

東海大學資訊管理研究所
碩士學位論文

個人風格手寫字形合成之研究-使用行動裝置
Study of Chinese Font Synthesis in Personal Handwriting Style
- Using Mobile Device



指導教授：林正偉 博士

研 究 生：謝承穎 撰

中華民國 105 年 7 月

東海大學資訊管理學系碩士學位

考試委員審定書

資訊管理學系研究所 謝承穎 君所提之論文

個人風格手寫字形合成之研究-使用行動裝置

經本考試委員會審查，符合碩士資格標準。

學位考試委員會 召集人：張瑞益 (簽章)

委員：

高宏宇

柯慶華

吳榮心

林正偉

中華民國 105 年 7 月 21 日

論文名稱：個人風格手寫字形合成之研究-使用行動裝置

校所名稱：東海大學資訊管理學系研究所

畢業時間：105 年 7 月

研究生：謝承穎

指導教授：林正偉

論文摘要：一套完整的中文字型通常涵蓋上千個比英文字母複雜的中文字體和符號，即使由字型專家來設計一套字型，也會耗費大量的時間和精力。儘管可嘗試從筆劃和部件合成中文字，但要一次就合成大量的中文字體並不容易。在我們先前研究中，透過截取受測者的手寫字來合成有個人風格的字型。根據我們的分析，一名使用者只需要手寫約 400 個左右的字就可以創造一套具備 3,914 個常用繁體字的個人字形。在這篇論文裡，我們提供一個方便的 APP 給使用者，讓他們透過 APP 一個字接著一個字的寫字，APP 會追蹤辨識使用者寫下的每一個筆劃，就可以截取這個字的各個部件。當我們想要合成某個字時，只要將合成此字所需的部件，根據此字的部件位置及大小等資訊，放置及縮放到適當的位置和大小就行了。實驗結果顯示，透過 APP，我們的系統可以輕鬆的擷取字的部件，讓使用者更快速地創造出自己的手寫字型。

關鍵詞 - 中文字型、個人字型、手寫字、字體、部件擷取

Title of Thesis : Study of Chinese Font Synthesis in Personal Handwriting Style – Using
Mobile Device

Name of Institute : Tunghai University, Graduate Institute of Information Management

Graduation Time : 07/2016

Student Name : Cheng-Ying Hsieh

Advisor Name : Jeng-Wei Lin

Abstract : To create a Chinese font, even for professional font designers, it still has to take a lot of time and efforts. A Chinese font typically packs glyphs of several thousand or more Chinese characters, and many of them are very complicated. Previously we had presented an easy and fast solution for an ordinary user to create a Chinese font of his or her handwriting style. We adopted the approach: synthesizing Chinese characters using components extracted from users' handwritings. According to our analysis, a user has to handwrite 400 or so Chinese characters to create a personal handwriting font, with commonly-used 3,914 traditional Chinese characters. In this thesis, we present a convenient APP for smartphone users, on which they can write one character by one character. The APP can track every stroke, recognize and extract components from the user's handwritings. Thus, target Chinese characters can be synthesized from these extracted components, by placing them properly according to their position and size information. The experiment results show that with the APP, our system can easily extract required components from users' handwritings. As a result, users can easily create a Chinese font in their personal handwriting style.

Keywords : Chinese font; personal font; handwrite font; glyph synthesis; component extraction.

目 錄

第一章 緒論	1
第一節 研究背景與動機	1
第二節 一般造字流程	2
第三節 研究目的	4
第二章 文獻探討	5
第一節 相關研究	5
第二節 字形合成的基本概念	7
第三節 中文編碼相關標準及專有名詞	9
第三章 研究方法	11
第一節 前置階段	11
第二節 生產階段	14
第四章 實驗設計與結果	25
第一節 實驗流程	25
第二節 驗證 APP 的使用結果	31
第三節 字形合成結果	32
第五章 結論	33
參考文獻	34

圖 次

圖 1-1 BIG5 中文筆劃分佈	3
圖 2-1 手寫字合成概念圖	7
圖 2-2 中文字部件拆解及合成概念圖	8
圖 3-1 漢字構形資料庫擷圖	12
圖 3-2 切字介面	13
圖 3-3 客製化稿紙	14
圖 3-4 手寫字 APP 示意圖	15
圖 3-5 部件組成資訊	16
圖 3-6 手寫字儲存結構(1).....	17
圖 3-7 手寫字儲存結構(2).....	19
圖 3-8 模擬合成	22
圖 3-9 部件縮放概念圖	22
圖 4-1 實驗架構	26
圖 4-2 APP 擷取使用者手寫字流程圖	29
圖 4-3 APP 修改畫面擷圖	30
圖 4-4 合成結果	32

表 次

表 1-1 中文碼標準字數	3
表 3-1 構字式表	11
表 3-2 手寫字儲存結構圖鍵值表(1).....	18
表 3-3 手寫字儲存結構圖鍵值表(2).....	20
表 4-1 資料庫 schema (1).....	27
表 4-2 資料庫 schema (2)	28
表 4-3 APP 使用統計結果統計表	31



第一章 緒論

第一節 研究背景與動機

中國書法被認為能反映一個人的情緒，教育程度和個性。許多華人一直在尋找個人化的中文字型，這樣不僅可以在數位創作中呈現它們的點子，也可以向讀者們展現他們的個性和感受。然而目前中文字型庫或產品的選擇並不多。如表 1-1 所示使用在電腦系統的中文字碼標準，通常會有上千個中文字，甚至許多的中文字比英文字母還要複雜。因此，一般人要製作有個人手寫風格的字形並不容易。直到現在，中文字型仍然由專業的設計師或公司製作。他們通常會花數個月至幾年的時間來設計及美化字型中所有中文字的外形。

2015 年 9 月 8 號，justfont(就是字股份有限公司)¹ 團隊設計的金萱體，在群眾募資平台 flyingV² 推出不到 80 分鐘就達到 150 萬元的募資目標，更在 11 個小時後，募資超過千萬台幣。近期也有不少手寫字被使用於市場上的案例，例如獨立音樂代表人物-陳綺貞，為了帶給歌迷不同的體驗以及展現風格，她本人前前後後共親自手寫了約 1 萬 8000 個字，在作品「時間的歌 巡迴演唱會影音記錄 DVD」中使用了個人手寫字做為字幕，也成為第一位擁有個人手寫字型的台灣音樂創作者。網路上也出現了由鋼筆手寫愛好者組成的 facebook 社團 - 鋼筆旅鼠本部連³，成員會在上面分享各種手寫字的作品和技巧，或推廣相關產品。

從這些例子可以發現，字形滿足了個人對於風格與感情的展現、概念的表達、文化性上等多種用途。雖然台灣有一些造字公司早已經開發出中文字型，但選擇性仍然不夠多元，而且這些公司通常有自己的中文造字工具及相關專利，也間接導致中文造字技術及相關知識在台灣並不普及的現象。雖然目前也有許多造字工具，但這些專業軟體對於想踏進造字領域的一般人而言，也是一種門檻。

因此，我們希望讓一般人也能夠快速輕鬆的打造自己的手寫中文字型。

¹ justfont <http://www.justfont.com/>

² <https://www.flyingv.cc/projects/8250>

³ <https://www.facebook.com/groups/676899475660255/>

第二節 一般造字流程

Justfont 的設計師也分享了他們製作字形的流程⁴，主要分為下列 4 個步驟：

Step 1 決定字型風格。

Step 2 決定字體的框架。

Step 3 試做 300 個基礎字，接著再製作 7000 字，此時已進入量產階段，最後可擴充到 1 萬。在此階段設計師也會進行不同字之間的混合編排，而進行此步驟時最好要使用專業的造字軟體，因為專業造字軟體能確保字體在統一標準尺寸下設計，以方便混排測試字與字之間的距離及視覺效果。

Step 4 無止盡的微調，這也是最花時間的部分。

造字需要牽涉到設計面及工程面，因此多人之間的協調和溝通也是成本之一。中文字形的數量及結構是造字成本高的兩個主要因素，表 1-1 顯示各種中文字集編碼標準所涵蓋的字數，要留意的是，新的中文字仍然不斷的被創造出來，這些字集中包含日常生活常用以及不同類型的中文字。而且中文字也比英文字母複雜。圖 1-1 為 BIG5 的中文字筆劃分佈圖，從中我們可以看出，在擁有 13,053 字數的 BIG5 裡，筆劃大於 10 的字數占了約 70%，而且這些字的組成都很複雜。因此，大部分中文字型仍由專業的文字設計師或公司製作，也會耗費數個月或年的時間修正及美化這些字，對一般人來說，要製作一套個人手寫字型更是困難的事。

⁴ justfont blog <http://blog.justfont.com/2015/07/check-this-out-b4-downloading-fonts/>

