Cyberon Voice Commander 多國語言語音命令系統

何泰軒、劉進榮
賽微科技股份有限公司
Cyberon Corporation
tai@cyberon.com.tw, AlexLiou@cyberon.com.tw

摘要

Cyberon Voice Commander (CVC) 是賽微科技自行研發的多國語言版本手機聲控軟體，應用於 Windows Mobile, WinCE, Symbian 等智慧型手機中，提供使用者語音撥號、語音指令、語音點歌、Text-to-Speech (TTS) 朗讀簡訊、電子郵件、和行事曆內容等功能。CVC 能支援 24 種語言版本，目前已有超過 30 款手機內建 CVC 銷售全球。本文描述 CVC 所使用的語音辨識和 TTS 技術、支援多國語言的作法、幾種語言的語音辨識實驗結果、以及我們對 CVC 未來改良的想法。

關鍵詞：語音辨識、語音合成、TTS、語音撥號、語音命令、多國語言語音技術

一、緒論

隨著來無線通訊、半導體技術的快速進步，以及手機製造商建立起完善的全球供應鏈大幅降低製造成本，促成了手機產業的快速發展，2006 年全球手機出貨超過 10 億台，Nokia 預估今年全球手機用戶數亦將突破 30 億，2010 年可達 40 億用戶，手機已成爲大部分現代人生活中不可或缺的裝置之一。

近年來語音聲控功能在手機上的重要性逐漸增加。小型化、攜帶方便爲多年來手機設計不變的趨勢，手機按鍵越來越少、螢幕大小有所限制，造成用手操作手機不夠方便，例如很多使用者電話簿中都有上百筆人名資料，用按鍵一筆一筆地瀏覽尋找聯絡人是非常沒有效率的，應用語音辨識技術就能很快的幫助使用者找到想查詢的聯絡人資料。另外，大部份人在開車中不可避免地會撥打手機，但這是相當危險的舉動，不僅危害自身安全，也帶給其他用路人威脅，爲避免撥打手機而造成交通事故，許多國家已經訂定嚴格的法規禁止開車時用手撥打手機，語音撥號功能就能讓使用者在開車的同時安全地撥號。自 1997 年 Philips 推出第一款語者相關 (speaker-dependent) 的聲控手機開始，手機上語音撥號、語音命令功能的需求即逐年增加，包括 Nokia、Motorola、Samsung、Sony Ericsson、LG 等前五大手機廠都開始採用此功能。2006 年底路透社 (Reuters) 粗略預估當年約有 1 億到 1.5 億台手機整合語音撥號功能，而 2007 年整合語音撥號的手機數量將成長一倍。

爲此，我們從 2002 年開始研究適合在手機平台上運行的語音技術，並開發 CVC 手機聲控軟體，以台灣手機製造商爲主要客戶。台灣手機製造商以代工爲主要營運模式，代工產品行銷全球，要打入手機製造商的供應鏈，產品和技術必須能支援多國語言，因此除了中文外，我們也開發多國語言的語音辨識與 TTS 技術，目前 CVC 可支援 24 種語言的語音辨識與 TTS，詳見表一。以 HP iPAQ 510 Voice Messenger [1][2] 爲例，即內建 13 種不同語言版本的 CVC 出貨全球。除此之外，包括 Nokia 6708、ASUS P535、

Fujitsu-Siemens Pocket Loox T830、HTC Touch 等多款暢銷手機都有內建多國語言版 CVC。

表一、CVC 支援的語言版本

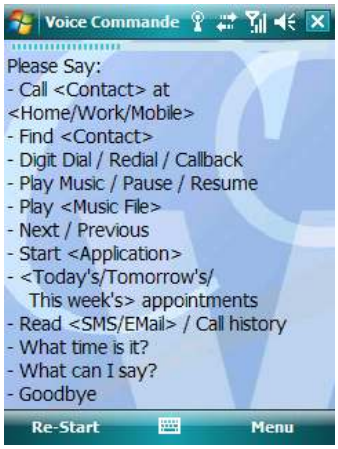
地區	語言版本
亞澳洲	台灣口音中文、大陸口音中文、粵語、韓語、日語、泰語、土耳其語、澳洲口音英語
美洲	美國口音英語、中南美口音西班牙語、巴西口音葡萄牙語
歐洲	英國口音英語、德語、法語、義大利語、西班牙語、葡萄牙語、俄羅斯語、荷蘭語、丹麥語、波蘭語、捷克語、瑞典語、希臘語

