

書字行動からみた言語の認知・学習過程に関する研究

—書字行動測定用ソフトウェア開発の試み—^{1,2}

河原純一郎・玉岡賀津雄³・中條和光・松見法男

(2001年9月28日受理)

Investigating Cognitive and Learning Processes of Languages through Writing Behavior:
Developing Software for Writing Behavior

Jun-ichiro Kawahara, Katsuo Tamaoka, Kazumitsu Chujo and Norio Matsumi

There is little extistent literature that has investigated the process involved in writing Japanese two-morpheme (kanji) compound words. One reason for this scarcity has been the difficulty in devising instruments to measure latencies for writing behaviour. To asnwere this problem, we developed software that enabled us to present auditory stimuli, to measure writing latencies, to track trajectories during the writing of kanji characters, and to assist data analysis. This software runs on the Microsoft Windows 98 Operating System with a commercially available pen-tablet. Technical details of this software will be explained in this paper.

Key words : writing process, Japanese two-morpheme compound words, Microsoft Windows Application

キーワード: 書字行動, 日本語漢字二字熟語, マイクロソフトウィンドウズ アプリケーション

1. 導入

"ほんごうのもくてきは、にほんごのしょじこうどうをそくていするためにかいはつしたじっけんかんきょうをかいつつすることである"という、ひらがなのみで表記された文を読むことは容易ではない。この例文を、漢字を使用して表記すると、"本稿の目的は、日本語の書字行動を測定するために開発した実験環境を解説することである"となり、読みやすさは向上する。この例

からわかるように、日本語を表記し、理解する上で漢字の果たす役割は極めて重要である。また、日本語の辞書に記載された語彙の約七割が漢字二字で構成されている(Yokosawa & Umeda, 1988)とされていることから、漢字二字熟語は日本語の表記と理解における基本単位のひとつといえるだろう。

この基本単位である漢字二字熟語をわれわれがどのように処理しているかという問題は、日本語の読み書きの認知機能を明らかにする上で解明しなければならない点である。漢字二字熟語の読み行動に関する研究は従来から幅広く行われている(Kayamoto, Yamada & Takashima, 1998; Tamaoka & Hatsuzuka, 1998; Wydell, Butterworth & Patterson, 1995)。具体的には、漢字二字熟語を単独で呈示し、その語彙判断を行う課題や種々のプライミング課題による読み行動の検討は数多く行われてきた(たとえば、廣瀬, 1992; Saito, Masuda & Kawakami, 1998)。また、文章中に他の機能語とともに呈示して眼球運動を測度として読み行動を理解するアプローチもある(たとえば、神部, 1998;

¹ 本研究は広島大学教育学部平成12年度 リサーチ・オフィス経費による助成を受けた。

² 本稿で紹介したソフトウェアの作成には西澤朋枝氏の助言を頂いた。ここに感謝する。このソフトウェアは非営利目的のためであれば自由に使用可能である。このソフトウェアの動作確認はWindows98でおこなった。このソフトウェアを使用したために生じたいかなる問題にも著者は責任を負わない。

³ 広島大学・留学生センター

Matsunaga, 1999).

一方、漢字二字熟語の書字行動については、実験的な手続きを用いた分析を行った研究は極めて少ない。それは書字時間を測定可能な実験装置の製作が困難であったことが主な理由として挙げられる。書字行動測定のためには、少なくとも、聴覚、または視覚的な刺激の呈示と、刺激呈示から書字行動の開始までの時間計測、書字軌跡の記録などが必要となる。また、行動記録は分析を簡便にするために、コンピュータ上で利用できる形態であることが望ましい。

