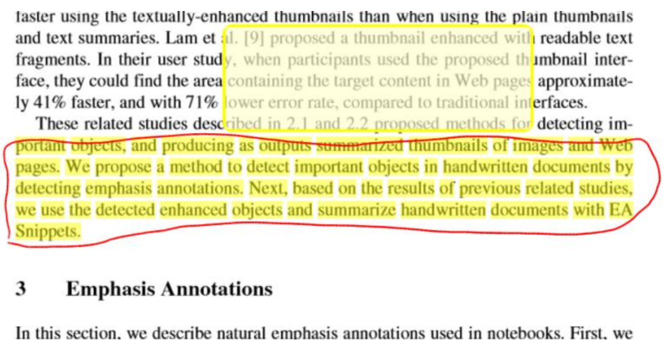
Marshall reported several types of annotations for targeting content; such as underline, circles (called "enclosure"), brackets (called "vertical") for the typical paper-based annotation [18]. Moreover, Golovchinsky et al. categorized such annotations into two categories: a part of the text within a line and selecting several lines [30]. According to these reports, our framework detects four types of targeting annotations (see Figure 3.3). Underline and enclosure are detected as "Targeting Content" which specifies the range of characters. On the other hand, multiline-enclosure and vertical are detected as "Targeting Content" which specify the range of text lines. These ranges are recognized by using coordinates of handwriting stroke, bounding boxes of characters in document and our collected annotation data set.


ea containing the target content in We
% lower error rate, compared to traditi
escribed in 2.1 and 2.2 proposed meth
cing as outputs summarized thumbnai
od to detect important objects in hand
ions. Next, based on the results of pre
ced objects and summarize handwritte

(e) Underline


e area containing the target content in Web
1 71% lower error rate, compared to traditio
s described in 2.1 and 2.2 proposed metho
oducing as outputs summarized thumbnails
ethod to detect important objects in handw
otations. Next, based on the results of prev
hanced objects and summarize handwritten

(f) Enclosure

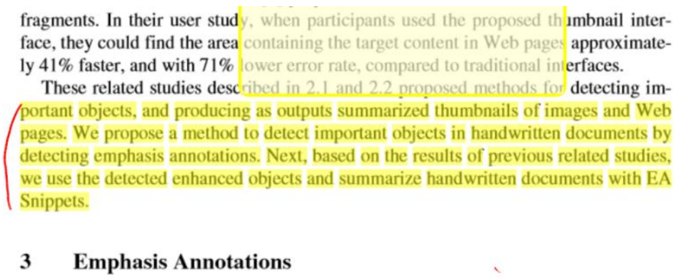

faster using the textually-enhanced thumbnails than when using the plain thumbnails and text summaries. Lam et al. [9] proposed a thumbnail enhanced with readable text fragments. In their user study, when participants used the proposed thumbnail interface, they could find the area containing the target content in Web pages approximately 41% faster, and with 71% lower error rate, compared to traditional interfaces.

These related studies described in 2.1 and 2.2 proposed methods for detecting important objects, and producing as outputs summarized thumbnails of images and Web pages. We propose a method to detect important objects in handwritten documents by detecting emphasis annotations. Next, based on the results of previous related studies, we use the detected enhanced objects and summarize handwritten documents with EA Snippets.

3 Emphasis Annotations

In this section, we describe natural emphasis annotations used in notebooks. First, we

(g) Multiline-Enclosure


fragments. In their user study, when participants used the proposed thumbnail interface, they could find the area containing the target content in Web pages approximately 41% faster, and with 71% lower error rate, compared to traditional interfaces.

These related studies described in 2.1 and 2.2 proposed methods for detecting important objects, and producing as outputs summarized thumbnails of images and Web pages. We propose a method to detect important objects in handwritten documents by detecting emphasis annotations. Next, based on the results of previous related studies, we use the detected enhanced objects and summarize handwritten documents with EA Snippets.

3 Emphasis Annotations

(h) Vertical

Figure 3.3 Targeting content annotation

3.3. Prototype Implementation

A prototype annotation system have been developed using our proposed framework running on Sony VAIO Duo 11. The resolution of the display is 1920*1080, and the display accepts both multi-touch input and stylus input. The software development environment is Windows store application on Windows8.1 (see Figure 3.4)

The prototype system shows a single page of PDF contents on the screen. Users can change pages by swiping finger to left or right on the screen. In addition, the system accepts handwriting by stylus input. Users can annotate documents by handwriting in the following methods.

3.3.1. Intelligent Ink Annotation Framework Mode

The prototype system implemented two types of annotation mode. One is automatic operation mode using our proposed framework. In this mode, the selected contents are highlighted followed by opening a comment window to add comments, when the system detects "Targeting Content." When the system detects "Commenting," the system accepts writing with no other feedbacks (see Figure 3.4).

3.3.2. Manual Operation Mode

On the other hand, the system also implemented traditional manual operation mode not using proposed framework to compare with proposed framework in the user study. In this mode, users can switch between two types of functions by selecting buttons on the screen. One is selecting function which can only accept selecting range of the contents. User can select contents by dragging pen just like selecting text by mouse (see Figure 3.5). The other function is commenting mode which can only accepts writing comments.

3.4. User Study

User study was conducted to compare usability between proposed intelligent ink annotation framework and traditional manual operation. Four participants (A to D) who are belonging to our laboratory were invited to our user study. They are bachelor and master course students majoring in computer science. The profiles of participants are shown in Figure 3.4.

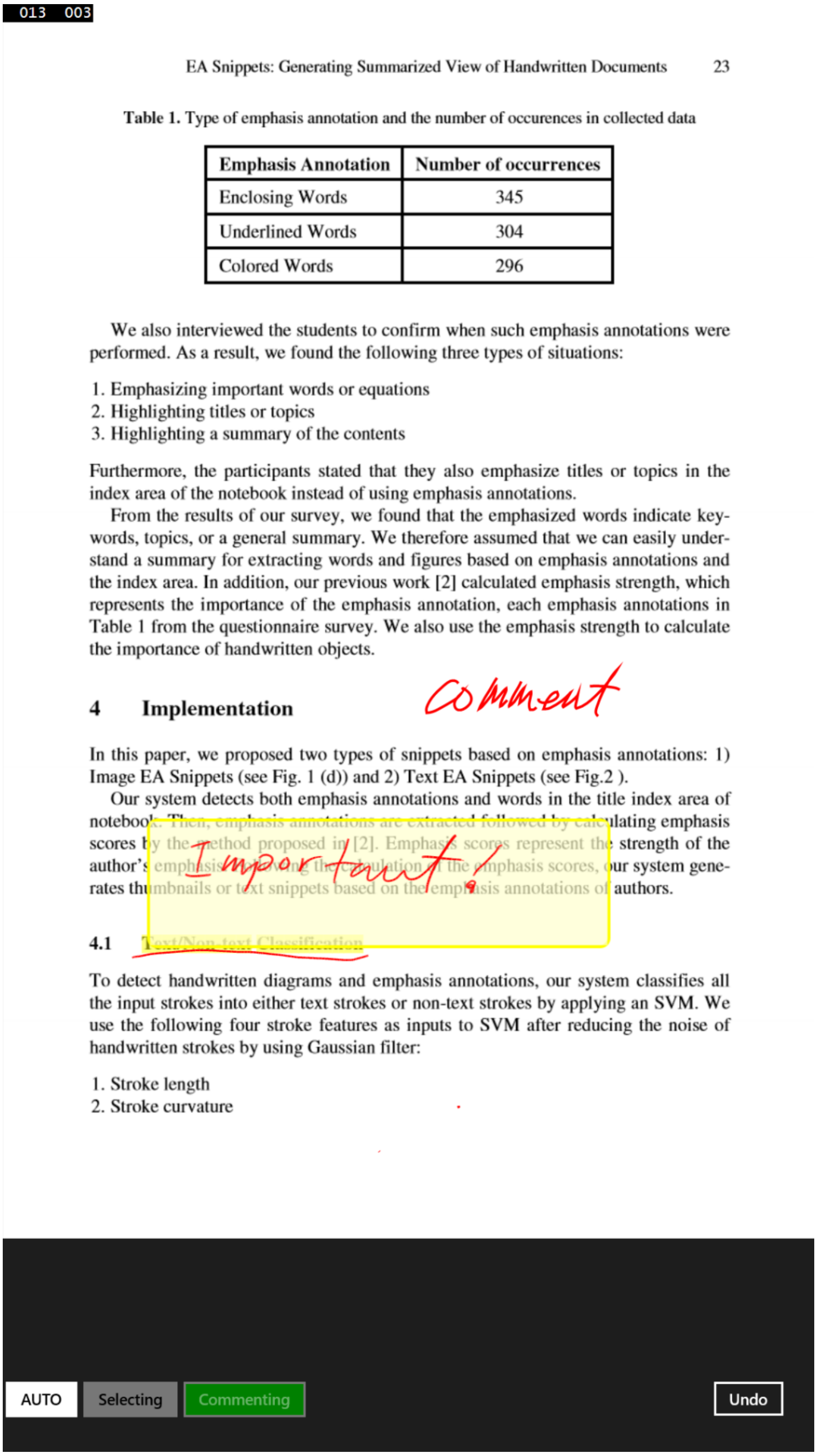


Table 1. Type of emphasis annotation and the number of occurences in collected data

Emphasis Annotation	Number of occurrences
Enclosing Words	345
Underlined Words	304
Colored Words	296

We also interviewed the students to confirm when such emphasis annotations were performed. As a result, we found the following three types of situations:

1. Emphasizing important words or equations
2. Highlighting titles or topics
3. Highlighting a summary of the contents

Furthermore, the participants stated that they also emphasize titles or topics in the index area of the notebook instead of using emphasis annotations.

From the results of our survey, we found that the emphasized words indicate keywords, topics, or a general summary. We therefore assumed that we can easily understand a summary for extracting words and figures based on emphasis annotations and the index area. In addition, our previous work [2] calculated emphasis strength, which represents the importance of the emphasis annotation, each emphasis annotations in Table 1 from the questionnaire survey. We also use the emphasis strength to calculate the importance of handwritten objects.

4 Implementation

Comment

In this paper, we proposed two types of snippets based on emphasis annotations: 1) Image EA Snippets (see Fig. 1 (d)) and 2) Text EA Snippets (see Fig.2).

Our system detects both emphasis annotations and words in the title index area of notebook. Then, emphasis annotations are extracted followed by calculating emphasis scores by the method proposed in [2]. Emphasis scores represent the strength of the author's emphasis. Following the calculation of the emphasis scores, our system generates thumbnails or text snippets based on the emphasis annotations of authors.

Important!

4.1 Text/Non-text Classification

To detect handwritten diagrams and emphasis annotations, our system classifies all the input strokes into either text strokes or non-text strokes by applying an SVM. We use the following four stroke features as inputs to SVM after reducing the noise of handwritten strokes by using Gaussian filter:

1. Stroke length
2. Stroke curvature

Figure 3.4 Prototype annotation system

Table 3.1 Profiles of participants

ID	A	B	C	D
Age	20s	20s	20s	30s
Sex	Male	Male	Male	Male
Nationality	Japan	Japan	Japan	Korea
Handedness	Right	Right	Right	Right

This study firstly introduced our prototype system in five minutes. After introduction, participants were instructed to read a research paper using our proposed framework at least 10 minutes. Then participants read the paper using manual operation mode not using the framework at least 10 minutes. The annotation behavior of participants was recorded by a video camera. After finishing these procedures, we discussed the prototype system with participants.

This study could obtain some findings from the comments of participants and the recorded videos. Three out of four participants (B, C and D) preferred the automatic mode using our framework because the commenting and targeting functions were switched automatically. In contrast, participant A preferred the manual operation mode not using the framework because the intention recognition accuracy was too low to use. Besides, the content-targeting by dragging stylus was acceptable. All the participants mentioned the recognition accuracy. Accordingly, this study found the recognition accuracy of user's intention is important factor whether the system is acceptable or not. Furthermore, participant B in addition to participant A said content-targeting by dragging stylus, which was implemented in manual operation mode, is useful. The participants also said targeting content gesture by dragging stylus should be also recognized as targeting content of proposed framework.

Moreover, participants A, B and C mentioned that the no system feedbacks while in writing "Targeting Content" was uncomfortable. The prototype system displays the highlight feedback only when the handwriting stroke is finished, i.e., pen up. For this reason, it is conceivable that our framework makes user uncomfortable when recognizing "Targeting Content" followed by no feedbacks.

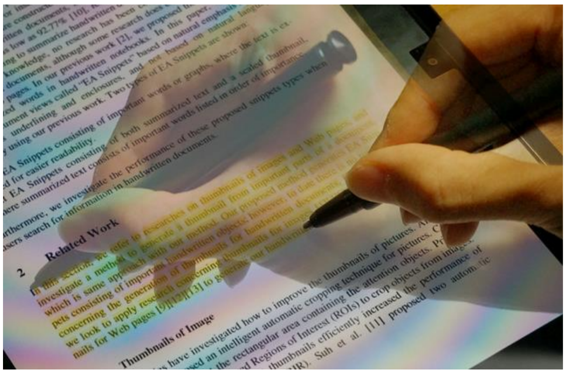


Figure 3.5 Targeting content by dragging a stylus

To summarize this user study, the result of user's evaluation and the future works to improve proposed framework are as follows:

- 3 out of 4 participants preferred the interface using proposed framework
- Finding points to improve proposed framework
  - Recognition accuracy

    The recognition accuracy of selected range on contents should be improved to reduce the frustration of users.
  - Visual feedback

    The visual feedback while writing "Targeting Content" can improve usability of proposed framework.

3.5. Conclusion

This chapter proposed intelligent ink annotation framework that helps handwriting annotation on electronic documents by using the recognition model described in Chapter 2. Our user study showed that our framework have a potential to reduce the users' burden of providing a system to annotation information. In addition, the study reveals two problems. One is our framework is required to the high-accuracy annotation recognition algorithm. Other is our framework needs to be equipped with an interactive feedback mechanism when it recognizes "Targeting Content."

Chapter 4. EA Snippets: A Summary of Handwritten Notebooks based on Emphasis Annotations

4.1. Introduction

Tablet and digital-pen devices that accept handwriting inputs have grown more common, and they are used as alternatives to traditional paper-based notebooks [39]. When digital notebooks replace paper-based notebooks, handwritten data will be changed from off-line to on-line formats. Off-line handwritten data, i.e., scanned data, are similar to pixel-based images. On the other hand, on-line handwritten data obtained by handwriting input devices, in addition to brushstroke coordinate information, include time-series data and pressure factors. As the number of on-line handwritten documents increases, the ability to search for information from such documents should be developed. One of the key ability in document search is how fast we can grasp the summary of each listed document. When we search document, we find the desired document effectively to scan through the summary list of documents called snippets, which is possible to grasp the summary of documents without scanning the actual contents. This chapter presents a new summarized view of on-line handwritten documents which improve searching own or others handwritten documents, such as thumbnails and summarized text required in search systems.

Displaying scaled thumbnails (scaled pictures of original content) is an effective way to scan through lists of documents (e.g., thumbnails are effective for Web searches [40] [41]). For example, Web image-retrieval services such as Google Images and Yahoo! Image Search output scaled thumbnails in a list view of the search result pages. From these thumbnails, users can see an outline of the original image. However, we cannot use traditional scaled thumbnails to understand a summary of handwritten documents since the text size is too small to read on small device screens.

In addition, text snippets, i.e., portions of original text, are commonly used in a list view of search result pages. For instance, results from Google Web Search display text snippets, which are constructed by extracting a series of words including the query word. In handwritten documents, however, the accuracy of recognizing handwritten characters is as low as around 92% [42] per a character, resulting in a difficulty in adopting natural language processing to summarize handwritten documents.

To the best of our knowledge, no research has been conducted on navigational views of handwritten documents, although some research does exist on search views of images and Web pages. Chapter 2 proposed the recognition model of "Content-targeting" annotation including emphasis contents intention. Based on the concept of the model, this chapter proposes handwritten-document views called EA Snippets based on natural emphasis annotations, such as underlining and enclosures, and note based on natural language processing by extracting content-targeting annotations. Two types of EA Snippets are shown in this chapter:

- **Image EA Snippets** consisting of important words or graphs, where the text is expanded for easier readability.
- **Text EA Snippets** consisting of both summarized text and a scaled thumbnail, where summarized text consists of important words listed in order of importance.

Furthermore, this chapter investigates the performance of these proposed view types when users search for information in handwritten documents.

4.2. Related Work

This section refers to researches on thumbnails of images and Web pages, and investigates a method to generate a thumbnail from important parts of a document, which is same approach with proposed method, In addition, since classifying and grouping handwritten objects is need to determine whether they are important or not, researches classifying handwritten strokes are referred.

4.2.1. Thumbnails of Image and Web Pages

Proposed method generates emphasized thumbnails consisting of important handwritten objects. However, to date there is no research concerning the generation of thumbnails for handwritten documents. Consequently, this section looks to apply research concerning thumbnails for images [43] [44] [45], and thumbnails for Web pages [46] [47] [48] to generate proposed handwritten Image EA Snippets.

Several studies have investigated how to improve the thumbnails of pictures. Amurtha et al. [43] proposed an intelligent automatic cropping technique for pictures. Cropping is used to extract the rectangular area containing the attention objects. Prior to shrinking an image, they used Regions of Interest (ROIs) to crop objects from images. Their experiments showed that thumbnails efficiently increased the performance of content-based image retrieval (CBIR). Suh et al. [45] proposed two automatic cropping techniques; the first detects salient portions of images, while the other is a method of automatic face detection. They generated thumbnails by cropping these detected areas. Their user study shows that these methods resulted in small thumbnails that can be easily resizing method, called Seam Carving, which supports content-aware image resizing. Seam Carving creates the energy map of an image, and then shrinks the image by removing the minimum energy path from left to right, or from top to bottom. Because Seam Carving does not discriminate between attention and other objects, as the image shrinks the attention objects become distorted.

Other studies aimed in improving thumbnails of Web pages. Teevan et al. [47] extracted title-texts, logo images, and salient images from Web pages, and produced thumbnails by compiling these component pieces. Their experiments showed that in re-finding tasks, their thumbnails enabled users to find Web pages faster than snippets of text and traditional thumbnails. Woodruff et al. [48] proposed textually enhanced thumbnails of Web pages. These enhanced thumbnails were created by enhancing screenshots of Web pages with query words. In their study, participants searched faster using the textually-enhanced thumbnails than when using the plain thumbnails and text summaries. Lam et al. [46] proposed a thumbnail enhanced with readable text fragments. In their user study, when participants used the proposed thumbnail interface, they could find the area containing the target content in Web pages approximately 41% faster, and with 71% lower error rate, compared to traditional interfaces.

These related studies proposed methods for detecting important objects, and producing as outputs summarized thumbnails of images and Web pages. This chapter proposes a method to detect important objects in handwritten documents by detecting emphasis annotations. Next, based on the results of previous related studies, our method use the detected enhanced objects and summarize handwritten documents with emphasized views.

4.2.2. Classification Methods of Handwritten Objects

Several methods that classify handwritten objects into text and non-text have been proposed. There are three types of classifying methods that adopt features:

1) Using stroke features [49]
2) Using both stroke features and the context of strokes [50] [51]
3) Using stroke features and the features of a stroke group [52] [53] [54]

Willems et al. [49] proposed a text/non-text classification method that adopts 12 kinds of stroke feature, such as length and curvature. A stroke is defined as a sequence of pen-down and pen-up actions. In addition to text/non-text classification, they applied a method to classify non-text strokes using four kinds of shapes, lines, and arrows.