CVC 讓使用者能用語音和手機對話互動，而達成在 hand-free、eye-free 操控手機的目的。CVC 具備的功能包括語音撥號、語音指令、語音查詢聯絡人、語音點歌、語音朗讀簡訊、電子郵件、行事曆內容等。圖一爲英文版 CVC 畫面，以下是透過 CVC 進行語音撥號的範例：

CVC: 請說指令
User: 打電話給何泰軒
CVC: 打電話到何泰軒，住家、公司、手機、或取消
User: 公司
CVC: 公司，撥號中請稍候

另一個範例是使用者直接說「打電話給何泰軒公司」:

CVC: 請說指令
User: 打電話給何泰軒公司
CVC: 打電話到何泰軒公司，確認或取消
User: 確認
CVC: 撥號中請稍候



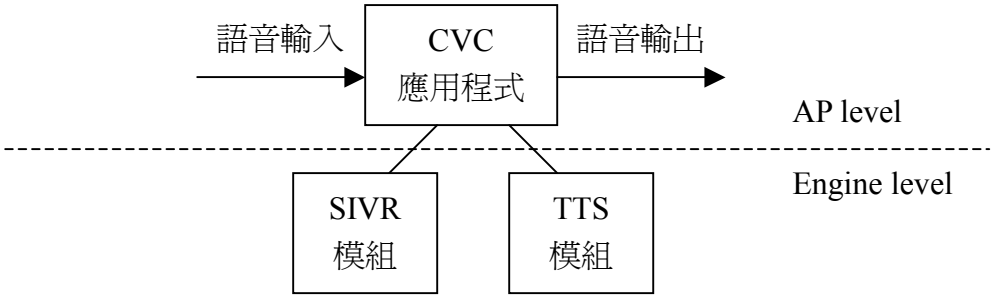
圖一、英文版 CVC 畫面

本論文將描述 CVC 的系統架構以及支援多國語言的作法，第二節將描述 CVC 系統架構，包括多國語言的語音辨識和 TTS 所使用的技術，第三節描述語音辨識實驗環境和幾個語言版本的實驗結果，第四節提出我們未來對 CVC 改良的幾個想法。

二、CVC 系統架構

CVC 包括一組應用程式，提供使用者介面、進行錄放音、控制整個和使用者對話的流程，底層爲 SIVR (speaker-independent voice recognition) 模組和 TTS 模組，如圖二所示。SIVR 模組可進行獨立詞彙辨識，或有文法限制的連續語音命令辨識，TTS 能將辨識的結果或甚至任意文字輸入轉換成 PCM 聲音，透過手機的喇叭播放放出來。由於手機資源較小，要能在上面運行，所有的運算必須改成整數運算，且使用的空間也不

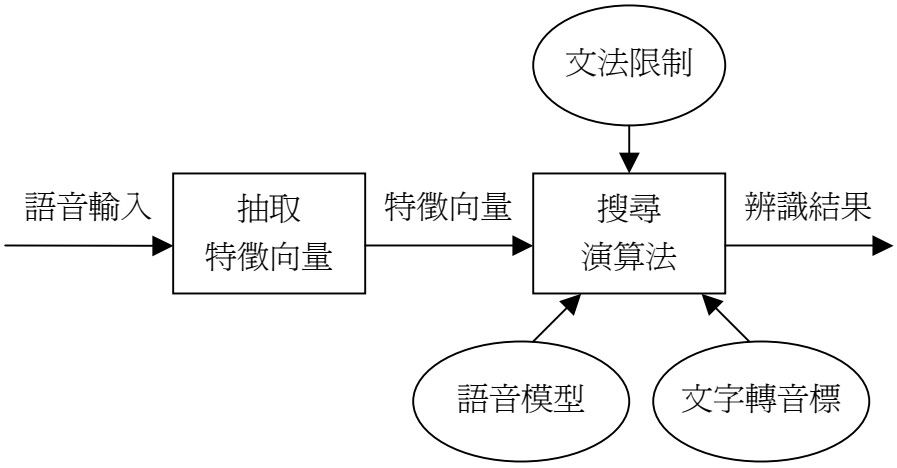
能太大，我們的 SIVR 和 TTS engine 一個語言加起來大約佔用 500KB 到 800KB，依語言不同而有所差異，RAM 大約需要 500KB。



圖二、CVC 系統架構

(一)、SIVR

SIVR 模組可分成幾個部份：一是從聲音中抽取特徵向量 (feature extraction)，二是辨識時所使用的語音模型(acoustic model)，三是將辨識的辭彙轉換成音標(word-to-phone conversion)，四是具文法限制的語音辨識搜尋演算法(search algorithm)，如圖三。



