國　立　交　通　大　學

資訊工程學系

碩士論文

語音辨識中語者調適方法之研究

The Study of Speaker Adaptation for Speech Recognition

研 究 生：　謝 宗 儒

指導教授：傅 心 家　教授

中華民國九十四年七月

語音辨識中語者調適方法之研究

The Study of Speaker Adaptation for Speech Recognition

研 究 生：謝宗儒　　Student：Zong-Ru Hsieh

指導教授：傅心家　　Advisor：Hsin-Chia Fu

國 立 交 通 大 學

資 訊 工 程 系

碩 士 論 文

A Thesis

Submitted to Department of Computer Science and Information Engineering

College of Electrical Engineering and Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Computer and Information Science

July 2005

Hsinchu, Taiwan, Republic of China

中華民國九十四年七月

語音辨識中語者調適方法之研究

研究生：謝宗儒　　指導教授：傅心家 教授

國立交通大學資訊工程學系碩士班

摘　要

在隱藏式馬可夫模型為語音辨識核心的語者調適方法中，最大相似度線性迴歸(Maximum Likelihood Linear Regression)調適法是一個有效且快速的方法。為了彈性地調整迴歸分類以達到調適參數的共享，其使用迴歸分類樹架構來定義那些欲調適的隱藏式馬可夫模型的參數應屬於同一個迴歸類別，使用相同的調適參數。然而，此分類樹的類別數，需要人為經驗才能做好的決定。針對此點，我們使用了貝氏資訊基準(Bayesian Information Criterion, BIC)，提出了由上而下的二元分裂法(Top-down binary splitting)來建立迴歸分類樹，其可以自動決定類別的個數，而不需人為的介入，而經過實驗的驗證，可以看出 Top-down binary splitting 方法所決定的類別數是合適的。另外我們也提出了由下而上的二元合併法(Bottom-up binary merging)來建立迴歸分類樹，其基於 Top-down binary splitting 的結果，建立更能代表資料在空間上分佈的迴歸分類樹，而對於語音辨識的效能，也能有效的加以提升。最後我們應用所提出的語者調適改進方法，實作在手持式設備上的語音辨識系統，用以辨識使用者的語音輸入，此系統以分散式的運算架構，以實現大字彙的語音辨識系統。經過多名使用者的測試後，觀察出加入語者調適技術後的語音辨識系統，正確率及準確率達到 90.09%及 87.21%。

The Study of Speaker Adaptation for Speech Recognition

Student: Zong-Ru Hsieh Advisor: Prof. Hsin-Chia Fu

Institute of Computer Science and Information Engineering

National Chiao Tung University

Abstract

In this paper, we focus on speaker adaptation technique for speech recognition. The main method we used is Maximum Likelihood Linear Regression (MLLR). MLLR makes use of regression classes to group model parameters, so that the parameters in the same group can share the same adaptation transformation. The Regression class tree is one approach to dynamically define number of regression class, but the construction of regression class tree need to determine manually. Therefore, we use the Bayesian Information Criterion (BIC) and propose a Top-down binary splitting algorithm. This algorithm can construct a deterministic regression class tree automatically and the experiment result is reasonable. We also propose a Bottom-up binary merging algorithm to refine the regression class tree constructed by Top-down binary splitting algorithm and has an improved result. Moreover, we apply the proposed methods and implement a distributed large vocabulary speech recognition system on handheld device. The correct rate and accuracy are 90.09% and 87.21%.

誌　謝

這本論文記載了這一年來的研究心得，希望它可以提供對語音辨識領域有興趣者一點幫助。這兩年的碩士生涯以此論文作一結尾，回想起來便有許許多多的感觸湧上心頭，在 NNLAB 裡，從大學專題生成為碩士班研究生，首先要感謝指導教授傅心家老師在過去對我的指導和教誨，讓我受益良多，學習做研究的方法及態度。另外還要感謝實驗室裡博士後研究及博士班的學長們，徐永煜學長，莊舜清學長，陳岳宏學長，曾政龍學長，賴柏伸學長及鄭士賢學長在諸多地方給予的幫助和指導，尤其是鄭士賢學長，在語音辨識中的領域裡幫助我找到研究方向，猶如大海中的燈塔，讓徬徨的小船找到港口靠岸，並給予我許多專業知識上的幫助和鞭策，讓我可以順利完成碩士論文。也要感謝已經畢業的聖育，俊銘，子源學長，在我們碩一時給予許多經驗分享。而一起實作語音辨識系統的戰友揚智和宜玲，一次又一次的挑燈夜戰，最後換來大家都能順利通過口試，這都是人生難得的經驗。還要感謝實驗室的學弟們，政邦，建榮和富評，在各方面的互相加油打氣，使原本平淡的生活增添了許多樂趣。我也要感謝我的父母和二個哥哥，在背後支持我，讓我沒有後顧之憂，可以專心在學業和研究上。

要感謝的人實在太多了，那就謝天吧。

目　錄

表　目　錄

圖　目　錄

第一章　論文簡介

1.1 研究動機

隨著時代的進步，資訊化的生活時代已經來臨，資訊設備的使用不再侷限在辦公室環境中，而是隨著許多行動計算設備的發展及普及化，大大地融入人們的日常生活中，人和資訊設備之間的人機介面因互動頻繁而顯得更為重要。傳統的人機介面中，常見的有鍵盤輸入，手寫裝置，語音輸入或是觸碰式螢幕等。然而很多使用者都是不熟悉電腦系統的操作，或是不熟悉中文文字之輸入方式，如能發展好的語音辨識系統，讓資訊設備可以接受人類平常用語的輸入，不僅可以方便一般人使用，也可讓資訊產品的使用更無障礙化，更普及化，使用上更為便捷【1】。

當使用者在使用一套新的語音辨識系統時，系統是否能夠學習和適應新使用者的語音特性，以提升對使用者的辨識準確率，就是一套語音系統好壞的重要考量。其中之一種技術為透過利用新使用者少量的調適語料，將原本適用於所有使用者的「語者不特定(Speaker Independent)」語音辨識系統加以調適成該使用者專用的「語者特定(Speaker Dependent)」語音辨識系統，以貼近該使用者的語音特性，增加對該使用者的辨識準確率，這種技術即稱為語者調適。

本論文希望對語者調適之技術作研究及發展，並期能對語者調適技術加以改進，以提升語音辨識系統的準確度。

1.2 語音辨識和語者調適之先前研究

在語音辨識的部分，目前較普遍及辨識效果較好之辨識核心大多使用隱藏式馬可夫模型（Hidden Markov Model），簡稱 HMM，其將整個發音過程視為狀態轉移之機率空間，由每個狀態產生的輸出來描述整個發音過程中的語音特徵，對於語音的多變性特色，很適合利用此種模型來描述【2】。

而隱藏式馬可夫模型參數的訓練，需要收集許多的訓練語料，如要對每個使用者都收集訓練語料以訓練該使用者專用的語者特定(SD)的辨識模型，是較為費時費力的。一般都是收集並使用許多不同特性的語者語料，訓練出語者不特定(SI)的平均辨識模型供所有使用者使用，但因為訓練語料和測試語料間聲學特性的不匹配，所以這種語者不特定模型和使用者專用的語者特定模型辨識相比，其正確率都會較低落。因此如能消除訓練語料和測試語料間的不匹配情形，即可以提升辨識的正確率。而目前的技術主要可以分為特徵向量式及聲學模型式兩大類。

特徵向量式的技術中最具代表性的為「語者正規化(Speaker Normalization)」技術【3】【4】，主要著重於對語音訊號作正規化，這種在訓練模型時，對輸入的訓練語料的語音訊號，作一正規化(normalize)的動作，將各語者不同的聲學特性消除後，再訓練出一個語者不特定辨識模型。而測試語料輸入時，也作正規化的動作後再加以辨識，如此訓練語料及測試語料皆被正規化，將各使用者不同的發聲特性去除，消除了兩者間的不匹配問題。

在聲學模型方面通常稱為語者調適技術，是利用使用者的調適語料，將語者不特定模型調適成該使用者的語者特定模型，再以此語者特定模型對該使用者的測試語料作測試。目前比較主要的研究可分成三大類方式，第一類為以最大後機率估測法（Maximum a Posteriori, MAP）為基礎的語者調適方法，其基本精神在結合了事前機率（prior probability）與調適語料，以對事後機率(posteriori

probability)進行最大化的估測，調適出新的模型參數，使語者不特定模型可依調適語料的數量，逐步調整成為語者特定模型，以適應新語者，此種方式需要較大量之語料，才能調整出效能較好的語者特定模型，適合在調適語料充足，且要對模型做精細調整時使用【5】【6】。第二類為以參數轉換為基礎的調適方法，最具代表性的方法是最大相似度線性迴歸法 (Maximum Likelihood Linear Regression, MLLR)，其利用轉換矩陣來對聲學模型之參數做轉換，以適應新的使用者，並透過許多不同的機制，使參數得以共享轉換矩陣，故只需少量語料即可調適，有快速調適的功能，但在語料充足時，效能卻有飽和情況，準確度到達一定瓶頸後即不會再有顯著的增加【7】【8】【9】。第三種為以向量空間為基礎的調適方法，代表方法為特性語音調適法（Eigenvoice），其在事前利用訓練語者的聲學參數建構參數空間，經主成份分析(Principle Component Analysis, PCA)將空間中重要的參數值抽出，建構出新的較精簡的空間，在進行語者調適時，其做法為在此精簡的空間中找出適合該語者特性的位置，再以其位置座標決定語者所有的聲學參數。其特性為只需極少量語料就可調適，但是很快即到達準確度的飽和，準確度很難再向上提升【10】【11】。

