

## 訓点資料を対象とした翻刻支援システムの構築および評価 Construction and Evaluation of a Transcription Support System for Classical Documents with Reading Marks

田中 勝<sup>\*1</sup>      村川 猛彦<sup>†</sup>      宇都宮 啓吾<sup>‡</sup>  
Masaru Tanaka    Takehiko Murakawa    Keigo Utsunomiya

### 1. はじめに

国語学や仏教学において、奈良・平安・鎌倉時代の言語や文化を知るために、訓点資料の解読が進められており、解読作業の効率化・高品質化を支援する環境が望まれている[1]。現在、様々な古写経の電子化が進められている中で、訓点資料については電子化がほとんど進んでいないのが現状である[2]。解読作業の電子的な支援や学生などの教育を支援する環境の整備が遅れており、訓点資料の電子画像化および電子テキスト化とともに、それを扱うデジタルアーカイブシステムの提供が求められている。

訓点資料のアーカイブ化にあたり、訓点資料の読解、およびテキスト化のそれぞれにおいて、課題が存在する。資料上には大きさが不揃いな漢字・送り仮名・ヲコト点が存在しており、墨を使った手書きに訓点が付いた資料であるため、計算機による文字認識は容易ではない。人手でテキスト化を行うにしても、漢字と訓点の組み合わせやヲコト点の読みなどが雑多に存在しているため、かなりの手間を要することとなる。

筆者らはこれまで、上述の課題に対応した解読支援環境の確立を目指し、訓点資料を対象としたデジタルアーカイブシステム、具体的には、訓点資料のテキスト化を支援する機能や、それを扱うデータベースシステムについて、実装および構築を行ってきた[3][4]。本論文ではその特徴ならびに主要な機能等を説明するとともに、実施した評価実験について報告する。

### 2. 準備

#### 2.1 訓点資料とは

訓点とは、漢文の訓読のために書き加えられた仮名や諸符号(句読点、返り点、ヲコト点)をいい、訓点資料とは、それらの訓点を付けられた資料のことである。諸符号のうち、ヲコト点とは、頻出する助詞や助動詞などの代用として用いられた表記で、その働き(点の読み)は、漢字に対する打点位置と点の形状によって決まる。

漢文に訓点を書き加えるのは、奈良時代の写経から始まり、平安、鎌倉時代へと続いている。中でも平安時代が顕著であり、この時代の資料だけでも 5,000 点以上現存していると言われている[5]。そういった歴史の中で定められてきたヲコト点の形式は、様々な宗派・流派等によってそれぞれ異なっており、それらを特定することで、資料の来歴などを突き止めることができる。



図 1 訓点資料の一部と文字の詳細

訓点資料の一例として、『千手千眼陀羅尼経残巻』を挙げる。これは、奈良時代の古写経であり、国宝として京都国立博物館に展示され、e 国宝の Web サイトにて公開されている[6]。図 1 は、この經典画像の一部と文字の詳細である。資料には、漢字・送り仮名・ヲコト点が見られる。この資料で用いられているヲコト点の形式は宝幢院点であることが判明している[7]。

頻出する形状における打点位置は、ヲコト点図として整理されている[5]。図 1 において漢字「求」の下部に打点されている「・」は、宝幢院点のヲコト点図と照合して、「む」と読むことができる。送り仮名とヲコト点の両方が見られるのは、漢文、送り仮名、ヲコト点を書いた(書き加えた)時期が異なるためである。

#### 2.2 翻刻の必要性

近年、IT 環境の整備と計算機の普及に伴い、古典籍のデジタルアーカイブ化が徐々に進んでいる。仏典の翻刻では、SAT 大正新脩大藏経テキストデータベース[8]がよく知られている。そこでは、漢訳仏典『大正新脩大藏経』の全テキストをデータベース化して公開しており、漢訳された仏典を対象とする研究を行う上で有用なシステムとなっている。

国内における過去の仏教を基に、当時の文化を研究する際、仏典が伝来してきた当時の解釈を知る必要があり、そのためには、人々が仏典を訓読する際に訓点を付けて書き写した訓点資料が有用となる。しかしながら、訓点資料は、漢訳仏典と比較すると、漢文に訓点が付与された複雑な文書構造となっているためか、翻刻や、そのための環境構築は、ほとんど進んでいないのが現状である[2][9]。そこで筆者らはこれまで、訓点資料を対象としたデジタルアーカイブシステムの構築に取り組んできた。

<sup>†</sup> 和歌山大学, Wakayama University

<sup>‡</sup> 大阪大谷大学, Osaka Ohtani University

\*1 現在, 大和ハウス工業株式会社, Presently with Daiwa House Industry Co.,Ltd.