表 1-1 中文標準字數表

	<i>Publish Year</i>	<i>No. of Chinese characters</i>	<i>Note</i>
BIG5	1984	13,051	Traditional Chinese
CNS 11643	1992	48,027	Traditional Chinese
GB 2312	1980	6,763	Simplified Chinese
GB 18030	2000	27,533	Simplified Chinese
Unicode 1.0	1993	20,902	Chinese, Japanese, Korean, etc.
Unicode 3.1	2001	70,195	

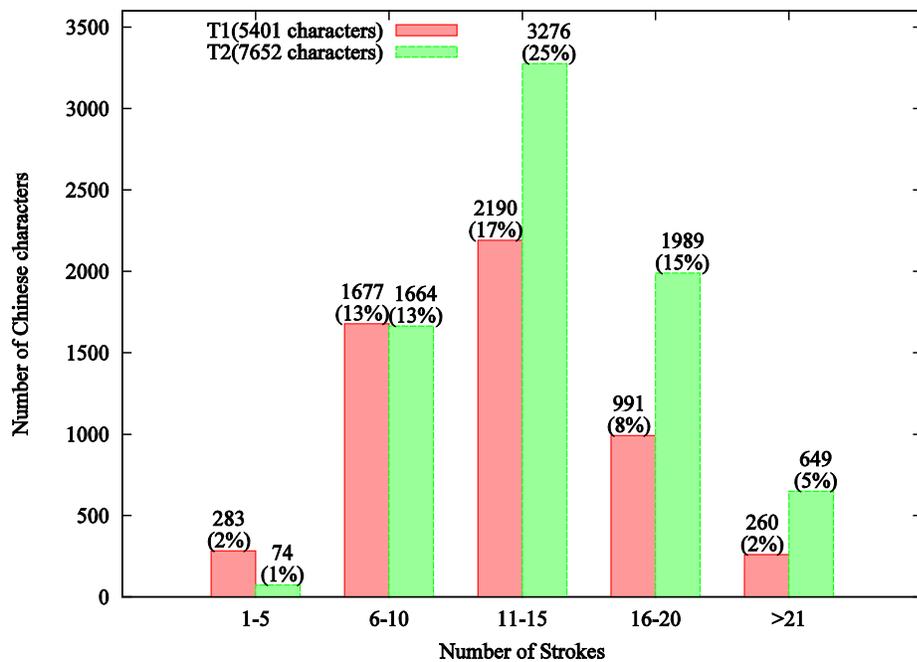


圖 1-1 BIG5 中文筆劃分佈(T1 和 T2 分別代表使用率最高及第二高的子集)

第三節 研究目的

根據上述研究背景及動機，為了能夠大規模生產中文手寫字型，並降低過程中的時間及人力成本，我們需要一個快速又有效率，人人都能使用的方法。本研究透過行動裝置擷取使用者的手寫字，抽取合成其他中文字所需的候選部件，加上以部件為基礎的方法來合成中文字，使用者只要透過行動裝置寫下 392 個字，就可以輕鬆擁有自己的手寫中文字型。

本文的結構如下。在簡介完後，我們將在第三章呈現解決方案，也就是前置準備和生產階段的實驗方法。在本文的第四章，我們將介紹 APP 以及初步的實驗結果。最後我們針對本研究提出結論及後續議題。



第二章 文獻探討

第一節 相關研究

雖然隨著科技及技術的進步，有許多相關的造字工具及平台被開發出來。例如 FontForge (G. Williams, 2003) 就是一套開源的字型編輯器，或者是 Glyphs⁵，為 MAC 平台上常用的字型設計軟體，它們可以創造或修改字體以及產生字型檔。然而，要使用這些工具，使用者必須具備一定程度的電腦繪圖及影像處理技術及相關知識，而且這些工具通常對中文支援並不完善。

Lim (1995) 和 Wong (2000) 等提出了利用中文字最基礎原件 - 筆劃，來合成字體的概念，來取代除了自己利用手動繪圖造字以外的方法來產生字型。除此之外，Wong (1995) 和 Xu (2005) 等則使用了比筆劃更高階的結構 - 元件(部件)，來合成文字。然而，這些研究通常是針對一次合成一個中文字的情況所設計。因此目前仍缺乏針對個人風格手寫字型有系統性的，一次合成大量中文字的解決方案。

Zhang (2001) 及 Zhang (2007) 介紹如何將古代書法數位化並向量化成字體。Zhu (2008) 和 Xia (2010) Zitnick (2013) 根據 Zhang 的方法來美化手寫中文字。

對設計師或創作者來說，他們可能一次只需要使用少量的中字體，因此研究團隊之前提出了一種根據需求，漸進式的方法產生中文字型(Lin 等,2015)。團隊開發出一個方便的雲端字型設計平台 FontCloud (目前更名為守寫字⁶)，透過這個平台，使用者不需要耗費大量時間手動畫完並製作所有的中文字。只要在他們需要時，創造夠用的中文字型就可以了，每次產生一些中文字型，積少成多，最後也能創造一套完整的中文字型。

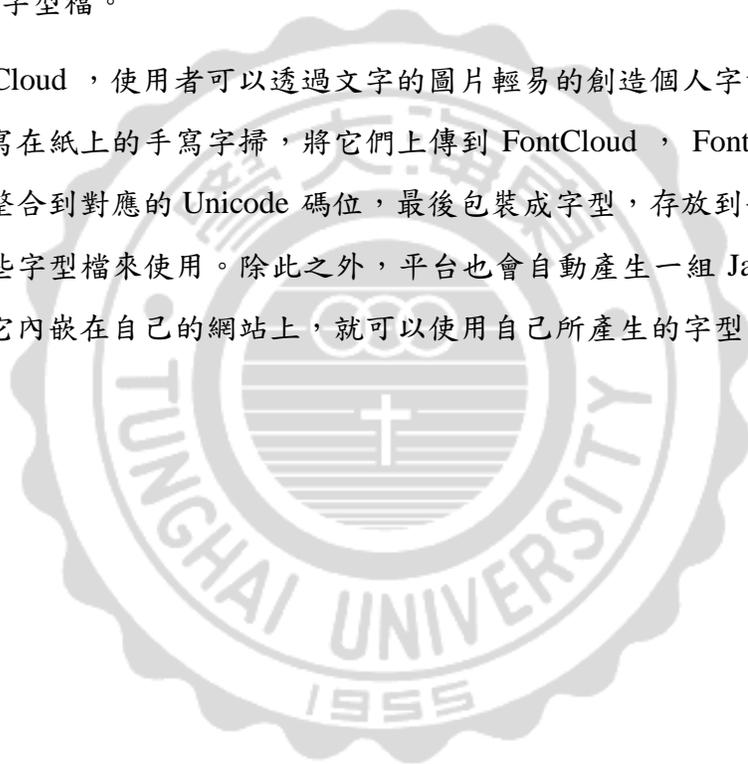
利用 FontCloud 產生個人手寫字型需要下列步驟:

⁵ Glyphs available at <https://glyphsapp.com>

⁶ 守寫字 www.writes.com.tw

1. 挑選想要使用的字，並將它們上傳到 FontCloud 。
2. 完成第一步驟後，FontCloud 就能讓你列印出特定的稿紙，上面的字就是剛剛所輸入的字。
3. 手寫完稿紙上的字
4. 掃描或拍照上傳回 FontCloud
5. 檢查、調整以及管理字形
6. 下載並安裝字型檔。

在 FontCloud ，使用者可以透過文字的圖片輕易的創造個人字型。我們可以掃描中文書法或寫在紙上的手寫字掃，將它們上傳到 FontCloud ， FontCloud 將這些檔案向量化，並整合到對應的 Unicode 碼位，最後包裝成字型，存放到平台中。使用者日後可下載這些字型檔來使用。除此之外，平台也會自動產生一組 JavaScript 程式碼，使用者只要將它內嵌在自己的網站上，就可以使用自己所產生的字型。



第二節 字形合成的基本概念

根據我們先前的研究 (Lin 等 2013,2015)，我們嘗試有系統的利用從使用者的手寫字中擷取出的部件，一次合成大量中文字。基本概念如圖 2-1 和圖 2-2 所示。經過初步分析，我們試圖回答以下問題。

假設我們要產生一組中文字：

1. 我們所需要的部件有哪些？如圖 2-1 所示，我們的系統必須知道「妍」可以被「女」和「开」合成。在大量合成中文字之前，我們必須知道合成這些字所需的部件。
2. 每個使用者該手寫多少個中文？如圖 2-2 所示，使用者只要手寫「落」、「份」、「是」這三個字，就有 30 個字可以被合成。我們希望使用者所需手寫的字數越少越好。
3. 如何從使用者的手寫字中擷取出部件？如圖 2-1 所示，當使用者寫了「媽」這個字，我們的系統必須從中擷取出「女」，供日後合成其他字時使用。
4. 給定一個想要合成的中文字，如何決定部件的擺放位置和大小？如圖 2-1 所示，若要合成「妍」這個字，我們必須將使用者手寫「女」和「开」的擺放在對的地方以及縮放到適當的大小。