Other studies proposed text/non-text classification methods that used note only stroke features, but also the context of strokes [50] [51]. In addition to using characteristics of strokes, Bishop et al. [50] used information provided by relations between strokes, such as the pen-up times between adjacent strokes in a time-series order, and the pen-down or pen-up locations of adjacent strokes. By using a bi-partite hidden Markov model (Bi-partite HMM), their system classified sequences of handwritten strokes into text and graphics with 95% accuracy. On the other hand, Zhou et al. [51] presented an approach for separating text and non-text handwritten strokes in on-line handwritten Japanese documents, based on Markov random fields (MRFs) which effectively utilize the spatial relationship between strokes. They used four features of the relationship between two neighboring strokes: minimum distance between to strokes, maximum and minimum distance between the endpoints of two strokes, and distance between the centers of the bounding boxes of two strokes. Their approach successfully classified handwritten stroke with 96.61% accuracy.

Several studies proposed the approach of stroke grouping [52] [53] [54]. Shilman et al. [54] presented an integrated approach for parsing textual structures in freeform notes. First, they grouped strokes by applying their layout analysis algorithm that uses robust statistics. Next, each stroke group was classified into text or graphics by using local and global stroke features, such as stroke length, curvature, and number of strokes in the group. Mochida et al. [53] proposed a method for separating on-line handwritten patterns into Japanese text, figures, and mathematical formulas. They applied a probabilistic model employing stroke features, such as stroke crossings, and stroke densities. To classify non-text strokes into figures and mathematical formulas, they grouped strokes using the length of off-strokes, defined as the distance between pen-up to pen-down. On the other hand, Ao et al. [52] proposed a method of structuralizing and classifying raw digital ink into text and graphs using multiple hierarchies. They used a link model [55] to group strokes, and classified the groups into text and graphs by suing support vector machine (SVM).

These studies on processing handwritten data enable to group strokes, and classify strokes into text and non-text. To detect emphasis annotation of authors, such as underlines and enclosing, proposed system references these methods to classify and group strokes.

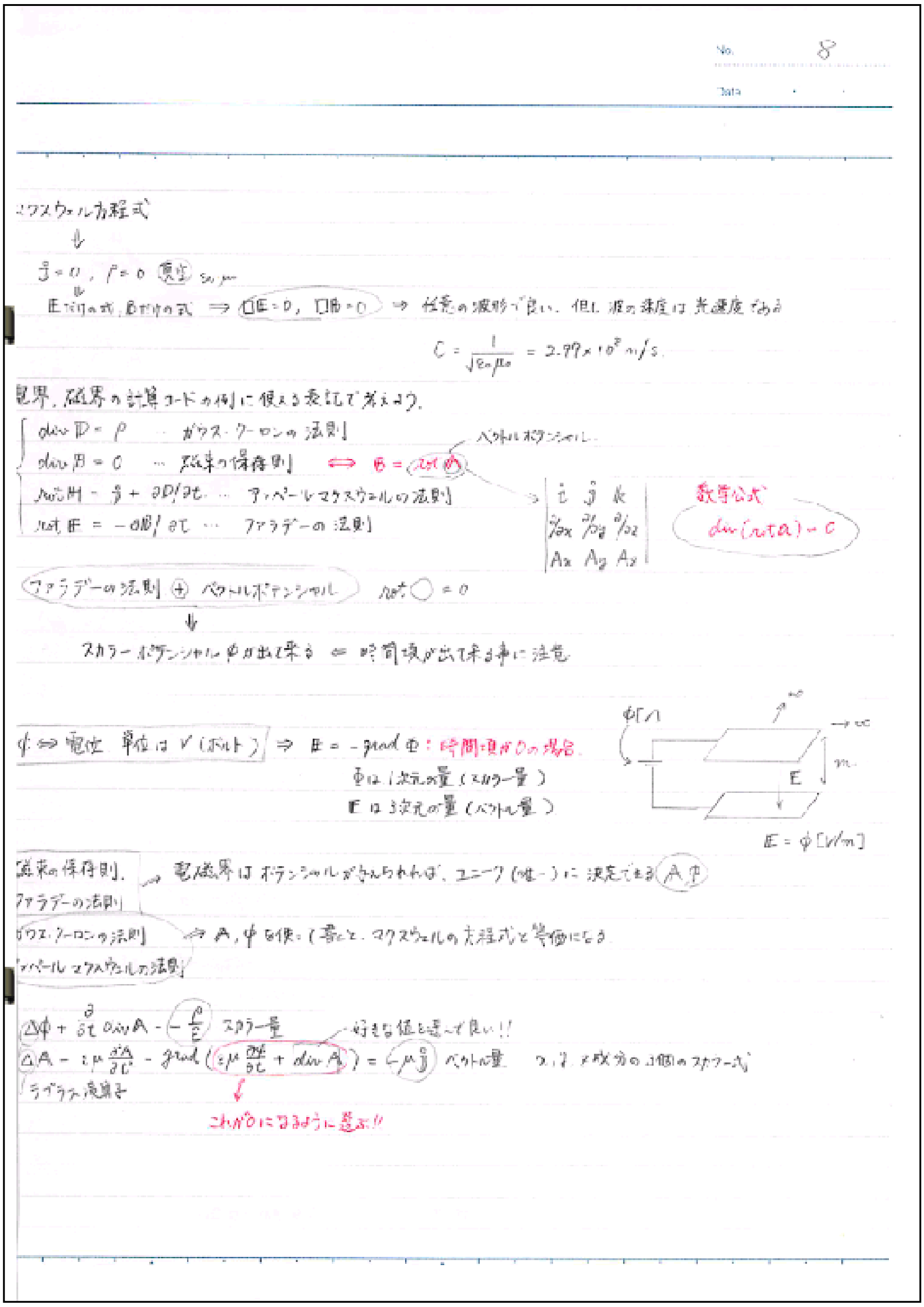


Figure 4.1 Example of collected notebook

4.3. Survey of Emphasis Annotations

In this section, we describe the survey on natural emphasis annotations used in notebooks. First, some frequently used natural emphasis annotations, which are often used in notebooks, are defined. Then the survey performed two investigations:

1) How often emphasis annotations are used in notebooks
2) Under which situations they are utilized.

The survey collected 278 handwritten pages from the notebooks of eight university students in their 20s, studying such subjects as mathematics, physics, chemistry, and programming for six months to 1 year (an example is shown in Figure 4.1). The notebooks were written in Japanese, and our analysis shows that they include three types of natural emphasis annotations:

1) Enclosing words
2) Underlined words
3) Colored words

The examples of these annotations are shown in Figure 4.2. In addition, the investigation found that emphasis annotations were performed 3.4 times per page on average. Table 4.1 shows the number of occurrences for each emphasis annotation.

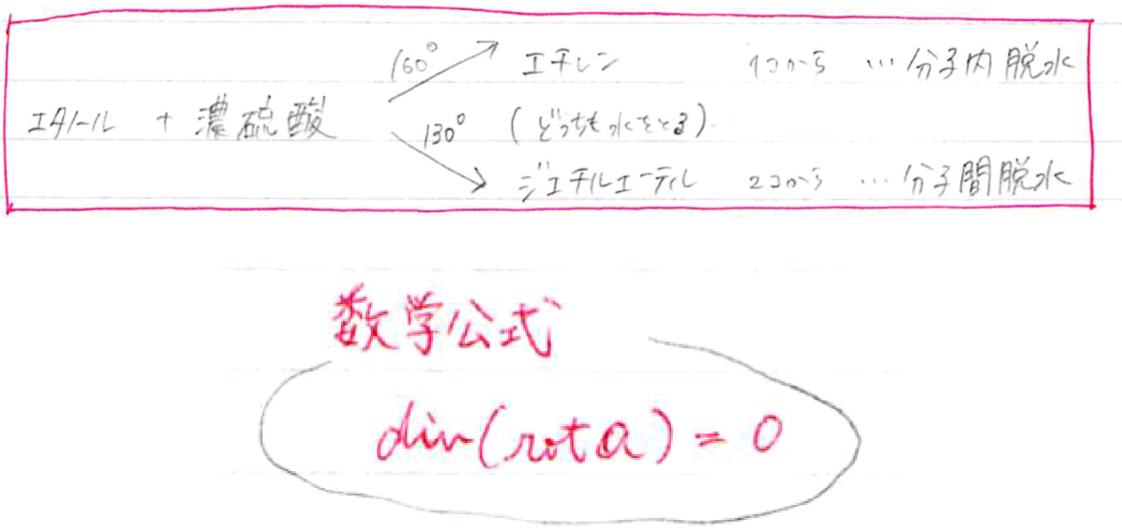
Table 4.1 Type of emphasis annotation and the number of occurrences in collected data

Emphasis Annotation	Number of occurrences
Enclosing Words	345
Underlined Words	304
Colored Words	296

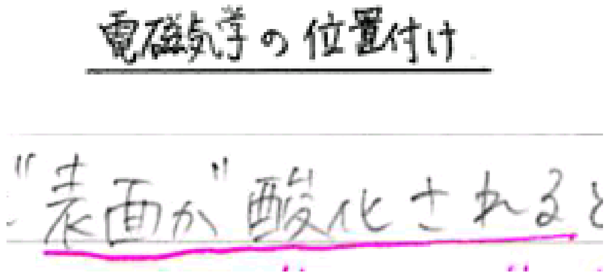
The survey also interviewed the students to confirm when such emphasis annotations were performed. As a result, the following three types of situations were found.

1) Emphasizing important words or equations
2) Highlighting titles or topics
3) Highlighting a summary of the contents

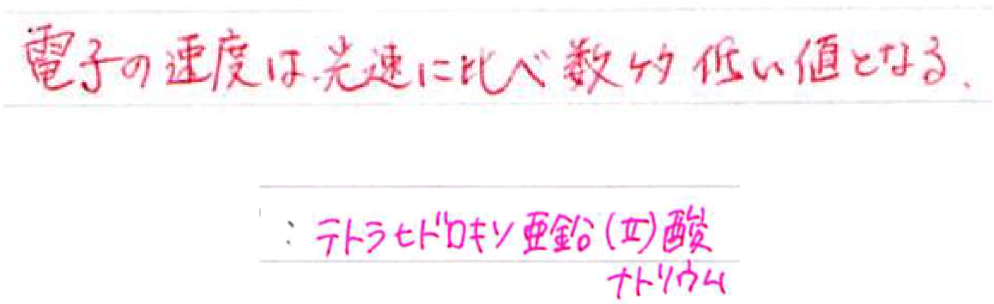
Furthermore, the participants stated that they also emphasize titles or topics in the index area of the notebook instead of using emphasis annotations.



1) Enclosing words



2) Underlined words



3) Colored words

Figure 4.2 Examples of natural emphasis annotations

The results of our survey found that the emphasized words indicate keywords, topics, or a general summary. Therefore, the findings are assumed that we can easily understand a summary for extracting words and figures based on emphasis annotations and the index area.

In addition, the questionnaire survey to calculate emphasis strength, which represents the importance of the emphasis annotations, was conducted. The additional survey invited 19 computer science students, including the authors of the notebooks we collected, all of whom were in their 20s. The participants were asked to assess the magnitude of the strength of each emphasized expression in Table 4.1. All answers were normalized to a unit scale by each participant and were perceived as the score. Furthermore, we defined Emphasis Strength by the following equation:

$$\text{Emphasis Strength} = \frac{\text{Mean of the Score}}{\text{SD of the Score}} \tag{9}$$

Table 4.2 shows the Emphasized Strength of each emphasized annotation, based on the results of the survey. Proposed method also uses the emphasis strength to calculate the importance of handwritten objects.

Table 4.2 Emphasis Strength of each emphasis annotation

Emphasis Annotation	Emphasis Score
Enclosed Words	4.924
Underlined Words	2.551
Colored Words	2.423

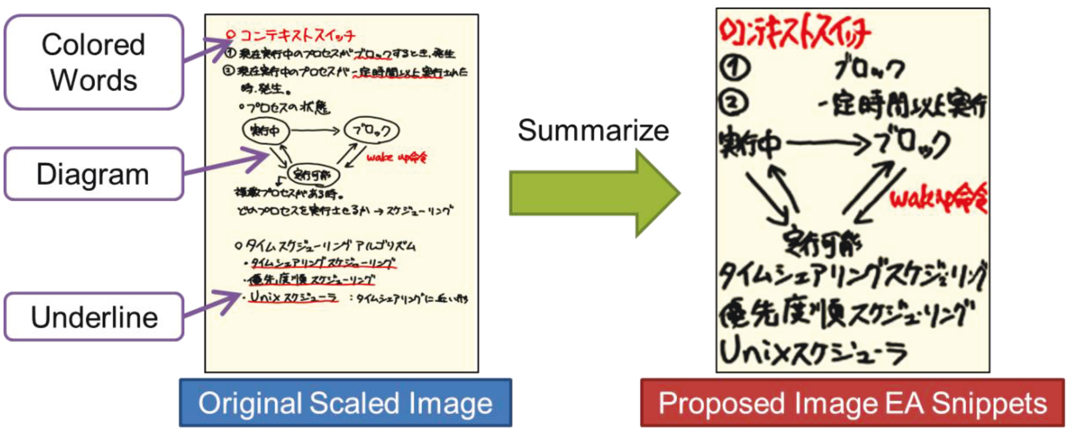


Figure 4.3 Example of Image EA Snippets

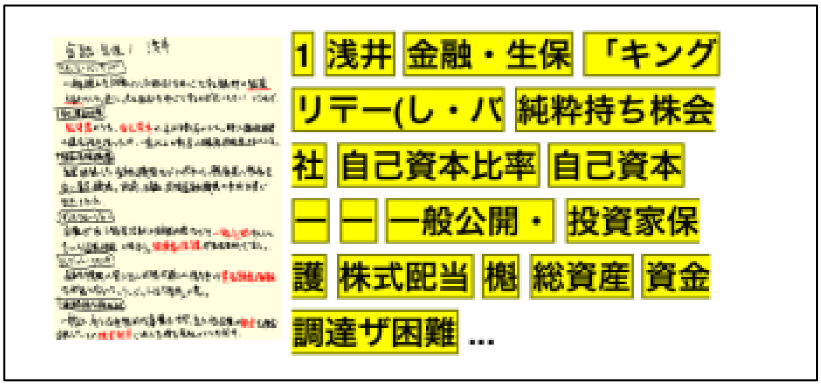


Figure 4.4 Example of Text EA Snippets

4.4. Implementation

This section proposes two types of emphasized view based on emphasis annotations:

1) **Image EA Snippets** (see Figure 4.3)
2) **Text EA Snippets** (see Figure 4.4)

Proposed system detects both emphasis annotations and words in the title index area of notebook. First, emphasis annotations are extracted, and emphasis scores are calculated based on the Emphasis Strength described in 4.3. Emphasis scores represent the strength of the author's emphasis. Following the calculation of the emphasis scores, the system generates Image EA Snippets and Text EA Snippets based on the emphasis annotations of authors.

Figure 4.5 shows the procedure for calculating emphasis scores. First, all input strokes are classified into either text or non-text strokes. From the text strokes, we extract words in the title index of notebook and colored words. From the non-text strokes, the system extracts words in the title index of notebook and colored words. From the non-text strokes, the system extracts enclosing words and underlined words. In addition, non-text strokes are classified into emphasized strokes and graph strokes. After classification, our system organizes strokes into displaying units, which are groups as a unit of display or hide, and for each group calculates the emphasized score based on the emphasis strength. Finally, the system generates thumbnails containing magnified words that can be read, or text snippets consisting of emphasized words.

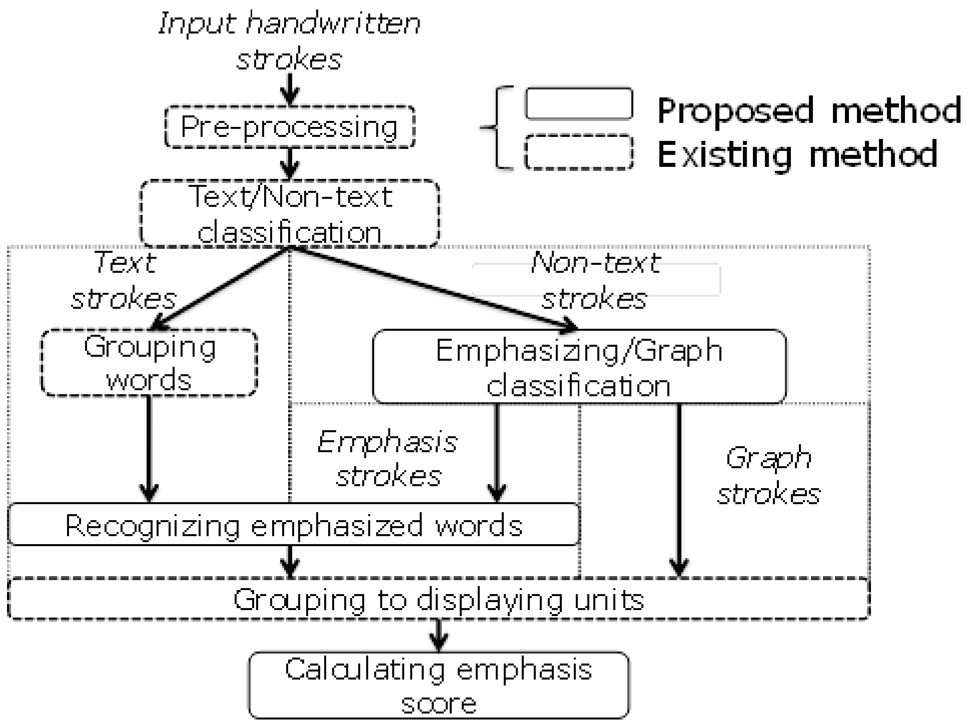


Figure 4.5 Procedure for calculating emphasis scores

4.4.1. Pre-processing

The on-line handwritten data obtained by the devices include the noise caused by hand-shake and accidental errors. To minimize the influence of the noise, the methods reduces the noise of handwritten strokes before classifying them. The pen locus of each handwritten stroke is smoothed with a Gaussian filter. The variable $\mathrm{P}_{\mathrm{N}_{\mathrm{new}}}$ represents the pen locus coordinates after smoothing, and is given by:

$$\mathrm{P}_{\mathrm{n}_{\mathrm{new}}} = \frac{P_{n-2} + 4P_{n-1} + 6P_n + 4P_{n+1} + P_{n+2}}{16} \qquad (10)$$

$$(2 < \mathrm{n} < \mathrm{N} - 1)$$

where P_{n} represents the coordinates of the nth pen locus of the stroke, and N is the number of coordinates in the stroke.

4.4.2. Text/Non-text Classification

To detect handwritten diagrams and emphasis expressions, our system classifies all input strokes into either text strokes or non-text strokes by applying an Support Vector Machine (SVM). We use the following four stroke features as inputs to SVM after reducing the noise of handwritten strokes by using Gaussian Filter:

1) Stroke length

$$\sum_{n=1}^{N-1} \sqrt{(x_{n+1} - x_n)^2 + (y_{n+1} - y_n)^2} \tag{11}$$

2) Stroke curvature

$$\frac{\sum_{n=2}^{N-1} \left(\frac{(x_{n-1} - x_n)(x_{n+1} - x_n) + (y_{n-1} - y_n)(y_{n+1} - y_n)}{\sqrt{(x_{n-1} - x_n)^2 + (y_{n-1} - y_n)^2}\sqrt{(x_{n+1} - x_n)^2 + (y_{n+1} - y_n)^2}} \right)}{N-2} \tag{12}$$

3) Long side of the stroke's bounding box

$$max((x_{max} - x_{min}), (y_{max} - y_{min})) \tag{13}$$

4) Number of other strokes crossed by the stroke being classified

Variables x_{n} and y_{n} are the *n*th x- and y-coordinates recorded in the stroke. Variable *N* represents the total number of coordinates included in the stroke. bles x_{max}, y_{max}, x_{min} and y_{min} are the maximum and minimum coordinate values in the stroke.

4.4.3. Emphasizing/Graph Classification

After text/non-text classification, non-text strokes are further classified into emphasis strokes and graph strokes. Emphasis strokes consist of both underlined and enclosing strokes.