圖三、SIVR 各組成部分

1. 特徵向量

進行語音辨識時，會先將錄進來的聲音轉成特徵向量，聲音可來自手機上的麥克風、有線耳機、或藍芽耳機，由於藍芽耳機只能傳輸 8kHz, 16-bit PCM 的聲音，因此我們使用的聲音輸入爲 8kHz, 16-bit PCM。每秒中聲音抽取出 100 個特徵向量，每個特徵向量爲 16 個維度，由 8 維的 MFCC (Mel-Frequency Cepstral Coefficients) 和 8 維的 delta MFCC 所組成。MFCC 用 CMS(Cepstral Mean Subtraction)做通道效應補償。

2. 語音模型

語音模型爲傳統以音素爲單位的隱藏式馬可夫模型(Phoneme-based Hidden Markov Model)，每個模型由 3 個由左到右的狀態(state)組成，我們使用三聯音素模型(Triphone)加強對連音的辨識。相較於 Triphone 數量，我們所收集的訓練語料相對不足，此時可用決策樹(decision tree) [3][4]來決定哪些類似的狀態(state)可共用參數和訓練語料，我們可調整共用的程度來控制參數量，一方面避免某些參數的訓練語料不足，一方面可依手機的資源調整出最佳的模型大小。

訓練語音模型使用 forward-backward 演算法 [5]。每個語言我們蒐集 100 到 800 人不等的聲音，每人念 200 到 300 個句子，大約 25 到 30 分鐘的語料，每個人唸的文稿皆不同，另外再唸 40 到 60 個單詞，這些單詞涵蓋該語言所有的音標，用來初始化語音模型 (boot model)。初始化模型的步驟如下：我們先取約 10%的單詞語料，用人工標出每個組成音標的邊界，並訓練 CI model (context-independent model)，再用此 CI model 自動標出所有單詞語料的音標邊界，並以人工重新校正調整，之後再拿校正後的語料重新訓練一次 CI model。我們的經驗是初始化模型的品質對最後語音模型的好壞有很大的影響，因此在這個階段投入較多的人力做語料的處理。訓練完成 CI model 後，再將其逐步擴充訓練成 RCD model (right-context-dependent model)和 Triphone model，最後再進行共用參數的調整，依手機平台的計算資源和客戶的需求，調整出最佳大小的語音模型。

3. 文字轉音標

辨識時須將欲辨識的文字轉成音標，再由語音模型中取出對應的 Phoneme HMM，串接組成以詞爲單位模型(word model)，辨識時將輸入的語音特徵向量和 word model 比對計算相匹配的機率值。

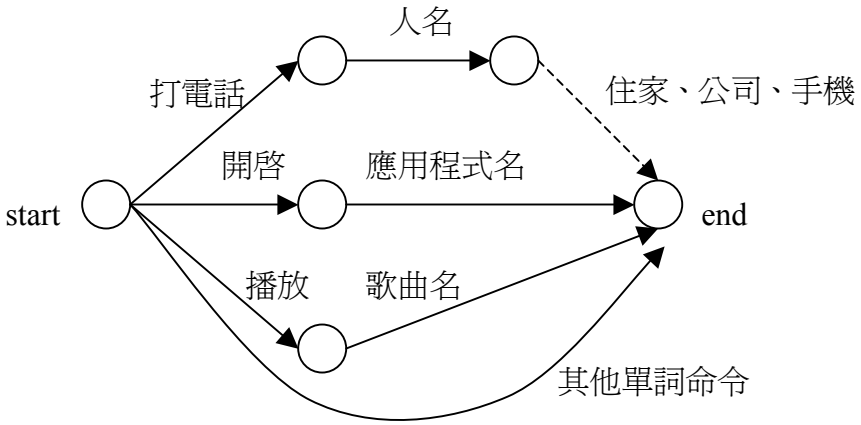
表二、CVC 各語言版本所使用的文字轉音標方法

方法	語言
發音規則加例外發音對應表	義大利語、西班牙語、葡萄牙語、捷克語、土耳其語、韓語、日語假名, 俄羅斯語、希臘語、泰語
文字音標對應表	中文、日語漢字、韓語漢字
決策樹演算法加例外發音對應表	英語、法語、德語、荷蘭語、丹麥語、瑞典語