1.3 研究目標

由上述的語音辨識和各種語者調適方法，對於不同場合之應用，各有優點及限制，本論文的研究著重於最大相似度線性迴歸調適法，此法會將模型參數作分類，以共用相同的轉換矩陣，針對於此，我們使用迴歸分類樹的架構，以動態的決定迴歸類別的數量，增加最大相似度線性迴歸調適法的使用彈性，並期能改良迴歸分類樹的架構建立方法，使語音辨識系統的辨識準確度能加以提升。

1.4 論文章節大要

本篇論文之結構，除本章為論文的簡介外，在第二章中將介紹語音辨識系統的核心模型，以及語者調適技術之概念和方法，以說明本論文提出之方法的基本理論和雛型。在第三章中，將介紹本論文針對語者調適技術中最大相似度線性迴歸法所提出的一些改進演算法。第四章則是介紹本論文方法的實驗部分，以驗證提出方法的可行性及效能。在第五章介紹應用本論文提出的方法實作之語音辨識系統。最後在第六章對本系統及實驗結果作一結論，也提出檢討建議及未來可行之研究方向，希望對日後的研究有所參考。

第二章　背景知識

本論文所提語音辨識系統主要之核心為隱藏式馬可夫模型(Hidden Markov Model, HMM)，此模型在語音辨識方面的應用研究中均有不錯的效能。我們首先在第一節中介紹隱藏式馬可夫模型的基本原理和架構。第二節介紹語者調適技術的概念和作用。第三節中介紹本論文所採用的最大相似度線性回歸調適法(MLLR)，以及在第四節中介紹迴歸分類樹(Regression Class Tree)的概念。

2.1 隱藏式馬可夫模型

隱藏式馬可夫模型是一個雙重隨機程序(doubly stochastic process) 【12】【13】，亦就是包含了一個隱藏的狀態變化，由狀態轉移機率(state transition probability)來決定下一時間的狀態是否留在原來狀態，或是轉移到別的狀態；也包含另一以狀態觀測機率(state observation probability)來描述在各狀態中觀察到的輸出。其可用三個參數來定義，分別為：

（1）初始狀態分佈向量

$$\pi = [\pi_1, \pi_2, ..., \pi_N] \tag{2.1}$$

（2）狀態轉移矩陣

$$A = [a_{ij}], 1 \le i, j \le N \tag{2.2}$$

（3）狀態觀測函數

$$B = [b_j(O_t)], \ 2 \le j \le N-1, 1 \le t \le T \tag{2.3}$$

其中

$$\pi_i = \Pr(S_0 = q_i) \tag{2.4}$$

$$a_{ij} = \Pr(S_{t+1} = q_j \mid S_t = q_i) \tag{2.5}$$

$$b_j(O_t) = \Pr(O_t \mid S_t = q_j) \tag{2.6}$$

S_t：在 t 時刻的狀態，O_t：在 t 時刻的觀測輸出，N為狀態的數量

連續性(Continuous)隱藏式馬可夫模型的狀態觀測函數為連續性的機率密度函數(Probability Density Function, PDF)，通常使用高斯混合分佈(Gaussian Mixture Distribution)來表示，其機率密度函數如下：

$$b_j(O_t) = \sum_{k=1}^{M} c_{jk} \mathrm{N}(o_t, \mu_{jk}, \Sigma_{jk}), 2 \le j \le N-1 \tag{2.7}$$

在式子(2.7)中 M 為高斯混合元件(Gaussian Mixture Component)的數量，c_{jk} 為在第 j 狀態中對第 k 個高斯混合元件的權重(weight)，$\mathrm{N}(\cdot,\cdot,\cdot)$ 代表為高斯混合模型(Gaussian Mixture Model, GMM)，包含平均值向量(mean vectors) μ_{jk} 和共變異數矩陣(Covariance Matrix) Σ_{jk}。如此，一個隱藏式馬可夫模型可以表示為：$\lambda = (\pi, A, B)$。若我們限制狀態轉移矩陣的型式為：

$$\begin{bmatrix} a_{11} & a_{12} & 0 & 0 & & 0 \\ 0 & a_{22} & a_{23} & 0 & & 0 \\ 0 & 0 & a_{33} & a_{34} & & 0 \\ 0 & 0 & 0 & a_{44} & .... & 0 \\ & & . & .. & & a_{N-1N} \\ 0 & 0 & 0 & 0 & 0 & a_{NN} \end{bmatrix}$$

即對於 a_{ij} 矩陣，只有當 $i = j$ 或 $i+1 = j$ 時 a_{ij} 才有數值，其餘位置為零，可限制隱藏式馬可夫模型的狀態為由左至右改變，其圖例如圖 2-1 所示。

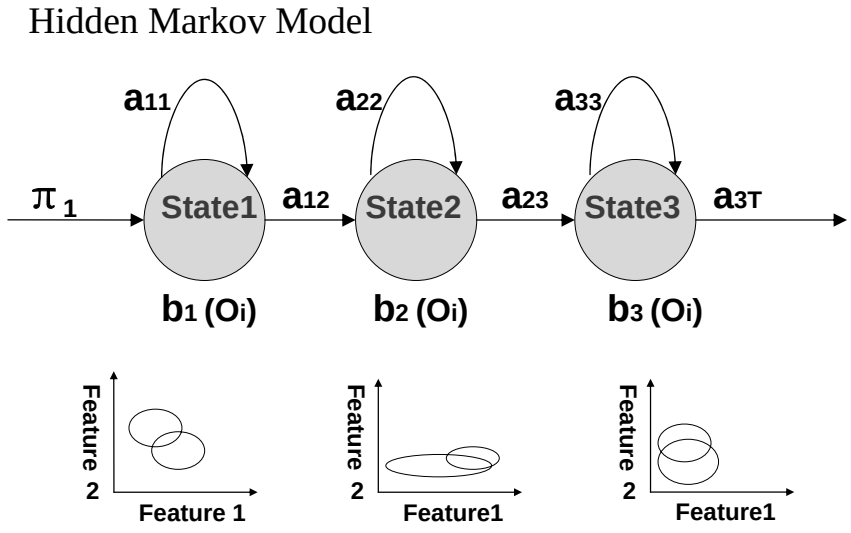

圖 2-1 限制狀態改變為由左至右的隱藏式馬可夫模型。其狀態觀測函數的分佈位在 Feature1, Feature2 組成的二維聲學空間中。

在語音辨識的應用中，可以想像聲道(vocal tract)處於有限個發聲形狀(articulatory configuration)，每個形狀以馬可夫模型中不同的狀態表示，而每個發聲形狀可發出之語音特徵參數，當成是該狀態下的觀測結果，即可將一段語音的

發音過程，看成是隱藏式馬可夫模型中狀態轉移所產生出的一串輸出。

如此，我們可以利用語料，訓練出語音基本單位(speech units)的隱藏式馬可夫模型（以本論文實作的中文語音辨識系統為例，共使用了一百五十一個語音基本單位，可參考附錄 A），而這些語音基本單位的隱藏式馬可夫模型(HMM)，可以多個加以串連組合成更巨大的隱藏式馬可夫模型(Super HMM)，來表示一段語音特徵。對於輸入的一段測試語料，只要找出最有可能產生出該段測試語料語音特徵的 Super HMM，則可得知該測試語料是由那些語音基本單位所組成，即達到語音辨識的目的。

2.2 語者調適技術

我們在將隱藏式馬可夫模型應用在語音辨識上時，首先要先利用大量語料去訓練隱藏式馬可夫模型的參數 $\lambda = (\pi, A, B)$。對於模型的建立，可分為兩種，一種為「語者特定(Speaker Dependent, SD)」模型，即專門針對某位或某群具有相同語音特性的語者所訓練與使用，另一種為「語者不特定(Speaker Independent, SI)」模型，即該模型並無針對給某位或某群具有相同語音特性的語者，而是對於所有語者都適用。這兩種類型的建立，決定在訓練語料的選擇上，如果訓練語料的選擇是只有單一特定的語者，或是只侷限在一群具有相同語音特性的語者，則訓練出來的模型對於該位或該群語者就會有最好的適合性，為其專用的 SD 模型。相反的，如果訓練語料的選擇是廣泛的收集各種不同的語者，包含各種不同的語音特性，則訓練出來的 SI 模型就會對於不同的語者都有不錯的適合性。這兩種模型在辨識率的比較上，SD 模型著重於特化性(specialization)，對於其特定的語者有最好的辨識率，而對於不在訓練語料中的語者，因訓練語料未包含其語音特性，故辨識率會很差；而 SI 模型著重在一般性(generalization)，其辨識率雖然不及 SD 對其特定語者的辨識率，但對於所有的語者都可有不錯的辨識率，對

於不在訓練語料中的語者，也可以有一定的辨識率。

這兩種類型的模型的選擇，取決於系統應用上的合適性。以本論文的針對連續中文語音的辨識系統為例，需要大量的訓練語料來訓練隱藏式馬可夫模型，如果要讓每個新使用者在使用系統前，先輸入所有訓練語料來訓練其專屬的 SD 模型，是非常耗時耗力，不切實際的，因此我們選擇了利用大量的不同語者之語料來訓練出系統所用的 SI 模型。對於 SI 模型雖然對所有使用者都可有一定的辨識率，但是其辨識準確度是不及 SD 模型的。其原因在於 SD 模型的訓練語料及測試語料的語音特性兩者間是相同的，而 SI 模型的訓練語料及測試語料的語音特性較為不匹配。而聲學模型式的語者調適技術，即是希望藉由改變 SI 模型的參數，使模型所代表的語音特性能和測試語料的語者能匹配，成為該語者的 SD 模型，以提高辨識率【14】，如圖 2-2 和圖 2-3 所示。