Here, a stroke is classified as an underlined stroke when the height of the stroke's bounding box located under the word's bounding box, is within the height of the word's bounding box. Specifically, non-text strokes satisfying the following two conditions are categorized as underlined:

1) Shape condition

$$\begin{cases} 2W_{WordAve} < W_{Stroke} \\ H_{WordAve} > H_{Stroke} \end{cases} \tag{14}$$

2) Neighborhood character count condition

When two or more neighborhood characters satisfy the followingconditions:

$$\begin{cases} \min(X_{Stroke}) < X_{WordG} < \max(X_{Stroke}) \\ Y_{WordG} - H_{WordAve} < \min(Y_{Stroke}) \\ \max(Y_{Stroke}) < Y_{WordG} \end{cases} \tag{15}$$

Variables H_{WordAve} and W_{WordAve} are the average height or width of the characters in the page. Variables H_{Stroke} and W_{Stroke} are the height and width of the target stroke. Variables X_{Stroke} and Y_{Stroke} are the sets of x- and y-coordinates of the target stroke. Variables X_{WordG} and Y_{WordG} are the x- and y-coordinates of the median point of the characters' bounding box (see Figure 4.6).

Conversely, the enclosing stroke is extracted if its bounding box encloses the word's bounding box. Specifically, non-text strokes satisfying the following two conditions are categorized as enclosing:

1) Shape condition

$$\begin{cases} 2W_{WordAve} < W_{Stroke} \\ \frac{1}{2}H_{WordAve} < H_{Stroke} \end{cases} \tag{16}$$

2) Comprehension character count condition

The bounding box of the target stroke contains the center point of character, and the number of characters in the bounding box of the target stroke is greater than or equal to

$$\max\left(2, \frac{\mathrm{S_{Stroke}}}{\alpha \mathrm{S_{WordAve}}}\right) \tag{17}$$

Variable $\mathrm{S_{Stroke}}$ represents the bounding box area of the target stroke. Variable $\mathrm{S_{WordAve}}$ is the average bounding box area of the characters in the page. Variable α is the threshold of the character's density, which we the method set to 6.0 to maximize detecting accuracy (see Figure 4.7).

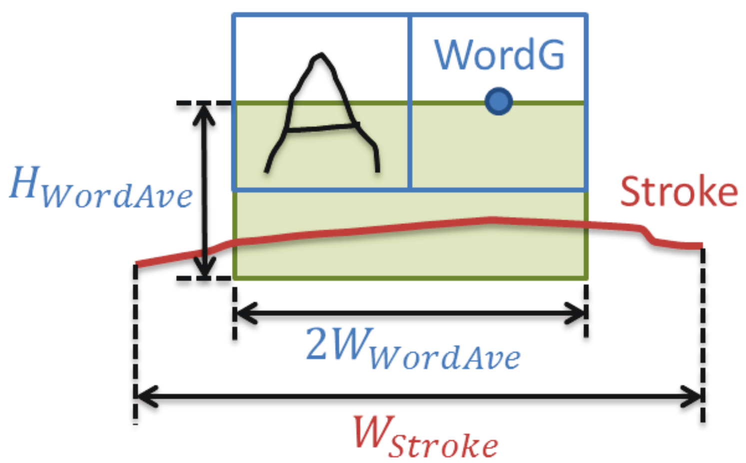


Figure 4.6 Detecting an underline stroke

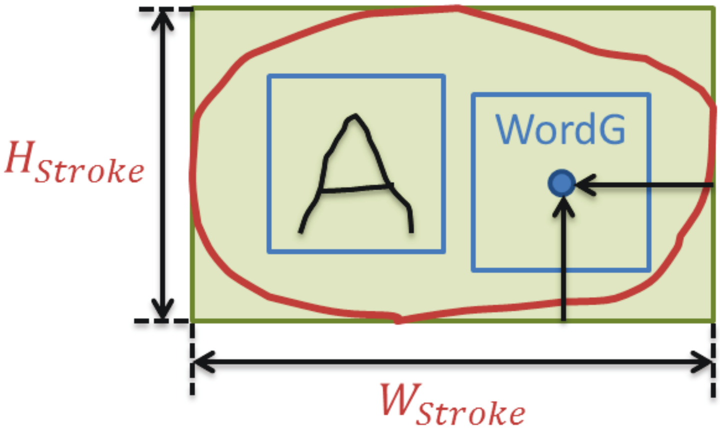


Figure 4.7 Detecting an enclosure storke

4.4.4. Recognizing Emphasized Words

After emphasizing/graph classification, the method detects which part of the text is emphasized by the author, and which patterns of emphasized expression are present in the text. First, our system splits text strokes into character groups. We use *.NET Ink Analyzer*1 for the character grouping. After grouping, we detect underlined and enclosed words by using the spatial relationships between character groups, and the underlined and enclosing strokes we extracted from non-text strokes.

Underlined words are located above the underline stroke. Thus, our method detects underlined words by extracting the words satisfying the following conditions:

$$\begin{cases} \min(X_{Underline}) < X_{WordG} < \max(X_{Underline}) \\ \min(Y_{Underline}) < Y_{WordG} \\ Y_{WordG} < \max(Y_{Underline}) + \frac{3}{2} H_{WordAve} \end{cases} \quad (18)$$

Variables $X_{Underline}$ and $Y_{Underline}$ represent the sets of x-and y-coordinates of the underline strokes we extracted from non-text strokes (see Figure 4.8).

Conversely, enclosed words are located within the area enclosed by the enclosing stroke. Thus, our method detects enclosed words by extracting words whose median points are within the bounding box of the enclosing stroke (see Figure 4.9).

---

1 Microsoft, .NET Ink Analyzer,
http://msdn.microsoft.com/en-us/library/microsoft.ink.inkanalyzer(v=vs.80).aspx

4.4.5. Stroke Grouping for Summarization

To avoid displaying handwritten strokes discretely, after detecting the emphasis of the author, our system groups strokes by the kind of handwritten object. First, we sequentially check strokes ordered in a time series. Two strokes adjacent in a time series are grouped together if the strokes satisfy the following conditions:

1) The distance between the x-coordinates of adjacent strokes is less than the threshold value of 80 pixels.
2) The distance between the y-coordinates of adjacent strokes is less than the threshold value of 60 pixels.
3) The type of emphasis of the adjacent strokes is the same.

The system selected threshold values that maximized detection accuracy. By using this method, the system organizes handwritten strokes by the kind of handwritten object, such as a text line (same emphasis type) and a graph. After stroke grouping, the system classifies groups into text and non-text groups. Groups containing more than 50% text-strokes are classified as text groups, and the remaining groups are classified as non-text groups. In addition to stroke grouping, the system reduces the incorrect recognition of text/non-text classification. To reduce the number of strokes recognized incorrectly as non-text strokes, the system changed all strokes in text groups into text strokes.

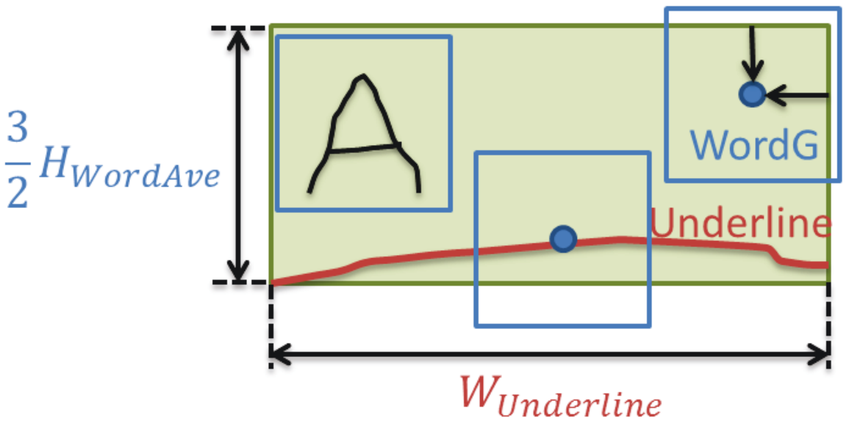


Figure 4.8 Detecting underlined words

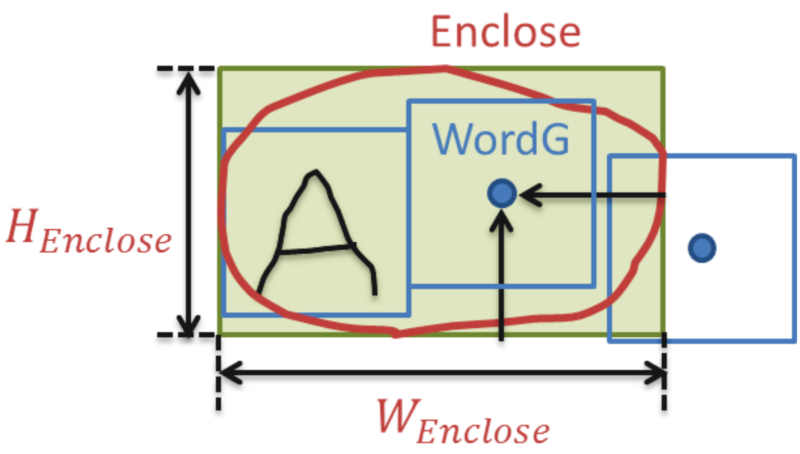


Figure 4.9 Detecting enclosed words

4.4.6. Calculating Emphasized Scores

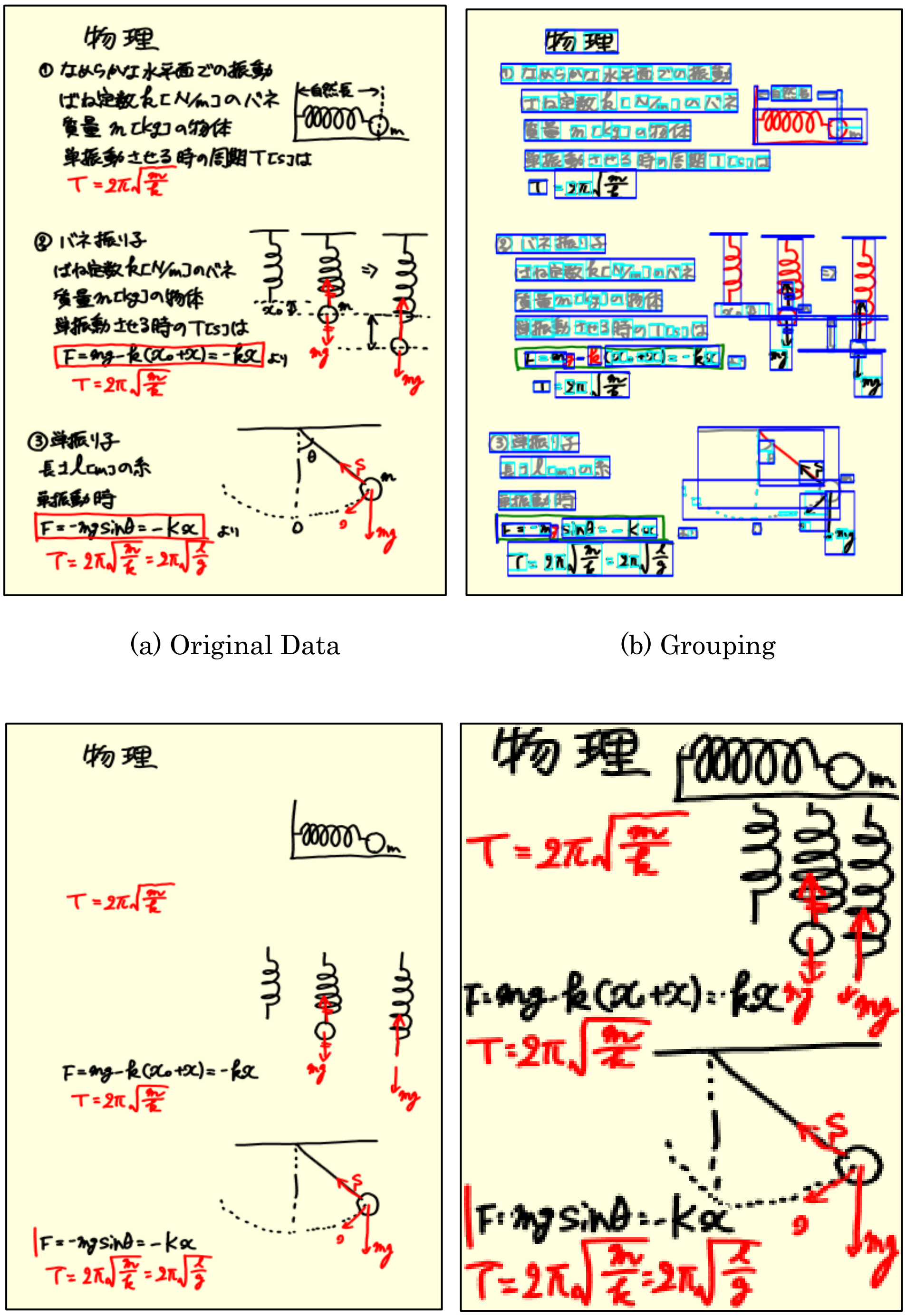
When a word is classified as an emphasized word, we calculate its emphasis score, indicating the importance of the word. The score is calculated based on the emphasis strength, as shown in Table 4.2. Using these processes, the system computes the emphasis score of each group based on the intended emphasis of the author. In the next section, we present a method for generating two types of EA Snippets.

4.4.7. Generating Image EA Snippets

Compared to traditional scaled thumbnails for images, we should take into account when used for handwritten documents:

1) The text in a scaled thumbnail is too small to read.
2) The amount of text, i.e., the amount of information, in a scaled thumbnail is not reduced compared to the original data. Due to this, the cognitive load of understanding contents is not reduced.

Therefore, this chapter proposes "Image EA Snippets" summarizing the intended emphasis of authors. Our proposed method summarizes the contents of handwritten data based on emphasis, such as underlines and enclosing, and increases the size of text in the contents of the thumbnail. Figure 4.10 shows the process to generate Image EA Snippets.



(a) Original Data (b) Grouping

(c) Object Selection (d) Zooming Objects

Figure 4.10 Process of generating proposed Image EA Snippets

By using the emphasis scores, the method generates thumbnails that improve information retrieval in handwritten documents. First, our system groups handwritten strokes, and calculate emphasis scores by applying proposed method (see Figure 4.10b). Second, the number of text stroke groups is reduced by removing groups with emphasis scores under a threshold (see Figure 4.10c). Note that non-text stroke groups, such as diagrams, are not removed. The threshold he method uses is the maximum value satisfying the following condition:

$$\sum_{n=1}^{N_{SG}} S_{group}(n) B_{thres}(n) < \beta S_{org} \tag{19}$$

where N_{SG} represents the number of stroke groups, and $S_{\mathrm{group}}(n)$ returns the area of the bounding box of the nth stroke group. If the emphasis score of the nth stroke group is more than the threshold, $B_{\mathrm{thres}}(n)$ returns one, otherwise it returns zero. The scaling rate of the thumbnail is denoted by β, and the area of the original contents is denoted by S_{org}.

Finally, the stroke groups are reallocated and expanded by using the Seam Carving method [44]. Using this method, proposed method can scale down a handwritten document by removing blank spaces, removing contents below the threshold, and maintaining the alignment of stroke groups. Figure 4.10d shows our proposed thumbnails scaled by the Seam Carving method. From this thumbnail, we can understand the summary of the contents.

4.4.8. Generating Text EA Snippets

Proposed system detects important words, and scores important words based on their importance. It is difficult for natural language processing methods to detect important words from handwritten documents, because handwritten recognition techniques do not perform well. Thus, this section presents a method that uses the intended emphasis of authors, and applies on-line handwritten recognition methods to generate text snippets.

First, the method applies a handwritten recognition method to text stroke groups. Specifically, the method uses the *.Net Ink Analyzer* for handwritten recognition. Next, the method sort text stroke groups by their emphasis scores, and clip at a maximum the 80 top-ranked words. Finally, the method displays scaled thumbnail to help users understand the layout and graphs of the contents in addition to the 80 top-ranked words. Figure 4.4 shows proposed Text EA Snippets summarized by the intended emphasis of the author. From the text snippet, we can understand the keywords in the contents.

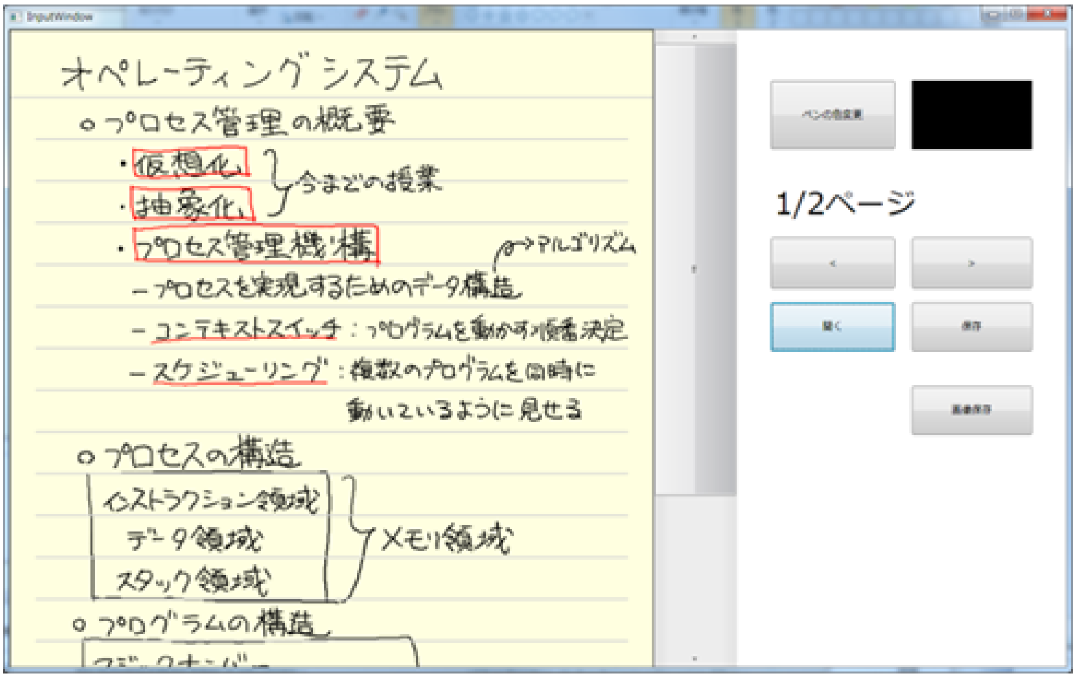


Figure 4.11 Collecting system of handwritten notebook

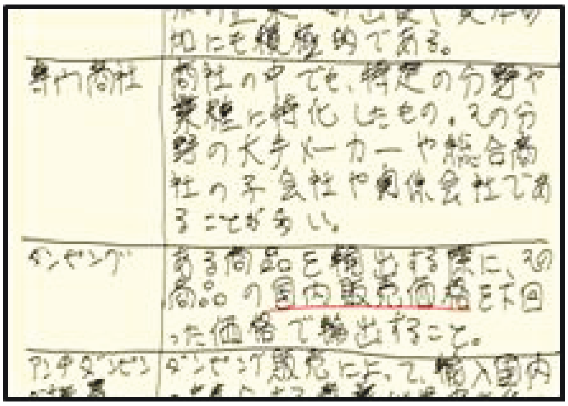
4.5. User Study

4.5.1. Collecting On-line Handwritten Data

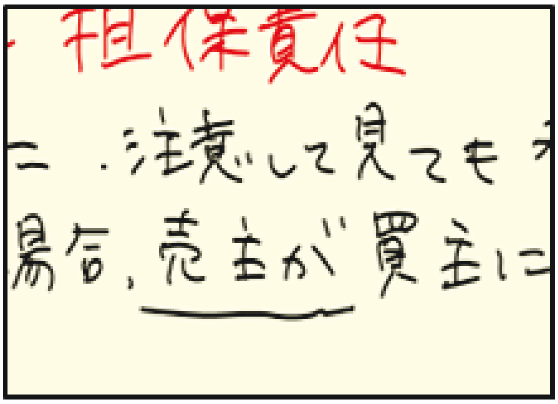
Compared to traditional paper-based notebooks, digital notebooks enable us to collect on-line handwritten data. This data, in addition to drawings representing handwritten information, include a time series of strokes, pressure, and writing speed. Using digital notebooks, we can analyze handwritten data in more detail. Proposed method uses on-line handwritten data to detect the intended emphasis of authors. Hence, this section describes an experimental system for Windows (using the pen tablet WACOM Cintiq 12WX[1]) to collect on-line handwritten documents.