井上・石原(1980)は潜時反応測定器(TKK製)を使用して漢字の筆記測度を測定しているが、この装置は現在は製造されておらず、入手して利用することは極めて困難である。また、この装置を用いて書字行動を記録しても、それを直接的に現在一般的に利用されているコンピュータに取り込むことはできない。Cottrell(1989)は英単語の書字時間を測定するために、圧力センサを埋め込んだ専用「パッド」を用いた。この装置は市販されておらず、一般には利用できない。最近では、玉岡・高橋(1999)はコンピュータ(Mac Classic II, Apple製)とペンタブレット(ワコム製)を組み合わせた環境で書字行動を測定する実験を行った。しかし、コンピュータの処理速度が極めて遅かったため、データの整理に手間がかかった。また、この環境では呈示する刺激を試行ごと、被験者ごとに無作為化する手続きが煩雑であるという問題があった。

このように、書字行動を測定するための実験環境を構築することは容易であるとは言いがたい。反応時間を測定するために専用のハードウェアを用いることによって、刺激呈示から書字行動開始までの時間計測は可能になる。しかし、この方法では、他の機能(刺激呈示や軌跡の記録)を何らかの形で追加しなければならないことや、反応時間測定専用のハードウェアは比較的高価であるという欠点に伴う。

そこで本稿では、一般的に利用可能な安価な装置を利用した、書字行動測定用のソフトウェアを紹介する。これは、現在九割を超えるといわれるシェアをもつWindows OS(Microsoft製)を搭載したコンピュータに、市販のペンタブレットを接続した環境で、刺激呈示、書字行動開始までの時間計測、書字軌跡の記録という機能を実現するソフトウェアである。

2. 書字行動測定用のソフトウェア

このソフトウェアはMicrosoft Windows98の動作するコンピュータを対象としている。このソフトウェア実行にはサウンドカード、スピーカ(聴覚刺激呈示

用)、ペンタブレット(書字行動記録用)を必要とする。これらのハードウェアは安価に入手できる。プログラミング言語はVisual Basic 6.0(Microsoft製)を使用した(ただし、本ソフトウェアの実行には、マウスさえ接続されていればペンタブレットは必ずしも必要ない。マウスと同様のポインティング装置として機能するため、左ボタンをクリックしたままマウスを動かすことでペンタブレット上で描画していることと等価になる)。

われわれが使用したペンタブレット(intuous model I-400, WACOM製)はインキングペンをもち、ボールペン芯を装着することができた。したがって、タブレットの上に紙を置いたままペン先の状態の記録することができ、実際に紙の上に筆跡を残しながら書字するため、自然な漢字書字事態をつくることが可能となった。

プログラムの機能

本ソフトウェアは、次の手順で動作する。起動後、Figure 1のようなフォームが現れる。スペースキーを押すと試行が開始される。各試行では、まず音声刺激を呈示し、被験者からの入力待。被験者は入力用のペンをもってタブレットに音声刺激で指定された漢字二字熟語を書く。このとき、刺激呈示から書き始めまでの時間、書き終わりまでの時間を自動的に記録される。全ての文字を書き終わったり、被験者がある漢字を書くことができずにあきらめた場合は、実験者はスペースキーを押してその試行を終了させる。スペースキー押下後2sec経過すると、自動的に次の試行が始まるように設定されている。全ての試行が終了すると、実験者はメニューバーからプログラム(P)、データの保存(A)を選択し、ファイル名を指定する。この操作