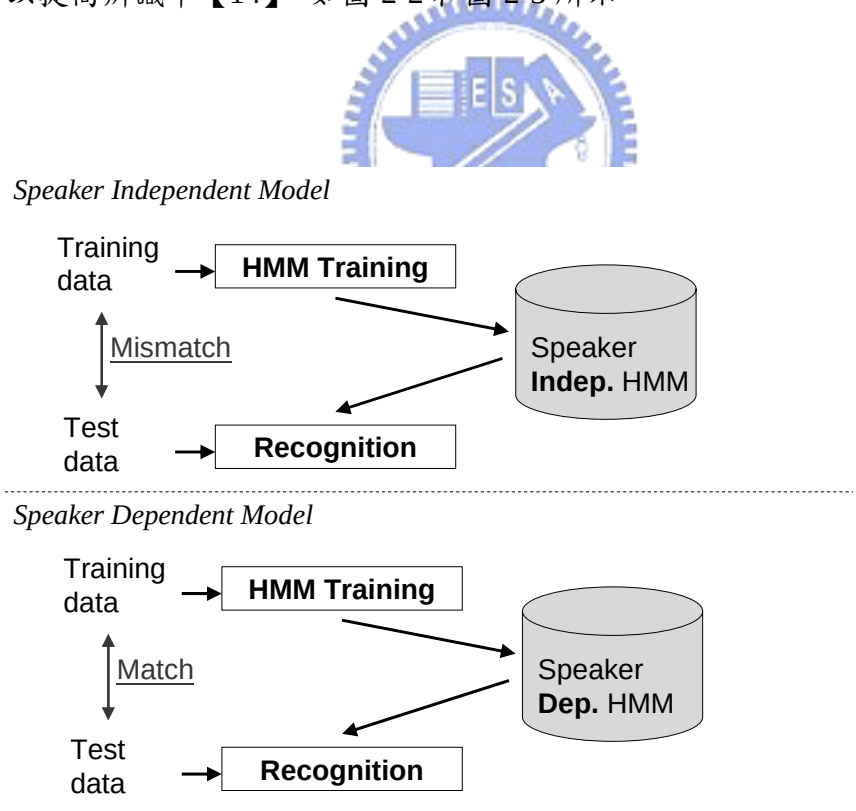

圖 2-2 使用語者不特定(Speaker independent, SI) 和語者特定(Speaker dependent, SD)模型在語音辨識上的差異。可由圖中看到語者不特定模型存有訓練語料和測試語料間語音特性不匹配的情形，而語者特定模型則否。

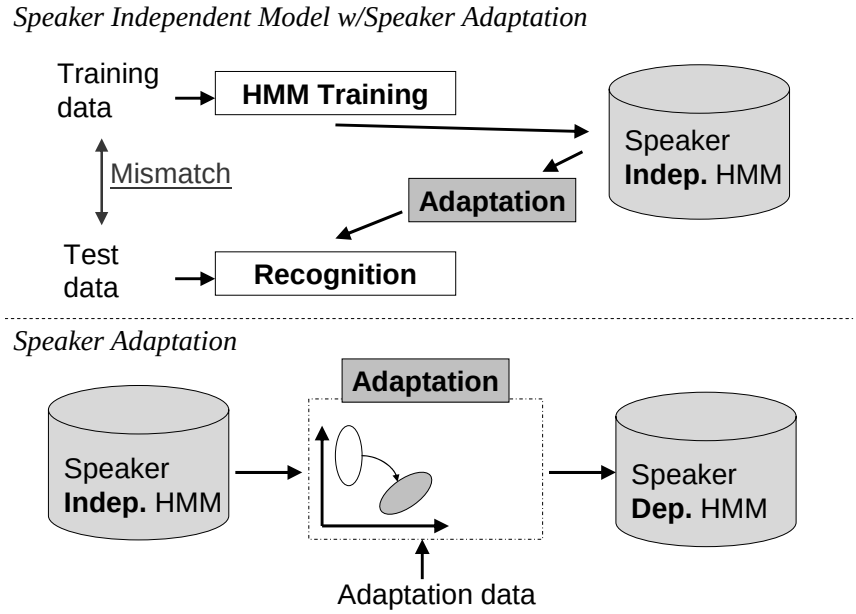

圖 2-3　圖中上方顯示了使用語者不特定(Speaker independent, SI)模型加上語者調適(Speaker Adaptation)技術後的語音辨識系統。圖中下方顯示了語者調適技術利用調適語料將原有之語者不特定模型之參數加以適整，使之成為語者特定模型，以消除訓練語料和測試語料間語音特性不匹配的問題。

如同 1.2 節中所介紹的，目前在聲學模型上，主要的語者調適技術主要有最大後機率估測法(MAP)，最大相似度線性迴歸法(MLLR)，以及特性語音(Eigenvoice)調適法。在本論文中，我們採用了最大相似度線性迴歸法作為語者調適的方法，因為其調適的速度較最大後機率估測法快速，雖比特性語音調適法慢，但是仍在系統可以接受的範圍，也不需如特性語音調適法要事前建立各類語者的 SD 模型，且飽和時的辨識效能也較特性語音調適法高【15】。

2.3 最大相似度線性迴歸調適法

本節茲對最大相似度線性迴歸法(Maximum Likelihood Linear Regression, MLLR)作一介紹。MLLR 法為基於參數轉換(transformation based)的調適方法，其主要概念在於將隱藏式馬可夫模型集合中的參數加以轉換，成為 SD 的模型，理論上應該要對模型中所有的參數都作調整，但是在實用上，調適的語料的數量是有限的，並不足夠調適所有的參數。依參考資料【16】可知，去調整狀態轉移機率或是狀態觀測函數的高斯混合元件(Gaussian Mixture Component)的權重都只有有限的效果，很少語者調適方法會轉換這些參數。較具影響性的是調整高斯混合元件的參數－共變異數矩陣(covariance matrix)及平均值向量(mean vectors)。但如果只單獨調整共變異數矩陣而保留原來的平均值向量，也沒有很好的效果，不如保留共變異數矩陣參數，調整較具代表性的平均值向量。圖 2-4 表示了在聲學特徵空間(acoustic feature space)中調整平均值向量的效果。

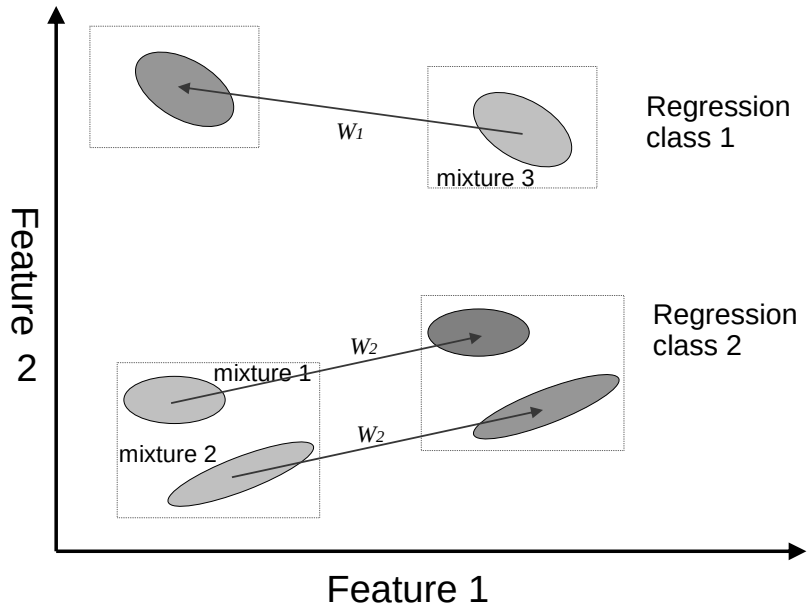

圖 2-4 此為包含了特徵 1 和特徵 2(Feature 1, Feature 2)的二維聲學特徵空間(acoustic feature space)，由三個高斯混合元件(Gaussian Mixture Components)來表示狀態觀測函數機率的分佈。在調整平均值向量(mean vectors)後，雖然因沒有調整共變異數矩陣(covariance matrix)，因此高斯混合元件本身的分佈形狀沒有改變，但是其在空間中的分佈位置可以作很大的改變。

其調整的方法為假設原有之 SI 模型和調適後的 SD 模型的參數之間，存在著一線性轉換的關係，如下列式子：

$$\hat{\mu}_s = W_s \cdot \xi_s \tag{2.8}$$

W_s 為對第 s 個高斯混合元件的轉換矩陣(Transform matrix)，$\hat{\mu}_s$ 為調適後的平均值向量，ξ_s 為第 s 個高斯混合元件的擴充平均值向量，其定義為：

$$\xi_s = [\omega, \mu_{s_1}, ..., \mu_{s_n}]' = [\omega : \mu_s]' \tag{2.9}$$

μ_s 為原來的平均值向量，n 為特徵向量(feature vectors)的維度，ω 為偏移向量(offset term)，其為一常數，可使原來之所有平均值向量加上相同偏移量，通常用來估計不同的錄音環境差異。