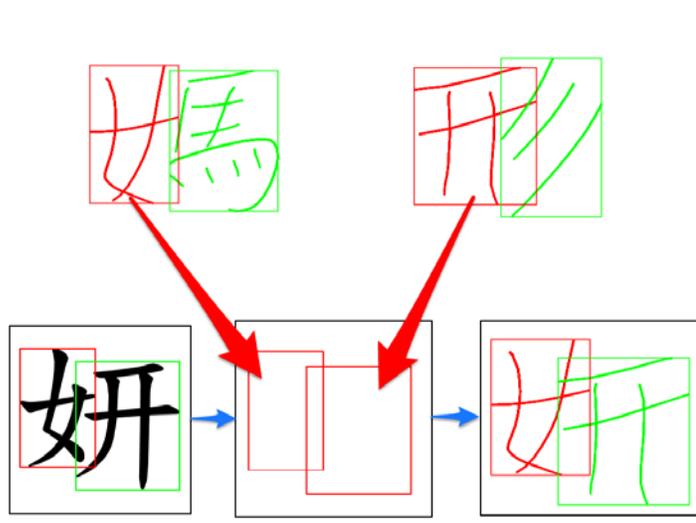


圖 2-1 手寫字合成概念圖

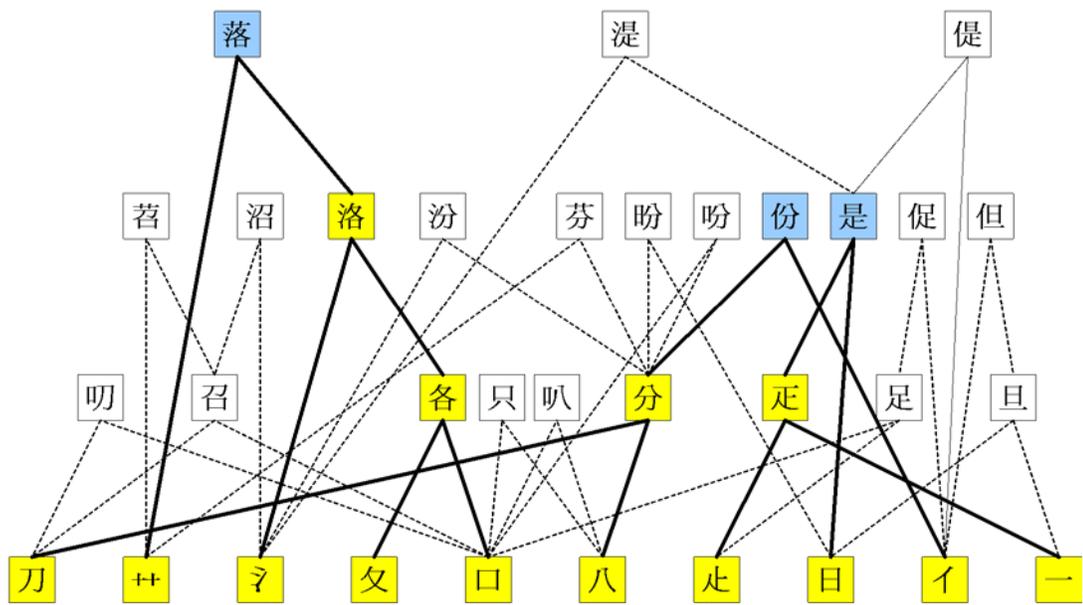


圖 2-2 中文字部件拆解及合成概念圖

根據之前的研究，要創造一套具備 3,914 個常用繁體字的個人手寫字型，一名使用者只需要手寫約 392 個左右的字（約 10% 左右）。實驗結果顯示，即使有少數的合成中文字還需要手動的調整，但使用者可以更容易且迅速的創造以自己手寫風格為基礎的中文字型。

第三節 中文編碼相關標準及專有名詞

壹、Big5

早期台灣尚未有能夠讓不同中文軟體相容的中文編碼，因此資策會為了文書、資料庫、試算表、通訊及繪圖五大套裝軟體而設計了 Big5，完成於 1983 年，又被稱為大五碼。但因為 Big5 並非正式的國家標準，僅僅是業界標準，再加上之後各廠商又推出不同的衍伸版本⁷，以及「許功蓋」之衝碼等問題，使得 Big5 逐漸被 Unicode 取代，然而它對早期台灣及繁體中文進入電腦產生深遠的影響。目前最新的版本為 Big5-2003，而該版本也被收錄到臺灣國家標準 CNS 11643 中，共收錄 13,060 個字。

貳、CNS 11643

國家中文標準交換碼(後改名為中文標準交換碼)，顧名思義，它是被用來資訊交換的編碼方案，為了因應 1980 年代業界出現了各種不同的編碼，例如倚天碼、王安碼、電信碼等等，導致資訊交換及整合困難，由學者及專家討論解決方法，最終在 1986 年，由中央標準局正式公布此標準。

參、GB 2312

為中華人民共和國使用的中文編碼，全名為《信息交換用漢字編碼字符集基本集》，共收 6763 個滿足日常生活所需的漢字，但對於罕用字及繁體字的支援不完善，因此有了後續的 GBK 及 GB 18030 以解決此問題。

肆、ISO 10646 / Unicode

為了解決今日跨國資料交換需求頻繁的問題，先由國際標準組織 ISO 與國際電工聯盟 IEC 提出了 ISO10640 草案，以滿足各國的需求，同時期另一個組織 Unicode Consortium 也推出 Universal Code(Unicode)，試圖解決相同問題。之後這兩個組織決定一起合作將各自的版本互相同步。目前 Unicode 9.0 共有 128,172 個字元，此版本也新增了 72 個繪文字⁸。

⁷ <http://www.cns11643.gov.tw/AIDB/encodings.do#encode4>

⁸ <http://emojipedia.org/unicode-9.0/>

伍、 外碼

當我們使用輸入法時，所輸入一連串的符號，就是外碼。不同輸入法有著不同的輸入方式，因此外碼的組合也就不一樣。例如注音符號，倉頡輸入法。

陸、 內碼

相較於外碼，內碼是指某個字元在電腦中所代表的碼，一個內碼可以表示一個字元。以 Big5 的中文字「一」為例，其內碼為 A440，但對 Unicode 而言則是 4E00。

柒、 碼位

又稱為 code point 或 code position，我們可以把它視為座位表，每個位子都可以有一個內碼的值被存放在裡面，以擁有 128 個碼位的 ASCII 為例，它的第 1 個碼位是 0_{16} ，第 2 個碼位是 1_{16} ，第 16 個碼為是 0F，一直到最後一個碼位 7F，也就是 127_{10} 。



第三章 研究方法

第一節 前置階段

研究方法分成三個階段，分別為前置、產生以及合成階段，在前置及產生階段，我們試著回答在第二章第二節所述的 4 個問題：

壹、 部件分析

要回答問題 1，也就是需要使用哪些部件，我們根據先前對於國際漢字編碼及各種漢字資料庫(Juang,2005;Lin & Lin,2012)的研究，進行了中文字部件的分析(Shu,1997; Juang,2005)。

Juang 等(2005)的研究成果，如表 3-1，是將所有被收錄進漢語大字典的中文字拆解成約 1,000 個部件。我們延伸了他們的研究成果，解構了其他屬於 Unicode3.1 的中文字而得到了更多部件(Lin & Lin, 2012)。

表 3-1 構字式表(漢語大字典內的中文字被分解為大約 1000 個部首)

	Operators	Explanation	Example
Two components	ㄩ	Left and Right	順 = 川 ㄩ 頁
	ㄟ	Top and Bottom	含 = 今 ㄟ 口
	ㄚ	Outer and Inner	圍 = 口 ㄚ 韋
More than two components	⊠	Components enclosed between Begin (⊠) and End (⊡)	牘 = ⊠ 片 戶 甫 ⊡
	⊡		
Duplications	∪	Two same components in vertical	炎 = ∪ 火
	∩	Three same components in vertical	彗 = ∩ 戶
	∞	Two same components in horizontal	朋 = ∞ 月
	∞∞	Three same components in horizontal	𠂔 = ∞∞ 去
	∪∪	Three same components in triangular	焱 = ∪∪ 火
	∞∞∞	Four same components in horizontal	
	∩∩∩	Four same components in vertical	
	∪∪	Four same components in 2x2 array	燚 = ∪∪ 火

我們修改了某些部件的拆解結果，因為我們的目的跟之前的研究 (Juang,2005;Lin & Lin,2012) 不一樣。例如「升」原本被拆解為「丿」和「升」，但在合成該字時的效果會不佳，因此將它拆解成獨立的「升」比較適合。

在實務上，我們利用漢字構形資料庫 (Chinese Character Component Searching System of Academia Sinica) 裡的資訊做為文字擷取及合成的重要原始資訊，如圖 3-1 所示。