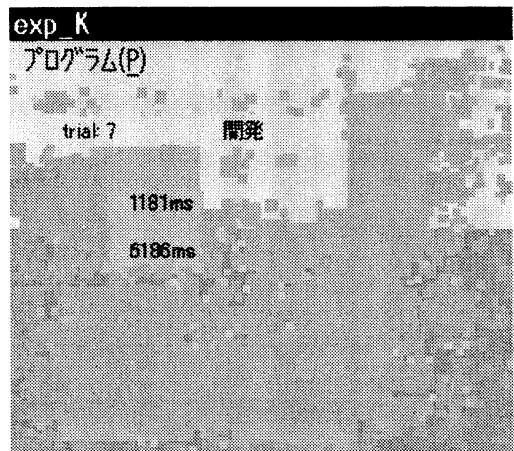


Figure 1 書字行動測定用ソフトウェアの画面例。上の測定値は書き始め、下は書き終わりを示す。

によって、被験者名、試行番号、呈示された音声刺激ファイル名、書き始めまでの時間、書き終わりまでの時間を記録したファイルが出力される。

プログラムの構造

本プログラムは3つのモジュールで構成されている。それらはフォームモジュール(FrmMain.frm)、標準モジュール(PlayWave.bas)、クラスモジュール(clsMouse.cls)である。フォームモジュールはフォーム(いわゆるWindowsアプリケーションソフトウェアの"窓")上に設定したさまざまな機能や手続きが書き込まれている。標準モジュールには、フォームモジュールに属さないプログラムが記述されている。クラスモジュールにはユーザが定義する一種の機能が記述されている。以下では、これら3つのモジュールの機能について説明する。

フォームモジュール(FrmMain.frm)

このモジュールの主要な機能は次の3つである:(1)刺激セットを準備する,(2)スペースキーが押されたら1つの試行を行う,(3)全ての試行が終了したら結果をファイルに出力する。このモジュールではマウスのボタンクリック(ペンの上げ下げ)の検出やマウスカーソル座標(ペン先の位置)取得を行うためのオブジェクトクラスを最初に定義している。これに続いて、timeGetTime関数を使用するのに必要なWindowsAPIの定義を行う。この関数は書き始めまでの時間を測定するために用いる。これ以降は一般的な変数と構造体用の定義である。

サブルーチン・プロシージャForm_Load: 刺激セットの準備、無作為化、フォームの初期化(試行数、呈示されるwaveファイルの表示、反応時間の表示)を行う。

サブルーチン・プロシージャForm_KeyPress: スペースキーを押したとき、試行が始まっていなければタイマによる計測を開始し、すでに試行が始まってタイマが動いていればそれを止め、データ構造体に記録する。

サブルーチン・プロシージャForm_MouseDown: 書き始めの検出を行う。刺激音声ファイル再生から書き始めまでの時間を計算し、データ構造体に記録する。

サブルーチン・プロシージャForm_MouseUp: 最後にペンが記録用紙から離れた時間を記録する。

サブルーチン・プロシージャmnuData_Click: "プログラム(P)"というメニュー項目が選択されたとき、ファイルへの結果出力などの処理を行う。

サブルーチン・プロシージャTimer1_Timer: 10msごとに現在のペン先の座標(x,y)を記録する。サンプリング間隔はタイマのプロパティで変更することができ

る。

サブルーチン・プロシージャTimer2_Timer: スペースキーを押してから2sec後にwave音声ファイルを再生する。この時間もタイマのプロパティで変更できる。

標準モジュール(PlayWave.bas)

このモジュールはwaveファイルの再生・停止を行うWindowsAPIを定義している。

クラスモジュール(clsMouse.cls)

このモジュールはマウスカーソルの取得に関するWindowsAPIを定義している。ここで定義されたWindowsAPIはfrmMainで使用される。

3. おわりに

先述したように、本ソフトウェアを用いて書字行動を測定する場合は、ペンタブレット以外には特別な装置を必要としない。したがって、このソフトウェアを用いることで、従来は測定と分析が難しかった書字行動の実験がきわめて容易に行うことができるだろう。

また、こうしたソフトウェアは簡便性という点だけでなく、拡張性という点でも意義がある。本ソフトウェアの開発に使用したVisual Basicは、オブジェクト志向言語のひとつで、ActiveX技術への対応、ネイティブコンパイラのサポート、データアクセス機能の充実などの機能拡張を図り、Windowsアプリケーションソフトウェアの開発用の主要言語として定着しつつある(河西, 1999)。このように汎用性の高いプログラミング言語を用い作成した書字行動の測定用プログラムは、他の実験に容易に転用できる可能性があり、書字行動に限定しない他の言語認知機能を調べるための実験的研究に貢献できるだろう。