依語言的不同我們使用三種文字轉音標的方法，第一種是有固定發音規則的語言，從文字本身即可對應出音標，例如西班牙文和義大利文，但有些外來語會出現例外的狀況，此時可建立一個例外發音的對應表來解決這個問題。第二種是完全沒有發音規則的語言，例如中文或日文的漢字，只能用一個對應表來儲存每個文字的發音。第三種是可從文字中大略猜出發音，發音有某種程度的規則性但不夠明確，例如英文，這類語言的發音難以用幾條明確的規則表達出來，我們則使用決策樹演算法 [6]來建立主要發音規則，同樣地，對一些例外的發音也可以用對應表來解決。表二是 CVC 現在所支援的語言所使用的文字轉音標方法。

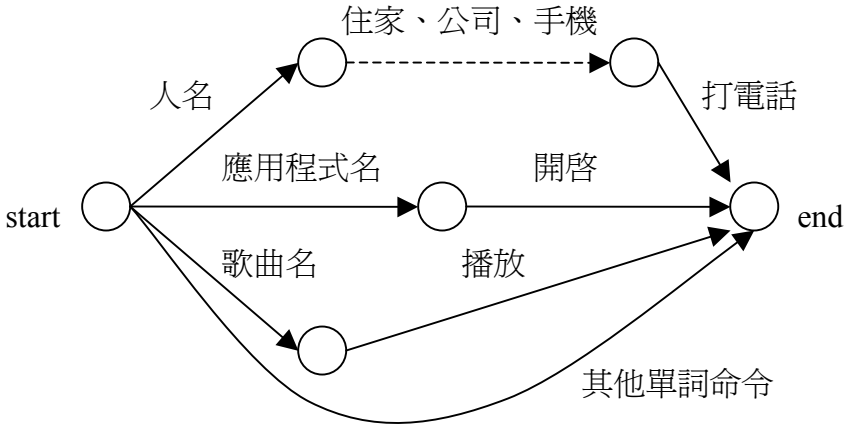
4. 搜尋演算法及文法限制

辨識使用的搜尋演算法是 Viterbi Search。SIVR 可進行獨立詞彙辨識，或有文法限制的連續語音命令辨識，以語音撥號、語音命令而言，輸入的連續語音命令通常是有規則的，例如「打電話給何泰軒住家」是由打電話(動作)、何泰軒(人名)、和住家(位址)所組成，「開啓 Windows Messenger」是由開啓(動作)、Windows Messenger(應用程式)所組成。以 CVC 而言，其語音命令的文法限制如圖四所示，其中虛線上的詞彙爲非必要的，使用者講「打電話給何泰軒」也是合法的語句。在辨識的過程中，Viterbi search 進行 word model 之間的狀態轉移(state transition)會參考此文法結構，只允許符合文法結構的狀態轉移，如此可降低連續語音辨識的搜尋複雜度，降低計算量，同時提高辨識的準確率。



圖四、CVC 連續語音命令的文法

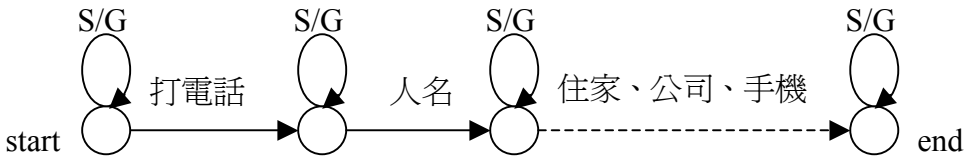
並不是所有語言版本都是使用圖四的文法結構，有些語言是後置動詞的，例如 SOV 語序的日文、韓文和土耳其文。以「打電話給何泰軒住家」爲例，韓文的念法爲「하태헌의 집으로 전화걸기」，對應到中文念法是「何泰軒（하태헌의）住家（집으로）打電話（전화걸기）」。對這類後置動詞的語言，我們使用如圖五所示的文法。



圖五、後置動詞語言版本 CVC 使用的文法

在 word model 之間的狀態轉移時，SIVR 允許先轉移至靜音模型(silence model)

或垃圾模型(garbage model)，圖六描述包含靜音和垃圾模型的打電話命令的文法，其中 S 代表靜音模型，G 代表垃圾模型。靜音和垃圾模型皆是 1 個狀態的 HMM，使用這兩個模型可以讓使用者的輸入語音中包含一些贅詞或稍微停頓不講話，例如使用者可以說「請幫我電話給…嗯…何泰軒他的…住家的…電話」，其中的「請幫我」、「給…嗯…」、「他的…」、和「的…電話」等不屬於辨識詞彙的語音有時可以被 S/G 過濾掉。不過我們實際使用的經驗發現，過濾贅詞的效果只有在辨識的詞彙量少的時候會比較好，如果某個人名和「給…嗯…」念起來的聲音很類似，就很容易發生誤判的情形。