轉換矩陣的計算方法通常利用最大相似度(Maximum Likelihood, ML)方法求出，我們所選擇的轉換矩陣$\hat{W}_s$是以會讓經此轉換矩陣$\hat{W}_s$調適後的模型產生出調適語料的機率為最大值，以下列式子表示：

$$\hat{W}_s = \max_{W_s} P(O \mid \hat{\lambda}) \tag{2.10}$$

其中 $O = \{o_1, o_2, ..., o_T\}$，為觀測語料的特徵向量集合，$\hat{\lambda}$ 為調適後的隱藏式馬可夫模型的參數。

我們通常使用 EM(Expectation-Maximization)演算法【17】來求出式子(2.10)中的$\hat{W}_s$。在 EM 演算法中的 E-step (Expectation Step)中，我們先定義輔助函數(auxiliary function)：

$$Q(\lambda, \hat{\lambda}) = \sum_{\theta \in \Theta} P(O, \theta \mid \lambda) \cdot \log(P(O, \theta \mid \hat{\lambda})) \tag{2.11}$$

(2.11)式中θ為狀態序列，Θ為所有長度為T的狀態序列。

而在M-step (Maximization Step)中，我們利用對$\hat{W}_s$一次微分等於零的方式來求極值，以最大化輔助函數(2.11)式。而當EM演算法以迭代(iterative)的方式調整$\hat{\lambda}$，重複對輔助函數(2.11)式求最大化時，會得到以下的性質：

$$Q(\lambda,\hat{\lambda}) \geq Q(\lambda,\lambda) \Rightarrow P(O|\hat{\lambda}) \geq P(O|\lambda) \qquad (2.12)$$

從(2.12)可得到當不斷最大化輔助函數(2.11)式時，會使得$P(O|\hat{\lambda})$的數值不斷遞增，即達到(2.10)式的目的，求出使$P(O|\hat{\lambda})$最大化的$\hat{W}_s$【9】。

2.4 迴歸分類樹

如果我們要對HMM中的每一個高斯混合元件的平均值向量參數都使用一個轉換矩陣去調適的話，一般來說，會需要非常大量的調適語料才足夠估計出每個轉換矩陣，使實用性大大降低。為了克服這點，在參考資料【8】中提出了使用迴歸分類(Regression Classes)的概念。迴歸分類為數個高斯混合元件的集合，這些高斯混合元件的參數會使用相同的轉換矩陣來調適。其優點在於對某個迴歸分類的轉換矩陣，可以使用該分類中所有高斯混合元件的調適語料來估計，因此如果有些高斯混合元件的參數沒有被包含在調適語料中，它仍然可以利用該分類的轉換矩陣來轉換。因此要如何將所有高斯混合元件分類成不同的迴歸分類，就成為很重要的問題。

將高斯混合元件分類成不同的迴歸分類，主要可分為下列兩類方法：

一、 根據語音學知識(Phonetic knowledge)：使用語音學上的專業知識，決定各高斯混合元件的語音種類為何，再分到定義成不同語音種類的分類，舉例來

說，附錄 B 顯示了在參考資料【18】的中文發音分類。

二、 聲學空間距離(acoustic space distance)：以高斯混合元件在聲學空間中某種距離量測的方法來決定如何分群，距離的量測方法取決於不同的應用和技術，距離相近的高斯混合元件視為有相近的特性，而分類在同一迴歸分類中【19】【20】。

第一種方法的缺點在於需要語音學上的專業的背景知識，否則無法依語音學的特性對高斯混合元件作良好的分類，另外語音學上的知識和語言本身的特性也有很大的相關性，因此如果辨識系統為不同的語系時（如英文辨識系統和中文辨識系統），這些語音學上的背景知識也會有差別，需要跟著作更動。而第二種方法較無這些缺點，因其為「資料驅動(data-driven)」的處理方式，和語系的差別較為無關，只需將所有高斯混合元件依距離件作分群，就可以找出各個不同的迴歸分類。

一般來說，如果調適語料的數量較多，則我們會將所有的高斯混合元件分成較多的迴歸分類，以求對模型作較精密的調整，如果調適語料的數量較少，我們會減少迴歸分類的數量，以讓每個分類的轉換矩陣分配到較足夠的調適語料作估計。以最極端的例子說明，如果我們把全部的高斯混合元件都分到同一個分類，即所有參數都只透過一個轉換矩陣作調整（稱為 global adaptation），那麼只需很少量的調適語料就可以對這個轉換矩陣作有效的估計。而如果我們把每個高斯混合元件都視為一個分類，即每個參數都有自己的轉換矩陣來調整，那麼我們就需要很大量的調適語料來對每個轉換矩陣作有效的估計。因此我們需要根據調適語料的多寡來決定迴歸分類的數量。但是在實作上，我們往往是無法知道調適語料的數量的。一個系統的調適語料數量，通常是隨著使用者的使用而由少到多漸漸增加的，如果我們依初期調適語料少的特性而使用較少的迴歸分類數量，那麼到後期語料多時，也無法對模型作更精細的調整，而限制了辨識率的上升；如果我們依後期調適語料多的特性而使用較多的迴歸分類數量，則在初期語料少時，對

每個分類的轉換矩陣都無法做有效的估計，可能會使得轉換後的模型的辨識率更差。不論那一種方法，都無法適合所有的情況，因此硬性的事先設定要使用多少個迴歸分類會顯得缺乏彈性。

為了能夠更有彈性的使用迴歸分類的概念，在參考資料【8】【19】中提出了迴歸分類樹(Regression Class Tree)的架構，以樹(tree)的層級(hierarchy)概念，來動態地決定迴歸分類的數量。將所有高斯混合元件依層級觀念，建立迴類分類樹，樹根節點(root node)包含所有的高斯混合元件，使用同一個轉換矩陣，而這些同分類的高斯混合元件再依特性分成其左子樹和右子樹，分別使用各自的轉換矩陣，依此類推。

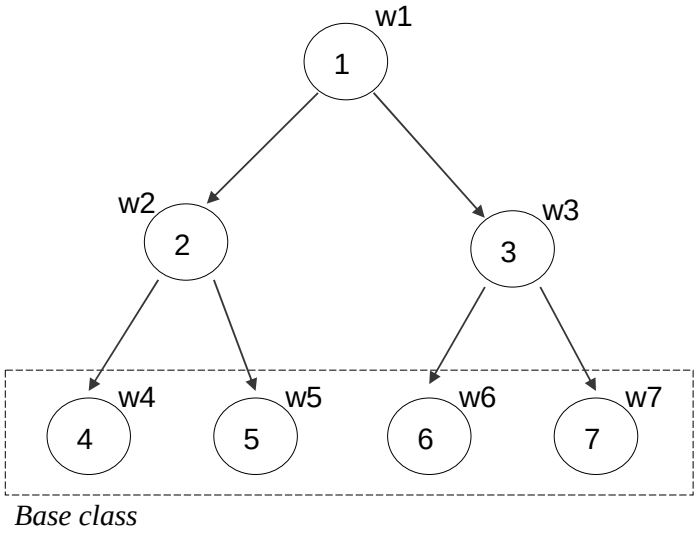

圖 2-5 迴歸分類樹，每個節點(node)都包含了一個轉換矩陣(Transform matrix)Wn，節點 1 為根(root)，節點 4~節點 7 為葉節點(leaf node)，稱為基底分類(Base class)。

圖 2-5 顯示了這樣的架構，根節點 1 包含了所有的高斯混合元件，接著細分成二群，分別屬於節點 2 和節點 3，節點 2 又分成節點 4 和節點 5，節點 3 分成

節點 6 和節點 7，每一個節點代表一個迴歸分類，分別為分類一至分類七(class1~class7)，各自以一個轉換矩陣W_n來調整，節點 4 至節點 7 為樹的葉(leaf)節點，代表分類最細的分類，稱作基底分類(base class)。當調整語料輸入時，會用來調整包含語料所屬的高斯混合元件的所有節點的轉換矩陣，舉例來說，如果輸入屬於 class4 的調適語料，則該語料會用來對 class4、class2 和 class1 的轉換矩陣作估計。而在使用時，因為轉換矩陣需要收集有足夠多的語料後，才能作有效的估計，所以我們會設定一個門檻值(threshold value)，當轉換矩陣的調適語料數量超過門檻值後才代表轉換矩陣的估計是有效的，以此法來決定要使用的轉換矩陣的層數。舉例來說，如果要使用 class4 中高斯混合元件的參數時，此參數會以轉換矩陣調適後再輸出，我們會先檢查 class4 的轉換矩陣 w4 的調適語料數量是否超過門檻值，如有表示矩陣已被有效的估計，可以用來作轉換調適模型，如果沒有，則會往上檢查其父節點 class2，如 class2 的轉換矩陣 w2 的調適語料數量超過門檻值，則會使用轉換矩陣 w2，如沒超過，則會再往上檢查其父節點 class1，依此類推。如此對於每一個高斯混合元件，我們都能依據調適語料的多寡來選擇適合的轉換矩陣作參數的調整。

第三章　建立最大相似度線性迴歸調適法的迴歸分類樹

本章介紹本論文對於最大相似度線性迴歸調適法(MLLR)中的迴歸分類樹(Regression Class Tree)之建構的一些改進方法。我們首先在第一節中介紹基本的迴歸分類樹的建構方法，為本論文提出方法之基礎。第二節中介紹以由上而下的二元分裂法(Top-down binary splitting)建立迴歸分類樹的概念，第三節中以介紹由下而上的二元合併法(Bottom-up binary merging) 建立迴歸分類樹的概念。

3.1 迴歸分類樹的建立方法

如 2.4 節所提及，我們在使用最大相似度線性迴歸調適法(MLLR)時，可以預先建立迴歸分類樹(Regression Class Tree)，以使得高斯混合元件的分類數量可以動態地選擇，在調適語料數量很少時，可以使用總體調適(global adaptation)方式使所有高斯混合元件共用同一個轉換矩陣，當調適語料增加時，也能增加轉換矩陣的數量，使每個轉換能更精確的調整，以得更佳的調適效果。