編號	字碼	連接	筆畫	部件序	字根序	字根組	Big5	Unicod
1674	時	1	10	日寺	日士寸	士寸日	AEC9	6642
1675	晉	2	10	廾日	一ムム一日	一一ムム日	AECA	6649
1676	晏	2	10	日安	日宀女	宀女日	AECB	664F
1677	晃	2	10	日光	日艹儿	儿艹日	AECC	6643
1678	晒	1	10	日西	日西	日西	AECD	6652
1679	晌	1	10	日向	日ノ門口	ノ門口日	AECE	664C
1680	暄	1	10	日亘	日二日	二日日	AECF	6645
1681	晁	2	10	日兆	日儿宀	儿日宀	AED0	6641
1682	書	2	10	聿日	聿日	日聿	AED1	66F8
1683	朔	1	10	艹月	艹中月	艹中月	AED2	6714
1684	朕	1	10	月关	月ノ一大	一ノ大月	AED3	6715
1685	朗	1	10	良月	、良月	、月良	AED4	6717
1686	校	1	10	木交	木宀父	宀木父	AED5	6821
1687	核	1	10	木亥	木宀夕人	人宀夕木	AED6	6838

圖 3-1 漢字構形資料庫擷圖

貳、找出手寫字的範圍

為了解決問題 2，我們設計了一個演算法來找出適合合成其它中文字的字集，因此使用者只要寫一小部份的中文字就行了。套用不同的篩選標準也可以找出不同特色的字集，例如涵蓋更多部件的字集或常用字優先的字集。在本研究，我們使用以常用字優先為標準來篩選手寫字，因為這些字很常出現，所以將它做為手寫字會比用合成的好。以這個標準經過貪婪演算法篩選的字數為 392 個字 (Chang & Hong, 2015)。

參、 部件的位置及大小資訊

針對問題 4，如圖 3-2，我們建立了一個方便的 Web 服務，可以標記中文字的部件位子和大小。因為這個部分只需要做一次就行了，所以我們雇用了多名學生來幫助我們標記出這些資訊，我們目前大約需要標記兩萬個字。

當我們要合成某個字時，我們的系統使用此字之前被標記過的部件位置及大小等資訊作為參考依據，我們也增加了修正部件的選項，在未來若有需要，可以根據需求修改文字的組成部件。



圖 3-2 切字介面

第二節 生產階段

壹、 部件擷取

一、 透過客製化稿紙擷取部件

要解決問題 3，也就是從使用者的手寫字中擷取部件，有很多方法。以 FontCloud 所採用的方法為例，使用者在圖 3-3 的特定稿紙上的空格內，依照每個格子左邊的範例寫字，再將這些手寫字掃描成影像檔並上傳到系統，接著使用影像處理的技術從圖檔擷取部件。如果使用者對其中某些字不滿意的話，就必須重新寫一張相同的字。



天	地	玄	黃	宇	宙	洪	荒	日	月	盈	
辰	辰	宿	列	張	寒	來	暑	往	秋	收	
冬	藏	閏	餘	成	歲	律	呂	調	陽	雲	
騰	致	雨	露	結	為	霜	金	生	麗	水	
玉	出	昆	岡	劍	號	巨	闕	珠	稱	夜	
光	果	珍	李	柰	菜	重	芥	姜	海	咸	
河	淡	鱗	潛	羽	翔	龍	師	火	帝	鳥	
官	人	皇	始	製	文	字	乃	服	衣	裳	
推	位	讓	國	有	虞	陶	唐	吊	民	伐	
罪	周	發	殷	湯	坐	朝	問	道	垂	拱	
平	章	愛	育	黎	首	臣	伏	戎	羌	遐	
邇	壹	體	率	資	歸	王	鳴	鳳	在	竹	
白	駒	食	場	化	被	草	木	賴	及	萬	
方	蓋	此	身	四	大	五	常	恭	惟	鞠	
養	豈	敢	毀	傷	女	慕	貞	潔	男	效	

圖 3-3 客製化稿紙

二、透過行動裝置擷取部件

在本研究中，我們開發了一款 APP，讓使用者在手機或平板上寫字，如圖 3-4，APP 每一次會顯示一個字，要求使用者透過螢幕手寫這個字。我們利用中文字的筆畫來擷取部件，因為 APP 已經知道所有手寫字的部件組成資訊，所以可以根據筆劃的起始點擊順序，輕易的追蹤並將筆劃分類到不同的部件，並且將部件以不同的顏色顯示。如果 APP 沒辦法將部件分類，使用者可以選擇微調或重寫該字。這個解決方案是漸進式的，使用者可以運用零碎的時間來寫字，我們注意到當你寫越多字時，越多候選的部件被擷取。而且我們也不用等到所有手寫字都寫完時，才開始合成其它字。



圖 3-4 手寫字 APP 示意圖

要透過筆畫數量來擷取使用者的手寫字部件，必須先產生每個手寫字的筆畫數量、部件數量、部件位置及大小等相關資訊，我們利用漢字構形資料庫所收錄的中文筆畫相關資訊，產生擷取 392 個手寫字所需要的資料給 APP 使用，如圖 3-5。

word: 黻	rad1: 黹	rad2: 友
<pre> { "unicode": "9EFB", "expr": { "type": "Buffer", "data": [220, 176, 201, 165] }, "operator": 1, "rad_order": "黹友", "root_order": "黹友", "allStrokes": "17", "rads": [rad1, rad2] } </pre>	<pre> { "unicode": "9EFB", "word": { "type": "Buffer", "data": [233, 187, 185] }, "rank": 1, "posx": 0.04508, "posy": 0.143382, "w": 0.48472, "h": 0.714436, "stroke": "12", "operator": 1, "rad_order": "黹友", "root_order": "黹友" } </pre>	<pre> { "unicode": "9EFB", "word": { "type": "Buffer", "data": [229, 143, 139] }, "rank": 2, "posx": 0.46392, "posy": 0.160473, "w": 0.4988, "h": 0.680218, "stroke": "5", "operator": 1, "rad_order": "黹友", "root_order": "黹友" } </pre>

圖 3-5 部件組成資訊

APP 靠著部件組成資訊來擷取手寫字，一旦使用者完成一個字後，按下寫完了的按鈕，該字的筆畫資訊會被 APP 提交到資料庫裡，以供日後讀取及合成字形時使用。如圖 3-6，我們透過 Firebase⁹的資料管理介面來觀察使用者手寫字的檔案結構，以「電」這個手寫字為例，它是手寫字清單中編號第 130 字，手寫字的筆畫資訊被存為 JSON 格式，APP 將使用者手寫該字的狀況紀錄在 state 屬性內，state 內又有更多其它屬性，例如 clearCount 代表重寫該字的次數。「電」有兩個部件，分別是「雨」和「电」，所以會有兩個部件陣列(strokeToRadical)，每個陣列裡的元素皆為被歸類為該部件的筆畫資訊。如圖 3-7，我們可以看出第一個部件有八畫，第二個部件有五畫，而圖中展開的部分為第二個部件的第一個筆畫，可以看出構成該筆畫的八個座標點以及相關資訊。

⁹ <https://www.firebase.com>

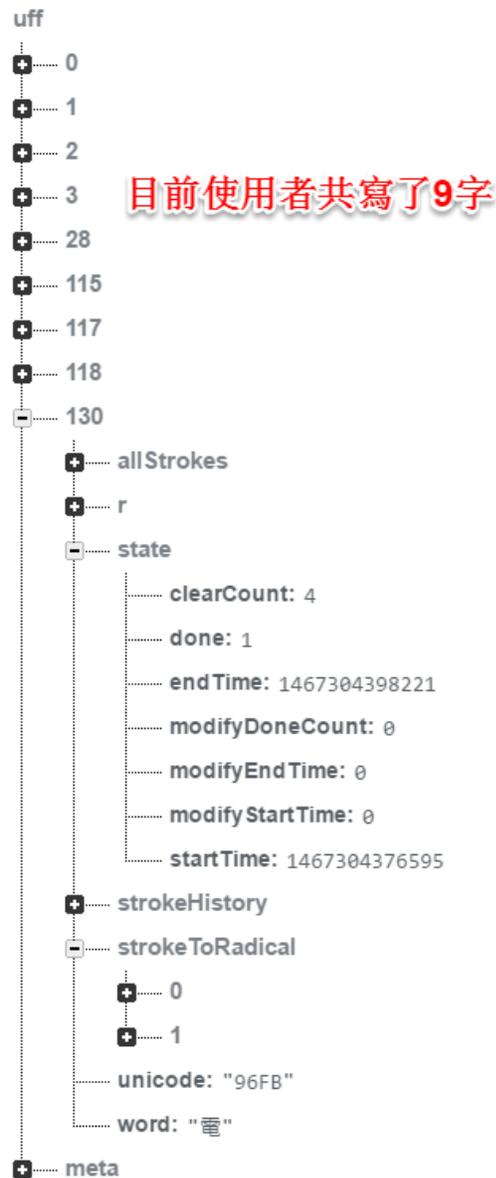


圖 3-6 手寫字儲存結構(1)