This system was developed in Visual C#, and accepts a pen tablet device to enter inputs by handwriting. This study collected 42 pages (consisting of 38,416 handwritten strokes) of on-line handwritten notebook data. Eleven university students majoring in computer science were used as notebook authors. This study gave them a document containing common topics and current events, and informed them about the important words in the documents. Participants were instructed to create a note summarizing the

[1] Wacom Cintiq 12WX, http://wacom.jp/jp/products/cintiq/12wx/



(a) Handwritten ruled line



(b) Far from text

Figure 4.12 Examples of underline recognition error

documents. Note that participants were not forced to follow any format, i.e., participants could emphasize important words using any emphasis expression they wanted, and were allowed to use the notebook in any way they chose. Figure 4.11 shows the screen capture of the collecting system.

4.5.2. Recognition of Emphasized Words

First, this study evaluated the recognition performance, i.e., precision and recall, of our detection method for emphasized words. Here, words in the title index and colored words were successfully detected from on-line handwritten data like color data and written area.

(a) Traditional Scaled Thumbnail (b) Traditional Head Text Snippet

(c) Proposed Image EA Snippets (d) Proposed Text EA Snippets

Figure 4.13 Screen captures of experimental system

The study investigated 38,416 strokes contained in the handwritten documents collected by using the collecting system. The manual classification of the documents resulted in 16 enclosing and 72 underlines. Proposed method detected all enclosing in the documents with no errors. Conversely, proposed system detected underlines with 85.71% precision rate, and 96.43% recall rate. Figure 4.12 shows examples of an underline recognition error. These results found that text written by hand above the ruled line was falsely recognized as underline (see Figure 4.12a). In addition, some underlines could not be detected, because the underline was located far from handwritten text (see Figure 4.12b).

4.5.3. Search Performance

This section describes a user study to compare the search time required for handwritten documents using both traditional thumbnails and proposed EA Snippets. To measure the search performance, an evaluation application that shows various views of handwritten documents and operates on various device types were developed. This study used an iPhone 3GS, and a screen capture of the experimental system is shown in Figure 4.13. In this study, the following four view types were compared:

1) Traditional Scaled Thumbnails, which are reduced versions of the original image (see Figure 4.13a).
2) Traditional Head Text Snippet + Scaled Thumbnail, which are generated by recognizing the first 80 characters of handwritten text in a document. Scaled thumbnails are also presented (see Figure 4.13b).
3) Proposed Image EA Snippets, which are summarized based on their emphasis scores (see Figure 4.13c).
4) Proposed Text EA Snippets + Scaled Thumbnails, which are also summarized based on their emphasis scores (see Figure 4.13d).

On the same screen, four pages are displayed together for (1) and (3). On the other hand, 2.5 pages are displayed for (2) and (4). The goal of this study is to verify which thumbnails enable us to find information more easily.

This study conducted three types of evaluation. The first study performed the comparison of the search time of four view types to answer the fill-in-blank question, on condition that the keyword of the question is included in the proposed view. After the first study, the additional studies in addition to the first study were also conducted because the first study leaves the two questions. One did not consider the situation in which the users searched document by using proposed view which is not include the keyword of the question. The other question is that there is no consideration of document's author, that is to say we did not consider the difference of the performance searching in own documents or other's documents. This study conducted the two additional user studies to evaluate the questions. The questions that participants answered are shown at Appendix.

Search by Keywords Related to Emphasized Word

First, this section evaluates the search performance by keywords related to emphasized word. This study invited twenty participants to participate in our user study, including eleven who were authors of the collected handwritten documents. All participants were university students in their 20s, two of them women. The user study was performed using the four view types shown in Figure 4.13, and measured the time required to finish answering the questions from each view. In each experiment, all participants were given twenty pages of handwritten documents each from the collected data, along with five questions. All participants were given the same questions and handwritten documents. The participants were required to answer the questions by navigating using the views generated from the documents. The questions were fill-in-the-blank types, and the answers were written directly on the original handwritten documents provided (see Table 4.3). In addition, the keywords of each question were indicated using emphasis annotations.

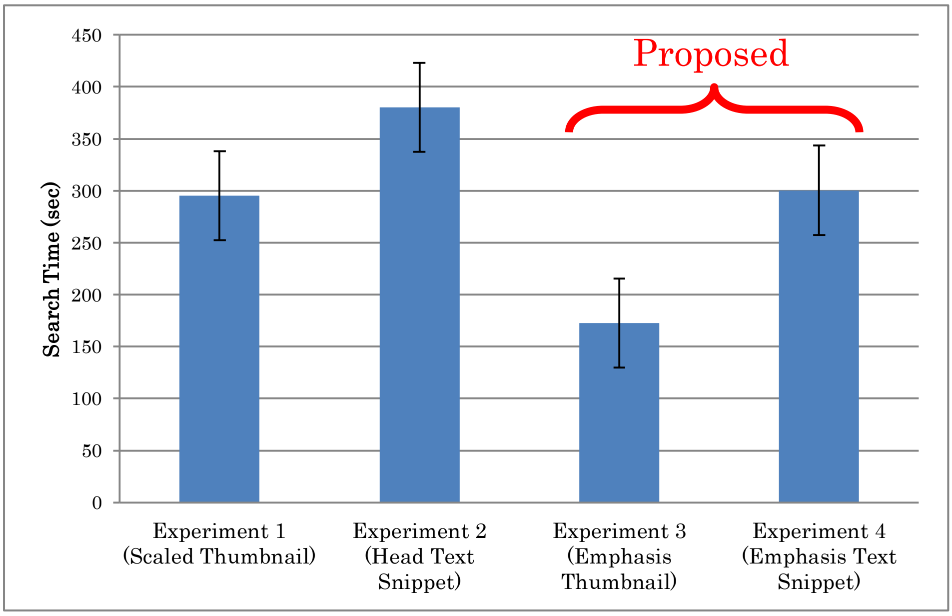


Figure 4.14 Results of search performance study

Table 4.3 Example of the fill-in-the-blank questions used in the user study

Question	Answer
Forward integration strategy means that a trading company conducts expansion of trade to cooperate with distribute firms closer to consumers, such as supermarkets and XXX.	Convenience store

Figure 4.14 shows the average search time and the standard error. In addition, this study performed a Kruscal-Wallis test, and conducted a pairwise comparison of the results. The results show that, on average, our proposed emphasized thumbnails result in the best search time among the four view types. Compared with the traditional scaled thumbnails, we found that proposed Image EA Snippets enable users to search 42% faster ($p < 0.001$) on average. On the other hand, compared to traditional head text snippets, proposed Text EA Snippets also enable users to search faster on average, although the difference is not statistically significant ($p > 0.1$). Moreover, the results show that proposed Image EA Snippets help users search faster than do proposed Text EA Snippets ($p < 0.0001$).

Search by Keywords Not Related to Emphasized Word

For the above task, the users searched for answers based on the emphasized keywords. In other words, the task did not consider situations in which the users searched for information not based on the emphasized words. This section therefore conducted another user study ($N = 10$) for searches not based on the emphasized keywords (e.g., "Forward integration strategy" is replaced by the blank boxes in Table 4.3). Herein, N is the number of subjects in this study. The results show that the proposed Image EA Snippets enable users to search 24% faster on average than traditional scaled thumbnails, although the difference is not statistically significant ($p > 0.1$), while proposed Text EA Snippets enable only a 9% speed increase ($p > 0.1$). After this study, some participants said that they could find the page if they can imagine the keyword from the words which include in the snippet. From these results showed that our proposed view is effective when we can imagine the information we want to know from the words or graph showing in snippet.

Comparison of the User's Own Notes and the Notes of Others

In addition, the study also investigated the difference between searching one's own notes and the notes of others (N = 4). The results show that, using traditional scaled thumbnails, the users found the answer 206% slower on average for notes other than their own ($p < 0.05$). In contrast, there are no statistical differences between searching one's own notes or the notes of another student for emphasized thumbnails and text snippets ($p > 0.1$). These result showed that our proposed view is effective for navigating pages of handwritten documents to find information regardless of the authors.

Discussion

After the user study, the study discussed with the participants proposed snippets. Some of them said that they could not understand what was written in traditional scaled thumbnails, because characters were too small to read. Conversely, they could guess the contents in our proposed emphasized thumbnails, and proposed thumbnails often helped them in searching for information in handwritten documents. Participants also reported that if the exact search keyword was not included in the thumbnail, they had trouble determining the contents of the thumbnail. On the other hand, some participants reported that text snippets occasionally did not help them understand the summary of the handwritten documents, because the accuracy of the handwritten character recognition was low. In addition, some of them said that they often looked scaled thumbnail only in the text snippet. This show the removal of these limitations could improve the searching speed.

4.6. Conclusion

This chapter discussed the ineffectiveness of traditional thumbnails in information retrieval when targeting handwritten documents, and presented a new approach, i.e., detecting natural emphasis annotation. This chapter proposed the use of emphasized index summarized by emphasis annotations. The user study showed that proposed Image EA Snippets enable users to search 42% faster on average compared with the traditional scaled thumbnails. In addition, this chapter found that thumbnails are more effective than text snippets for searching handwritten documents because handwritten data are hard to recognize that results in defective structural analysis.

Chapter 5. Conclusion

Document annotation such as writing comments by handwriting on paper-based documents is indispensable interaction between human and documents. Currently, the demand of handwriting annotation on electronic documents has been increasing due to the replacement from paper-based documents to electronic documents. The handwriting annotation on electronic documents requires the availability maximizing the advantage of electronic documents on computers. However, the user interfaces of traditional researches related to handwriting annotation have the problems either the lack of the document availability or the lack of the learnability in the system. Therefore, this part proposed the user interfaces for handwriting annotation on electronic documents to both increase the learnability of the annotation system and enhance the availability of electronic documents.

Chapter 2 proposed the recognition model of handwriting annotation on electronic documents. The recognition model generated by human annotation data achieved improving the availability of documents to recognize the annotation selecting range on contents while traditional research using heuristic model had difficulty to recognize the exact selected rage. The study found the proposed model can find user's content-targeting intention for 95% precision and 33% recall, and estimate targeted contents for 70 to 88% accuracy.

Chapter 3 proposed the intelligent ink annotation framework that uses user's intention of document annotation based on the annotation recognition model proposed in Chapter 2. Traditional ink annotation systems require users to perform system-defined gestures to improve the availability of annotated documents. However, this results in decreasing the learnability of the system. The chapter therefore proposed the framework that can improve learnability of the annotation system without decreasing the learnability. The user study revealed the recognition accuracy and visual feedbacks are important to improve proposed framework in addition to find 75% of participants preferred proposed framework.

Chapter 4 proposed the EA Snippets which improves the search performance of handwritten documents based on the annotation recognition model proposed in Chapter 2. Snippets like the thumbnails that are the reduced image of original contents and text snippets which is the summarized text of original contents improves the performance of document navigation. However, the snippets of handwritten documents have problems. One is that we cannot grasp the summary from the scaled thumbnail image of handwritten document because its character can be too small to read. The other is that it is difficult to summarize the text of handwritten documents because of the insufficient recognition accuracy of handwritten characters. Therefore the chapter proposed the snippets of handwritten documents that are summarized based on emphasis handwriting annotation. The user study found proposed snippets improve search time 42% faster on average.

Thus, this part proposed the methods increasing the learnability of the handwriting annotation system and improving the availability of handwritten documents in digital handwriting environment through the proposal of the recognition model and two types of the application related to the user interfaces of digital handwriting environment. However, there is future work about the proposal. This part proposed the digital ink framework reducing learnability and the snippets improving availability, and then evaluated on the prototype systems by using the data collected by assuming actual situation. It is necessary to develop the system implementing proposed methods, and then evaluate in real-life situations. I believe these concept and initial works improve the user interfaces of digital handwriting environment in the future.

Part II. Data Analysis for Effective Information Extraction

Chapter 1. Introduction

This part focuses on the extraction of effective information from digital handwritten data as the study of the data analysis. On-line handwritten data, which is obtained by computers, includes additional features such as pressure and velocity in addition to paper-based handwriting. The data have the potential to extract effective information. This part proposes the two kinds of methods that extract effective information from on-line handwritten data.

One is extracting the psychological state of students described in Chapter 2. The student's psychological information is indispensable for teachers to understand the understandings of student and teach students suitable for their understanding. However, the traditional studies did not try to detect such information from online handwritten data though previous studies revealed the extraction of cognitive load from on-line handwritten data. The proposed method detects the psychological state of students such as frustration and need help from on-line handwritten data.

The other is the estimation of human memory level described in Chapter 3. Rote learning, which is a memorization technique based on repetition like memorizing Kanji and English word in Japanese students, is required to grasp the incompletely memorized items for efficient memorization since to learn completely memorized items waste their time. Handwriting behavior of learners have the potential that can improve the estimation of human memory level while traditional researches detect that by using the result of recall test and subjective evaluation. The method estimates the degree of human memory by using their on-line handwritten data to realize the effective rote learning system.

Chapter 2. Student Frustration Detection using Handwriting Behavior

2.1. Introduction

Good teachers typically expend great effort to identify states of student frustration during learning activities. Indeed, many will simply walk around the classroom during learning exercises, observing student behavior directly. Given the inefficiency of this approach, it is clear that a more automated technique for detecting student frustration would be a tremendous benefit to teachers and to the many Intelligent Tutoring Systems (ITS) now emerging.

Research on computer-assisted instruction (CAI), which helps students to learn by using computers, have been conducted for a long time [56] [57]. In such studies, detecting student frustration is important to help students suitable for individual comprehension. Some researchers have provided theoretical and practical foundations for developing such technique based on the log of the learning system on computer [58] [59] [60] [61] and some sensor data such as video, behaviors on computer and biosensors [62] [63] [64]. However these researches increase the burden of students, and are difficult to apply to the pen-based learning that is currently employed widely in primary and secondary education. Furthermore, there is a research to report that GUI interfaces using mouse and keyboard prevent students from thinking than pen-based interfaces [65].

Therefore, this chapter examines the relationship between student frustration and pen activity with the aim of providing information for teaching assistance tools and intelligent tutoring systems that use handwritten digital input. This chapter presents our findings regarding discriminative features of pen activity, as well as an explanation of proposed detection method.

2.2. Related Work

To the best of our knowledge, there is no research related to extract student frustration from on-line handwritten data but are researches detecting student frustration from other data and some findings to use on-line handwritten data for the detection. Lazard et al. [58] proposed a teacher support tool monitoring both students' learning activities (Positive, Negative and Neutral) and their progress in mathematical learning. Kapoor et al. proposed an automated technique for predicting student frustration [62], based on input from a video camera, pressure-sensitive mouse, skin conductance sensor, and pressure-sensitive chair. These researches detect student frustration by using additional sensors such as a video camera. This can result in increasing the burden of students and requires additional equipment. We therefore propose the method detecting student frustration by using the data obtained from a digital handwriting.

On the other hand, there is a finding related to detecting the status of writers from on-line handwritten data. Yu et al. have investigated a number of handwriting features for evaluating cognitive load [17]. They collected on-line handwritten data in answering to question the problems of English composition which take 3 levels of cognitive loads. The paper reported that the maximum value of pressure factor and the minimum value of velocity are effective to estimate cognitive load from the results of analyzing features like pressure factor, velocity and length of stroke extracted from the collected data. The results revealed that the psychological state of human appears in the on-line handwritten data. Hence, detecting student frustration from on-line handwritten data can be possible since the frustration of students, which is the target of this chapter, is also related to the psychological state of human. This chapter examines the features of on-line handwritten data including the features using this research.

Based on these related works, this chapter examines the relationship between student frustration and pen activity with the aim of providing information for teaching assistance tools and intelligent tutoring systems that use handwritten digital input.

2.3. Method and Task Design

2.3.1. Task Description and Procedure

This section conducted a user study in which participants answered mathematics problems as their handwriting data was collected. Nine participants, all local university students between the ages of 22 and 24, were asked to answer three mathematic problems from high school and university entrance examinations on a WACOM 12WX. Figure 2.1 shows a screen capture of the developed collecting system and the collected data. The data includes timestamp, pen status (writing, erasing, or hovering), pressure, and motion coordinates. The system also provided buttons with which participants could express their learning status directly—specifically, an “I'm frustrated” button and an “I need help” button.