訓点資料を構成している要素としては、漢字、送り仮名、ヲコト点といった資料から直接見て取れる一次情報が挙げられるが、それらを組み合わせることで、本文テキスト(漢字)の行番号・列番号、漢字と訓点の関連(組み合わせ)情報、ヲコト点の打点位置・読みなどのメタデータともいえる二次情報が生成される。本論文では、これらの二次情報のことを「記述情報」と呼ぶ。

### 2.3 翻刻にあたっての課題

訓点資料を読解する際の課題として、大きさが不揃いな漢字・送り仮名・ヲコト点が混在していること、訓点資料のテキスト化が十分になされていないこと、学生など専門家以外の作業への参加が難しいことが指摘されている[10]。

また、テキスト化における課題として、記述情報が雑多であるため一つ一つデータ入力することがかなりの手間となること、ヲコト点の打点位置が曖昧で機械的に読みを確定しにくいこと、墨を使った手書きに訓点に加わった資料であるため計算機による文字認識は容易ではないことが挙げられる。

### 2.4 Web アプリケーション開発における既存技術

本研究で開発に用いた既存技術について記す。HTML5 Canvas [11]は、HTML 上で図を描画するために開発された技術である。Canvas は、HTML5 に適合するブラウザであればどのブラウザにおいても使用可能であり、Flash や Java と異なり、あらかじめプラグインを導入しておく必要がないのが特徴である。また、JavaScript を併用することで、HTML ページの Canvas 上で画像・図形などを動的に描画することができる。

jQuery は、Web アプリケーション開発を対象に、JavaScript をより容易に記述するための JavaScript ライブラリである。マルチブラウザに対応しており、様々な Web ブラウザに対して動作保証をしている。また、ブラウザごとの複雑な例外処理を行う必要がなく、システム開発の負担を軽減できる。本研究での実装にあたり、jQuery を積極的に使用した箇所は、入力機能に用いるマウスのイベント処理と、メニュー表示に用いるフローティングウィンドウ、次に述べるサーバとクライアントの非同期通信である。

Ajax とは、Asynchronous JavaScript + XML の略で、非同期通信を行うシステムを構築するための技術の一つである。非同期通信を用いると、クライアントからサーバに通信を行う際に、サーバからのレスポンスを待つことなく、他の処理に移ることができる。この特徴を利用することで、画面遷移なしに操作できるインターフェイスを構築することができる。本システムでは、座標データの送受信には Ajax を用いているため、ページの再読み込みを行うことなく、動的なデータの表示を可能としている。

### 3. 関連研究

本節では、訓点資料またはその他の古典籍に関するこれまでのデジタルアーカイブシステムについて、特徴を述べ、本研究で開発したシステムと比較を行う。

訓点資料を対象としたデジタルアーカイブシステムの先行研究として、岡本[12][13]がある。本研究と同様に、資料画像上で文字領域をマウス操作により入力して抽出し、領域に対して記述情報を付与する仕組みを用いている。本

研究との違いとして、領域指定は正方形の固定サイズで行っており、様々な字形に対応するために、領域幅を大きく取って入力に対応していること、および、各漢字の文字画像の取得を主目的としているために、領域抽出は漢字の文字領域に留めていることが挙げられる。このため、対象の漢字以外の情報も領域内に多く含まれることとなり、また、訓点における記述情報に関しては、漢字の文字領域と関連付けて人手による入力を余儀なくされる。それに対して本研究では、任意サイズの矩形の領域を入力することができるのに加えて、資料中の文字領域や点座標を用いて、各要素間の位置関係を割り出すことで、記述情報を自動取得する仕組みを実現している。

翻刻支援に関するデジタルアーカイブシステムの関連研究には、高橋ら[14]がある。古代エジプトの文献を対象としたデジタルアーカイブシステムを構築している。扱う文書は神官文字で構成されており、その字形は多種多様である。対象とする文字の領域は矩形では対応できないため、ポリゴンの形式で文字領域を扱っている。ユーザインターフェイスの実装には、コンテンツ管理システムの一つである Drupal を使用することで、文字情報表示のためのフローティングウィンドウなど、豊富な機能拡張を実現している。また、データベースでは、1文字領域ごとに ID を付与し管理することで、文字の機能や読み、一つの単語を構成する文字の集合、単語の意味などのデータを柔軟に関連付けている。それに対して本研究では、訓点資料を対象としているが、漢字または送り仮名に対しては、入力の負担などを考慮し、文字領域は矩形の形式で十分に対応できると判断した。また、インターフェイスの実装には、HTML5 canvas, jQuery, Ajax などの既存技術を用いることで、機能の充実を図ってきた。データベースにおいても、1領域・1座標ごとに ID を付与し管理することで、訓点資料の記述情報に対して柔軟な対応を図った。

ヲコト点の読み入力に関する先行研究には、田島ら[15]および堤ら[16]がある。田島ら[15]では、テキストベースのインターフェイスを採用しており、あらかじめ用意しておいた本文テキストにヲコト点の記述情報を付与する仕組みを用いている。インターフェイスには、入力対象のテキストとテキストを分割するグリッドが表示される。利用者は、資料画像のヲコト点とテキスト上のグリッドを見て、同じ位置と判断したグリッドの箇所を選択し、打点位置を入力する。それにより、ヲコト点の読みは、入力された打点位置から機械的に確定される。そのため、資料上の読みとテキスト上の打点位置の関係がずれてしまうことが、課題として指摘されている。本研究では、画像ベースのインターフェイスを採用しており、資料画像上の領域・座標にヲコト点の記述情報を付与する仕組みを用いている。本システムで実装した機能には、各文字領域・点座標の位置関係を基にヲコト点の読みを機械的に推定する機能と、推定されたヲコト点の読みの確認と修正を行う機能がある。利用者は、画面上の文字画像と対応表を見て、適切な打点位置や読みかどうかを判断し、必要であれば修正を行う。田島ら[15]では、打点位置を手動で入力し読みを自動で確定させる仕組みを用いているのに対して、本研究では、打点位置の自動推定を行い、それを基に読みを確定し、必要に応じて読みを手動で修正できる仕組みを用いた。