根結點 uff 下有多個屬性(child property)或鍵(key)，代表每個使用者手寫過的字，每個鍵都有相對應的值(key)，而屬性底下也可能有多個子屬性及值。相關鍵與值的解釋如表 3-2 所示。

表 3-2 手寫字儲存結構圖鍵值表(1)

key	註解
allStrokes	紀錄該字最新且所有筆畫的座標
r	紀錄該字的組成資訊，以方便參考
state.clearCount	清除該字的次數
state.done	是否已經完成該字，也就是寫完字並已擷取部件
state.endTime	紀錄寫完該字的時間
state.modifyDoneCount	修改次數
state.modifyEndTime	修改完成時間
state.modifyStartTime	開始修改時間
state.startTime	開始寫某個字的時間
strokeHistory	紀錄所有筆畫的歷史座標
strokeToRadical	紀錄筆畫被分類到各部件後的結果，如果該字有兩個部件，代表此鍵底下會有兩個子鍵，它們分別記錄了兩組連續座標，代表兩組被分類到不同部件的筆劃資訊。
unicode	該字的 unicode
word	該字的字碼
meta.lastSave.timestamp	記錄了上次使用 APP 的時間點
meta.lastSave.wordIndex	紀錄上次手寫字的進度

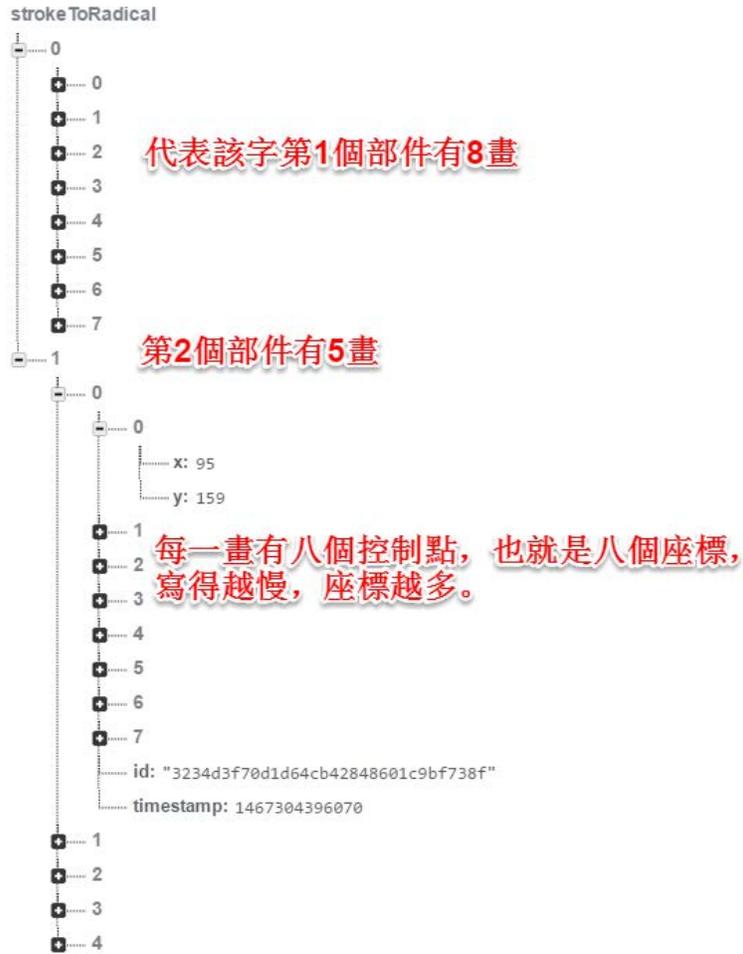


圖 3-7 手寫字儲存結構(2)

以圖 3-7 為例，strokeToRadical 底下更細部的結構，我們可以看出第一層有兩個子鍵分別為 0 和 1，分別代表該字的兩個部件，它們各自又有更多的子鍵，代表每一個筆畫，而每一個筆畫下又有子鍵，裡面記錄了該筆畫的座標，相關鍵的解釋如表 3-3 所示。

表 3-3 手寫字儲存結構圖鍵值表(2)

key	註解
strokeToRadical[0]	代表該字的第 1 個部件
strokeToRadical[1]	代表該字的第 2 個部件
strokeToRadical[1][0]	被歸類在該字第 2 個部件的第 1 個筆劃
strokeToRadical[1][0][n].x	被歸類在該字第 2 個部件的第 1 個筆劃中，第 n 個點的 x 座標
strokeToRadical[1][0][n].y	被歸類在該字第 2 個部件的第 1 個筆劃中，第 n 個點的 y 座標
strokeToRadical[1][0].id	紀錄筆畫的 id，以方便版本控制
strokeToRadical[1][0].timestamp	紀錄筆畫的完成時間，以方便版本控制



貳、字形合成

一、合成字挑選部件的選擇考量

當蒐集足夠合成某個字的部件時，就可以合成該字，並透過 Selinger(2009) 提出的演算法將該字向量化，接著再利用 FontForge(Williams,2003)。將字體合併成字型檔(Lin 等,2015)。

一個部件可能會出現很多次。例如當我們採用使用率高的字作為貪婪演算法的篩選標準時，「的」和「時」這兩個字經常被選到，因為他們有共同的部件「日」，但分別位於不同位置。「妻」和「姊」都有共同的部件「女」，「作」與「便」也有相同的部件「亻」。Lin 等(2015)也提出了各種從候選字挑選適當部件的方式。

(a) 相似位置的部件較佳。如果要合成「姑」這字，從「姊」和「媽」這兩個字裡挑選部件「女」會比從「妻」挑選恰當。

(b) 完整的手寫部件比合成的還要好，以「侍」為例，從此字擷取的「寺」，比用「士」和「寸」合成的「寺」好。

(c) 當有多個滿足條件(a)和(b)的部件時，我們會隨機從中選擇一個。

透過以上條件，我們可以確保合成的字體具有使用者的手寫風格，如圖 2-1 所示，我們根據從前置處理階段中所使用的標準楷體字型中，所擷取的大小和位置資訊擺放後選部件。

二、遞迴拆解部件及合成

在大量批次合成字之前，我們建造一個 Web 介面方便觀察合成時的部件選擇結果，如圖 3-7，「府」剛開始被拆解為「广」和「付」兩個部件，接著程式會從候選字，也就是使用者的手寫字，挑出有涵蓋部件「广」的字，「麼」被列為最優先，因為「麼」有部件「广」，而且該部件位置與「府」的「广」相似。第二個部件也是依此類推，然而，部件「付」沒有找到適合的候選手寫字時，便會根據該字的組成資訊，遞迴分解成下一層部件，因此「付」會被拆解成「亻」和「寸」。

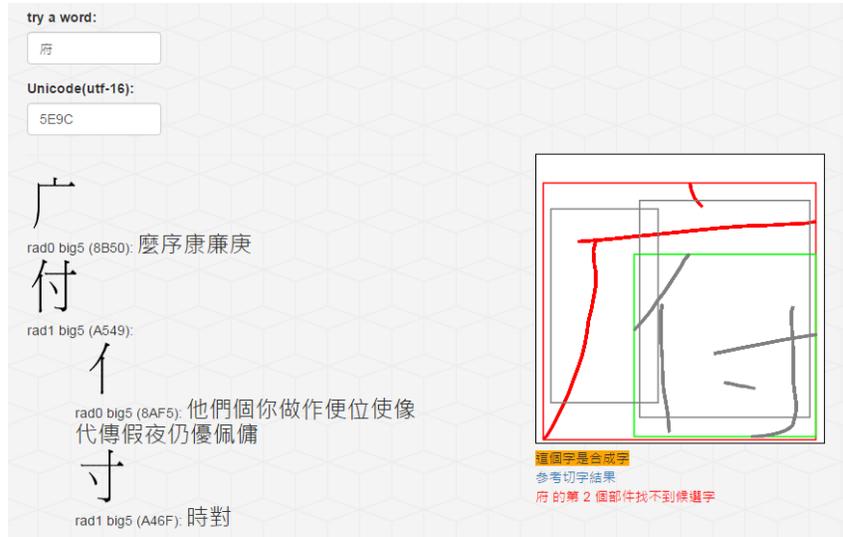


圖 3-8 模擬合成

接著在合成字形時，我們必須將候選部件擺放到正確的位置以及適當的大小，因此參考該字以標楷體為主的部件位置資訊，我們使用一個簡單的等比例縮放及平移的公式將部件適當的擺放及調整大小，同樣的，如圖 3-8，在遞迴拆解部件後，將子部件（「亻」、「寸」）再次合成時也會使用到此公式，概念如圖 3-9:

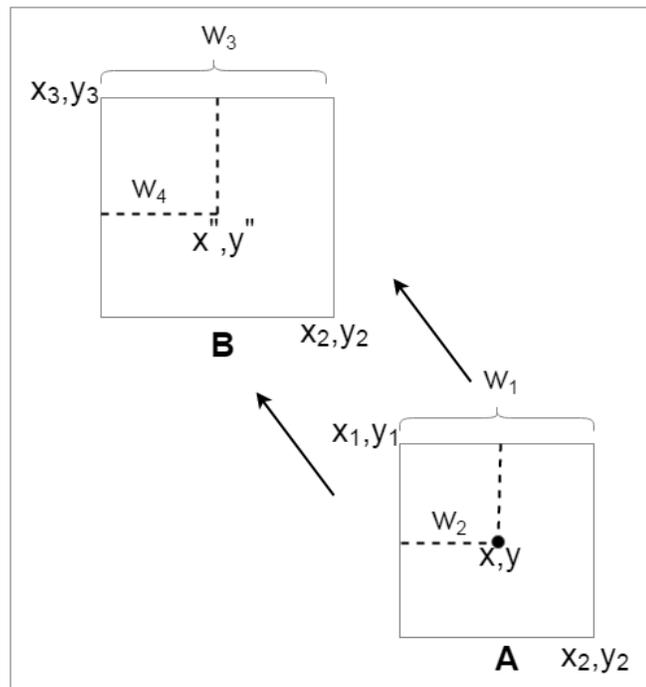


圖 3-9 部件縮放概念圖

從圖 3-7 中可看出，部件係由筆畫構成，每個筆畫裡有多個原始座標，因此縮放時必須將原始部件轉移到新的位置及適當的大小，遞迴合成時也一樣會用到此方法。如圖 3-9，給定兩個矩形 A 及 B，A 代表候選字的某個部件，B 代表目標部件，也就是候選部件 A 將會被擺放的位置及被縮放的大小。

圖 3-9 所使用的公式如下：

$$\frac{(x - x_1)}{(x_2 - x_1)} = \frac{(x'' - x_3)}{(x_4 - x_3)}$$

$$\frac{(y - y_1)}{(y_2 - y_1)} = \frac{(y'' - y_3)}{(y_4 - y_3)}$$

改寫成以下公式可清楚看出 x'' 及 y'' ：

$$x'' = \frac{(x - x_1)(x_3 - x_4)}{(x_2 - x_1)} + x_3$$

$$y'' = \frac{(y - y_1)(y_3 - y_4)}{(y_2 - y_1)} + y_3$$

X'' 及 Y'' 代表 B 的任一點，我們必須將 x 及 y ，也就是原始部件中的其中一點，移動到 x'' 及 y'' ，除此之外也要考慮到部件能夠等比例被重新縮放，因此藉由 $(x - x_1)$ 與 $(x_2 - x_1)$ ，也就是 A 的寬度的比例 ($w1$ 及 $w2$) 等於 $(x'' - x_3)$ 與 $(x_4 - x_3)$ 的比例 ($w3$ 及 $w4$)，就可以求出 x'' 及 y'' ，最後將結果套用到部件 A 每個筆畫的座標。

三、批次合成

批次合成程式也利用跟模擬合成同樣的方法進行批次合成，最後將合成結果輸出成影像檔。我們可以指定程合成單一文字或多個字組成的句子，也可以讀取文字清單已進型大量合成。目前批次合成程式需要使用到 2 個外部檔案，分別是 392 個手寫字的組成資訊檔案以及使用者手寫字的筆畫資訊檔案，前兩者必須在分析階段時取得，後者則是在使用者利用 APP 完成手寫字實驗時取得。



第四章 實驗設計與結果

在本章，我們呈現實驗及實驗結果。我們主要針對 3,914 個常用繁體中文，這些字被收錄在 BIG5 (BIG5-2003) 和 CNS (CNS 11643-1992) 的 T1 plane，作為實驗的素材。

第一節 實驗流程

我們利用先前的研究成過 (Juang,2005;Lin & Lin,2012) 分析部件結構。使用一個貪婪演算法(Lin 等,2015)，以常用字作為篩選標準，找出 392 個中文字，並請數位自願者透過 APP 手寫這些字。

實驗架構如圖 4-1 所示，流程包括部件位置大小等資訊的收集、手寫清單資訊組成檔產生、手寫字 APP 使用及驗證以及合成結果。如表 4-1 及表 4-2 所示，在實驗過程中所使用的主要資料表為目標字(word)、切字結果(subword)、輔助查詢用的字集資料表(hwref)以及常用字資料表(view_uff)。

圖 4-2 為 APP 擷取使用者手寫字部件的流程圖。當 APP 開始執行時，它一次會顯示上述手寫字集的一個字給使用者看，接著使用者開始透過螢幕手寫該字。APP 會追蹤每一個筆劃，並且根據部件分析資料庫 (Juang,2005;Lin & Lin,2012) 將筆劃歸類到不同部件，各部件由不同顏色的筆劃所組成，以方便區分及修改。當使用者寫完一個字後，他們可以檢查部件的分類是否有錯誤。如果部件分類錯誤，使用者只要點擊各個筆劃就可以變換該筆劃的顏色，顏色也會依序變換成紅、綠、藍色等不同顏色，各自代表著不同的部件，藉此將該筆劃分類到不同部件。當所有筆劃的分類結果都正確時，使用者可以將提交結果。如果使用者對自己寫的某個字或部件分類結果不滿意，也可以選擇重寫該字。

到目前為止，APP 僅使用使用者寫字的筆畫順序作為分類部件的主要參考資訊。圖 4-3 為兩張 APP 的擷圖，當使用者寫「時」這個中文字，也就是 392 個 APP 指定的手寫字中的第 29 個，剛開始有兩個筆劃被分類到錯誤的部件，使用者可以輕鬆的點擊各個筆劃來更換筆劃所屬的部件。

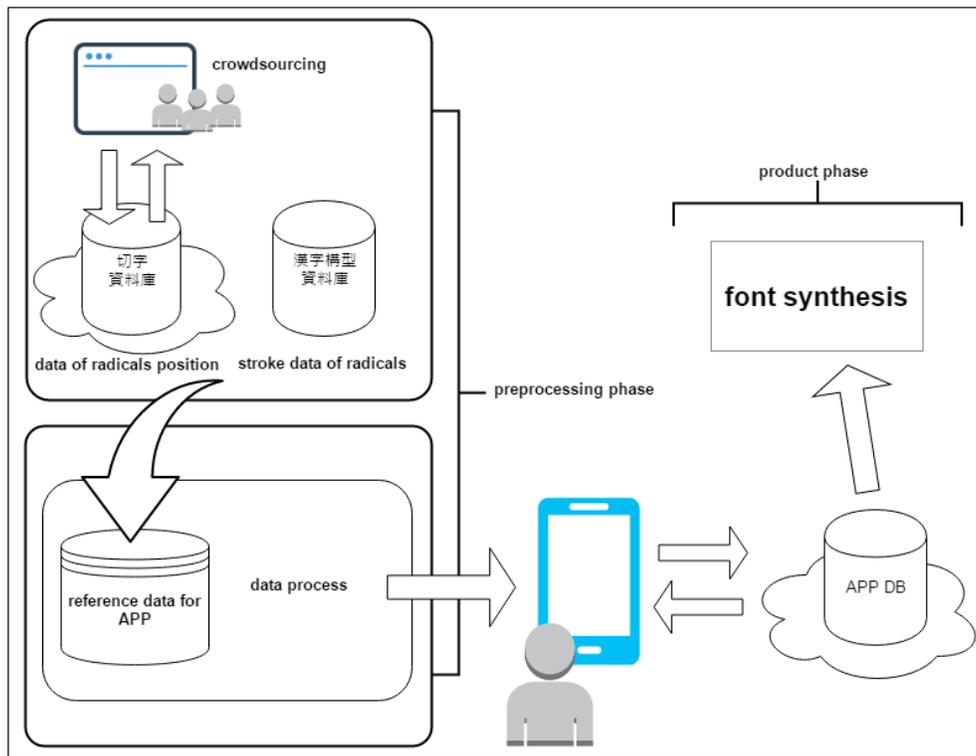


圖 4-1 實驗架構

如圖 4-1 所示，包含前置及生產流程的所有實驗步驟為：

step1 切字 & 合併外部參考資料(只需進行一次)

step2 產生手寫字結構

step3 使用者手寫

step4 產生手寫自結構

step5 字形合成

表 4-1 資料庫 schema (1)

word		
column	type	comment
id	int(5)	目標字的 id
unicode	varchar(6)	代表該字的 UTF-16 內碼，例如: 7684
expr	varbinary(20)	該字的部件組合，以 Big5 表示，例如「的」的組成部件為白 (0xa5d5) 和勺 (0xa463)
userid	varchar(20)	使用者 id
update_at	timestamp	切字記錄更新時間
rect_done	tinyint(1)	該字的部件是否都被框選完成
status	varchar(1)	該字的框選結果以及部件組成是否都正常
subword		
id	int (5)	子部件的 id
unicode	varchar (6)	代表該字的 UTF-16 內碼，例如: 7684
rank	int (1)	該部件的順序
word	varbinary (5)	該子部件的字碼，以 UTF-8 表示
posx	float	框選起點的 x 座標
posy	float	框選起點的 y 座標
w	float	部件被框選之矩形的寬
h	float	部件被框選之矩形的高
userid	varchar (20)	使用者 id
updated_at	timestamp	該部件被框選的最後新時間
history	varchar (60000)	框選部件的歷史紀錄