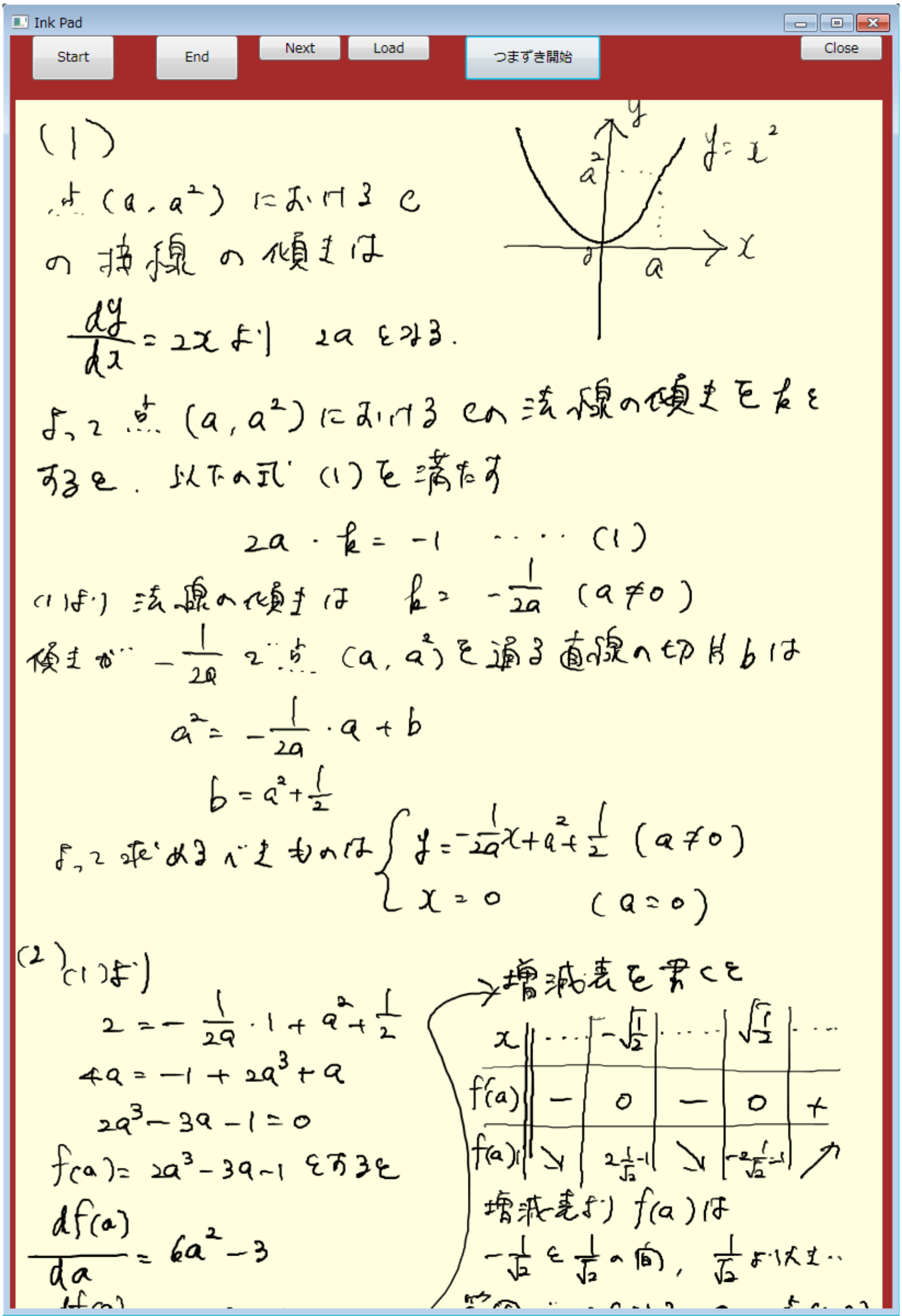


Figure 2.1 Screen capture of collecting system and example of collected data

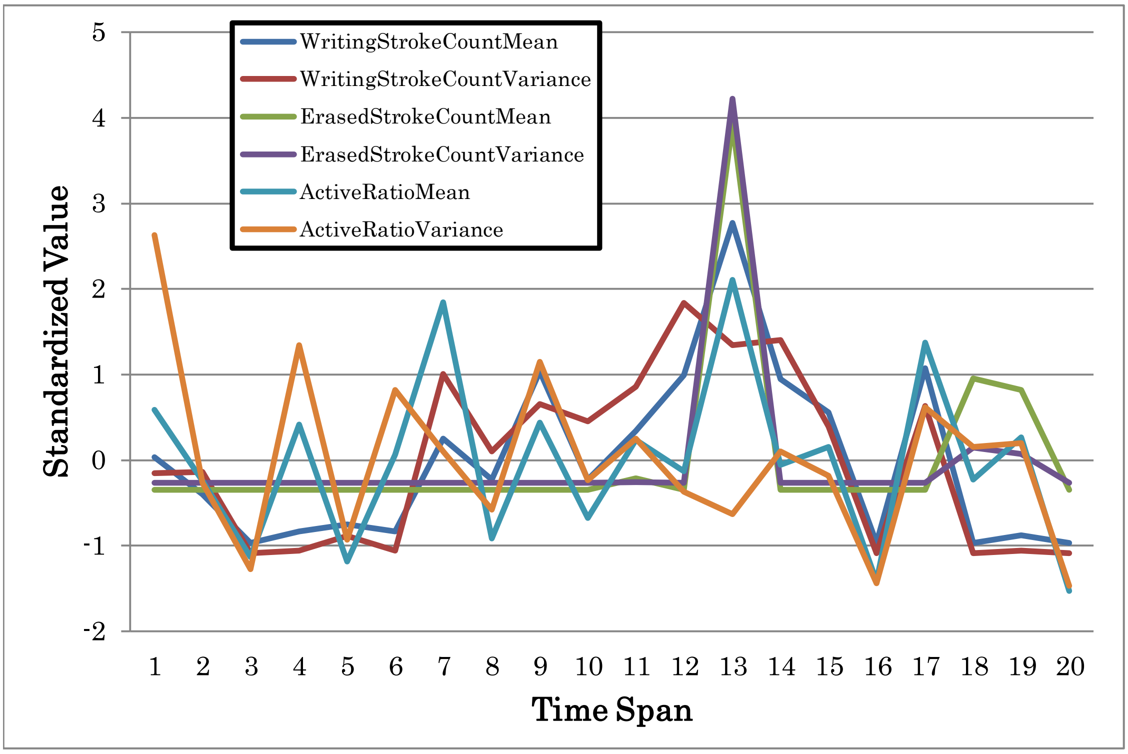
2.3.2. Feature Extraction and Detecting Method

Given that providing the frustration information to teachers or ITS, the detection should be complete within around 60 seconds. Accordingly, the method extracted 12 features within each 60-second window to obtain pen activity data. Each window was further divided into 6 time spans of which is 10 seconds each to extract the following 6 local features:

1) Writing Stroke Count:
   The number of writing strokes.
2) Erased Stroke Count:
   The number of erasure strokes.
3) Active Ratio:
   The ratio of time during which the pen is moving at more than 5 pixels/sec.
4) Pressure Factor:
   The mean value of pen's pressure factor
5) Stroke Speed:
   The mean value of writing stroke speed.
6) Air Speed:
   The mean value of air-stroke, i.e., non-contact, speed.

After extracting these features, the method calculated the mean and variance values for each feature within each window. Figure 2.2 shows the example of collected data.

For training data, the study tagged windows in which the "I need help" button was pushed as *NeedHelp* windows and windows in which the "I'm frustrated" button was pushed as *Frustration* windows. Windows that included the last writing stroke in an answer but were not tagged as *NeedHelp* windows were tagged as *Working* windows. With these tagged windows the method then trained a Support Vector Machine (SVM) with RBF kernel.



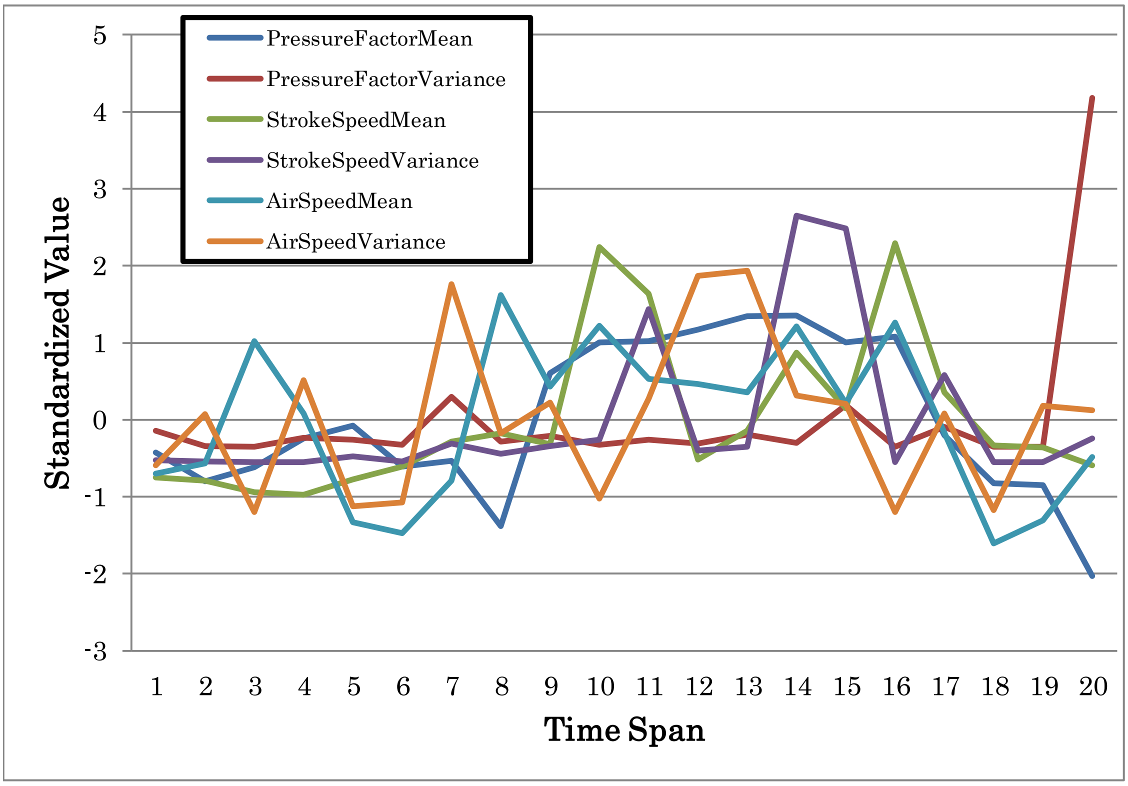


Figure 2.2 Example of collected handwritten data

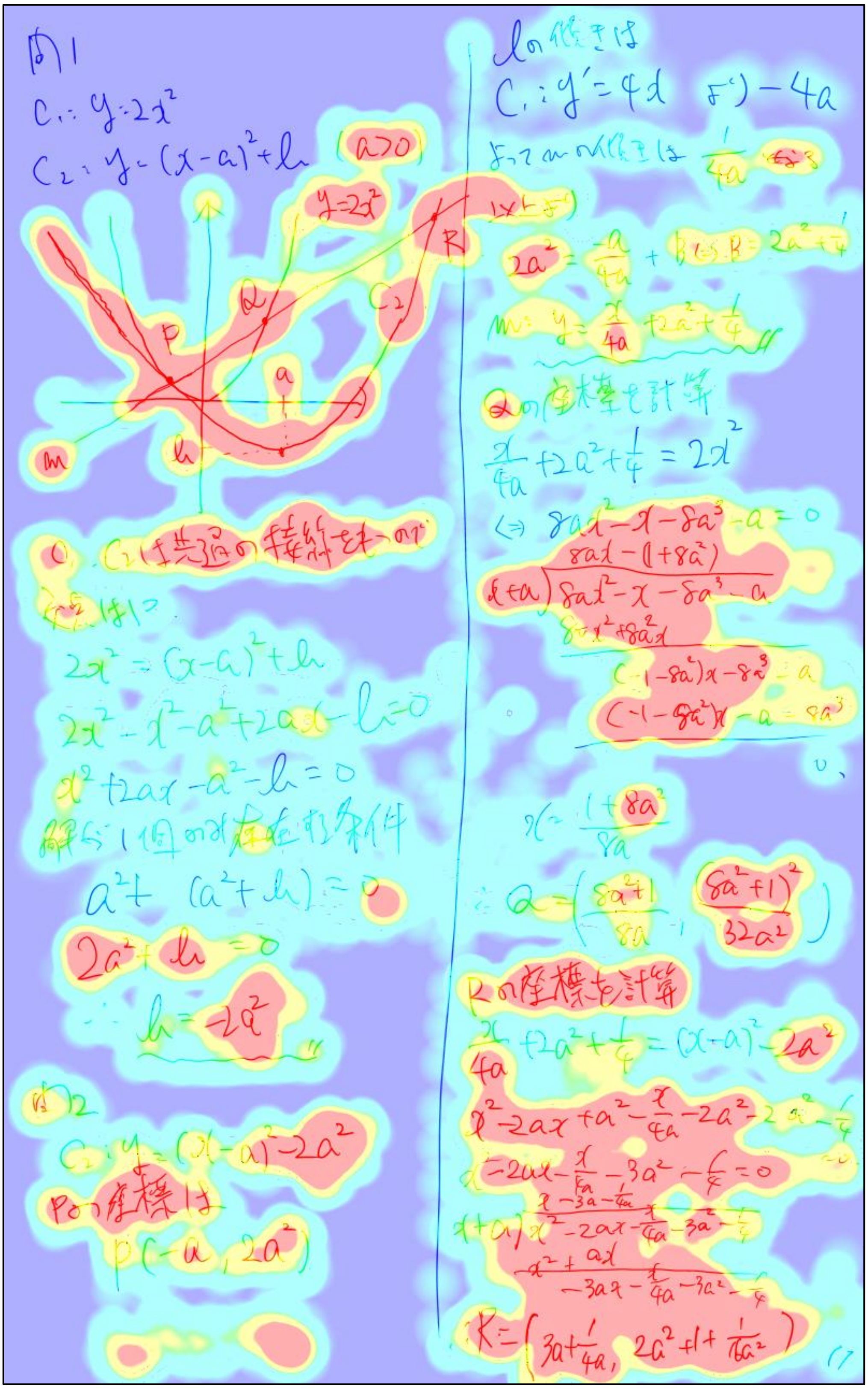


Figure 2.3 Visualization of frustrated state probability. This heat map overlay indicating zones of probable frustration

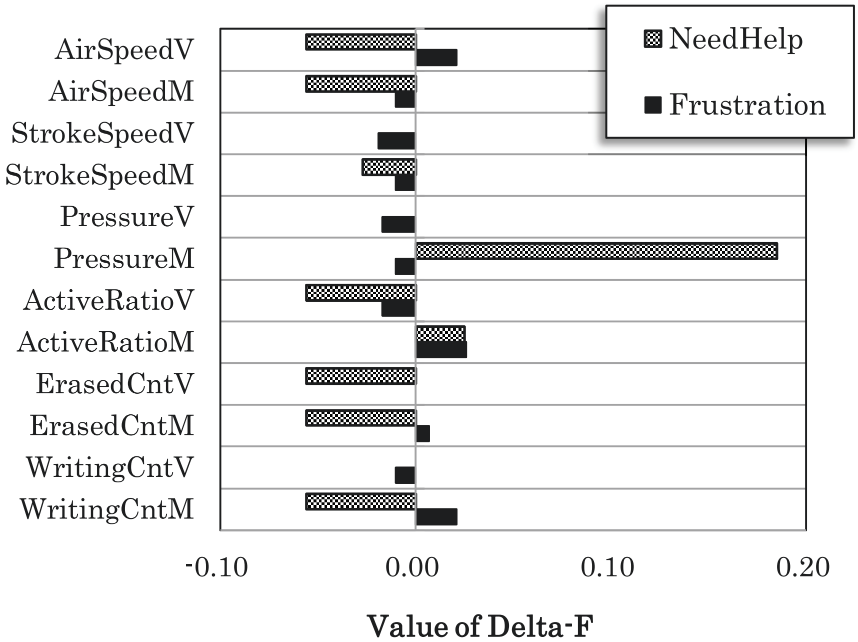


Figure 2.4 Delta-F values for each feature
Indicating the difference in F-measure between use of all features and use of all features except the given feature

2.4. Experimental Analysis

2.4.1. Feature Selection

Using the above detection method, this section investigated which features were most discriminative of the target states by leave-one-feature out cross validation. Figure 2.4 shows the result of this investigation, based on differences in F-measure. In the graph, we see that *PressureFactor Mean* and *ActiveRatio Mean* were the most discriminative features for detecting the *NeedHelp* state, and *AirSpeed Variance*, *ActiveRatio Mean*, *ErasedStrokeCount Mean* and *WritingStrokeCount Mean* were the most discriminative features for detecting the *Frustration* state.

2.4.2. Detection Accuracy and Visualization

Using the above feature combinations, the method then calculated the accuracy of proposed method using leave-one-window-out cross validation. The results are shown in Table 2.1. A sample visualization of the SVM probability score (we used "e1071" package of R) for the *Frustration* tag is shown in Figure 2.3.

Table 2.1 State detection results

State	# of window	Precision	Recall
NeedHelp	14	0.72	0.57
Frustration	59	0.87	0.90

2.5. Conclusion

Detecting student frustration is indispensable for teachers to know students' understandings. However, there was no research related to detect the frustration from handwritten data which are widely used in primary and secondary education. This chapter examined several extractable features of pen activity and determined which of these features were most related to states of frustration and need in problem solving. The examination found that AirSpeed Variance, ActiveRation Mean, ErasedStroke-Count Mean and WritingStrokeCount Mean were the most discriminative features for detecting the frustration, and then the proposed system achieve to detect the frustration in 87% precision and 90% recall. Based on our findings, we have developed a teacher assistant application, and are now considering the potential for other applications, such as smart user help systems for pen-based applications.

Chapter 3. Human Memory Level Estimation using Handwritten Data

3.1. Introduction

Memorizing certain things, which is one of the key ability of human being, is an indispensable for carrying out life in society. In educational situation, rote learning like memorizing Chinese characters is performed in everyday life. The goal of such rote learning is fixing memory of target things to remind at any time. Consequently, it is effective to realize the learning system that can fix memory in shorter learning time span.

The dual storage model proposed by Atkinson et al. [66] is well known as the studies about human memory level. In case of applying to Chinese character learning, the short term memory of Chinese characters obtained by recognizing in visual moves to long term memory when we perform a rehearsal which repeats outputs by writing of the memory. However, the forgetting curve proposed by Ebbinghaus [67] known as the study of forgetfulness, described that the memory will lost drastically in an hour, and then the increasing of the forgetfulness will be stable in a week if we do not perform a rehearsal.

The iterative learning which repeats same learning is required to prevent such forgetfulness. Implicit and explicit memory proposed by Graf et al. [68] are taken as the general idea to explain the iterative learning. Explicit memory represents the memory which involves conscious recollection, and implicit memory represents the memory which does not involves conscious recollection. As Terasawa et al. advocates that not explicit memory which is obtained by cramming but implicit memory which is obtained by the iterative learning is important to measure the effect of English word learning [69], it is thought that implicit memory obtained by the iterative learning is important to establish memory. In addition, Edge et al. proposed the learning system for mobile devices that schedules the iterative learning based on spacing repetitions theory [70]. Mizuno pointed out that it is effective to learn the item as priority which is not memorized completely [71] in the point of a reactivation theory [72].

With this in mind, the way to shorten time required for memorizing completely should be considered. The method to extract the memory which is not explicit memory, that is, the memorizing items which are not established is required to learn effectively. However, the item which can be recalled but not established, that is, the item which cannot be recalled in the future cannot be extracted by only using the result of recall test because the test can only extract the items that cannot be recalled in the point in time to answer the test. Moreover, the result of recall test cannot decide the priority of the iterate learning since only the binary value like correct and incorrect can be obtained from the test.

Therefore, this chapter focuses on not only the result of the recall test but also on-line handwritten data obtained by the device such as tablet PC and digital pen. On-line handwritten data includes not only pen moving coordinates but timestamp and pressure factor that inform the behavior of writer, and then the memorizing status of learners can appear in the data. The proposed system calculates the remembrance level which represents the establishing degree of learner's memory by using the results of recall test and on-line handwritten data. The effective rote learning system can be implemented since learning items which is completely memorized can be avoided to learn items which indicate low remembrance level as priority.

3.2. Related Work

This section initially describes the researches related to the scheduling for the iterative learning, and then describes the knowledge from past researches and the difference from proposed method. Next, this section describes the researches about the estimation of learner status to estimate establishment of memory from on-line handwritten data using proposed method, and then considers the feature extraction from on-line handwritten data.

3.2.1. Scheduling for Iterative Learning

The scheduling method to achieve effective iterative learning is required for the effective rote learning. However, the result of recall test is insufficient for the scheduling because only the forgotten items in the point of time can be extracted. Low-First spaces learning method [71] proposed by Mizuno pointed out as the research on the scheduling of determining the priority of iterative learning.

Low-First spaces learning method schedules iterative learning to refer the past results of recall test in addition to current recall test. The priority of iterative learning is calculated as the following weighted cumulative percentage of correct answers.

$$P_n = \sum_{i=1}^{n} 2^{-(n-i+1)} \times P_i$$

Where P_n represents the weighted cumulative percentage of correct answers after nth learning. n represents the number of learning. P_i represents ith answer rate.

It is possible to learn the items which currently indicate lower incorrect rate as priority in order from small P_n value.

Furthermore, Terasawa et al. proposed micro-step measuring method [69] which measures the invisible small outcome of English word learners in each learner. Micro-step measuring method is scheduled to learn items at regular intervals, and then measures the outcome of individual learner by using long term learning data such as the results of recall test and subjective evaluation values. The method is not targeted on the effective rote learning but measuring the outcome of learners. In addition, the method does not measure the memorizing status in each item but measure the learning outcome in each learner.

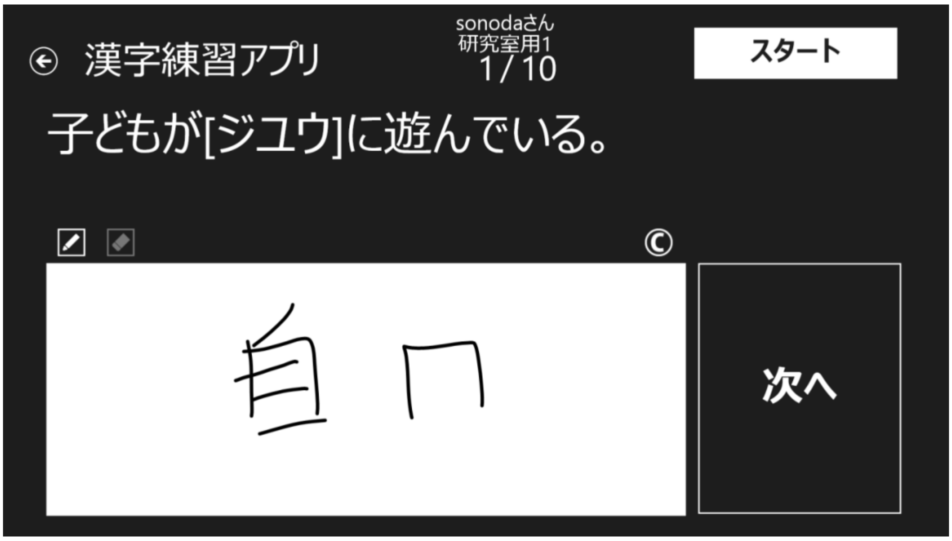
Such researches use cumulative the results of recall test and subjective evaluation values. This chapter, on the contrary, describes the estimation of the items that will forget in the future by using on-line handwritten data in addition to the results of recall test.