而迴歸分類樹的建立方法，可利用語音學上的知識，或是聲學空間上的距離量測，我們採取了後者，以求和語系有最大的無關性，以及避免語音學背景知識的偏差導致分類代表性不足。建立的方法為將在聲學空間上，距離較近的高斯混合元件分在同一群(clustering)，如此性質相似的高斯混合元件中的參數就可以用相似的方法被轉換。要注意到的是我們是利用原本的語者不特定模型中的高斯混

合元件的參數來建立迴歸分類樹，因此迴歸分類樹亦為和語者無關，可以適用在任何新加入的語者。

在聲學空間上建立迴歸分類樹的方法，較具代表性的為【9】【20】所利用歐基里德距離(Euclidean distance)作為量測標準的方法，一般以 Centroid splitting 演算法來建立，將所有高斯混合元件的平均值向量參數作為資料(data)點，來建立二元迴歸分類樹，其建立的演算法如下：

1. 選擇要被分裂的末端節點 P。
2. 計算 P 節點所包含所有的資料的平均值(mean)及變異數(variance)。
3. 產生兩個子節點 C1 和 C2，以父節點 P 的平均值加上及減去變異數作為兩子節點的初始平均值設定。
4. 對 P 節點中包含的每個資料點，計算其到兩個子節點 C1 和 C2 的歐氏距離，並分配到距離較近的子節點，以此法將所有資料點分成 C1 和 C2 兩群。
5. 將節點 P 中所有資料點分成 C1 和 C2 兩子節點中後，重新計算子節點 C1 和 C2 的平均值和變異數。
6. 回到第 4 步驟，再次分配節點 P 中的資料點到 C1 和 C2，直到分配情形不再變化為止。
7. 回到第 1 步驟，再選定要分裂的節點，直到所有節點的數量達到需求時才停止。

分析此演算法可以得知，我們在使用時，必須要決定我們要產生有多少節點的迴歸分類樹，如此在第 7 步驟才會達到停止的條件。假設我們最後要產生有 32 群基底分類(base class)的迴歸分類樹，那麼總共就會有 32+(32-1)=63 個節點（葉結點數量加上中間節點數量），也就是有 63 個轉換矩陣（每個節點包含了一

個轉換矩陣）來對這 63 個迴歸分類作調整。如果基底分類的數量設定的較多，會產生有更多的節點的迴歸分類樹，在語料充足時，可以對模型做更精密的調整。這個數量的決定，理論上是愈多愈好，但是如果設定過多的轉換矩陣，可能會因為調適語料的不足而使得許多的轉換矩陣都無法做到有效的估計而不會被使用到，如此會造成系統複雜度的上升和執行效能的下降。因此，這部份通常需要靠經驗和測試來做設定。

3.2 由上而下的迴歸分類樹建立方法

基於 3.1 節所介紹的 Centroid splitting 迴歸樹建立方法，需要以經驗或是多次的試驗後才能決定要使用的基底分類數目，我們認為在實際應用上並不方便，且如果基底分類的數目選擇不當，可能會使得辨識系統的準確率較為低落。因此，我們提出了一個由上而下的二元分裂法 (Top-down binary splitting) 來建立迴歸分類樹，其使用了 BIC (Bayesian Information Criterion, 貝氏資訊基準)【21】【22】，來自動決定迴歸分類的數量，產生具確定性(Deterministic)的迴歸分類樹，而不需人為判斷的介入。

BIC 可對模型的相似度(likelihood)和模型複雜度間作衡量，以決定適合用來估計資料的模型。其一般定義如下：

$$BIC(M_d) = -2 \cdot \log likelihood + (\log N) \cdot d \qquad (3.1)$$

方程式(3.1)中 M 代表模型種類，*likelihood* 為資料對此模型的相似度，N 為資料個數，d 為模型的參數個數，參數個數愈多，表示模型愈複雜。

方程式(3.1)的第一項為此模型(model)對資料(data)的相似度(likelihood)，相似度愈大表示此模型對資料分佈的描述愈好，但以最極端的例子來說，如果我們把每一個資料都用一個模型去估計(approximate)，那麼得到的 likelihood 就會是

最大了，但是這些模型複雜度會非常高，就失去了用模型來估計資料的意義。因此式子(3.1)加上了第二項的複雜度懲罰(penalty)部分，我們期望得到較為簡單合適的模型來估計資料分佈，因此對於較複雜的模型，我們施以較多的 penalty，以避免選擇到一個過於複雜的模型，而計算出的 BIC 數值愈小，就代表對這群資料 N 愈適合以模型 M 來估計。有了 BIC 這個標準後，我們定義了 ΔBIC (Delta BIC)，其方程式如下：

$$\Delta BIC(M_a, M_b) = BIC(M_b) - BIC(M_a) \tag{3.2}$$

其意義在於比較兩模型間的優劣，對於一群資料，我們可以去計算其 $\Delta BIC(M_a, M_b)$ 值，由前述 BIC 值的定義可知，BIC(M)的數值愈小，代表這群資料愈適合以模型 M 來估計，因此如果 $\Delta BIC(M_a, M_b) > 0$，就表示對此群資料，我們使用模型 a 估計會比使用模型 b 來得適合。

對於迴歸分類樹的建立上，我們利用訓練高斯混合模型(GMM)來對資料作估計和分群，高斯混合模型的機率密度函數(PDF)為：

$$f_K(x \mid \Theta) = \sum_{i=1}^{K} w_i \cdot \left[\frac{1}{(2\pi)^{\frac{d}{2}} \mid V_i \mid} \exp(-0.5(x-\mu_i)^T V_i^{-1}(x-\mu_i)) \right] \tag{3.3}$$

其中

K 為高斯混合元件的個數

μ_i 和 V_i 為第 i 個高斯混合元件的平均值向量和共變異數矩陣

w_i 為第 i 個高斯混合元件的權重

$\Theta = \{w_i, \mu_i, V_i \mid i = 1,2,...,K\}$ 為此高斯混合模型的參數集(parameter set)

給定資料 $X = \{x_1, x_2, ..., x_N\}$，所謂對 X 訓練高斯混合模型，即是去估計模型的參數集 Θ，使資料對模型有最大的相似度(likelihood)。我們利用 EM 演算法，

以迭代(iterative)的方式調整 Θ，以求得 $\log(likelihood(X \mid \Theta))$ 的最大值。

訓練完成後的高斯混合模型，我們將每筆資料，計算其和每個高斯混合元件的相似度，求出相似度最大的高斯混合元件 i，將資料標記成第 i 群，以將所有資料分成 K 群。我們使用了 K=1 和 K=2 的高斯混合模型，稱為 GMM_1 和 GMM_2，分別可以將所有資料分成 1 群和 2 群。

我們使用了 $\Delta BIC(GMM_1, GMM_2)$ 來當成我們判斷的基準。首先對於要分類的所有資料，以 GMM_1 和 GMM_2 加以估計，並計算 $BIC(GMM_1)$ 和 $BIC(GMM_2)$，$BIC(GMM_1)$ 表示資料以分成單群的模型來估計的合適度，$BIC(GMM_2)$ 則表示資料以分成兩群的模型來估計的合適度。我們再以此計算 $\Delta BIC(GMM_1, GMM_2)$，如 $\Delta BIC(GMM_1, GMM_2) > 0$，表示此群資料較適合以單群來表示，反之，如 $\Delta BIC(GMM_1, GMM_2) < 0$ 則表示此群資料較適合以分成兩群來表示。

有了這個判斷方法後，我們把隱藏式馬可夫模型中要分類的高斯混合元件的平均值向量當成資料，將原有之 Centroid splitting 的迴歸分類樹建立方法加以改良，基本的步驟如下：

1. 初始化，將所有資料點分配至同一節點 R(root)，設節點 R 為可分裂點。
2. 選擇任一可分裂點作為節點 P。如所有節點都為不可分裂節點時則停止，表示建立完成。
3. 對於 P 節點所包含所有的資料 X，分別以 GMM_1 和 GMM_2 模型來估計，並計算 $\Delta BIC(GMM_1, GMM_2)$ 值。
4. 如 $\Delta BIC(GMM_1, GMM_2) > 0$，表示資料 X 較適合以單群表示，故節點 P 不需分裂，將節點 P 設為不可分裂點，並回到步驟 2.。
5. 如 $\Delta BIC(GMM_1, GMM_2) < 0$，表示資料 X 較適合分成兩群，以 GMM_2 模

型所估計的結果對資料作分群成 X1 和 X2。

6. 產生兩個子節點 C1 和 C2，其分別包含資料 X1 和資料 X2，記錄節點 P 之子節點為 C1 和 C2，節點 P 設為不可分裂，C1 和 C2 設為可分裂節點。

7. 回到步驟 2.，直到所有節點都為不可分裂節點時停止。

由此法我們可以看到，並沒有需要人為判斷或靠經驗設定的地方，此演算法會將資料不斷分裂，直到每一群的資料都無法分裂為止，即自動決定了分群的數目，而且最後的分群結果中的每一群資料，都代表了該群資料都不適合再細分成兩群。