表 4-2 資料庫 schema (2)

hwref		
column	type	comment
word	varchar(10)	字碼
operator	int(1)	代表該字部件的連接方式。例如 0:單一個體字，1:左右，2:上下，3:內外)
rad_order	varchar(10)	部件序
root_order	varchar(16)	字根序
strokes	varchar(2)	該字的筆畫數量
unicode	varchar(6)	代表該字的 UTF-16 內碼
big5	varchar(6)	代表該字的 Big5 內碼
view_uff		
id	int(5)	
unicode	varchar(6)	代表該字的 UTF-16 內
expr	varbinary(20)	該字的部件組合，以 Big5 表示
userid	varchar(20)	使用者 id
update_at	timestamp	切字記錄更新時間
rect_done	tinyint(1)	該字的部件是否都被框選完成
status	varchar(1)	該字的框選結果以及部件組成是否都正常

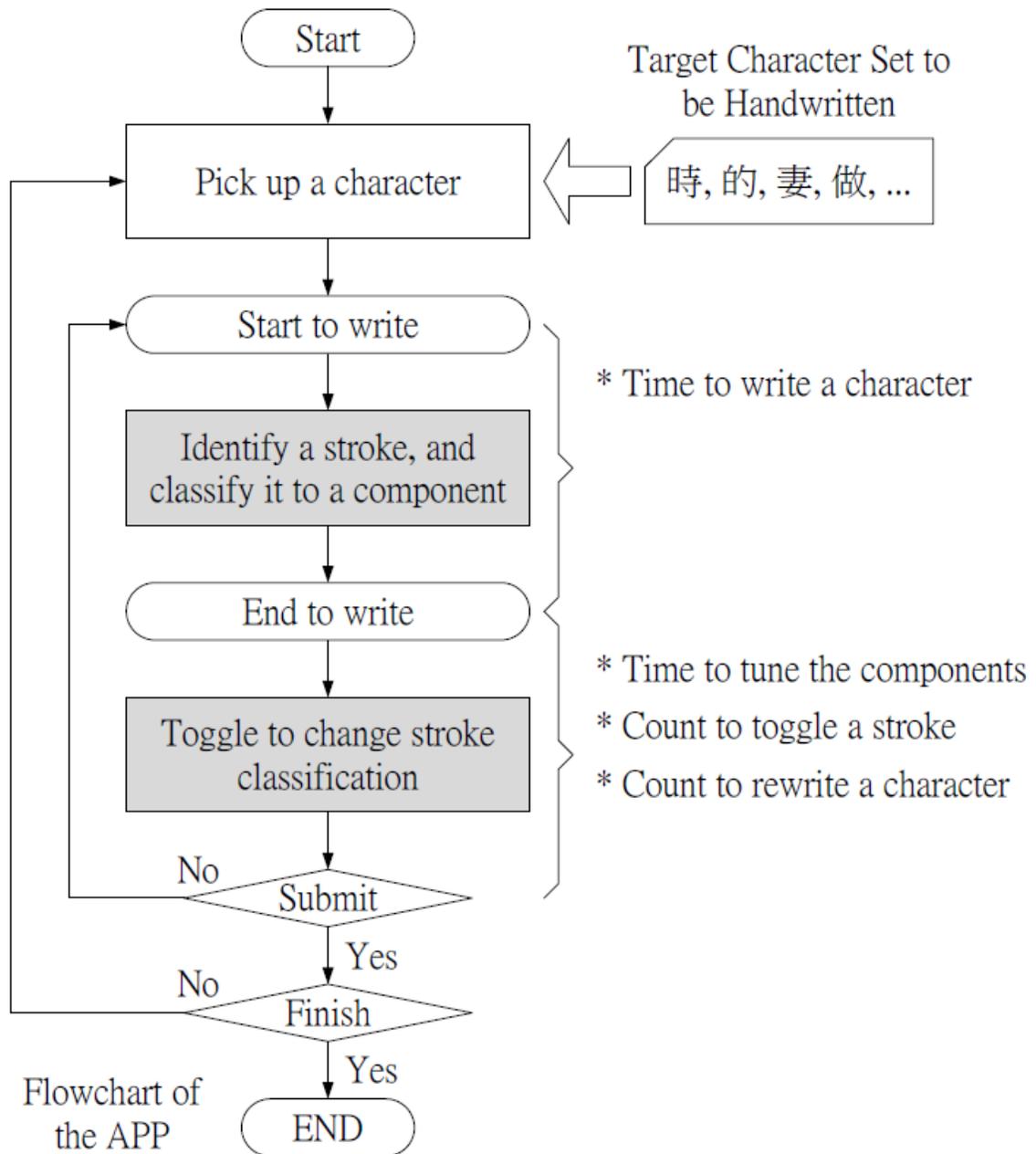


圖 4-2 APP 擷取使用者手寫字流程圖

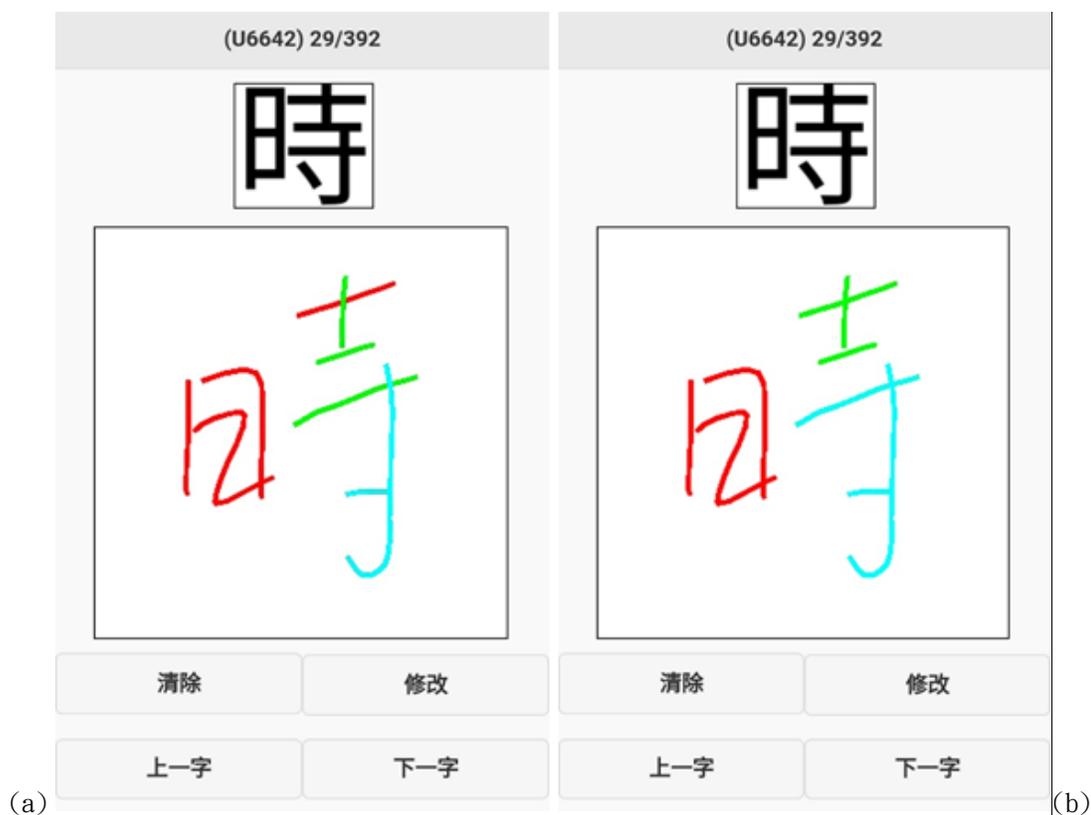


圖 4-3 APP 修改畫面擷圖

(a) 如圖 4-3 所示，一開始有兩個部件的擷取結果是錯誤的。因為使用者用 3 個筆畫寫完第一個部件「日」，但它其實有 4 畫。因此第二個部件「寸」的第一畫被分類到部件「日」(也就是紅色的部分)，相同的第三個部件的第一畫「寸」也被分類到「士」(綠色的部件)。