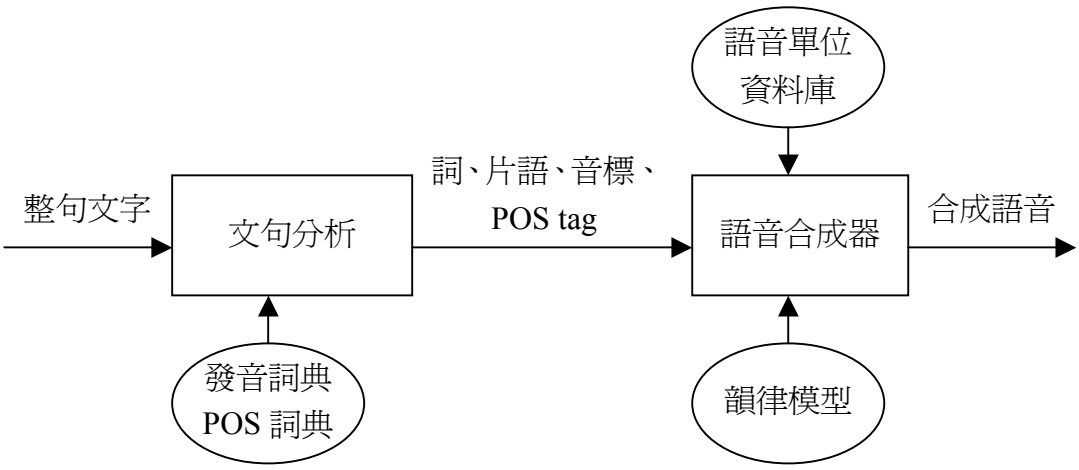


圖六、包含靜音和垃圾模型的打電話命令的文法

使用者有時會忘記語音指令的確切說法，要解決這個問題，我們可以爲固定的語音命令加入其他常用的說法，例如「打電話」也可以說成「撥電話」和「打給」，「開啓」可以說成「打開」和「執行」，把這些不同說法加入辨識的詞彙，可以讓使用者覺得產品更容易使用、更聰明、表現更穩定。

(二)、TTS

由於 CVC 主要應用的平台爲手機，相對計算量、記憶體比較小，因此我們必須選擇使用較精簡的 TTS 技術。在中文和粵語我們以音節(syllable)爲發音單位，只存放聲調爲一聲的音節，其他語言則以 diphone 爲單位。我們的聲音資料庫存的是 16kHz, 16-bit 的聲音，透過壓縮，一個語言的 TTS 大小可降低到 300KB 到 600KB，依語言不同而有所差異。我們的語音合成方法須對聲音做不少的處理，因此合成出來的聲音失真也比較嚴重，一般使用者對 CVC 語音合成品質的評價是「機械音較重，不過還是能聽得懂。」



圖七、TTS 各組成部分

TTS 的架構如圖七所示，整句文字輸入後先進行文句分析處理：對整句文字進行斷句(phrasing)，同時找出對應的音標和每個詞的 POS (part-of-speech) tag，有些語言還需要先進行斷詞處理。根據文句分析結果，韻律模型(prosodic model)建立每個片語的韻律參數，語音合成器(speech synthesizer)則一次合成一個片語的語音：由語音合成單位資料庫(speech unit database)取得片語中所需的合成單元語音資料，並參考韻律模型提供的資訊，調整片語的韻律(prosody)輸出合成語音。以下描述 TTS 的各個組成部分運作過程。

1. 文句分析

整句文字輸入後，我們須先將每個詞標註 POS，由於在詞典中每個詞可能有多種 POS，我們使用 Viterbi search 和 POS n-gram 找出機率最高的 POS tag 組合，POS n-gram 是事先由人工標示過 POS tag 的大量文字資料中訓練而得，再由此標註出的 POS tag 找出最可能的片語邊界(phrase boundary)，找尋的方法，同樣地也是使用 Viterbi search 和 boundary n-gram 找出機率最高的 boundary 位置 [7]，boundary n-gram 也是事先由人工標示過 POS 和 boundary tag 的大量文字資料中訓練而得。但有些語言我們的字典缺乏 POS，也缺乏標註好 POS tag 的文字資料，這類語言我們使用簡單的規則做斷句：先以標點符號斷句，若句子仍太長則設定片語音節數的最大值，超過最大值即強迫斷句，但斷句的位置必須落在詞邊界(word boundary)上，不能把一個詞分開放在兩個片語中。在歐洲語系詞和詞之間會用空格分開，沒有斷詞的問題，但亞洲語言如中文、泰文就必須先做斷詞，找出詞邊界，泰文一般不使用標點符號，斷詞在此處更爲重要。這種簡單規則斷句的品質並不好，不過在片語間的靜音停頓不要太長的情況下，合成語音聽起來還不至於太突兀。