引用文献

- Cottrell, D. 1989 Temporal processes in spelling production. Unpublished doctoral dissertation, University of New South Walse, Australia.
- 廣瀬等 1992 熟語の認知過程に関する研究—プライミング法による検討 心理学研究, 63, 303-309.
- Kayamoto, Y., Yamada, J., & Takashima, H. 1998 The consistency of multiple-pronunciation effects in reading: The case of Japanese logographs. *Journal of Psycholinguistic Research*, 27, 619-637.
- 神部尚武 1998 読みの視覚情報処理の基礎 芋阪直行

- (編) 読み-脳と心の情報処理(pp. 1-16) 朝倉書店
- 河西朝雄 1999 Visual Basic Ver.6.0 初級プログラミング入門[上] 技術評論社
- Matsunaga, S. 1999 Universality in reading processes: Evidence from an eye-tracking study. *Psychologia*, 42, 290-306.
- Saito, H., Masuda, H., & Kawakami, M. 1998 Form and sound similarity effects in kanji recognition. In C. K. Leong & K. Tamaoka (Eds.), *Cognitive Processing of the Chinese and the Japanese Languages* (pp. 169-203). London: Kluwer Academic Publishers.
- Tamaoka, K., & Hatsuzuka, M. 1998 The effects of morphological semantics on the processing of Japanese two-kanji compound words. In C. K. Leong & K. Tamaoka (Eds.), *Cognitive Processing of the Chinese and the Japanese Languages* (pp. 139-168). London: Kluwer Academic Publishers.
- 玉岡賀津雄・高橋登 1999 漢字二字熟語の書字行動における語彙使用頻度および書字的複雑性の影響 心理学研究, 70, 45-50.
- Wydell, T. N., Butterworth, B., & Patterson, K. E. 1995 The inconsistency of consistency effects in reading: The case of Japanese kanji. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 21, 1155-1168.
- Yokosawa, K. & Umeda, M. 1988 Processes in human Kanji-word recognition. *Proceedings of the 1988 IEEE international conference on systems, man, and cybernetics*, 377-380.

```
'***** FrmMain.frm *****
```

```
' 書字行動測定用プログラム
```

```
' Jun Kawahara
```

```
August, 2001
```

```
Option Explicit
```

```
' オブジェクトの生成と初期化
```

```
Dim objMouse As New clsMouse
```

```
Dim DrawNow As Integer
```

```
' 反応時間を取るためにいるもの
```

```
Private Declare Function timeGetTime Lib "winmm.dll" () As Long
```

```
Dim StartTime As Long
```

```
Dim StopTime As Long
```

```
Dim RacTime As Long
```

```
Dim EndTime As Long
```

```
Dim trialcounter As Integer
```

```
Dim Cnt As Integer
```

```
Dim wav(12) As String
```

```
Private Type Data
```

```
trial As Integer
```

```
wavfile As String
```

```
startRT As Integer
```

```
endRT As Integer
```

```
End Type
```

```
Dim Data(12) As Data
```

```
Private Sub Form_Load()
```

```
Dim i, j, k As Integer
```

```
Dim temp As Data
```

```
' これらは音声ファイル名。実際のファイル名には.wav が続きます。
```

```
wav(0) = "a": wav(1) = "b": wav(2) = "c": wav(3) = "d": wav(4) = "e"
```

```
wav(5) = "f": wav(6) = "g": wav(7) = "h": wav(8) = "i": wav(9) = "j"