3.2.2. Estimation of Lerner Status

There are researches about estimating the outcome of iterative learning using Event-Related Potential (ERP) [73] and evaluating reading proficiency using functional Magnetic Resonance Imaging (fMRI) [74] as the researches of evaluating learning status by using biological information of learners not using learning data. However, it is difficult to apply daily learning since this equipment takes a burden on the learners. Due to this, this chapter estimates the learner status by using the information obtained by a pen which learner use in daily.

In connection with the information that can be obtained from a pen, Yu et al. mentioned the relationship between handwritten information and the cognitive load of writers [17]. The research collected on-line handwritten data in answering to question the problems of English composition which take 3 levels of cognitive loads. The paper reported that the maximum value of pressure factor and the minimum value of velocity are effective to estimate cognitive load from the results of analyzing features like pressure factor, velocity and length of stroke extracted from the collected data.

Moreover, on-line handwritten data also provides time stamp information. Ueno proposed the method detecting the problem which indicates abnormal reaction of learners by using learning time on e-learning contents [60] as the research of learning and time. This method can detect the self-report of abnormal by applying on-line outlier detection algorithm to learning time data.



(a) Answer input phase

(b) Answer confirmation phase

Figure 3.1 Screen captures of collecting system

Based on these researches, the features related to pressure factor, velocity and time can be effective for calculating the remembrance level. This chapter tries to estimate the remembrance level by using these features in addition to the result of recall test.

3.3. Data Collection about Human Memory

This section describes the experiment about collecting on-line handwritten information that needs to implement the calculation model of the remembrance level. The purpose of the experiment is collecting on-line handwritten data related to established and non-established memory of learners. The definitions of established and non-established memory in this chapter are shown in Table 3.1. The model estimating the non-established items, that is, the items which cannot recall in a week is implemented by using the collected data.

Table 3.1 Definitions of established memory

Terms	**Definitions**
Established Memory	The memory which the user can recall it over one week from learning
Non-Established Memory	The memory which the user forgets it within one week from learning

3.3.1. Experimental Environment

In the beginning, the experimental environment was implemented to collect the data. VAIO Duo 11, which is the tablet PC by SONY, is used for collecting on-line handwritten data. The tablet equipped with 1920 * 1080 pixels multi-touch display which accepts touch and stylus input. The software is implemented as Windows Store App on Windows 8.1. The experiment of the data collection was conducted as following procedure.

STEP 1: Answer Input Phase (Figure 3.1a)

The problem of Kanji dictation is presented after the 3 seconds count down. The reason of presenting count down is to obtain the feature related to answer time. Participants write the answer to the center area on the display. Participants can erase the answer for removing incorrect handwriting. The system moves to STEP 2 by touching the NEXT button after answering the problem.

STEP 2: Answer Confirmation Phase (Figure 3.1b)

The system moves to answer confirmation phase after the STEP 1. This phase collects the information whether participants could recall answer or not, and whether participants feels the memory is established or not. Due to this, participants input the subjective evaluation and the result of the recall test. Participants initially confirm the answer by checking the displayed answer. After that, participants select the result in the three types of self-report about memorizing status. The kinds of the button are as follows:

1. "Already remembered" button: Subjective established memory
   This button is selected when the answer of the participant is correct and participants feel the memory is established. The answer selected this button is labeled as subjective established memory.
2. "Learn again" button: Subjective non-established memory
   This button is selected when the answer of the participants is correct and participants feel the memory is non-established. The answer selected this button is labeled as subjective non-established memory.
3. "Incorrect" button: Not memorized
   This button is selected when the answer of the participants is incorrect. The answer selected this button is labeled as not memorized.

STEP 2 is completed when participants select the button. Then it moves to STEP 1 if there are non-answered problems, and it is finished if there is no non-answered problem. By the above flow, the system collects the on-line handwritten data in each problem and the remembrance level information of participants.

3.3.2. Experimental Procedure

This section describes the experiment using the collecting system described in 3.3.1. 11 university students who belong to Waseda University were invited in the experiment. The experiment collected the data in the following procedure to collect established and non-established memory of participants. Note that the operating procedure was instructed to participants before starting the experiments.

Confirmation of Problems (3 minutes)

Participants confirmed the problems and its answer. The given 50 problems were in the Japan Kanji Aptitude Test Grade 2. The purpose of this phase is generating non-established memory that is obtained by short time learning to indicate the answers of the problems in advance. All of participants were shown the answers in 3 minutes.

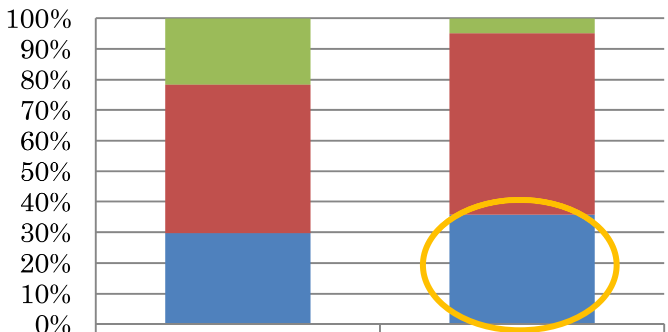
Implementation of Test (After the Confirmation)

The experiment conducted recall test using the collecting system after finishing the confirmation of problems. The purpose of this phase is collecting answer data including non-established memory that is occurred in previous phase.

Re-Implementation of Test (One Week Later)

The experiment conducted same recall test again after one week from the previous test. The result of this test detects objectively non-established memory. For instance, the problem can be decided as objectively non-established memory if the problem that is corrected in the test just after the confirmation will be incorrect in the test after one week from the confirmation. In contrast, the problem can be decided as established memory if both the test just after the confirmation and the test after one week from the confirmation are correct. The forgetting curve proposed by Ebbinghaus [67] reported the amount of the forgetting was stable after one week from learning while the amout of the forgetting was dramatically increasing just after learning. Consequently, the experiment set interval as one week between two recall tests since the non-established memory obtained by short term learning almost can be forgotten in one week.

By the above procedure, the experiment collected on-line handwritten data related to non-established and established memory.



# of problems with	First Time	Second Time
Subjective Non-Established Memory	119	27
Subjective Established Memory	267	326
Not Memorized	164	197

Figure 3.2 Changes of recall test results

3.3.3. Analysis of Collected Data

This section describes the statistics about the collected data. Figure 3.2 shows the changing of the recall test results. The graph represents the number of the problems in subjective non-established memory in second time decreases at 27 against the number of the problems in subjective non-established memory in first test are 119. This is considered that the established memory have changed to non-established memory or not memorized since the amount of subjective non-established memory increases.

Figure 3.3 shows the memory status breakdown at the first time in “Not Memorized” problems at the second time (the target item set is represented in Figure 3.2 with an orange color circle). This graph represents around 60% of the problems are “Not Memorized” at the first time in “Not Memorized” problems at the second time. That is, only 60% of not-established memory can be extracted by extracting incorrect problems in recall test. On the other hand, the rest 20% of the non-established memory is subjective established memory at the first recall test. The other rest 20% of the non-established memory is subjective non-established memory at the first recall test. That represents at most 80% of the non-established memory can be extracted without taking into account the precision if the rote learning system which requires users to report subjective evaluation is implemented. That is, the subjective judgment of memory level is not necessarily correct since the memory which learners think subjectively established includes 20% of non-established memory in fact. This chapter proposes the method predicting non-established memory without relying on the subjectively evaluated value.

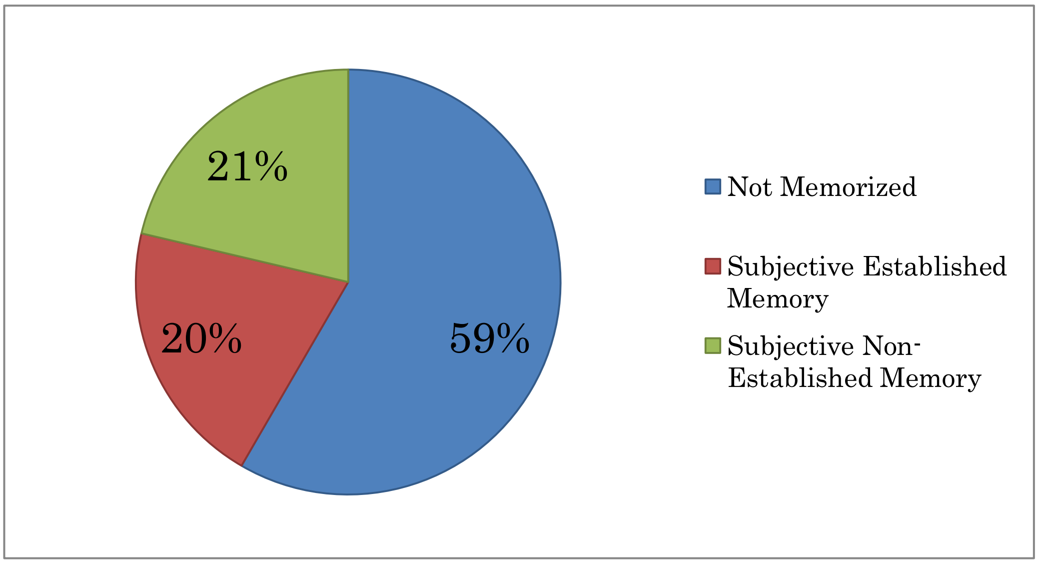


Figure 3.3 Memory status breakdown at first recall test in “Not Memorized” problems at the second recall test

3.4. Remembrance Level Calculation

This section describes the method calculating the remembrance level based on the collected data.

3.4.1. Outline of Remembrance Level Calculation

The purpose of proposed model is calculating the remembrance level which indicates the degree of establishing memory from the data when learners answer the recall test. Figure 3.4 shows the calculation flow of the remembrance level. Proposed model uses the result of recall test and on-line handwritten data when learners answer the test as the input data. The handwritten features are extracted from on-line handwritten data, and then input the calculation model with the results of recall test. The calculation model of the remembrance level output continuous value from 0 to 1 by using support vector machine (SVM) with RBF kernel.

3.4.2. Generation of Learning Dataset

Proposed model uses the answer data collected in 3.3 as the learning data. To use as the learning data, established memory (positive) and non-established memory (negative) are generated as the dataset from the collected data. The generating process is as follows.

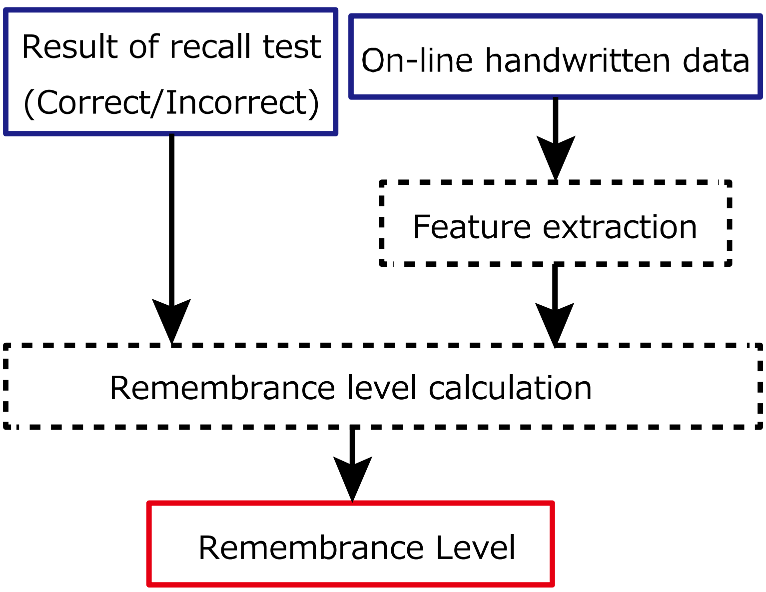


Figure 3.4 Calculation flow of remembrance level

Established Memory (Positive)

The answer which is correct in the second recall test, that is, the memory which enables to recall for one week is treated as established memory. The dataset of established memory is generated to extract the result and the on-line handwritten data at the first recall test in the problem which is correct at the second recall test in spite of the result of the first test.

Not-Established Memory (Negative)

The answer which is incorrect at the second recall test, that is, the memory which does not enables to recall after one week from learning is treated as not-established memory. The dataset of not-established memory is generated to extract the result and the on-line handwritten data at the first recall test in the problem which is incorrect at the second recall test in spite of the result of the first test.

3.4.3. Feature Extraction

This section describes the features using proposed model. The following features are extracted as the candidate of using proposed model.

A) The number of using eraser
B) <u>Time from giving problem to first writing</u>
C) <u>Time from last writing to answer completion</u>
D) Average time between handwritten strokes
E) Maximum time between handwritten strokes
F) <u>Maximum value of pressure factor</u>
G) <u>Average value of pressure factor</u>
H) Variance value of pressure factor
I) Maximum velocity value of handwritten stroke
J) Minimum velocity value of handwritten stroke
K) Average velocity value of handwritten stroke
L) <u>Variance velocity value of handwritten stroke</u>

The result of recall test is also as the candidate of using proposed model in addition to these extracted features. Future selection using machine learning is conducted to select discriminative features for estimating not-established memory in the candidates. Figure 3.5 shows the result of forward stepwise feature selection based on F-value by using SVM. The six features selected by the feature selection are the result of the test and handwritten features shown in underline (B, C, F, G and L).

Figure 3.5 Result of forward stepwise feature selection

3.4.4. Remembrance Level Calculation

This section describes the remembrance level calculation model using the features in the result of recall test and on-line handwritten data as the input data. The model applies SVM and calculates probability value of SVM.

First, the classification model of SVM is generated by using the dataset of established and non-established memory described in 3.4.2. Next, the model detect non-established memory by inputting the features such as the result of recall test and on-line handwritten features extracted in 3.4.3. Furthermore, the model calculate class-probability estimates as the remembrance level.

3.5. Evaluation of Remembrance Level Estimation

This section describes the experiment evaluating the proposed non-estimated memory estimation.

3.5.1. Evaluation of Not-Established Memory Detection

In the beginning, this section describes the accuracy of non-established memory detection. The data collected in 3.3 is used to the evaluation. Table 3.2 shows the detection performance of non-established memory detection.

Table 3.2 Results of non-established memory detection

Feature	Result of recall test	Subjective evaluation	Proposed method
Precision	70.12%	55.48%	68.14%
Recall	58.38%	79.70%	70.56%
F-value	0.6371	0.6542	0.6932

Leave-one-subject-out (LOSO) cross validation is used to calculate precision and recall in the results. "Result of recall test" represents the detection method extracting incorrect problems at the recall test as non-established memory. "Subjective evaluation" represents the detection method extracting the problems that a learner thinks it will be forgotten in the future as non-established memory. "Proposed method" represents the method proposed in 3.4. Table 3.2 shows "Subjective evaluation" indicates the best recall value. On the other hand, the table represents "Result of recall test" indicates the best accuracy value. The proposed method shows the highest F-value which indicates the performance of the detection.

3.5.2. Performance of Proposed Model

The proposed method finally outputs the remembrance level in addition to the detection of non-established memory. The traditional method using the result of recall test or subjective evaluation value can only output binary decision. In contrast, proposed method enables to output the remembrance level which is continuous value from 0 to 1. Figure 3.6 shows the ROC curve indicating the detection performance of non-established memory by proposed method.

The graph represents the remembrance levels which is proposed method can select false positive rate freely achieving almost same performance of traditional method. The graph also represents the true positive rate shows 80% if the false positive rate is set to 40%, and the true positive rate shows 70% if the false positive rate is set to 20%.

From the above results, proposed method enables to determine the priority of iterative learning in each memorizing item. The effective rote learning system can be achieved by using the remembrance level.

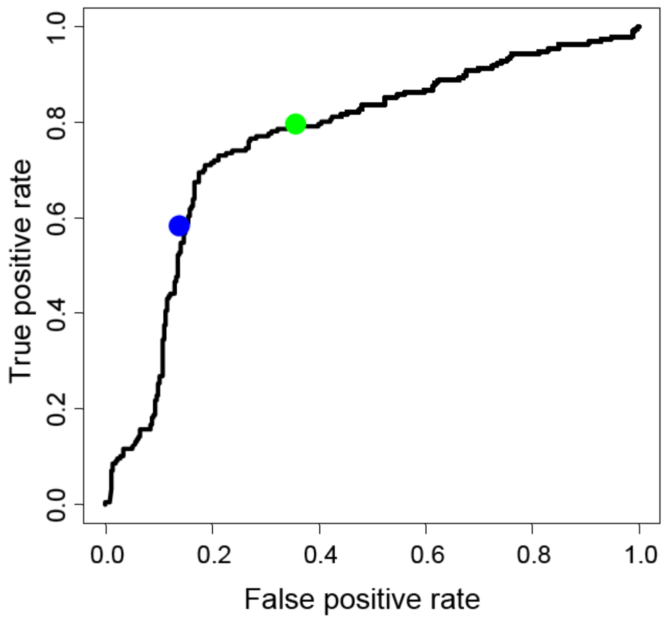


Figure 3.6 ROC curve indicating the detection performance of non-established memory (Blue represents the estimation using result of recall test Green represents the estimation using subjective evaluation)

3.6. Conclusion

Detecting human memory level of memorizing items is important for the effective rote learning. Traditional method using the result of recall test and subjective evaluation cannot detect all non-established memory. This chapter described the method calculating the remembrance level which represents the degree of establishing memory by using on-line handwritten data. Proposed model can output the continuous remembrance level from 0 to 1 while traditional method using the result of recall test and subjective evaluation output binary decision only. The scheduling of iterative learning which learns non-established memory as priority can be achieved by using the continuous remembrance level. In addition, the experiment shows proposed model achieved the best performance with 0.69 F-value in comparison with traditional methods.

Chapter 4. Conclusion

On-line handwritten data, which is obtained in digital handwriting environment, includes additional features such as pressure factor and velocity in addition to paper-based handwriting. The data have the potential to extract effective information by some data mining methods. This part proposed two methods that extract effective information from on-line handwritten data.

Chapter 2 proposed the method extracting the psychological state of students. The student's psychological information when they are learning is indispensable for teachers to both understand the understandings of student and teach students suitable for their understanding. However, there is no research detecting such information from on-line handwritten data. The proposed method achieved to detect two types of student's status from on-line handwritten data: frustration and need help.

Chapter 3 proposed the method estimating human memory level. Rote learning, which is a memorization technique based on repetition like memorizing Kanji and English word learning of Japanese students, is required to grasp the incompletely memorized items for efficient iterative learning since learning completely memorized items wastes their time. Traditional methods used the result of recall test and subjective evaluation to detect them. The chapter proposed the method estimating the remembrance level from on-line handwritten data that was not used in traditional method. As the result, the method achieved to both improve the detection accuracy of non-established memory and calculate the continuous remembrance level of memorizing items, that is, the method enables to calculate the priority in each memorizing item of iterative learning.

In this way, this part described the effective information extraction from invisible on-line handwritten data through proposing two methods of on-line handwritten data analysis. However, this part did not propose the application systems using proposed extraction method. In the future, we have to develop the systems and evaluate the effectiveness of proposed extraction method. Though there is still remaining future work, I believe these analyzing methods disclose the new research field that is effective information extraction from on-line handwritten data, and then I believe the field provides new values to digital handwriting environment in the future.

Conclusion of This Thesis

While mouse and keyboard have been adopted as the input device of desktop PC, multi-touch screen has been adopted as the input device of tablet PC which is currently developing in the world. In the multi-touch interface on a tablet device, users navigate information by touching displayed objects and input information by touching the keyboard displayed on the display; however, though suitable for navigating information, the interface is unsuitable for inputting information such as diagrams and free-form note-taking. Pen-based computing is a promising technology for input device of tablet PCs.