2. 韻律模型

在韻律模型方面，中文和粵語都是具有聲調(tone)的語言，中文包含輕聲共有五種聲調，粵語則有九種聲調，我們用人工設計出各種聲調的 F0 形狀，透過合成器改變音節的 F0 形狀，即可合成出其他聲調的音節。除此之外，我們讓整個片語的 F0 由高到低變化，遇到有聲調的音節時將該聲調的 F0 形狀加入片語的 F0 中，組合出整個片語的 F0 變化。另一個韻律訊息是音節的長度變化，在中文和粵語我們使用固定的規則，根據音節在片語和詞內位置的變化給予不同的權重(weight)。

其他語言沒有聲調，但有重音的訊息，我們使用 CART (Classification and Regression Trees) [8]演算法預測片語中重音的位置。我們找一位語者唸大約 1 到 1.5 小時的語料，用人工標示出片語邊界、詞邊界、音節邊界、音標邊界、重音位置、重音種類、以及每個詞的 POS，從中取出若干的特徵參數，建立預測音節重音的 CART。常用的特徵參數包括：該音節在片語內的位置、在詞內的位置、該音節前後若干個音節的類別、該音節所在的詞的 POS 等，在不同的語言這些特徵參數對韻律的影響也不同，因此每種語言會使用適合該語言的特徵參數。合成時我們從片語的音標和 POS tag 中取得每個音節的特徵參數，由 CART 依據這些特徵預測哪個音節有重音、以及其重音種類，配合線性回歸法(linear regression)去訓練出預測整段片語的 F0 模型，即可組合出片語的 F0 變化，

我們也將 CART 應用在預測片語中每個語音單元的長度上，我們用 VR 的語音模型將所有訓練語料自動切出每個音標邊界，從中取出特徵參數建立 CART，常用的參數和預測重音用的 CART 類似，包括該音標在片語內的位置、在詞內的位置、在音節內的位置、該音節前後若干個音標爲何、該音標的重音種類、以及該音標所在的詞的 POS 等等。

預測重音的 CART 需要大量人力進行人工標註，目前由於人力不足，只有先針對英文和韓文建立，其他非中、粵、英、韓的語言則是將音標對應成英文的音標，用英文的 CART 預測該語言的重音位置和形狀，但每種語言會使用不同的片語 F0 變化規則，也有該語言的自己的音標長度 CART。這種方法合成的語音聽起來還可以被接受，不過韻律聽起來怪怪的，以德文爲例，德國的客戶曾提過 CVC 的 TTS 聽起來「像外國人在說德文」。

3. 語音合成器

語音合成器是基於線性預測編碼(LPC, Linear Predictive Coding)。每個語言我們找一位以該語言爲母語的女性語者進行錄音，從所錄製的語音中切割出所需要的語音合成單元(speech unit)，同時決定出每個語音單位的 pitch 位置。我們對每個 pitch 做 LPC 分析，再對 LPC 係數和 residual 壓縮儲存於資料庫中。合成時韻律模型(prosody model)會預測出整個片語的 F0 形狀和每個語音單位的長度(duration)，合成器可透過改變 LPC residual 的長度調整 F0，和透過增加或減少合成的 pitch 數目調整語音的長度。

雖然我們選擇使用精簡的 TTS 方法，不過我們的經驗發現，合成語音的品質和語音資料好壞、人工處理的品質有很大的關係，包括錄音者聲音的特性、所錄的聲音是否有瑕疵、音標邊界和 pitch 位置是否切得準確等。如果能仔細檢查資料和人工切音的結果，也是能合成出品質不錯的語音。例如我們的英語語音合成只佔用不到 400KB 的大小，但其合成英文單詞和 4 個詞以內組成的片語品質就還不錯，我們將其應用於手機英漢字典的英文發音功能中，於 2006 年下半年推出產品「賽微隨身典」，得到相當不錯的評價，終端使用者一般認爲已經達到接近真人發音的水準。

三、多國語言語音辨識實驗

本論文只有對語音辨識進行比較科學性的實驗。語音合成的驗證方法目前仍不夠嚴謹，驗證方法主要是找幾位以該語言爲母語的聽者試聽，以聽者主觀認定聽得懂爲合格的標準，目前 CVC 所支援的語言版本仍有幾個語言無法達到這個標準，細節則不在本論文中討論。