3.3 由下而上的迴歸分類樹建立方法

由 3.2 節中，我們提出了由上而下的二元分裂法(Top-down binary splitting)，其可以提供一個自動化的方法建立迴歸分類樹。以由上而下的二元分裂法產生的結果來看，可能會有些資料，其實在性質上接近，但是在一開始較少群數的分群時，被分裂至不同的子樹，而在分類樹的架構中分離的很遠。如圖 3-1 中的迴歸分類樹例子所示，(2-1)類和(1-1-2)類這兩個分類在空間上的分佈很接近，但是在 Iteration 1 時，這兩類的資料就被分在不同的兩個子節點(2)和(1)，而最後產生的迴歸分類樹中，(2-1)類的節點對 (1-1-2)類的節點的路徑距離為 5，反而在空間中離(2-1)類較遠的(2-2-1)類，路徑距離只有 3，類似這種情形的特殊例子，會讓迴歸分類樹中的某些節點，無法真正的表示出其在空間上的關係。這也是使用二元分裂法(binary splitting)來建立樹狀結構所無法避免的問題。因此我們提出了由下而上的二元合併法(Bottom-up binary merging)來建立迴歸分類樹，此法基於 3.1 節中介紹的由上而下的二元分裂法所產生的分類結果，加以調整，以求能建立更

具代表性的迴歸分類樹。

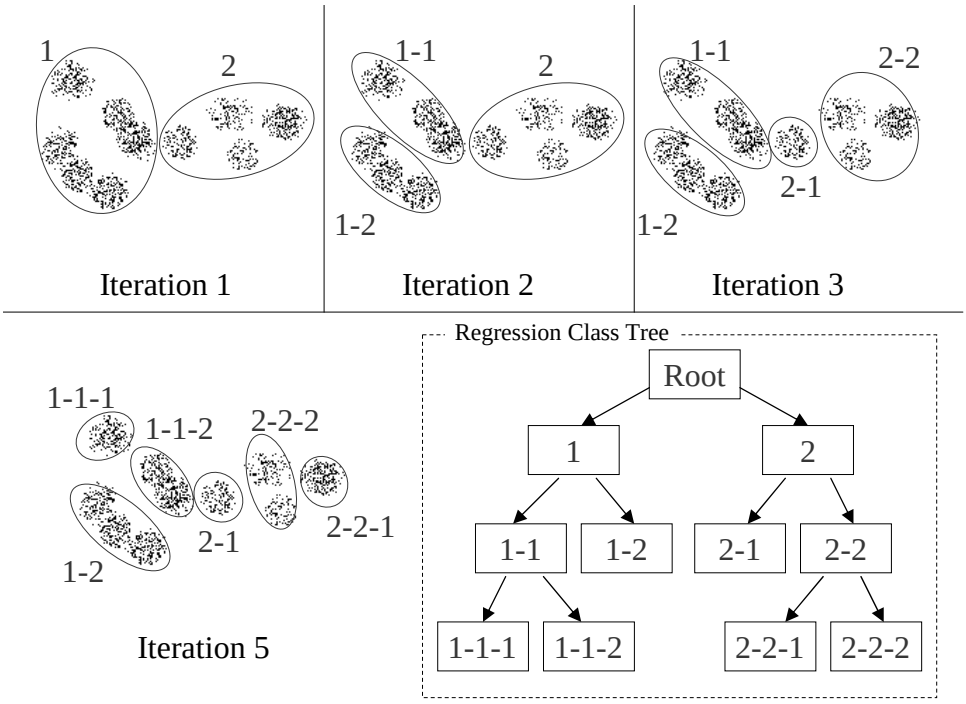

圖 3-1　使用 Top-down binary splitting 方法對資料作二元分割過程及最後產生的迴歸分類樹。

首先我們會先利用 3.2 節中提出的 Top-down binary splitting 方法，對隱藏式馬可夫模型中要分類的高斯混合元件參數作處理，決定出要分類的群數和分類結果。因 Top-down binary splitting 方法所產生的樹狀結構可能產生一些無法真正表示分類在空間中關係的問題，因此我們將分類的結果，利用由下而上的方法，將性質接近的分類節點加以合併，並記錄其關係，直到所有節點資料都合併成一個根節點，來以重建出迴歸分類樹的架構。要合併的分類節點的選擇標準我們一樣是採用了計算 BIC 數值來比較。其建立的方法如以下步驟：

1. 初始化，將使用 Top-down binary splitting 決定的分群輸入，建立各 Base class 節點 $C_1, C_2, ..., C_N$ 。

2. 對所有的節點的資料，對每兩節點的資料合起來計算 $\Delta BIC(GMM_1, GMM_2)$ 數值，建立對照表記錄任兩節點資料的 $\Delta BIC(GMM_1, GMM_2)$ 數值。

3. 對所有的 $\Delta BIC(GMM_1, GMM_2)$ 數值，找出最大值，即表示該兩節點的資料最適合以單群來表示，故將此兩節點加以合併成單一節點。

4. 合併之後，節點總數 $N = N - 1$，更新記錄任兩節點資料的 $\Delta BIC(GMM_1, GMM_2)$ 數值的對照表。

5. 如果 $N \neq 1$，回到第 3 步驟。如 $N = 1$ 表示所有節點已合併至根節點(Root)，表示建立完成。

經過以上步驟，就可以將分群好的節點 $C_1, C_2, ..., C_N$ 依 BIC 的判斷，將適合合併為一群的節點依序合併，直到只剩根(Root)節點，即可依此建立起迴歸分類樹的架構，且建立好的迴歸分類樹中的節點距離，較 Top-down binary splitting 的方法更具有空間上代表性。如以圖 3-2 的例子中，是以圖 3-1 的 Top-down binary splitting 的結果作為初始值，再以 Bottom-up binary merging 方法加以調整的結果，在圖 3-1 中原本距離為 5 的(2-1)類和(1-1-2)類的節點，在經過調整之後的距離為 2，表現了這兩類在空間中的相似性。因此以 Bottom-up binary merging 方法，可以建立較好的階層式架構，較能代表資料在實際空間的關係。

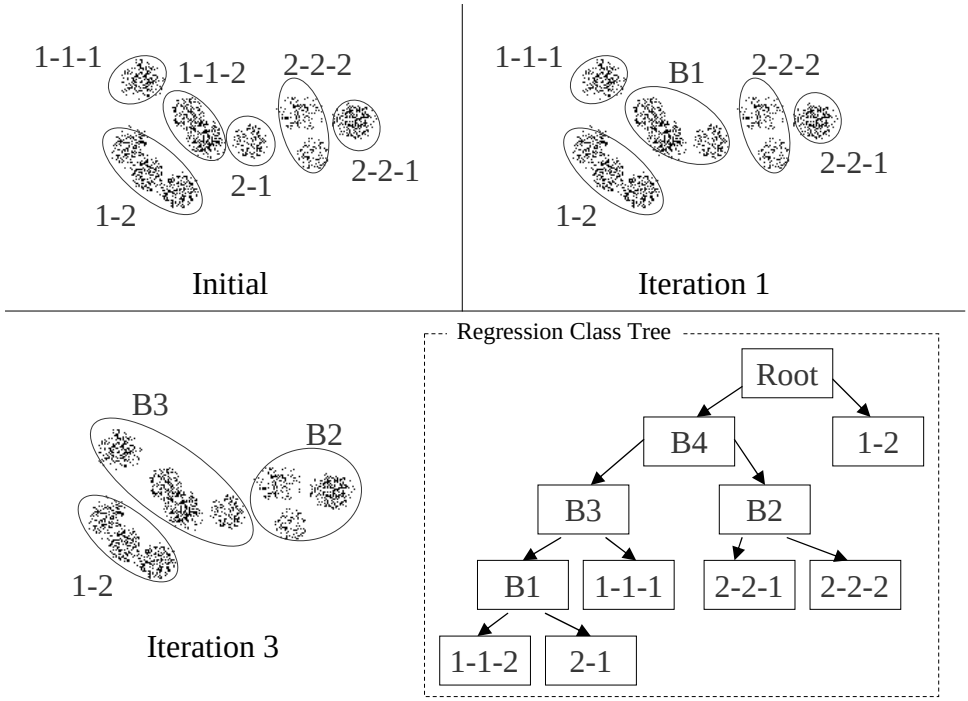

圖 3-2　利用 Bottom-up binary merging 方法將圖 3.1 的結果作調整的過程以及最後重新建立的迴歸分類樹。

第四章　實驗及討論

在這章中我們對於第三章所提之演算法，加以實作，並設計實驗，以評估提出之方法對於語音辨識系統的效能，並和現行之方法作比較。

4.1 實驗設定

對於實驗的平台，我們在硬體方面使用了以 Intel Pentium4 2.4Ghz 為中央處理器的個人電腦，搭配有 512 Megabytes 主記憶體，作業系統為 Microsoft Windows XP 專業版。語音辨識系統方面，我們使用了劍橋大學工程系(Cambridge University Engineering Department)所發展的 Hidden Markov Model Toolkit (HTK) 第 3.2.1 版來建立實驗用的語音辨識系統【23】。

在實驗的語料方面，我們使用了 TCC300 語音資料庫【24】。TCC300 為國立台灣大學，國立交通大學，國立成功大學各自之語音資料庫所集合而成，各校錄製之目的為語音辨認研究，屬於麥克風朗讀語音。其中台大資料庫主要包含短句和字詞，由 100 人錄製而成，交大成大資料庫為長句，長句的文章內容由中央研究院所提供之 500 萬詞詞類標示語料庫中選取，各由 100 人錄製而成，TCC300 為此三校之資料庫集合，有共 300 人的語音資料。我們將此 300 個人的語料，以約略 8.5：1.5 的比例將之分開，85%用作語音辨識引擎的訓練語料，其餘 15%的語料中的每個語者語料中的 5%，約 200 秒的語料用來作為語者調適之用，剩下