文字領域情報を用いた応用例には、先に述べた岡本[12][13]のシステムにおける文字領域単位の画像抽出機能の他に、安岡[17][18]が開発した拓本文字を対象としたデジタルアーカイブシステムが挙げられる。このシステムでは、拓本の資料画像上でマウスを使って文字領域を取得し、それを基に文字画像の抽出を行っている。そして、各文字領域に対して、文字テキストと抽出した文字画像を付与することで、文字データベースを構築している。このシステムには、文字領域上に文字テキストを透過表示させ入力データを確認できる機能や、資料画像上の文字領域をクリックすることで同様のテキストが付与されている文字画像を一覧表示できる文字画像比較機能が実装されており、データ出力に複数の機能を備えている。本システムにおいては、文字領域・点座標を用いて、自動関連付け機能やワコト点の読み推定機能を実装することで応用を図った。

訓点資料のデータベース構築に関する先行研究として、高田[19]を挙げる。この研究では、訓点資料特有の記述情報への対応を行うために、資料の情報は漢字を中心とした階層構造とみなすことで、XML 記法を用いてデータベース上に表現しており、テキストでデータを取り扱う。訓点資料の電子化において、データの共有手段として普及させることを狙いとしている。本研究では、資料の情報は各要素の領域・座標で扱うことで、訓点資料への柔軟な対応を図ってきた。座標データを用いて 1 文字 1 箇所単位の文字領域・点座標をデータベースで管理しており、各要素に ID を付与することで訓点資料の複雑な記述情報に対応を図った。資料の情報を座標で取り扱うことで、データの共有・加工・復元を容易に行うことができる。

## 4. システム概要

### 4.1 開発方針

本研究では、訓点資料の解読支援環境の構築を目指し、デジタルアーカイブシステムの開発・提供を行う。そのためには、訓点資料のテキスト化を支援する機能や、それを扱うデータベースシステムが求められる。構築するシステムの開発方針として、訓点資料の記述情報を簡易な入力操作で、柔軟かつスムーズに取得できること、および取得した記述情報の保存性と応用性を確保することが挙げられる。

テキスト化の課題解決にあたって、本研究では、利用者と計算機で役割分担を行うことで、訓点資料の記述情報の取得を目指す。利用者は、画像上の複雑な表記を読み解き、単純な形(本研究においては文字領域や点座標)に置き換えてデータ入力を行う。それに対して計算機は、利用者が入力したデータを基に、記述情報の取得を行い、得られた記述情報をデータベースに登録する。この仕組みにより、複雑な記述情報を含む訓点資料に対応しつつ、利用者の入力負担を軽減できる。

システムの動作環境について、今回構築したシステムでは、複数の利用者による並列作業は考慮せずに、利用者は 1 人を想定している。また、システムの使用環境は、クライアントとサーバが同一の LAN (Local Area Network) 内にあることを想定している。

### 4.2 システム構成

構築した翻刻支援システムの構成について説明する。システム構成を図 2 に示す。クライアントは、記述情報の取得と表示、または座標情報の処理を行い、サーバは、記述情報の保存とデータベース管理を行う。

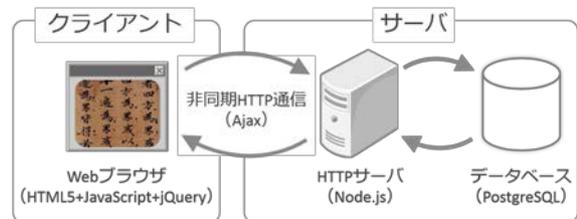


図 2 システム構成

システムは、2.4 節で述べた既存技術と、4 ファイル約 3,000 行の自作プログラムで構成される、Web アプリケーションとして実現した。その特長として、画面遷移なしで記述情報の取得と表示を行うことができるほか、マルチブラウザに対応していることが挙げられる。また、サーバとの通信には Ajax を用いることで、非同期通信によるスムーズな操作を可能とした。

サーバは、Linux 上で稼働する HTTPサーバとデータベースで構成される。HTTPサーバは Node.js を用いており、クライアントからのリクエストに応じてデータベースに問い合わせをし、結果を返す。これを用いるメリットとして、サーバプログラムに JavaScript を用いることができる点と、簡易な HTTPサーバであれば少量のコード記述で導入可能という点が挙げられる。データベースには PostgreSQL を用いており、訓点資料に対応したデータベース設計をし、データ管理を行っている。

### 4.3 データベース設計および座標の扱い

データベースの設計について説明する。訓点資料の特徴として、本文中の漢字に対して、送り仮名やワコト点などの訓点が付与されていることが挙げられる。この特徴をデータベースに対応するために、取得する文字領域・点座標に対して 1 文字 1 箇所単位でレコードを設定し、漢字と訓点のそれぞれの関係を ID で紐付ける。これにより、どの訓点がどの漢字に属するものかが定まり、1 つの漢字に対して複数の訓点が付与されている場合においても対応することが可能である。