語音辨識的實驗分成兩種，第一種是模擬實驗(simulation)，我們收集以該語言爲母語的語者的聲音作爲測試語料，訓練語料中不包含測試者的聲音，我們以這些測試語料驗證我們系統的辨識率，並在開發過程中做爲改進技術的依據。測試語料爲人的名字，包含姓(last name)和名(first name)，每種語言至少收集 4 到 6 人的聲音，男女各半，測試者須把每個測試人名唸 2 次，以國內宏達國際電子公司(HTC)所生產的 Universal

PocketPC Phone 錄音，在安靜的辦公室環境中錄製。我們實驗的辨識詞彙量爲 200 個人名，由於手機聲控主要在開車時使用，我們把測試語料加入 AURORA 汽車噪音，進行 S/N 爲 15dB 到 0dB 的噪音實驗。有時候我們爲了配合客戶出貨時程而趕工，以致有些語言的實驗並不完整，在此我們選擇列出 13 種實驗做得較完整的語言，辨識率結果列於表三。

表三、各語言於 AURORA 汽車噪音下模擬實驗的準確率(%)

S/N / 語言	安靜	15dB	10dB	5dB	0dB
台灣口音中文	98.03	97.04	96.37	93.09	75.33
大陸口音中文	96.62	96.21	95.21	90.33	71.67
粵語	95.36	94.01	93.97	88.01	71.62
美國口音英語	98.90	97.90	96.68	92.58	79.40
英國口音英語	93.88	94.85	94.21	91.45	77.79
德語	95.17	95.17	93.65	87.81	75.29
法語	94.83	95.02	94.08	90.25	76.62
義大利語	95.77	94.15	93.64	91.56	81.73
西班牙語	96.18	95.37	92.83	89.28	78.00
巴西口音葡萄牙語	96.20	97.15	95.49	93.35	80.29
荷蘭語	94.25	93.12	92.62	88.12	74.75
日語	96.55	96.10	92.40	90.40	81.10
俄語	97.15	95.60	93.62	87.07	75.47
平均	96.07	95.51	94.21	90.25	76.85

在安靜環境下，各個語言基本上都可達到 95%以上的正確率，同時我們的辨識核心也有相當的穩健性，S/N 爲 10 dB 以內的噪音只會稍微降低辨識效果 2%左右。隨著噪音程度加強，辨識率隨之呈現較高的下降趨勢，在 S/N 爲 5 dB 時辨識率維持在 90%左右，0 dB 時平均辨識率則下降到約 76%，此時各個語言也呈現出較大的差異，我們推測可能和測試人名的長度有關(中文人名爲 3 個音節，其他語言人名大多高於 4 個音節)。

另一種是實地測試(field test)，CVC 每種語言版本出貨前，我們會找以該語言爲母語的測試者來實際使用，我們從旁觀察使用狀況，並記錄準確率。每個語言我們找 4 到 6 名測試者，男女各半。測試的環境爲辦公室內、馬路邊(北二高新店交流道旁新店市中興路上)、以及行進在高速公路上窗戶關閉的汽車內，汽車平均時速爲 90 到 100 公里，測試車輛是 2000 年生產的 Nissan Sentra 1.6。測試手機包括 HTC、ASUS、BenQ 所生產的 PocketPC Phone，手機的設定爲電話簿中有 200 個人名，手機系統內大約有預設 20 到 30 個應用程式，每個測試者在一種測試環境下唸 100 句的連續語音測試語句，包括下列 4 種語音命令：

1. 打電話給 <人名>
2. 打電話給 <人名> <住家、公司、手機>
3. 查詢 <人名>
4. 開啓 <應用程式>

測試結果列於表四。

表四、各語言實地測試的準確率(%)

場所 / 語言	辦公室	馬路邊	行進汽車內
台灣口音中文	98.6	92.8	93.5
大陸口音中文	96.2	90.4	92.3
粵語	94.8	89.7	91.5
美國口音英語	93.7	85.2	90.5
英國口音英語	93.2	83.7	88.5
德語	95.7	86.3	93.8
法語	96.5	91.4	92.6
義大利語	97.5	92.3	94.0
西班牙語	97.1	89.4	91.2
巴西口音葡萄牙語	95.3	87.6	88.7
荷蘭語	92.4	84.0	91.3
日語	96.2	88.3	91.2
俄語	96.3	88.4	92.8
平均	95.63	88.42	91.68