There are mainly two kinds of studies about pen-based computing; 1) interaction technique, and 2) recognition methods. The previous interaction techniques about pen-based computing reduce the learnability of the system in exchange of implementing additional functions. Therefore, Part I proposed the intelligent user interfaces that help users performing digital handwriting to detect user's intention from input data. On the other hand, most of previous recognition methods of digital handwritten data focused on visible handwritten data. In digital handwriting environment; however, computers also obtain "invisible" handwritten data like a pressure factor and velocity. These invisible handwritten data have a potential to extract effective information. Hence, Part II attempted to extract effective information from invisible handwritten data.

Part I. User Interfaces for Digital Handwriting Annotation

Part I focused on handwriting annotation on electronic documents which is indispensable interaction between human and documents. The handwriting annotation on electronic documents requires the availability maximizing the advantage of electronic documents on computers. However, the user interfaces of traditional research related to handwriting annotation have the problems either the lack of the document availability or the lack of the learnability in the system. Therefore, this part proposed the intelligent user interfaces for handwriting annotation on electronic documents both to increase the learnability of the annotation system and to enhance the availability of electronic documents.

Chapter 2 proposed the recognition model of handwriting annotation on electronic documents. The recognition model generated by human annotation data achieved improving the availability of documents to recognize the annotation selecting range on contents while traditional research using heuristic model hand difficulty to recognize the exact selected range. The study found the proposed model can find user's content-targeting interaction for 95% precision, and estimate selected range of contents for 70 to 88% accuracy.

Based on the annotation recognition model, Chapter 3 proposed the intelligent ink annotation framework that uses user's intention of document annotation. Traditional ink annotation systems require users to perform system-defined gestures to improve the availability of annotated documents. However, this results in decreasing the learnability of the system. The chapter therefore proposed the framework that can improve learnability of the annotation system without decreasing the learnability. The user study found proposed framework was preferred to 75% of participants.

On the other hand, Chapter 4 proposed the EA Snippets which improves the search performance of handwritten documents based on the recognition model. Snippets like thumbnail images improve the performance of document navigation. However, the snippets of handwritten documents are that we cannot grasp the summary from the scaled thumbnail image of handwritten document because its character can be too small to read. In addition, it is difficult to summarize the text of handwritten documents because of the insufficient recognition accuracy of handwritten characters. Therefore the chapter proposed the snippets of handwritten documents that are summarized based on emphasis handwriting annotation. The user study found proposed snippets improve search time 42% faster on average.

Part II. Data Analysis for Effective Information Extraction

On-line handwritten data includes additional "invisible" features such as pressure factor and velocity in addition to paper-based handwriting. The data have the potential to extract effective information by some data mining methods. This part proposed two methods that extract effective information from on-line handwritten data.

Chapter 2 proposed the method extracting the psychological state of students. The student's psychological information when they are learning is indispensable for teachers to both understand the understandings of student and teach students suitable for their understanding. The proposed method achieved to detect two types of student's status from on-line handwritten data: frustration and need help. The experiments showed proposed method detects the frustration in 87% precision and 90% recall.

Chapter 3 proposed the method estimating human memory level. Rote learning, which is a memorization technique based on repetition, is required to grasp the incompletely memorized items for efficient iterative learning. The chapter proposed the method estimating the remembrance level from on-line handwritten data that was not used in traditional method. As the result, the method enabled to calculate the priority in each memorizing item of iterative learning. The experiments showed proposed model achieved the best performance with 0.69 F-value.

Discussion and Future Work

Part I focused on digital handwriting on electronic documents as the use case of intelligent user interfaces that help user to perform digital handwriting by detecting user's intention from input data. The part proposed the methods increasing the learnability of the handwriting annotation system and improving the availability of handwritten documents in digital handwriting environment through the proposal of the recognition model and two types of the application related to the user interfaces of digital handwriting environment. Meanwhile, the part didn't do enough discussed with respect to how degree the interface intervene user's operation. Moreover, we have to apply and verify proposed method to other use cases as the future work.

On the other hand, Part II focused on the extraction of the effective information from on-line handwritten data through proposing two methods of on-line handwritten data analysis. These proposed methods enabled to extract effective information in educational situations. We have to verify the effectiveness to develop the application system using this extracted information as the future work.

While there is still remaining future work, I believe these concept and initial work provide new values both the user interfaces of digital handwriting environment and the data analysis of on-line handwritten data.

References

[1] Charles C. Tappert, Ching Y. Suen, and Toru Wakahara, "The State of the Airt in On-Line Handwriting Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 8, pp. 787-808, 1990.

[2] Rejean Plamondon and Sargur N. Srihari, "On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 63-84, 2000.

[3] Cheng-Lin Liu, Stefan Jaeger, and Masaki Nakagawa, "Online Recognition of Chinese Characters: The State-of-the-Art," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 26, no. 2, pp. 198-213, 2004.

[4] Joseph J. LaViola and Robert C. Zeleznik, "MathPad2: A System for the Creation and Exploration of Mathematical Sketches," *In ACM SIGGRAPH 2006*, no. 33, 2006.

[5] Yuji Kasuya and Hayato Yamana, "MathBox: Interactive Pen-based Interface for Inputting Mathematical Expressions," *In Proc. of the 12th International Conference on Intelligent User Interfaces*, pp. 274-277, 2007.

[6] Alexandra Mendes, Roland Backhouse, and Joao F. Ferreira, "Structure Editing of Handwritten Mathematics: Improving the Computer Support for the Calculational Method," *In Proceedings of the 9th ACM International Conference on Interactive Tabletops and Surfaces*, pp. 139-148, 2014.

[7] Gem Stapleton, Beryl Plimmer, Aidan Delaney, and Peter Rodgers, "Combining Sketching and Traditional Diagram Editing Tools," *ACM Transactions on Intelligent Systems and Technology*, vol. 6, no. 1, p. No. 10, 2015.

[8] Levent Burak Kara and Thomas F. Stahovich, "Hierarchical Parsing and Recognition of Hand-Sketched Diagrams," *In Proc. of ACM SIGGRAPH 2007*, p. No. 17, 2007.

[9] Wayne Citrin: Mark D. Gross, "Distributed Architectures for Pen-based Input and Diagram Recognition," *In Proc. of the workshop on Advanced Visual Interfaces*, pp. 132-140, 1996.

[10] Hyunyoung Song et al., "PenLight: Combining a Mobile Projector and a Digital Pen for Dynamic Visual Overlay," *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI2009)*, pp. 143-152, 2009.

[11] Adriano Scoditti, Renaud Blanch, and Joëlle Coutaz, "A Novel Taxonomy for Gestural Interaction Techniques based on Accelerometers," *In Proceedings of the 16th international conference on Intelligent user interfaces (IUI2011)*, pp. 63-72, 2011.

[12] David Lee, KyoungHee Son, Joon Hyub Lee, and Seok-Hyung Bae, "PhantomPen: Virtualization of Pen Head for Digital Drawing Free Form Pen," *In Proceedings ot the 25th annual ACM Symposium on User interface software and technology (UIST2012)*, pp. 331-340, 2012.

[13] Hyunyoung Song et al., "Grips and Gestures on a Multi-Touch Pen," *In Proc. of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1323-1332, 2011.

[14] Ken Hinckley et al., "Pen + Touch = New Tools," *In Proceedings of the 23nd annual ACM symposium on User interface software and technology (UIST2010)*, pp. 27-36, 2010.

[15] Huawei Tu, Xiangshi Ren, and Shumin Zhai, "A Comparative Evaluation of Finger and Pen Stroke Gestures," *In Proc. of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1287-1296, 2012.

[16] Ken Hinckley et al., "InkSeine: In Situ Search for Active Note Taking," *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI2007)*, pp. 251-260, 2007.

[17] Kun Yu, Julien Epps, and Fang Chen, "Cognitive Load Evaluation of Handwriting using Stroke-level Features," *In Proceedings of the 16th international conference on Intelligent user interfaces (IUI2011)*, pp. 423-426, 2011.

[18] Catherine C. Marshall, "Annotation: from paper books to the digital library," *In Proceedings of the second ACM international conference on Digital libraries*, pp. 131-140, 1997.

[19] Sacha Zyto, David Karger, Mark Ackerman, and Sanjoy Mahajan, "Successful Classroom Deployment of a Social Document Annotation System," *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI2012)*, pp. 1883-1892, 2012.

[20] Craig S. Tashman and W. Keith Edwards, "LiquidText:A Flexible, Multitouch Environment to Support Active Reading," *In Proceedings of the SIGCHl Conference on Human Factors in Computing Systems (CHI2011)*, pp. 3285-3294, 2011.

[21] Dongwook Yoon, Nicholas Chen, and François Guimbretière, "TextTearing: Opening White Space for Digital Ink Annotation," *In Proceedings of the 26th annual ACM symposium on User interface software and technology (UIST2013)*, pp. 107-112, 2013.

[22] Beryl Plimmer, Samuel Hsiao-Heng Chang, Meghavi Doshi, Laura Laycock, and Nilanthi Seneviratne, "iAnnotate: Exploring Multi-user Ink Annotation in Web Browsers," *In Proceedings of the Eleventh Australasian Conference on User Interface (AUIC2010)*, vol. 106, pp. 52-60, 2010.

[23] Maneesh Agrawala and Michael Shilman, "DIZI: a Digital Ink Zooming Interface for Document Annotation," *In Proceedings of the 2005 IFIP TC13 international conference on Human-Computer Interaction (INTERACT 2005)*, pp. 69-79, 2005.

[24] Daniel Kaneider, Thomas Seifried, and Michael Haller, "Automatic Annotation Placement for Interactive Maps," *In Proceedings of the 2013 ACM international conference on Interactive tabletops and surfaces (ITS2013)*, pp. 61-70, 2013.

[25] Nicholas Chen, Francois Guimbretiere, and Abigail Sellen, "Designing a Multi-slate Reiding Environment to Support Active Reading Activities," *ACM Transaction on Computer-Human Interaction (TOCHI)*, vol. 19, no. 3 (18), 2012.

[26] Muhd Dzulkhiflee Hamzah, Shun'ichi Tano, Mitsuru Iwata, and Tomonori Hashiyama, "Effectiveness of Annotating by Hand for Non-alphabetical Languages," *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI2006)*, pp. 841-850, 2006.

[27] Xin Wang, Michael Shilman, and Sashi Raghupathy, "Parsing Ink Annotations on Heterogeneous Documents," *In Proceedings of the Third Eurographics conference on Sketch-Based Interfaces and Modeling (SBM2006)*, pp. 43-50, 2006.

[28] Bill N. Schilit, Gene Golovchinsky, and Morgan N. Price, "Beyond Paper: Supporting Active Reading with Free Form Digital Ink Annotations," *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI1998)*, pp. 249-256, 1998.

[29] Jr. Dan R. Olsen, Trent Taufer, and Jerry Alan Fails, "ScreenCrayons: Annotating Anything," *In Proceedings of the 17th annual ACM Symposium on User interface software and technology (UIST2004)*, pp. 165-174, 2004.

[30] Gene Golvchinsky and Laurent Denoue, "Moving Markup: Repositioning Freeform Annotations," *In Proceedings of the 15th annual ACM symposium on User interface software and technology (UIST2002)*, pp. 21-30, 2002.

[31] Michael Shilman and Zile Wei, "Recognizing Freeform Digital Ink Annotaions," *Document Analysis Systems, LNCS*, vol. 3163, pp. 322-331, 2004.

[32] Manuel J. Foseca, César Pimentel, and Joaquim A. Jorge, "CALI: An Online Scribble Recognizer for Calligraphic Interfaces," *In Proceedings of the 2002 AAAI Spring Symposium*, vol. 2, pp. 51-58, 2002.

[33] David Bargeron and Tomer Moscovich, "Reflowing Digital Ink Annotations," *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI2003)*, pp. 385-393, 2003.

[34] Mortimer J. Adler and Charles Van Doren, *How to Read a Book.*: Touchstone, 1972.

[35] Xin Wang and Sashi Raghupathy, "Ink Annotations and their Anchoring in Heterogeneous Digital Documents," *In Proceedings of Ninth International Conference on Document Analysis and Recognition (ICDAR2007)*, vol. 1, pp. 163-167, 2007.

[36] Kenton O'Hara and Abigail Sellen, "A Comparison of reading paper and on-line documents," *In Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems (CHI1997)*, pp. 335-342, 1997.

[37] Fabrice Matulic and Moria C. Norrie, "Supporting Active Reading on Pen and Touch-operated Tabletops," *In Proceedings of the International Working Conference on Advanced Visual Interfaces (AVI2012)*, pp. 612-619, 2012.

[38] Fabrice Matulic and Moria C. Norrie, "Pen and Touch Gestural Environment for Document Editing on Interactive Tabletops," *In Proceedings of the 2013 ACM international conference on Intaractive Tabletops and Surfaces (2011)*, pp. 41-50, 2013.

[39] Peter Brandl, Christoph Richter, and Michael Haller, "NiCEBook: Supporting Natural Note Taking," *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI2010)*, pp. 599-608, 2010.

[40] Anne Aula, Rehan M. Khan, Zhiwei Guan, Paul Fontes, and Peter Hong, "A Comparison of Visual and Textual Page Previews in Judging the Helpfulness of Web Pages," *In Proceedings of the 19th international conference on World wide web (WWW2010)*, pp. 51-60, 2010.

[41] Susan Dziadosz and Raman Chandrasekar, "Do Thumbnail Previews Help Users Make Better Relevance Decisions about Web Search Results?," *In Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR2002)*, pp. 365-366, 2002.

[42] Zhu Blian and Masaki Nakagawa, "On-line Handwritten Japanese Characters Recognition using a MRF Model with Parameter Optimization by CRF," *In Proceedings of International Conference on Document Analysis and Recognition (ICDAR)*, pp. 603-607, 2011.

[43] I. S. Amrutha, S. S. Shylaja, S. Natarajan, and K. N. Balasubramanya, "A Smart Automatic Thumbnail Cropping based on Attention Driven Regions of Interest Extraction," *In Proceedings of the 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human (ICIS2009)*, pp. 957-962, 2009.

[44] Shai Avidan and Ariel Shamir, "Seam Carving for Content-aware Image Resizing," *ACM Transactions on Graphics*, vol. 26, no. 3(10), 2007.

[45] Bongwon Suh, Haibin Ling, Benjamin B. Bederson, and David W. Jacobs, "Automatic Thumbnail cropping and its Effectiveness," *In Proceedings of the 16th annual ACM symposium on User interface software and technology (UIST2003)*, pp. 95-104, 2003.

[46] Heidi Lam and Patrick Baudisch, "Summary Thumbnails: Readable Overviews for Small Screen Web Browsers," *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI2005)*, pp. 681-690, 2005.

[47] Jaime Teevan et al., "Visual Snippets: Summarizing Web Pages for Search and Revisitation," *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI2009)*, pp. 2023-2032, 2009.

[48] Allison Woodruff, Andrew Faulring, Ruth Rosenholtz, Julie Morrsion, and Peter Pirolli, "Using Thumbnails to Search the Web," *In Proceedings of the SIGCHl Conference on Human Factors in Computing Systems (CHI2001)*, pp. 198-205, 2001.

[49] Don Willems, Stéphane Rossignol, and Louis Vuurpijl, "Mode detection in on-line pen drawing and handwriting recognition," *In Proceedings of Eighth International Conference on Document Analysis and Recognition (ICDAR)*, vol. 1, pp. 31-35, 2005.

[50] C.M. Bishop, M. Svensen, and G.E. Hinton, "Distinguishing Text from Graphics in On-line Handwritten Ink," *In Proceedings of the ninth international workshop on Frontiers in Handwriting Recognition (IWFHR2004)*, pp. 142-147, 2004.

[51] Xiang-Dong Zhou and Cheng-Lin Liu, "Text/Non-text Ink Stroke Classification in Japanese Handwriting Based on Markov Random Fields," *In Proceedings of Ninth International Conference on Document Analysis and Recognition (ICDAR2007)*, pp.

377-381, 2007.

[52] Xiang Ao, Junfeng Li, Xuang Wang, and Guozhong Dai, "Structuralizing Digital Ink for Efficient Selection," *In Proceedings of the 11th international conference on Intelligent user interfaces (IUI2006)*, pp. 148-154, 2006.

[53] Keisuke Mochida and Masaki Nakagawa, "Separating Figures, Mathematical Formulas and Japanese Text from Free Handwriting in Mixed On-Line Documents," *International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI)*, vol. 18, no. 7, pp. 1173-1187, 2004.

[54] Michael Shilman, Zile Wei, Sashi Raghupathy, Patrice Simard, and David Jones, "Discerning Structure from Freeform Handwritten Notes," *In Proceedings of Seventh International Conference on Document Analysis and Recognition (ICDAR)*, vol. 1, pp. 60-65, 2003.

[55] Takeo Igarashi, Satoshi Matsuoka, and Toshiyuki Masui, "Adaptive Recognition of Implicit Structures in Human-Organized Layouts," *In Proceedings of 11th IEEE International Symposium on Visual Languages (VL1995)*, pp. 258-266, 1995.

[56] Jaime R. Carbonell, "AI in CAI: An Artificial-Intelligence Approach to Computer-Assisted Instruction," *IEEE Transactions on Man-Machine Systems*, vol. 11, no. 4, pp. 190-202, 1970.

[57] Tsuneo Kuwabara et al., "Support Functions for Stalled Students and Their Effect in a Multi-Media Assisted Education System with Individual Advance (MESIA)," *IEICE Journal (D)*, vol. 83, no. 9, pp. 1013-1024, 2000.

[58] Darren Pearce-Lazard, Alexandra Poulovassilis, and Eirini Geraniou, "The Design of Teacher Assistance Tools in an Exploratory Learning Environment for Mathematics Generalisation," *Lecture Notes in Computer Science*, vol. 6383, pp. 260-275, 2010.

[59] Norikatsu Fujita, Hiroki Zushi, Toshihiro Hayashi, and Toshinori Yamasaki, "Learning management system including functions for detecting the learner's stalled situation in e-learning," *IEICE Technical Report ET*, vol. 105, no. 632, pp. 63-68, 2006.

[60] Maomi Ueno, "Online Outlier Detection for e-Learning Time Data (in Japanese)," *IEICE Journal (D)*, vol. 90, no. 1, pp. 40-51, 2007.

[61] Yoshihiro Nakamura, Norio Akamatsu, Tsuneo Kuwabara, and Motonori Tamaki, "Method for Detecting a Learner's Stalled Situation Using a Dispersion of Operation Time Intervals," *IEICE Journal (D)*, vol. 85, no. 1, pp. 79-90, 2002.

[62] Ashish Kapoor, Winslow Burleson, and Rosalind W. Picard, "Automatic Prediction of Frustration," *International Journal of Human-Computer Studies*, vol. 65, no. 8, pp. 724-736, 2007.

[63] Tomoya Kurokawa, Kiyoshi Nosu, and Kiyoyuki Yamazaki, "A Research on the Time Sequential Estimation of Emotions of e-Learning Users by the Face and Biometric Information," *ITE Transactions on Media Technology and Applications*, vol. 61, no. 12, pp. 1779-1784, 2007.

[64] Kazuaki Nakamura, Koh Kakusho, Masayuki Murakami, and Michiko Minoh, "Estimating Learners' Subjective Impressions of the Difficulty of Course Materials by Observing Their Faces in e-Learning," *IEICE Journal (D)*, vol. 93, no. 5, pp. 568-578, 2010.

[65] Sharon Oviatt, Alex Arthur, and Julia Cohen, "Quiet Interfaces that Help Students Think," *In Proceedings of the 19th annual ACM symposium on User interface software and technology (UIST2006)*, pp. 191-200, 2006.

[66] R.C. Atkinson and R.M. Shiffrin, "Human Memory: A Proposed System and its Control Processes," *Psychology of Learning and Motivation, Academic Press*, vol. 2, pp. 89-195, 1968.