扱うデータの形式について説明する。資料本文における漢字・送り仮名・ワコト点は、画像上の座標をベースに、文字領域や点座標に置き換えてデータベースに保存している。これらを座標ベースで扱うことで、曖昧な手書き文書を計算機上での扱いを可能とした。また、これらを座標ベースで扱うことによるメリットとして、システムの更新を行っても、座標データと座標データの基となる資料画像があれば、復元が可能であること、データの加工・編集が容易であることなどが挙げられる。

本システムでは、資料画像上の座標を定義し、クライアントで座標を取得した上で、サーバを介してデータベースに格納する方法を取ることにした。座標は、日本語文の縦

書きに合わせて、資料画像における座標軸、横軸を左向き、縦軸を下向きに取り、Canvas から得られる座標と資料画像上の記述情報を関連付けて、座標の取得を行った。画像移動機能で資料画像、入力データの表示を移動させるには、資料画像における座標系の原点を移動させ、Canvas の固定座標系と比較することで実現した。

Canvas 上の座標取得方法について説明する (図 3)。JavaScript の `event.pageX`, `event.pageY` の関数を用いることで、ブラウザ上のマウスイベント (ボタンを押す, ボタンを放す, マウスを移動させるなど) が発生した際に、ブラウザ画面の左上隅を基点としたマウスの位置座標を取得できる。さらに、`canvas.offsetLeft`, `canvas.offsetTop` の関数を用いることで、ブラウザ画面の左上隅を基点に canvas の左上隅の座標を取得することができる。ゆえに、Canvas 上におけるマウスの位置座標 (X, Y) は、 $X = \text{event.pageX} - \text{canvas.offsetLeft}$ ,  $Y = \text{event.pageY} - \text{canvas.offsetTop}$  の演算により導き出すことができ、これにより、資料画像における任意の入力座標を取得できる。また、画像移動は、Canvas 上において、画像を (a, b) 移動させたことを記憶しておき、座標演算時に加減算することで、画像移動前の本来の座標を導き出すことができる。これにより、1 枚の Canvas 上で画像移動とそれに伴った記述情報の取得・表示を自在に行うことができる。

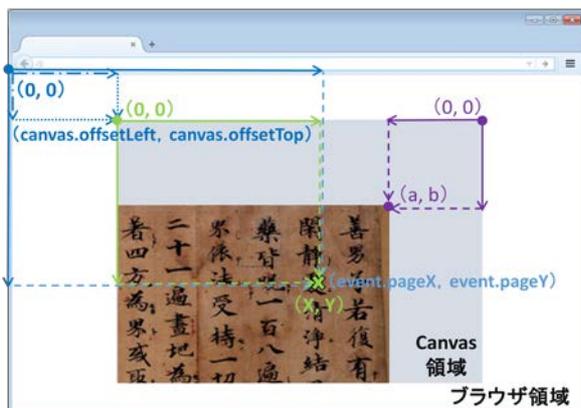


図 3 座標系と座標取得例

座標形式でデータを取り扱うことによる応用例として、得られた座標から漢字と訓点の位置関係を推定し自動で記述情報を得られることや、1 領域ごとの画像を容易に取得できることなどが挙げられる。前者は、後述の自動関連付け機能としてシステムに反映しており、後者は、HTML5 Canvas や jQuery の機能を利用することで、1 領域単位の画像抽出を行うことができる。また、座標情報をデータベースに格納しておくことで、これらの応用機能は任意のタイミングで使用することが可能になる。

それぞれの領域座標に対して ID を付与することで、漢字に対する訓点の関連付けや、様々な記述情報の紐付けを行うことが可能となる。それぞれの漢字は、レコードの ID のほか、矩形領域の左上 X 座標・Y 座標、右下 X 座標・Y 座標の 5 つのカラムで構成した。送り仮名における文字領域についても同様である。ヲコト点については、ID のほか、打点位置の X・Y 座標と、形状テーブルの ID (参照キー) の 4 つのカラムで構成した。

ヲコト点の形状データの取り扱いについては、計算機で使用可能な文字・記号を用いた。また、必要に応じて、CSS (Cascading Style Sheet) の機能を用いてそれらを回転させて表示することで、柔軟に対応を図った。データベースに登録した、ヲコト点形状に使用する文字は、表 1 に示す 20 種類である。また実際に打点されているヲコト点と、それに対応するデータ構造の例を、表 2 に示す (データベースには下線部の情報のみを登録する)。

表 1 ヲコト点形状に使用する文字

・	..	∴	—	L	フ	ソ
人	⊥	⊥	イ	七	ヒ	=
リ	コ	ヨ	ム	O	j	／

表 2 ヲコト点の形状表現例

	「L」を 0 度回転
	「..」を時計回りに 45 度回転
	「—」を時計回りに 315 度回転

#### 4.4 ユーザーインターフェイス

開発したインターフェイスでは、HTML5 Canvas を用いてデータの入出力を画像上に表現した。Canvas と呼ばれる描画領域を用意し、資料画像および諸情報を表示させている。元の資料画像は 1 枚の資料に対して複数の画像に分かれているが、本システムでは、複数に分かれた資料画像をあたかも 1 枚の画像のように取り扱うことができる。また、資料の拡大縮小、表示領域の移動なども行うことができる。訓点資料画像上の文字領域は四角形の実線・破線で、また点座標と点形状は文字・記号を用いて、表示を行った。文字領域の入力において、マウスのドラッグ操作中には、破線表示させたり、漢字と訓点の関連情報を破線で結びつけて表示させたりするなどの処理も行っている。訓点資料には、紙の色や文字の色、ヲコト点の色などに様々な組み合わせが存在しており、それに対応するために、データの表示色を細かく変更できる機能を備えている。