雖然和模擬測試的指令語法略爲不同，實地測試得到的結果和模擬測試相近，辦公室環境下有 95%的正確率，汽車內的辨識率 91%則介於模擬測試 S/N 爲 10 dB 和 5 dB 之間。馬路邊的噪音種類較多且較不穩定，因此辨識效果也略低一些，約爲 88%。實地測試使用的各款手機錄音頻率響應和通道效應或許不盡相同，但只要錄音音量控制得宜且無特殊的通道雜訊，都可得到相似的辨識結果。

四、結論與改良 CVC 的想法

本論文介紹 CVC 商用嵌入式語音命令系統，CVC 是國內唯一、也是全球少數能支援多國語音辨識與 TTS 的手機聲控軟體，本論文特別描述了 CVC 多國語言版本的開發經驗，和一些語言的語音辨識實驗結果。以手機聲控軟體而言，CVC 的辨識率已經能讓全球大部分語言的使用者所接受，特別是讓使用者在開車的時候，能更安全地用語音操作手機，保障自己和他人的生命財產安全。

我們於 2002 年中開始 CVC 產品的研發，於 2004 年初產品上市開始銷售，經過多年不斷地改良技術，語音辨識準確率和 TTS 發音品質才逐漸爲使用者所接受，並獲得國內外各大手機製造公司採用，內建於其手機產品中行銷全球。雖然如此，CVC 還是有很大的改進空間，在全球市場競爭仍然十分激烈的今天，我們絕對不能怠慢，以下我們提出一些 CVC 可以改善的地方：

1. 我們在製作某些語言版本時遭遇一些困難，例如阿拉伯文因爲文化的關係，很難找到女性的語者來錄音，有些語言我們收集的資料也明顯不足。我們未來將加強這些語料的收集，提升這些語言的語音辨識率。

2. 有時爲了配合客戶出貨而趕工，以致有些語言的實驗做得不夠嚴謹，可能造成消費者抱怨辨識率不好的風險，帶給客戶困擾。我們正積極招募更多具語音相關研究背景的人才，解決目前人力不足的問題，並爲每個語言建立更嚴謹更完善的實驗流程。

3. TTS 的品質還有很大的改進空間，目前有三個改進的方向，一是收集更多語言的訓練資料，爲每個語言建立韻律模型，二是使用更多的語音單位和 unit selection 的方法，減少對語音資料的調整，降低合成語音失真的程度，三是建立嚴謹的驗證方法和流程，確保產品的品質。

4. 我們發現很多使用者剛開始對產品不熟悉，唸出錯誤的語音指令而誤以爲 CVC 聽不懂他們說的話，幾次辨識失敗後產生的挫折感讓他們不願再使用。要讓產品更容易使用，我們須提升 CVC 語音辨識模組的容錯能力，甚至從使用者不完整的語音命令中，了解使用者可能的意圖，再透過對話的過程解決使用者不會操作的問題。

5. Barge-in，此功能可讓熟悉 CVC 的使用者直接切斷冗長的系統語音提示，讓 CVC 的操作更有效率。

參考文獻

[1] *HP iPAQ 510 Voice Messenger series - overview and features*. Available: http://h10010.www1.hp.com/wwpc/us/en/sm/WF05a/215348-215348-64929-314903-3352590-3360087.html

[2] Bonnie Cha, *HP iPaq 510 Voice Messenger*, CNET Asia Review, February 2007. Available: http://asia.cnet.com/reviews/mobilephones/0,39051199,40151061p,00.htm

[3] Hwang, M. Y., *Subphonetic Acoustic Modeling for Speaker-Independent Continuous Speech Recognition*, PhD thesis, CMU-CS-92-230, Carnegie Mellon University, 1993

[4] Odell, J.J., *The Use of Context in Large Vocabulary Speech Recognition*, PhD thesis, Cambridge University, 1995

[5] Rabiner, L. and Juang, B., *Fundamentals of Speech Recognition*. Prentice-Hall, 1993.

[6] Black, A., Lenzo, K., and Pagel, V., “Issues in building general letter to sound rules”, *ESCA Workshop on Speech Synthesis*, Jenolan Caves, Australia. 1998.

[7] Taylor, P., and Black, A. “Assigning Phrase Breaks from Part of Speech Sequences”, *Computer Speech and Language*. Vol. 12, pp. 99-117, 1998.

[8] Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, P. J., *Classification and Regression Trees.* The Wadsworth Statistics/Probability Series, Wadsworth and Brooks, 1984.