的 10%語料為測試語料。用來作為實驗測試的語者的語料都沒有包含在訓練語料中。

4.2 實驗方式

我們使用 TCC300 中的 85%，約 260 人的語料，訓練出的語者無關語音辨識模型為基礎，其辨識正確率在以下表各以 Baseline 表示之，接著針對第三章中所提的演算法，建立 MLLR 語者調適法所需的迴歸分類樹，並設計實驗以比較各種不同演算法所建立的迴歸分類樹。

我們使用了八名測試語者的語料，將每個測試語者的調適語料，平均約每 5 秒鐘分成一段，共分成 25 段，編號為調適語料 1 至調適語料 25，總長約 2 分鐘的調適語料，以依序增加的方式，對 Baseline 模型作調適。第一次以調適語料 1 進行語者調適，第二次以調適語料 1 加上調適語料 2 進行語者調適，以此類推，第二十五次以所有調適語料 1~25 進行語者調適。再對經過不同語料量調適的 HMM 模型以測試語料測試之，以藉此觀察出在不同數量的調適語料下，使用不同迴歸分類樹的各系統對辨識結果的好壞差異。

對於辨識結果的優劣，我們利用兩種不同的計算數據來加以評估，分別是

正確率(Percentage number of labels correctly recognized)，簡稱 Corr.：

$$\%Correct = \frac{H}{N} \times 100\%, \tag{4.1}$$

精確率(Accuracy)，簡稱 Acc.：

$$Accuracy = \frac{H - I}{N} \times 100\%, \tag{4.2}$$

其中 N 為測試語料文稿中所有 label 的數量，H 為辨識結果中正確的 label 數量，I 為插入性錯誤的數量(Insertion error)，即表示多辨識出不存在於文稿中的 label 的數量。除此之外，我們以 D 表示為刪除性錯誤的數量(Deletion error)，即少辨識出存在於文稿中的 label 的數量；以 S 表示為置換性錯誤的數量 (Substitution error)，即辨識出的 label 和文稿中的 label 不符合的數量，則這些數據彼此之間的關係為：

$$N = H + S + D - I \tag{4.3}$$

4.3 實驗結果

我們實作了第三章所提出的 Top-down binary splitting 和 Bottom-up binary merging 的演算法，將 MLLR 調適法中會調整到的隱藏式馬可夫模型中高斯混合模型的平均值向量(mean vectors)做分群，建立出迴歸分類樹，以供 MLLR 調適法來進行動態決定迴歸分類之用。Top-down binary splitting 法最後建立出有 34 群葉節點(leaf node)之迴歸分類樹，之後簡稱為 TD34，而 Bottom-up binary merging 法對 Top-down binary splitting 的結果作調整後建立的迴歸分類樹稱為 BU34。為了比較此二法之分群結果及分群數目的正確性，我們使用了 HMM tool kit 中的 Centroid splitting 演算法建立不同群數的迴歸分類樹，分別設定 2 群、8 群、16 群、34 群、64 群、200 群等不同數量葉節點群數來與之比較，簡稱為 HTK2、HTK8、…、HTK200。我們將不同測試語者語料的測試結果加以平均作為最後的實驗結果，並於表 4-1 列出正確率實驗結果，於表 4-2 列出精確率實驗結果，並分別於圖 4-1 和圖 4-2 畫出圖表。

表 4-1　語者不特定模型(Baseline)和經本論文提出之方法(TD34, BU34)與 HTK 方法進行語者調適的模型之正確率(Corr.)實驗結果，橫列為迴歸分類樹的架構，直行為調適語料的數量。

	Baseline	HTK2	HTK4	HTK8	HTK16	HTK200	HTK34	HTK64	TD34	BU34
1	70.00714	70.00714	70.00714	70.00714	70.00714	70.00714	70.00714	70.00714	70.00714	70.00714
2	70.00714	70.91429	70.91429	70.91429	70.91429	70.91429	70.91429	70.91429	70.91429	70.91429
3	70.00714	71.52571	71.50286	71.50286	71.50286	71.50286	71.50286	71.50286	71.19857	71.53429
4	70.00714	72.37429	72.64857	72.50714	72.50714	72.50714	72.50714	72.50714	72.35714	72.02571
5	70.00714	72.87143	72.50429	72.68571	72.68571	72.68571	72.68571	72.68571	72.85143	72.18714
6	70.00714	73.1	73.13571	73.13857	73.13857	73.13857	73.13857	73.13857	73.13	73.36857
7	70.00714	73.16714	73.40714	73.15857	73.15857	73.15857	73.15857	73.15857	73.15571	73.50143
8	70.00714	73.00571	73.58857	73.61286	73.67714	73.67714	73.67714	73.67714	73.64286	73.62857
9	70.00714	73.12429	73.44143	73.96286	74.01143	74.01143	74.01143	74.01143	73.53571	74.19286
10	70.00714	73.14286	73.64714	74.17571	74.48571	74.48571	74.48571	74.48571	74.08714	73.95
11	70.00714	73.40857	73.83857	74.60571	74.94714	74.94714	74.94714	74.94714	74.35	74.79
12	70.00714	73.55429	74.12	74.59	74.92429	74.92429	74.92429	74.92429	74.75286	74.87
13	70.00714	73.46286	73.96429	74.41857	74.86429	74.97143	74.97143	74.97143	74.95286	74.76714
14	70.00714	73.53571	74.10286	74.59143	74.81857	74.79714	74.79714	74.79714	75.09	74.87143
15	70.00714	73.82857	74.02143	74.41	74.77571	74.83286	74.83286	74.83286	75.08857	74.67857
16	70.00714	73.63429	73.93857	74.59429	75.02857	74.89857	74.89857	74.85286	75.25857	75.74571
17	70.00714	73.45857	74.00429	74.79714	75.41429	75.39714	75.3	75.42143	75.34143	75.86143
18	70.00714	73.33	74.04571	74.82	75.45	75.71714	75.57857	75.67	75.74857	76.12571
19	70.00714	73.64429	73.93571	74.95286	75.58429	75.61	75.53286	75.56286	76.37286	76.20429
20	70.00714	73.69286	73.96286	75.00429	75.64714	75.68857	75.75	75.61714	75.86857	76.33571
21	70.00714	73.60714	74.39286	75.01714	75.97571	75.99286	75.98571	75.96857	76.26857	76.62714
22	70.00714	73.69571	74.20429	75.15571	75.86143	76.21429	76.17143	76.14429	76.22857	76.63714
23	70.00714	73.65571	74.15286	75.04	76.15143	76.70714	76.68143	76.66	76.39286	76.85714
24	70.00714	73.58429	74.34714	75.55571	76.34286	76.92429	76.91143	76.84571	76.73	76.90286
25	70.00714	73.61714	74.57571	75.45857	76.61429	76.66857	76.64	76.59714	76.68	77.26

表 4-2　語者不特定模型(Baseline)和經本論文提出之方法(TD34, BU34)與 HTK 方法進行語者調適的模型之精確率(Acc.)實驗結果，橫列為迴歸分類樹的架構，直行為調適語料的數量。

	Baseline	HTK2	HTK4	HTK8	HTK16	HTK200	HTK34	HTK64	TD34	BU34
1	66.20143	66.20143	66.20143	66.20143	66.20143	66.20143	66.20143	66.20143	66.20143	66.20143
2	66.20143	68.56429	68.56429	68.56429	68.56429	68.56429	68.56429	68.56429	68.54286	68.54286
3	66.20143	69.35286	69.28286	69.28286	69.28286	69.28286	69.28286	69.28286	69.10429	69.35
4	66.20143	70.07714	70.40286	70.05	70.05	70.05	70.05	70.05	69.98429	70.14429
5	66.20143	70.77	70.38143	70.35429	70.35429	70.35429	70.35429	70.35429	70.67857	70.39143
6	66.20143	71.09857	71.09	71.04857	71.04857	71.04857	71.04857	71.04857	71.05714	71.50857
7	66.20143	71.11	71.47429	71.01286	71.01286	71.01286	71.01286	71.01286	71.11286	71.5
8	66.20143	70.93571	71.51857	71.45286	71.51714	71.51714	71.51714	71.51714	71.54	71.75857
9	66.20143	70.94286	71.46857	71.74143	71.79714	71.79714	71.79714	71.79714	71.58143	72.34571
10	66.20143	71.04286	71.80286	72.13857	72.49714	72.49714	72.49714	72.49714	72.12143	72.07143
11	66.20143	71.18429	71.98714	72.60143	72.98571	72.98571	72.98571	72.98571	72.43143	72.91
12	66.20143	71.43857	72.24429	72.54429	72.90571	72.90571	72.90571	72.90571	72.88286	72.94143
13	66.20143	71.26571	72.02571	72.35	72.89286	73.00143	73.00143	73.00143	73.05286	72.88143
14	66.20143	71.38286	72.09	72.62429	72.85714	72.83714	72.83714	72.83714	73.18143	73.05143
15	66.20143	71.62429	72.01571	72.36571	72.75286	72.84143	72.84143	72.84143	73.36429	72.81571
16	66.20143	71.27429	72.01857	72.61286	73	72.96714	72.92286	72.92	73.39857	73.89286
17	66.20143	71.12286	72.00571	72.89714	73.40714	73.39857	73.25714	73.42143	73.71571	74.13
18	66.20143	71.07857	72.13714	72.93143	73.59571	73.96714	73.80429	73.94429	74.21	74.50714
19	66.20143	71.34857	72.02714	73.11857	73.84286	73.89286	73.79429	73.84571	74.74286	74.60714
20	66.20143	71.49	71.97	73.10571	73.92571	74.01714	73.99571	73.94571	74.27429	74.70286
21	66.20143	71.44857	72.47143	73.07429	74.25143	74.32	74.25286	74.29714	74.53714	74.93571
22	66.20143	71.48857	72.28	73.23857	74.05714	74.50143	74.39571	74.45143	74.53571	74.98429
23	66.20143	71.41429	72.11857	73.14429	74.37286	74.95	74.88429	74.92286	74.72429	75.23857
24	66.20143	71.32143	72.33429	73.69857	74.58714	75.18143	75.14857	75.14571	75.07429	75.26
25	66.20143	71.32714	72.63143	73.56571	74.90286	74.93429	74.93143	74.86429	75.01286	75.63143