各機能の操作はマウスで行う。各機能を使用するには、図 4 に示すメニュー選択欄を用いる。なお、メニュー欄はフローティングウィンドウとなっており、ブラウザの画面内を自由に動かすことができるほか、作業に応じてメニュー欄を開いたり閉じたりすることもできる。

入力機能においては、入力モード、削除モード、移動モードを使用することで入力作業を行うことができる。文字領域・点座標の入力に応じて、自動関連付け機能が働き、記述情報が自動的に取得される。誤って記述情報が取得された場合は、修正モードを使用して、自動関連付け情報の修正機能を用いることで、記述情報の修正作業を行うこと

ができる。これらの入力・修正作業は、ブラウザに表示された画像上で行う。



図 4 メニュー選択欄

また、漢字の文字領域とヲコト点の点座標・形状入力に応じて、ヲコト点の自動読み推定機能が働き、ヲコト点の読みが自動的に取得される。取得されたヲコト点の読みの確認・修正作業は、点編集モードを使用し、ヲコト点の読み修正機能を用いて行うことができる。点編集モードでは、ヲコト点の打点位置・読み対応表と、文字拡大画像が表示され、それらを見ながら確認・修正作業を行う。

メニュー上で「入力」が選択されている状態のとき、文字の領域をドラッグしたり、ヲコト点の位置をクリックしたりするといったマウス操作を通じて、位置・領域情報を容易に入力することができる。漢字領域とヲコト点の入力表示例を図 5 に示す。



図 5 漢字領域およびヲコト点の入力表示例

画像上でマウスホイールを使うことで、画像の拡大・縮小ができるほか、メニューで「移動」を選択し、画像移動機能を使うことで、マウスを画像上でドラッグして資料画像を自在に移動させることができるようにしている。フローティングウィンドウのメニュー選択欄より、資料画像上の表示（データの表示色、ヲコト点のマーカーサイズ、関連付け情報の表示有無など）を変えることができ、資料に合わせた表示や利用者の使用感に適した表示設定を行うことができる。

#### 4.5 自動関連付け機能

資料画像上に入力された文字領域・点座標より、漢字の行・列番号を自動的に推定する機能と、漢字と訓点の組み合わせを自動的に関連付ける機能を実装した。図 6 は関連

付けられたデータの例で、行・列番号は「列番号/行番号」の形式で表示され、漢字と訓点の関連情報はそれぞれ点線で結びつけられて表示される。

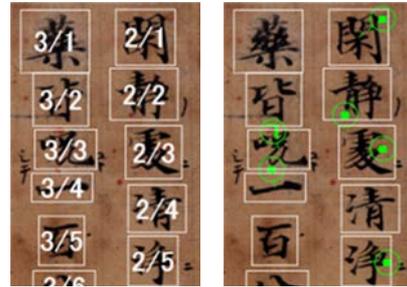


図 6 漢字領域およびヲコト点の関連付け機能例

漢字と訓点の関連情報を推定する方法について、ヲコト点の付与対象となる漢字は、打点座標から各漢字領域の中心までの距離が最短のもの、送り仮名の付与対象となる漢字は、漢字に対して左下または右下に書かれる傾向が見られることから、送り仮名領域の上辺中心から各漢字領域の中心までの距離が最短のもの、として推定を行った。

手書き資料であること、データ入力人が人の手によるものであることなどから、漢字と訓点の関連情報の自動推定には誤りが生じることがある。そこで、本システムでは関連情報の修正を容易に行える機能を実装した。誤った漢字と訓点の関連情報は、メニューで「修正」を選び、修正モードに切り替えて、資料画像上の正しい漢字と訓点の組をそれぞれクリックすることで直ちに修正することができる。修正例を図 7 に示す。「無」の右上と誤認識されたヲコト点を、「滅」に対する打点とするよう修正している。



図 7 修正機能の使用例

#### 4.6 自動読み推定機能

漢字とヲコト点の領域・座標データより、ヲコト点の打点座標を推定し、推定した打点位置と入力された形状より読みを導出・提示することで、翻刻を支援する機能を実装した。打点位置の推定は、各座標データの入力に応じて自動的に行われ、導出された読みは、画面上で確認することができる。

打点位置の推定方法について、漢字領域を縦 3 分割、横 3 分割の 9 箇所領域に分け、ヲコト点の打点座標がどの領域に属するかを判断して打点位置を導き出すことにした。ただし、傾出する形状「・」においては、打点位置と読みの関係が複雑なため、中央部をさらに分け、図 8 により判