```

```
wav(10) = "k": wav(11) = "l"
```

```
*** randomize ***
```

```
Randomize
```

```
For i = 0 To 12
```

```
Data(i).wavfile = wav(i)
```

```
Data(i).endRT = -1
```

```
Data(i).startRT = -1
```

```
Next i
```

```
// shuffle
```

```
For i = 0 To 120
```

```
    j = Int(Rnd() * 12)
```

```
    k = Int(Rnd() * 12)
```

```
    temp = Data(k)
```

```
    Data(k) = Data(j)
```

```
    Data(j) = temp
```

```
Next i
```

```
For i = 0 To 12
```

```
    Data(i).trial = i
```

```
Next i
```

```
trialcounter = 0
```

```
label1 = "trial: " & trialcounter + 1: label2 = Data(trialcounter).startRT & "ms"
```

```
label3 = Data(trialcounter).endRT & "ms": label4 = Data(trialcounter).wavfile
```

```
End Sub
```

```
Private Sub Form_KeyPress(KeyAscii As Integer) '何かキーを押したときの
```

```
' イベント (IME を OFF にしておく必要があります)
```

```
' スペースキーを押したら、タイマーの使用をはじめる or やめる。
```

```
If trialcounter < 12 Then ' 12 = t_trials
```

```
    If KeyAscii = 32 Then
```

```
        If Timer1.Enabled = False Then 'はじめ
```

```
            Cnt = 0
```

```
            Timer2.Enabled = True
```

```
            label1 = "測定中"
```

```
            txtOutput.Text = txtOutput.Text & "--[" &
```

```
            Data(trialcounter).wavfile & "]" & vbCrLf
```

```
        Else 'やめ
```

```
            Call sndPlaySound("", 0)
```

```
            Timer1.Enabled = False
```

```
            txtOutput.Text = txtOutput.Text & EndTime & vbCrLf
```

```
        Data(trialcounter).endRT = EndTime
```

```
        txtOutput.Text = txtOutput.Text & Data(trialcounter).startRT & vbCrLf
```

```
        txtOutput.Text = txtOutput.Text & Data(trialcounter).endRT & vbCrLf
```

```

Label2 = Data(trialcounter).startRT & "ms"
Label3 = Data(trialcounter).endRT & "ms"
Label4 = Data(trialcounter).wavfile

trialcounter = trialcounter + 1
label1 = "trial: " & trialcounter + 1
Label4 = Data(trialcounter).wavfile

        If trialcounter = 12 Then      ' 12 = t_trials
            label1 = "終了しました"
        End If
    End If
Else
    Exit Sub
End If
Else
    label1 = "終了しました"
End If
End Sub
Private Sub Form_MouseDown(Button As Integer, Shift As Integer, _
x As Single, y As Single)
    DrawNow = True
    Cnt = Cnt + 1
    If Cnt = 1 Then
        StopTime = timeGetTime()
        RacTime = StopTime - StartTime
        Debug.Print RacTime
        Data(trialcounter).startRT = RacTime
    Else
        Exit Sub
    End If
End Sub
Private Sub Form_MouseUp(Button As Integer, Shift As Integer, x As Single, _
y As Single)
    DrawNow = False

    StopTime = timeGetTime()
    EndTime = StopTime - StartTime
End Sub
Private Sub mnuData_Click()