[67] Hermann Ebbinghaus, *Über das Gedächtnis.*: Leipzig: Duncker & Humblot, 1885.

[68] P. Graf and D.L. Schacter, "Implicit and explicit memory for new associations in normal and amnesic subjects," *Journal of Experimental Psychology: Learning, Memory and Cognition*, vol. 11, pp. 501-518, 1985.

[69] Takafumi Terasawa, Tetsuya Yoshida, and Nobuo Ohta, "Solving the essential problem for behavioral predictions with longitudinal big data: Application of the scheduling principle controlling temporal dimension factors to education (in Japanese)," *In Proceedings of IPSJ/IEICE FIT Forum*, pp. 557-564, 2013.

[70] ArthurM. Glenberg and ThomasS. Lehmann, "Spacing repetitions over 1 week," *Journal of Memory and Cognition*, vol. 8, no. 6, pp. 528-538, 1980.

[71] Rika Mizuno, "A Test of the Effectiveness of the Low-First Spaces Learning Method with CAI (in Japanese)," *Journal of Japan Society for Educational Technology*, vol. 24, no. 2, pp. 111-120, 2000.

[72] Rika Mizuno, "Exploring the cause of spacing effects: A test of a reactivation theory by experiments and simulations (in Japanese)," *Cognitive studies: bulletin of the Japanese Cognitive Science Society*, vol. 4, no. 2, pp. 20-38, 1997.

[73] Mariko Funada, Tadashi Funada, Miki Shibukawa, and Kanji Akahori,

"Quantification and Analysis of the Efficiency of Iterative Learning by Using Event Related Potentials (in Japanese)," *Journal of Japan Society for Educational Technology*, vol. 35, no. 3, pp. 193-203, 2011.

[74] Satoru Yokoyama, IPutu Susila, Takeshi Osawa, and Ryuta Kawashima, "Assessment of English learners' reading skills by utilizing functional brain imaging (in Japanese)," *IEICE Technical Report (TL)*, vol. 109, no. 297, pp. 57-62, 2009.

[75] Mo Chen, Jian-Tao Sun, Hua-Jun Zeng, and Kwok-Yan Lam, "A Practical System of Keyphrase Extraction for Web Pages," *In Proceedings of the 14th ACM international conference on Information and knowledge management (CIKM2005)*, pp. 277-278, 2005.

[76] Craig S. Tashman and W. Keith Edwards, "Active Reading and its Discontents: the Situations, Problems and Ideas of Readers," *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI2011)*, pp. 2927-2936, 2011.

[77] Darren Edge, Stephen Fitchett, Michael Whitney, and James Landay, "MemReflex: Adaptive Flashcards for Mobile Microlerning," *In Proceedings of the 14th international conference on Human-computer interaction with mobile devices and services companion (MobileHCI 2012)*, pp. 431-440, 2012.

Appendix

Questions of User Study (Chapter 4)

Search by Keywords Related to Emphasized Word

TASK1

1. 川下戦略とは、商社が「川下」すなわちスーパーや○○○○○○○○○○○など消費者に近い流通企業と提携することで取引の拡大を図ること。
2. フリークエント・ショッパーズ・プログラムとは、ポイントカードや会員カードを発行することで顧客の○○○を高め、優良な顧客との長期的な関係を築き、「生涯顧客」を確保するマーケティング手法である。
3. ベンチャーキャピタルとは、ベンチャービジネスに対して投資する企業や、その○○○○のことをさす。
4. ネット銀行とは一般にインターネット専業銀行をさす。ネットによって決済できるため、○○も使わず人件費も抑えられ、預金金利の上乗せがしやすい。

　流通系列化とは，メーカーや卸売店や小売店をグループ内に取り込み、販路や○○のコネクションを作ること。

TASK2

1. エクステリアとは，○○○○○の逆のこと
2. 公定歩合とは，日本銀行が，金融機関に資金の貸し出しを行う際に用いる○○金利.
3. バリューチェーンとは○○○○のこと.
4. PFI とは民間の資金，ノウハウを使う○○○○整備.
5. バイヤーとは，店の方針に従って，消費者に提供するための商品をメーカーや○○から仕入れる担当者のこと.

TASK3

1. プロダクト・ライフサイクルとは、ひとつの商品が開発され市場に投入されてから姿を現すまでの一連のプロセスを○○の一生にたとえて表した言葉。
2. イールド・スプレッドとは金利を目安とする株価水準の判断指標で○○○○ともいわれる。
3. インテリジェントビルとは、高度な情報通信かに対応する機能と、最先端のビルオートメーション機能などが整備された、○○○○○オフィスビルのこと。
4. マネーサプライとは、国内に流通している○○の量のことをさす。

　チェーンストアオペレーションとは，発注や売上管理，在庫管理などの事務処理を

○○○○本部が集中して行い，店舗では販売だけに専念するという経営手法.

TASK4

1. ディスクロージャーとは，企業が自ら経営活動や財務内容などを○○○○することをいう.
2. 瑕疵担保責任とは，住宅に注意して見ても発見が難しい欠陥があった場合．○○が買主に負う責任のこと.
3. プラントエンジニアリングとは，石油，化学，電力，通信，鉄鋼などの○○○○(プラント)の計画から建設運転までを一貫して請け負う業種.
4. クロスマーチャンダイジングとは．関連した商品をカテゴリーにとらわれず，一つの売り場や○○○○に集中して配置することで，相乗効果による売上増を図る手法.
5. 純粋持ち株会社とは，一般に自らは実務的な事業をせず，主に他企業の○○を保有支配してその株式配当で収入を得る会社のことを指す.

Search by Keywords Not Related to Emphasized Word

TASK1

1. ○○○○とは、商社が「川下」すなわちスーパーやコンビニエンスストアなど消費者に近い流通企業と提携することで取引の拡大を図ること。
2. ○○○○○○○○○○○○○○○○○○○○○○とは、ポイントカードや会員カードを発行することで顧客の再来率を高め、優良な顧客との長期的な関係を築き、「生涯顧客」を確保するマーケティング手法である。
3. ○○○○○○○○○○○とは、ベンチャービジネスに対して投資する企業や、その投資資金のことをさす。
4. ○○○○○とは一般にインターネット専業銀行をさす。ネットによって決済できるため、店舗も使わず人件費も抑えられ、預金金利の上乗せがしやすい。
5. ○○○○○とは，メーカーや卸売店や小売店をグループ内に取り込み、販路や流通のコネクションを作ること。

TASK2

1. ○○○○○○○○○○○○○○○とは、ひとつの商品が開発され市場に投入されてから姿を現すまでの一連のプロセスを生物の一生にたとえて表した言葉。
2. ○○○○○○○○○○○とは金利を目安とする株価水準の判断指標で利回り差ともいわれる。
3. ○○○○○○○○○○○とは、高度な情報通信かに対応する機能と、最先端のビルオートメーション機能などが整備された、高付加価値オフィスビルのこと。
4. ○○○○○○○とは、国内に流通している貨幣の量のことをさす。
5. ○○○○○○○○○○○○○○○○とは，発注や売上管理，在庫管理などの事務処理をチ

ェーン本部が集中して行い，店舗では販売だけに専念するという経営手法.

TASK3

1．〇〇〇〇〇〇〇〇〇〇とは，企業が自ら経営活動や財務内容などを一般公開することをいう.

2．〇〇〇〇〇〇とは，住宅に注意して見ても発見が難しい欠陥があった場合．売主が買主に負う責任のこと.

3．〇〇〇〇〇〇〇〇〇〇〇〇〇〇とは，石油，化学，電力，通信，鉄鋼などの生産設備(プラント)の計画から建設運転までを一貫して請け負う業種.

4．〇〇〇〇〇〇〇〇〇〇〇〇〇〇〇とは．関連した商品をカテゴリーにとらわれず，一つの売り場やコーナーに集中して配置することで，相乗効果による売上増を図る手法.

5．〇〇〇〇〇〇〇とは，一般に自らは実務的な事業をせず，主に他企業の株式を保有支配してその株式配当で収入を得る会社のことを指す.

Comparison of the User's Own Notes and the Notes of Others

TASK1

1．IMAX とは，超大型の〇〇〇〇〇〇〇のこと．巨大スクリーンの投影で，臨場感と立体感が味わえる．(S)

2．官製談合とは，国の職員（〇〇〇〇〇，特殊法人）が落札業者の決定に関与すること（F)

3．瑕疵担保責任とは，住宅に注意してみても発見が難しい欠陥があった場合，〇〇が飼い主に負う責任のこと．(F)

4．川下戦略とは、商社が「川下」すなわちスーパーや〇〇〇〇〇〇〇〇〇〇〇など消費者に近い流通企業と提携することで取引の拡大を図ること。(I)

5．バリューチェーンとは〇〇〇〇のこと．開発・生産・販売といった一連の業務において，それぞれの段階を付加価値を生み出すステップと捉えるビジネスモデル．(I)

TASK2

1．構造計算とは，建築物が重さ（自重や積載荷重など）や〇〇（地震，風など）に対して，どの程度の強度があるかを客観的な数値として表すために行う計算．(K)

2．プラントエンジニアリングとは，石油，化学，電力，〇〇，鉄鋼などの生産設備（プラント）の計画から建設，運転までを一貫して請け負う業種．(K)

3．映倫とは，映画における〇〇〇〇の審査機関．(S)

4．在来工法とは，日本の伝統的な工法．柱，はり，筋交い・・・〇〇〇〇〇（F)

5．ダンピングとは，ある商品を輸出する際に，その商品の〇〇〇〇〇〇を下回った価格で輸出すること．(I)

TASK3

1．免震構造とは，建物と地盤との間に，特殊な○○○をつけることで，地震が起きた時の地面の揺れが建物に伝わりにくくするように設計した構造のこと．（K）

2．都市再生緊急整備地域とは，2002 年 6 月に施行された○○○○○○○○○法に基づき，都市再生の拠点として，都市開発事業等による緊急かつ重点的に市街地整備を行うべき地域．（K）

3．デジタルシネマとは，映像を○○○○○○で記録・保存した映画のこと．（S）

4．住宅性能表示制度は，10 分野○○項目で客観的評価を行う．（F）

5．コンバージョンとは，オフィスビル等に，部屋の区切りを作り，マンションとして転売するなど既存の建物の○○○○○を変更して，引き続き活用すること．（K）

Publications

Journals

1. Hiroki Asai and Hayato Yamana, “A Recognition Model of Selected Regions Indicated by Handwritten Annotations on Electronic Documents,” IPSJ Transactions on Databases, Vol.7, No.4, pp.1-12, 2014. (in Japanese)
2. Shota Tezuka, Hiroki Asai and Hayato Yamana, “An Image Blur Alert System for Mobile Device Cameras,” DBSJ Journal, Vol.13-J, No.1, pp.58-63, 2014. (in Japanese)
3. Hiroki Asai and Hayato Yamana, “Extraction of Emphasized Words from On-line Handwritten Notebooks,” DBSJ Journal, Vol.10, No.1, pp.67-72, 2011. (in Japanese)

International Conferences, Symposiums and Workshops

1. Hiroki Asai and Hayato Yamana, "Detecting Learner's To-Be-Forgotten Items using Online Handwritten Data," In Proc. of the 15th Annual ACM SIGCHI_NZ Conference on Computer-Human Interaction (CHINZ), Hamilton, New Zealand, 2015.
2. Hiroki Asai and Hayato Yamana, "Intelligent Ink Annotation Framework that uses User's Intention in Electronic Document Annotation," In Proc. of the 9th ACM International Conference on Interactive Tabletops and Surfaces (ITS), Dresden, Germany, 2014.
3. Shunya Okuno, Hiroki Asai and Hayato Yamana, "A Challenge of Authorship Identification for Ten-thousand-scale Microblog Users," 2014 IEEE International Conference on Big Data (IEEE BigData), Washington DC, USA, 2014.
4. Hiroki Asai and Hayato Yamana, "EA Snippets: Generating Summarized View of Handwritten Documents based on Emphasis Annotations," In Proc. of the 16th International Conference on Human-Computer Interaction (HCI International), Heraklion, Greece, 2014.
5. Hiroki Asai and Hayato Yamana, "Detecting Student Frustration based on Handwriting Behavior," In Proc. of the 26th Annual ACM Symposium on User Interface Software and Technology (UIST), St Andrews, UK, 2013.
6. Hiroki Asai, Takanori Ueda and Hayato Yamana, "Legible Thumbnail: Summarizing On-line Handwritten Documents based on Emphasized Expressions," In Proc. of the 13th International Conference on Human-Computer Interaction with Mobile Devices and Services (Mobile HCI), Stockholm, Sweden, 2011.

Domestic Forums, Symposiums and Workshops

1. Chen Wang, Hiroki Asai and Hayato Yamana, "視線情報を利用した理解できる言語の推定," IEICE General Conference, ISS-SP-231, 2015. (in Japanese)
2. Kazuya Uesato, Hiroki Asai and Hayato Yamana, "Twitter ユーザを対象とした属性推定手法の精度向上－周辺ユーザの属性補完を利用して－," The 7th Forum on Data Engineering and Information Management (DEIM), 2015. (in Japanese)
3. Syunya Okuno, Hiroki Asai and Hayato Yamana, "マイクロブログを対象とした 100,000 人レベルでの著者推定手法の提案," The 7th Forum on Data Engineering and Information Management (DEIM), 2015. (in Japanese)
4. Kazuya Onizawa, Hiroki Asai and Hayato Yamana, "字配りの平均化による手書き文章の魅力的な文字配置方法," The 7th Forum on Data Engineering and Information Management (DEIM), 2015. (in Japanese)
5. Zhuo Lu, Chen Wang, Hiroki Asai and Hayato Yamana, "3 軸加速度計を用いたデスクワーク中の割り込み可能性の推定," The 7th Forum on Data Engineering and Information Management (DEIM), 2015. (in Japanese)
6. Hiroki Asai and Hayato Yamana, "オンライン手書き情報を用いた未定着記憶推定システム," IPSJ SIG Technical Report (CE), 2014-CE-127(1), pp.1-6, 2014. (in Japanese)
7. Shunya Okuno, Hiroki Asai and Hayato Yamana, "マイクロブログを対象とした著者推定手法の提案－10,000 人レベルでの著者推定－," IPSJ SIG Technical Report (DBS), 2014-DBS-159(12), pp.1-6, 2014. (in Japanese)
8. Guanyung Zhou, Hiroki Asai and Hayato Yamana, "Topics and Influential User Identification in Twitter using Twitter Lists," IPSJ SIG Technical Report (DBS), 2014-DBS-159(13), pp.1-6, 2014.
9. Kazuya Uesato, Masahiro Tanaka, Hiroki Asai and Hayato Yamana, "メンション情報を利用した Twitter ユーザプロフィール推定における単語重要度算出手法の考察," IPSJ SIG Technical Report (DBS), 2014-DBS-159(22), pp.1-6, 2014. (in Japanese)
10. Hiroki Asai and Hayato Yamana, "電子ドキュメント上での書き込みを支援する手書きアノテーション認識モデル," The 6th Forum on Data Engineering and Information Management (DEIM), 2014. (in Japanese)
11. Shogo Sonoda, Hiroki Asai and Hayato Yamana, "認知心理学的記憶調査に基づく記憶支援システムの構築," The 6th Forum on Data Engineering and Information

Management (DEIM), 2014. (in Japanese)

12. Chen Wang, Kazuya Onizawa, **Hiroki Asai** and Hayato Yamana “入力ストローク数削減による高速手書き入力手法,” The 6th Forum on Data Engineering and Information Management (DEIM), 2014. (in Japanese)
13. Shota Tezuka, **Hiroki Asai** and Hayato Yamana, “携帯端末に適用可能なモーションブラー警告システム,” The 6th Forum on Data Engineering and Information Management (DEIM), 2014. (in Japanese)
14. Shunya Okuno, **Hiroki Asai** and Hayato Yamana, “マイクロブログを対象とした5,000 人レベルでの著者推定手法の提案－5,000 人レベルでの著者推定－,” WebDB Forum, 2013. (in Japanese)
15. Kazuya Uesato, Takashi Okutani, **Hiroki Asai**, Shunya Okuno, Masahiro Tanaka and Hayato Yamana, “文体及びツイート付随情報を用いた乗っ取りツイート検出,” IPSJ SIG Technical Report (DBS), 2013-DBS-158(21), pp.1-8, 2013. (in Japanese)
16. **Hiroki Asai** and Hayato Yamana, “A Summarization Technique for Handwritten Document Search using Emphasis Annotations,” iDB Workshop, 2013.
17. Shunya Okuno, **Hiroki Asai** and Hayato Yamana, “マイクロブログを対象とした1,000 人レベルでの著者推定手法構築に向けて,” IPSJ SIG Technical Report (DBS), 2013-DBS-157(7), pp.1-6, 2013. (in Japanese)
18. **Hiroki Asai**, Sayaka Akioka and Hayato Yamana, “きたああああああああああああああああ!!!!!11: マイクロブログを用いた教師なし叫喚フレーズ抽出,” The 5th Forum on Data Engineering and Information Management (DEIM), 2013. (in Japanese)
19. Akari Nozawa, Shogo Sonoda, **Hiroki Asai** and Hayato Yamana, “jQuery プラグインに特化したアニメーションサムネイル自動生成手法,” The 5th Forum on Data Engineering and Information Management (DEIM), 2013. (in Japanese)
20. Shogo Sonoda, **Hiroki Asai** and Hayato Yamana, “オンライン手書きデータによる記憶支援システムの構築,” The 5th Forum on Data Engineering and Information Management (DEIM), 2013. (in Japanese)
21. Yusuke Yamamoto, **Hiroki Asai**, Takanori Ueda, Sayaka Akioka and Hayato Yamana “テレビ番組に対する意見を持つ Twitter ユーザのリアルタイム検出,” The 5th Forum on Data Engineering and Information Management (DEIM), 2013. (in Japanese)
22. Takanori Ueda, **Hiroki Asai**, Shino Fujiki, Yusuke Yamamoto, Hiromasa Takei, Sayaka Akioka and Hayato Yamana, “ソーシャルメディアを含む多メディアビッグデ

ータの統合的解析による情報抽出," IPSJ SIG Technical Report (DBS), 2012-DBS-156(8), pp.1-6, 2012. (in Japanese)

23. Hiroki Asai, Akari Nozawa, Shogo Sonoda and Hayato Yamana, "筆記情報と時系列モデルを用いた学習者つまずき検出," IEICE Technical Report (ET), 112(269), pp.65-70, 2012. (in Japanese)
24. Hiroki Asai and Hayato Yamana, "強調表記を利用した手書きドキュメント検索スニペット生成," IPSJ SIG Technical Report (DBS), 2012-DBS-154(8), pp.1-7, 2012. (in Japanese)
25. Hiroki Asai, Akari Nozawa, Shogo Sonoda and Hayato Yamana, "オンライン手書きデータを用いた学習者のつまずき検出," The 4th Forum on Data Engineering and Information Management (DEIM), 2012. (in Japanese)
26. Shogo Sonoda, Hiroki Asai and Hayato Yamana, "オンライン手書きデータによる記憶度推定システムの構築," The 4th Forum on Data Engineering and Information Management (DEIM), 2012. (in Japanese)
27. Yuta Takahashi, Yoshiki Kataoka, Hiroki Asai, Yusuke Yamamoto, Sayaka Akioka and Hayato Yamana, "繰り返し表現を用いた感情を含むツイートの抽出手法," The 4th Forum on Data Engineering and Information Management (DEIM), 2012. (in Japanese)
28. Hiroki Asai, Daisuke Kobayashi and Hayato Yamana, "筆記者の強調表現に基づいたオンライン手書きノートの圧縮サムネイル生成手法," The 3th Forum on Data Engineering and Information Management (DEIM), 2011. (in Japanese)

Patents

1. Hayato Yamana, Shogo Sonoda and **Hiroki Asai**, “記憶度推定装置および記憶度推定プログラム,” 特開 2014-032320.