定を行うことにした。実線が漢字領域、点線が判定境界線である。

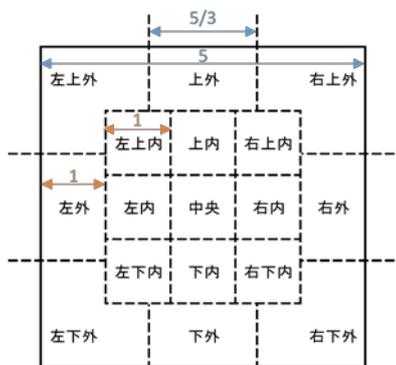


図 8 形状「・」の打点位置判定

推定された読みの確認，または修正のための機能を実装し，インターフェイスに点編集モードを設けた。

## 5. システム評価

本システムの利用により，翻刻を支援できることを確認するために，操作性に関する利用者実験，および訓読に関する評価実験を行った。

### 5.1 操作性に関する利用者実験

利用者を交えて，円滑に入力でき，漢字・送り仮名・ヲコト点の入力漏れが少ないことを確認するための実験を実施した。システムの利用者は計算機を使える専門家および専門を学ぶ学生と想定する。実験参加者は，和歌山大学の学生（計算機が使って，訓読の知識のない利用者）6人で、『千手千眼陀羅尼経残巻』を対象にデータ入力を行った。入力対象の本文は，巻頭から1段落22行の367字（以下，本文A），巻頭から5行の86字（以下，本文B）の2種類を用意し，本文Aは3人，本文Bは6人のデータを取得した。なお，本文Aのうち，本文Bの領域についても本文Bとみなして集計を行った。実験に用いた機器は，実施者が用意したWindows7のデスクトップPCである。

実験手順は次の通りである。それぞれの参加者に対し，実験概要について説明した後に，インターフェイスの資料画像上に漢字・送り仮名・ヲコト点の文字領域・点座標と点形状を入力してもらい，入力を終えた時点で入力データの最終確認と修正をもらった。

利用者および本文領域のデータ入力数を表3に示す。漢字領域の入力においては，本実験では，漏れ無く入力できた。送り仮名領域，ヲコト点座標・形状において，各参加者の入力データ数が異なったのは，これらを正確に見分けたり，これらの領域・座標指定の正否を判断したりするのに，専門的知識を要するためである。

表 3 データ入力数

参加者	領域	漢字	送り仮名	ヲコト点
1	本文A	367	231	324
2	本文A	367	234	382
3	本文A	367	232	366
1	本文B	86	53	71
2	本文B	86	55	89
3	本文B	86	51	88
4	本文B	86	54	86
5	本文B	86	53	86
6	本文B	86	51	84

漢字・送り仮名・ヲコト点の入力時間を，表4に示す。本実験における入力所要時間は，修正作業を含めており，漢字・送り仮名・ヲコト点において，それぞれ1箇所の入力が完了するまでの時間を計測した。最大時間は，修正・削除・確認作業と思考のための時間が含まれている。最小時間は，効率よく操作できた場合の時間と考えられる。平均入力時間は，本文Bより本文Aが短くなっており，これは入力操作の熟達化によるものと考えられる。

表 4 1件ごとの入力所要時間

領域	入力時間	漢字	送り仮名	ヲコト点
本文A	平均	2.8	2.9	3.5
	最大	21.3	60.4	51.8
	最小	1.2	0.9	0.5
本文B	平均	3.9	3.5	4.3
	最大	38.9	29.9	61.5
	最小	1.1	1.0	0.6

(単位：秒)

各実験の作業後に，アンケートを実施した。「領域指定の操作性」「座標・形状指定の操作性」「関連情報修正機能の操作性」「システム全体の操作性」をそれぞれ5段階の評点（5が最高，1が最低）で回答してもらったとともに，アンケート記入の際はヒアリングも同時に行い，システムの操作上で気づいたことがあるかを質問し，コメントがあれば自由記入欄に書かせた。

各項目の5段階評価について，「領域指定の操作性」で1名，2と回答した者がいた他は，どの項目も4～5の評点を付けていた。ヒアリングにおいて，ヲコト点の形状指定において，形状を頻りに切り替える上で，選択欄がラジオボタンだと操作性が良くない，マウスホイールによる画像の拡大・縮小操作に戸惑った，などのコメントが得られた。

以上の結果より，本研究において実装した本システムのマウスを使った入力に関して，課題はいくつか残されているものの，主機能である文字領域・点座標入力については特に問題なく操作できたと判断できる。

## 5.2 訓読に関する評価実験

自動関連付け機能の有用性を評価し、訓点資料の記述情報を取得できることや、ヲコト点の打点位置の妥当性を中心に、本システムの入力結果を基に適切な訓読が行えることを評価するための実験を行った。実験には、『千手千眼陀羅尼残巻』のほか、喜多院点の訓点が付された『金剛頂一切如来真実撰大乘現証大教王経』[20]（『金剛頂経』と表記する）を用いて、それぞれの巻頭から 5 行分を対象に、和歌山大学において文字領域・打点座標の入力を行い、自動関連付け機能とヲコト点の自動読み推定機能を用いて記述情報を取得し、結果を画像上に表示した。対象に出現するそれぞれの漢字について、自動推定で得られた記述情報を参考に、漢字 1 文字ごとの訓読を行った。

評価にあたり、「行番号」「列番号」「漢字」「送り仮名」「ヲコト点の読み」「ヲコト点の形状」「漢字の打点位置」からなるリストを作成した。「漢字」「送り仮名」については、画像を見てテキスト化し、「ヲコト点の読み」については、ヲコト点の読み修正機能より求めた。作成したリストと、対象箇所画像、実施者の入力データに基づく画像を、大阪大谷大学に送付し、そこで自動推定および訓読の適切性を判断した。本システムにおいては、返り点の取得・判別ができないため、訓読文の作成そのものは行わず、漢字単位で妥当性評価を試みた。また『金剛頂経』上に白色で書かれた「へ」および「下ヒ」の送り仮名は対象外とした。『千手千眼陀羅尼残巻』の 2 行目（1 行目は誤りなく訓読できた）の訓読結果を表 5 に示す。この表の下線は、大阪大谷大学における判断で誤りと判明し修正した箇所である。