```

```

Dim Save
Dim i As Integer

Save = InputBox("ファイル名を入力してください", "データの保存")
If vbOK Then
    Dim tDoc As String
    tDoc = txtOutput.Text
    'ファイルを開く
    Open "data¥" & Save For Append As #1
    'ドキュメントをファイルに書き込む
    Print #1, "-----"
    Print #1, tDoc
    'ファイルを閉じる
    Close #1
    txtOutput.Text = ""
    'ファイルを開く
    Open "data¥" & Save & ".dat" For Append As #2
    'ドキュメントをファイルに書き込む
    Print #2, "Subject Trial wav StartRT EndRT"
    For i = 0 To 12 - 1      ' 12 = t_trials
        Print #2, Save, Data(i).trial + 1, Data(i).wavfile, Data(i).startRT,
Data(i).endRT
    Next i

    'ファイルを閉じる
    Close #2
    txtOutput.Text = ""
Else
    Exit Sub
End If
End Sub
Private Sub mnuExit_Click()
    End
End Sub

'プログラムを終了する
Private Sub mnuStart_Click()
' Form_KeyPress (32)
End Sub

```

```

Private Sub Timer1_Timer()
    Timer2.Enabled = False
    If DrawNow = True Then

        Dim mY
        Dim mX

        mX = objMouse.PositionX
        mY = objMouse.PositionY
        'テキストボックスに出力
        txtOutput.Text = txtOutput.Text & mX & "," & mY & vbCrLf
    Else
        Exit Sub
    End If
End Sub

Private Sub Timer2_Timer()
    Dim fname As String

    StartTime = timeGetTime()
    fname = Data(trialcounter).wavfile & ".wav"
    Call sndPlaySound(fname, SND_ASYNC Or SND_NOSTOP)
    Timer1.Enabled = True
End Sub

'***** PlayWave.bas *****
Option Explicit
'Wave ファイルの再生・停止を行なう API
Public Declare Function sndPlaySound Lib "winmm.dll" _
    Alias "sndPlaySoundA" _
    (ByVal lpszSoundName As String, _
    ByVal uFlags As Long) As Long

'sndPlaySound 関数の uFlags で使用する定数
Public Const SND_SYNC = &H0 '再生終了後、制御を戻す
Public Const SND_ASYNC = &H1 '関数実行後、直ちに制御を戻す
Public Const SND_LOOP = &H8 '停止を命令するまで再生を繰り返す
                              '(SND_ASYNC も一緒指定する必要あり)

```

```

Public Const SND_NODEFAULT = &H2 '指定した Wave ファイルが見つから
                                  'なかった場合、デフォルトの Wave ファイルを再生しない
Public Const SND_MEMORY = &H4 'メモリ上の Wave ファイルを再生する
Public Const SND_NOSTOP = &H10 '現在 Wave ファイルが再生中の場合、
                                  '再生を中止する

'***** clsMouse.cls *****
Option Explicit
Private lngAPIReVal As Long
'グローバル変数
Private lngScaleMode As Long '設定する単位
Private lngPositionX As Long 'カーソルの位置 x 座標
Private lngPositionY As Long 'カーソルの位置 y 座標

'座標を設定する構造体
Private Type POINTAPI
    x As Long 'x 座標
    y As Long 'y 座標
End Type

Private Declare Function GetCursorPos Lib "user32" _
    (lpPoint As POINTAPI) As Long 'カーソルの表示位置を取得する API
Private Declare Function SetCursorPos Lib "user32" _
    (ByVal x As Long, ByVal y As Long) As Long 'カーソルの表示位置を設定する API
Private Sub Class_Initialize()
    lngScaleMode = vbPixels '初期化処理
End Sub

Public Property Get PositionX() As Long
    'カーソルの位置情報を取得
    Call GetPosition
    PositionX = lngPositionX
End Property

Public Property Let PositionX(lngNewValue As Long)
    'カーソルの位置情報を取得
    Call GetPosition
    lngPositionX = lngNewValue

```

```
Call SetProperty 'カールの位置情報を設定
End Property

Public Property Get PositionY() As Long
'カールの位置情報を取得
Call GetPosition
Position Y = lngPosition Y
End Property

Public Property Let PositionY(lngNewValue As Long)
Call SetProperty 'カールの位置情報を取得
lngPositionY = lngNewValue
Call SetProperty 'カールの位置情報を設定
End Property

Private Sub GetPosition()
Dim typPointAPI As POINTAPI
'カール位置を取得
lngAPIReVal = GetCursorPos(typPointAPI)

'取得した値の振り分け
With typPointAPI
lngPositionX = .x
lngPositionY = .y
End With
End Sub

Public Sub SetProperty()
'カール位置を設定
lngAPIReVal = SetCursorPos(lngPositionX, lngPositionY)
End Sub
```