(b) 使用者觸碰錯誤的筆畫後，它們的顏色被循環切換成紅色、綠色或藍色(有三個部件所以有三種顏色)，最後調整成正確的顏色。

第二節 驗證 APP 的使用結果

為了驗證 APP 的表現，我們徵求了 4 位志願者，請他們透過 APP 手寫 392 個經過挑選的中文字。我們簡單統計了 APP 的使用狀況，如表 4-3 所示。平均而言，每位使用者必須寫 3,743 劃來完成 392 個字。手寫字的差異是會自然而然的發生。每人完成所有字之時間約為 2.5 天。然而，如果扣掉使用 APP 以外的時間，平均上他們只花了 54 分鐘就能完成了。這代表利用 APP 擷取使用者的手寫部件，比傳統的方法更省時。

平均而言，使用者寫一個字的時間為 9 秒。因為目前 APP 只使用筆劃順序將筆劃分類為各部件，在 392 字中大約有 94 個字，也就是大約有 24% 的字需要微調。而大家平均花在微調的時間小於 1 秒。另一個有趣的發現是，大部分使用者對自己的手寫字不滿意時，比較喜歡重寫整個字。這也表示使用 APP 會比使用紙張方便。

表 4-3 APP 使用統計結果統計表

	<i>Min</i>	<i>Average</i>	<i>Max</i>
Total no. of strokes written	3,717	3,743	3,764
Total no. of characters that have been rewritten	143	195	250
Total no. of characters that need fine tuned	70	94	140
Total time to complete this job	6 hours 43 minutes	1 day 9 hour 41 minutes	2 days 11 hours 49 minutes
Total time to handwriting all characters	-	54 minutes	-
Average time to finish a character	6 seconds	7.7 seconds	9 seconds
Average time to fine tune a character	0.4 seconds	0.5 seconds	0.6 seconds

第三節 字形合成結果

我們利用 APP 擷取使用者手寫字的部件，也就是候選部件，當作字形合成的材料。目前合成結果雖然不是非常完美，但仍可接受，如圖 4-4。



圖 4-4 合成結果

第五章 結論

在本研究，我們呈現了一個簡單又快速的方法，讓一般人只需要在手機或平板上寫下約 400 個字，就能擁有個人手寫風格的中文字型。APP 會追蹤並分類筆劃到各部件，就能從使用者的手寫字中擷取部件。一個字接著一字，APP 將能收集更多有用的手寫部件，之後就可以使用它們來合成更個人手寫風格的字。

目前 APP 仍然對使用者的手指及感壓螢幕很敏感。因此在本研究我們只考慮普通手寫的狀況下的使用情況。也就是說，手寫時，每一個筆畫順序應該要正確，起始點也必須明確，連續筆劃之間也需要空間。為了讓 APP 更好，我們未來也會使用部件位置的資訊，讓 APP 更精準的分類部件。

我們預計會針對以 Unicode 1.1，其包含 20,901 個 CJKV 編碼(中文、日文、韓文以及越南文)組成的各人風格手寫字，進行一次規模更大的實驗。

本研究仍有一些細節及項目尚未全部完成，雖有遺珠之憾，但我們未來仍會針對下列幾點進行改善及研究：

- 1.改善切字結果之修正及判斷方法。
- 2.更準確的擷取使用者的手寫字。
- 3.調整及美畫合成字，有些合成字有筆畫重疊或筆畫粗細等美化字形相關問題。
- 4.更新及改善部件資料庫的資訊。
- 5.嘗試利用手機或平版的感壓能力或透過繪圖板手寫，讓筆畫看起來更像是手寫字。
- 6.可以嘗試透過機械學習的方式，將手寫字的個人化方式以各種書體延伸變化。
- 7.我們也有收集使用者手寫字的版本資料，所以可以利用這些資料進一步研究手寫習慣或字體美化的部分。
- 8.有關於法律上的議題，例如將合成字濫用於偽造文書之可能性是值得思考的。

參考文獻

英文文獻

1. BIG5-1984, or simply BIG5, a de facto standard for Traditional Chinese character.
2. BIG5-2003, in appendix of CNS-11643 expansion, 2003.
3. Chang, Ray-I and Hong, Chian-Ya (2015), Generating Compound Chinese Character by Chinese Character Description Language for Cloud-Based Font Service System
4. Chinese Character Component Searching System of Academia Sinica. Available at <http://cdp.sinica.edu.tw/cdphanzi/>
5. CNS 11643-1992, “Chinese Standard Interchange Code,” Obsoletes CNS 11643-1986. Chinese National Standard, 1992.
6. CNS 11643-1986, “Standard Interchange Code for Generally-Used Chinese Characters,” Chinese National Standard, 1986.
7. FontCloud (FontWalker), available at <http://fontwalker.iis.sinica.edu.tw/>.
8. GB 18030-2000, “Information technology - Chinese Ideograms Coded Character Set for Information Interchange - Extension for the Basic Set.” Technical Standards Press, People's Republic of China, 2000.
9. GB 2312-80, “Code of Chinese Graphic Character Set for Information Interchange Primary Set.” Technical Standards Press, People's Republic of China, 1981.
10. International Encoded Han Character and Variants Database. Available at <http://chardb.iis.sinica.edu.tw/>.
11. ISO/IEC 10646-1:2000(E). International Standard - Information technology -- Universal Multiple-Octet Coded Character Set (UCS) - Part 1: Architecture and Basic Multilingual Plane, 2000.
12. Juang, D.-M., Wang, J.-H., Lai C.-Y., Hsieh, C.-C., Chien, L.-F. and J.-M. Ho (2005), Resolving the unencoded character problem for Chinese digital libraries, 5th ACM/IEEE Joint Conference on Digital Libraries.
13. Lim, S. B. and Kim, M. S. (1995), Oriental character font design by a structured composition of stroke elements, *Computer Aided Design*, vol. 27, no. 3, 193–207.
14. Lin, J.-W., Hong, C.-Y., Chang, R.-I, Wang, Y.-C., Lin, S.-Y., Ho, J.-M. (2015), Complete Font Generation of Chinese Characters in Personal Handwriting Style, the 34th IEEE International Performance, Computing, and Communications Conference, Nanjing, China.
15. Lin, J.-W., Lin, F.-S., Wang, Y.-C., Ho, J.-M., Chang, R.-I.(2015). FontCloud: Web Font Service for Personal Handwritten, Ancient, and Unencoded Characters, *Future Information Technology - II, Lecture Notes in Electrical Engineering*, vol. 329, pp. 113–119.
16. Lin, J.-W., Wang, C.-Y., Ting, C.-L. and Chang, R.-I (2013). Font Generation of Personal Handwritten Chinese Characters, 5th International Conference on Graphic and Image Processing, Hong Kong, China.
17. Lin, J.-W. and Lin, F.-S. (2012), Unicode Han character lookup service based on similar radicals, *International Journal of Smart Home*, vol. 6, no.3, pp. 99–106.
18. Phinney, T. W. (2004), TrueType, PostScript Type 1, & OpenType: What’s the difference, version 2.36.

19. Selinger, P. (2009), Potrace: a polygon-based tracing algorithm. Available at <http://potrace.sourceforge.net/>.
20. Shu, H. and Anderson, R. C. (1997), Role of radical awareness in the character and word acquisition of Chinese children, *Reading Research Quarterly*, vol. 32, pp. 78–89.
21. The Unicode Consortium, “The Unicode Standard.” Available at <http://www.unicode.org/unicode/standard/standard.html>.
22. Williams, G. (2003), “Font creation with FontForge,” *Proceedings of EuroTeX 2003, TUGboat*, vol. 24, pp. 531–544.
23. Wong, H. T. F. and Ip, H. H. S (2000), Virtual brush: a model-based synthesis of Chinese calligraphy, *Computers & Graphics*, vol. 24, pp. 99–113.
24. Wong, P. Y. C and Hsu, S. C. (1995), Design Chinese typeface using components, *Computer Software and Applications Conference*.
25. Xia, W. and Jin, L. (2010), A Kai style calligraphic beautification method for handwriting chinese character, *International Conference on Document Analysis and Recognition*.
26. Xu, S., Lau, F. C. M. , Cheung, W. K., Y. Pan (2005), Automatic generation of artistic chinese calligraphy, *IEEE Intelligent Systems*, vol. 20, no. 3, pp. 32–39.
27. Zhang, J., Mao, G., Lin, H., Yu, J., C. Zhou (2001), Outline font generating from images of ancient Chinese calligraphy, *Transactions on Edutainment V, Lecture Notes in Computer Science*, vol. 6530, pp. 122–131.
28. Zhang, J., Yu, J., and Lin, H. (2007), Capturing character contours from images of ancient Chinese calligraphy, the 2nd Workshop on Digital Media and its Application in Museum & Heritage.
29. Zhu, X., Jin, L. (2008), Calligraphic beautification of handwritten Chinese characters: a patternized approach to handwriting transfiguration, *Proceedings of ICFHR 2008*, pp. 135–140.
30. Zitnick, C. (2013), Handwriting beautification using token means, *SIGGRAPH*