表 5 訓読結果例

行番号	列番号	漢字	送り仮名	ヲコト点	形状	位置
2	1	閑				
2	2	静	ノ	の	・	左下外
2	3	處	ニ			
2	4	清				
2	5	淨	ニ	に	・	右外
2	6	結				
2	7	界	シ	せよ	フ	左
2	8	呪	シテ	し、て	・、・	中央、右下外
2	9	衣	ヲ			
2	10	著	キヨ	き	\	中央
2	11	若	ハ	は	・	下外
2	12	水				
2	13	若	ハ	は	・	下外
2	14	食				
2	15	若	ハ	は	・	下外
2	16	香				
2	17	若	ハ	は	・	下外

行番号 2・列番号 8 の「呪」について、入力した漢字の領域およびヲコト点の位置を図 9 に示す。漢字のうち「し」の部分が大きく右に伸びており、そのため自動関連付け機能は、中央付近に打点されたヲコト点を、図 8 の区分のうち左上内にあるものと誤認識した。

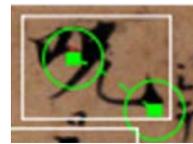


図 9 ヲコト点の位置の誤認識例

2つの訓点資料について訓読の正誤数を表 6 および表 7 に示す。ここで、『金剛頂経』のヲコト点関連付け機能で誤りが多くなっているが、これは、返り点をヲコト点として入力してしまったためである。この項目の正答率は、2つの資料で大幅に異なるため除外し、本実験において、送り仮名の自動関連付け機能における正答率は 98.6%、ヲコト点の自動読み推定機能における正答率は 88.9% という結果が得られた。

表 6 『千手千眼陀羅尼経残巻』における訓読の正誤

	総データ数	正しい	誤り
送り仮名関連付け	53	52	1
ヲコト点関連付け	57	57	0
ヲコト点読み推定	57	51	6

表 7 『金剛頂経』における訓読の正誤

	総データ数	正しい	誤り
送り仮名関連付け	16	16	0
ヲコト点関連付け	26	15	11
ヲコト点読み推定	15	13	2

本実験の結果より、専門家が各推定結果を修正することで訓読作業を補完でき、従来の翻刻作業に比べて労力が半分近くに低減できており、支援システムとして実用的なレベルに達していると考えられる。ヲコト点の読解が複雑なことから、システムが座標から打点位置を推定し、利用者に読みを提示する機能はきわめて有益である。課題としては、訓点資料は手書きであるため、打点位置と読みが必ずしも一致するとは限らず、機械的に読みを決定することに限界があるケースがあることが挙げられる。また、今回は読みを中心としたヲコト点の取り扱いを行ったが、アクセント符号、区切り点、返り点にも対応が必要となる。

## 5.3 考察

今回行った 2 つの評価実験について、それぞれ考察を行う。

まず、操作性に関する利用者実験について、入力時間の結果より考察すると、漢字・訓点 1 箇所に対して、およそ 3 秒の操作でこれらの領域・座標情報を登録でき、それに伴い記述情報が得られることがわかる。よって、本システムにおいて取得している記述情報の他に、テキストの自動認識や、領域画像の取得、訓読文の自動生成などの機能を

実装し、自動取得できる記述情報を増やすことで、少ない操作から得られる情報がさらに増えるため、より有用な翻刻支援システムになることが期待できると考えられる。

また、得られたコメントより考察すると、資料画像の表示における画像処理（明るさ・コントラスト・シャープネスなどの調整）機能の実装の必要性が高いように考えられる。これについては、HTML5 Canvas の機能を活用することで実現可能と思われる。その他コメントを加味すると、領域入力におけるドラッグ操作以外の手段で入力を行える方法の検討、形状指定方法の改良、ヲコト点の点座標において他の入力方法の検討、マウスにおける機能割り当ての設定機能の検討、などへの対応が挙げられるが、これらの機能追加・改善についても、操作性を高める上で検討すべき事項と考える。

次に、本システムを用いた訓読に関する評価実験について考察する。自動関連付け機能については、正答率 98.6% という結果が得られており、推定誤りについては、修正機能を用いて、システム利用者が補完することで、十分に対応可能と考えられる。また、自動読み推定機能についても同様に、正答率が 88.9% という結果が得られており、9箇所の入力に対して約 1 箇所の推定誤りを起こすことになるが、修正機能とシステムの補完作業により、対応可能であると考えられる。

推定誤りへの改善の方針には、次の 2 つが考えられる。一つは、送り仮名のテキストとヲコト点の読みが一致する傾向にあるため、送り仮名のテキスト入力を基にヲコト点の読みを推薦する機能を開発することである。もう一つは、漢文訓読辞書において、各漢字に対して訓読例が載せられているため、漢字のテキスト入力を基に訓読例を検索表示できる機能を提供することである。テキスト入力機能の実装や、訓読例を反映させたデータベースシステムの実装を行うことで、翻刻支援機能をさらに改良できるものと考えられる。

## 6. おわりに

本論文では、訓点資料の解読を支援するシステムの開発とそれを扱うデータベースシステムの構築、およびその評価について述べた。

資料画像の文字領域や点座標を抽出する仕組みにより、訓点資料に対する電子的対応を実現することができた。また、システムにおいて、利用者と計算機で翻刻作業の役割分担を図った。利用者が最低限の入力を手動で行い、計算機がそれに基づいて文書情報を自動で取得することで、入力作業の省力化や、手書き文字、ヲコト点に対する柔軟な対応を行うことができた。データベースの設計にあたっては、文字領域・点座標を一つ一つ管理することで、記述情報を文字・訓点の単位で扱えるようにした。また、ヲコト点の読み情報の入力については、座標情報を用いた読みの自動推定と、文字画像の拡大表示、読みと打点位置の対応表示などの機能により、入力支援を図った。

今後の課題として、考察で述べたような支援機能の追加を行うために、クライアント、サーバ、データベースの改善が挙げられる、それらに対応することで、少ないデータ入力と短い入力時間で、多くのデータ出力と記述情報を得られることが期待できる。

## 謝辞

本研究は JSPS 科研費 26284065 の助成を受けたものです。

## 参考文献

- [1] 林寺正俊, “日本古写経データベースの構築とその意義”, 情報処理学会人文科学とコンピュータシンポジウム論文集, Vol.2012, No.7, pp.11-16 (2012).
- [2] 小助川貞次, “デジタル化時代に対応した漢文訓読研究の社会的共有システムの構築”, 富山大学人文学部紀要, No.52, pp.87-101 (2010).
- [3] 田中勝, 村川猛彦, 宇都宮啓吾, “訓点資料を対象としたデジタルアーカイブシステムの構築”, 2015年電子情報通信学会総合大会情報・システム講演論文集 1, D-4-13 (2015).
- [4] 田中勝, 村川猛彦, 宇都宮啓吾, “訓点資料における翻刻支援システムの構築”, 人文科学とコンピュータシンポジウム論文集, 情報処理学会シンポジウムシリーズ, Vol.2015, No.2, pp.263-268 (2015).
- [5] 築島裕, “平安時代訓点本論考 ヲコト点図仮名字体系”, 汲古書院 (1986).
- [6] e 国宝『千手千眼陀羅尼経残巻』 . <http://www.emuseum.jp/detail/101016/000/000> (参照 2016-04-13).
- [7] 宇都宮啓吾, “宝幢院点の成立に関する一考察—源信・寂照・延殷・皇慶を巡って”, 訓点語と訓点資料, Vol.2009, No.123, pp.59-70 (2009).
- [8] SAT 大正新脩大藏経テキストデータベース. <http://21dzk.l.u-tokyo.ac.jp/SAT/index.html> (参照 2016-04-13).
- [9] 高田智和, “訓点資料積文制作における構造化記述の試み”, 情報処理学会研究報告 人文科学とコンピュータ, 2010-CH-85, pp.1-8 (2010).
- [10] 小助川貞次, “漢文訓読史概説の構想”, 富山大学人文学部紀要, No.56, pp.109-121 (2012).
- [11] David Geary (永井勝則 訳), “プログラミング HTML5 Canvas — ウェブ・モバイル・ゲーム開発者のためのアプリケーション開発ガイド”, オライリージャパン (2014).
- [12] 岡本隆明, “古文書・典籍を対象とした文字管理システムとその可能性”, 情報処理学会研究報告 人文科学とコンピュータ, 2008-CH-078, pp.77-84 (2008).
- [13] 岡本隆明, “コンピュータによる訓点資料の整理について”, 情報処理学会人文科学とコンピュータシンポジウム論文集, Vol.2008, pp.275-282 (2008).
- [14] 高橋洋成, 永井正勝, 和氣愛仁, “画像, TEI, LOD を用いた文字研究・言語研究のためのプラットフォームの構築”, 情報処理学会研究報告 人文科学とコンピュータ, 2015-CH-105, pp.1-5 (2015).
- [15] 田島孝治, 堤智昭, 高田智和, “ヲコト点電子化のためのデータ構造と入力支援システムの試作”, 情報処理学会人文科学とコンピュータシンポジウム論文集, Vol.2012, No.7, pp.211-216 (2012).
- [16] 堤智昭, 田島孝治, 高田智和, “点図情報入力支援ツールによるヲコト点図の電子化”, 人文科学とコンピュータシンポジウム論文集, 情報処理学会シンポジウムシリーズ, Vol.2015, No.2, pp.185-190 (2015).
- [17] 安岡孝一, “拓本文字データベースの設計とその応用”, 漢字字體史研究, pp.116-128, 勉誠出版 (2012).
- [18] 安岡孝一, “拓本文字データベースの現状と課題”, 情報処理学会研究報告 人文科学とコンピュータ, 2013-CH-97, pp.1-6 (2013).
- [19] 高田智和, “訓点資料の電子化について”, 国語研プロジェクトレビュー, Vol.4, No.1, pp.36-42 (2013).
- [20] 国立国語研究所蔵 金剛頂一切如来真実撰大乘現証大教王経 (嘉応二年加本) . <http://dglb01.ninjal.ac.jp/kongochokyo/> (参照 2016-04-13).