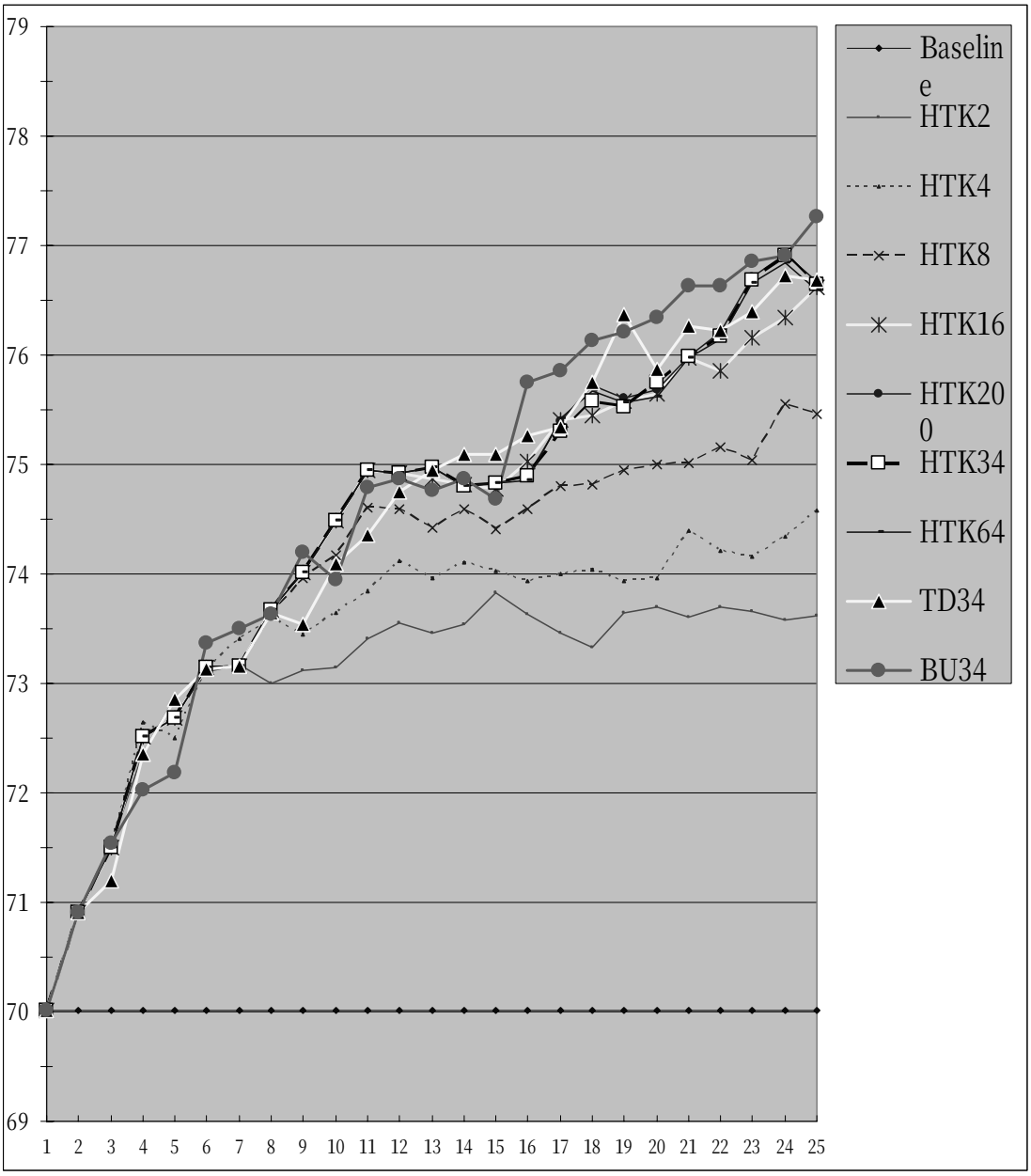

圖 4-1 語者不特定模型(Baseline)和經本論文提出之方法(TD34, BU34)與 HTK 方法進行語者調適的模型之*正確率(Corr.)比較圖，橫軸為調適語料數量，縱軸為正確率百分比，其中 HTK34、64、200 的正確率曲線幾乎是重疊在一起的。*

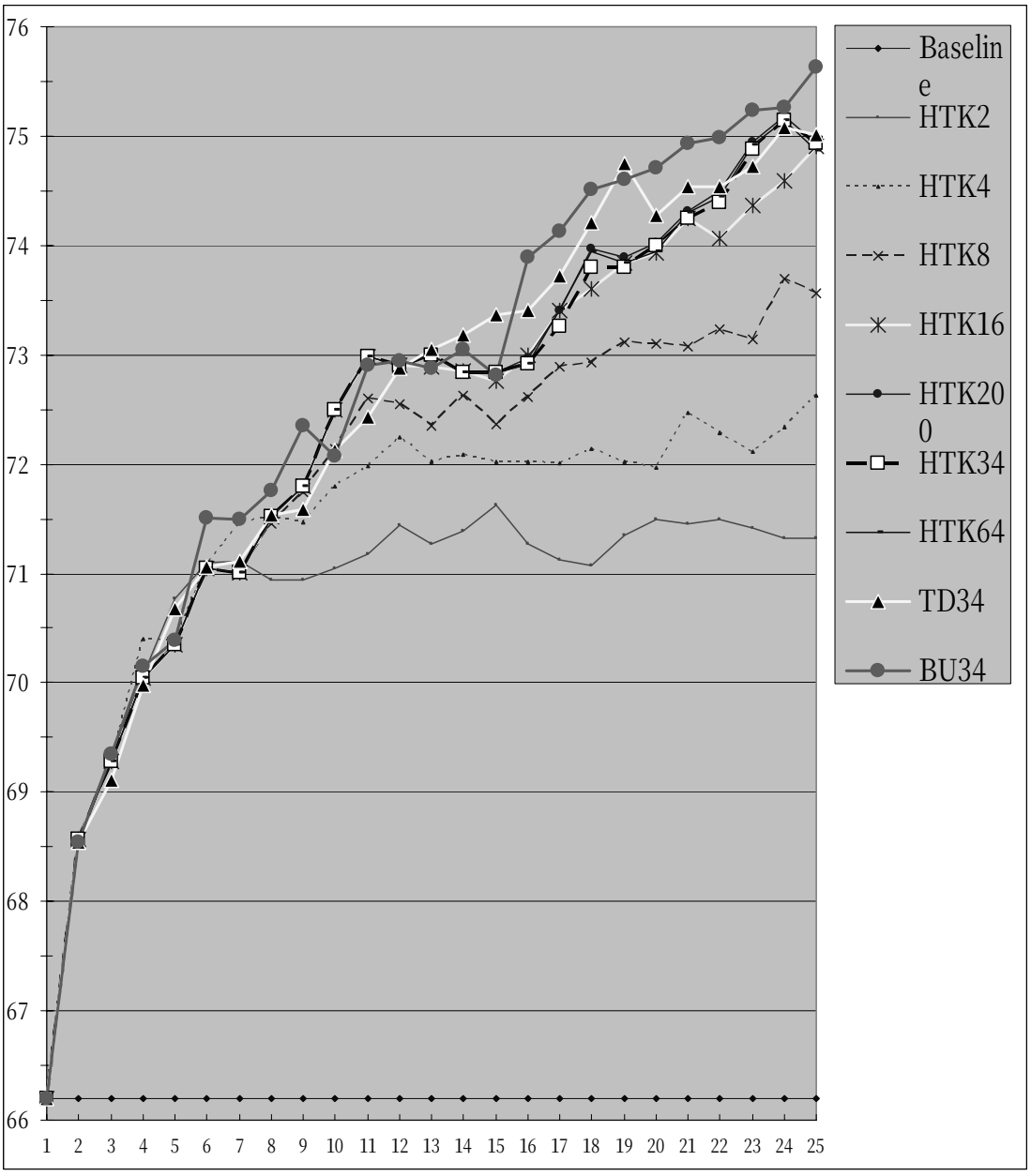

圖 4-2 語者不特定模型(Baseline)和經本論文提出之方法(TD34, BU34)與 HTK 方法進行語者調適的模型之*準確率(Acc.)比較圖，橫軸為調適語料數量，縱軸為準確率百分比，其中 HTK34、64、200 的準確率曲線幾乎是重疊在一起的。*

4.4 實驗討論

由表中或圖中的數據可以看出，在分群數較少的情況下，如 HTK2、HTK8，辨識的正確率和精確率都較易達到飽和，使調適語料數量增加後，辨識率也無法提升。這是因為在較少的分群情況下，較多的參數共用轉換矩陣，比較不能精密地對 HMM 作調整。

而當分群數超出一定數目時，如 HTK34、HTK64、HTK200 的辨識率曲線都無顯著的差別，幾乎是相同的，這表示在實驗中的調適語料數量下，只足夠調適某個上限數量分群數的轉換矩陣，而大於上限數量的分群數會使得迴歸分類樹的分類太細，沒有足夠的調適語料作調適，使過於精細分類的轉換矩陣都不會被使用到(可以參見第三章中的例子)。雖然如果再增加調適的語料，是可能可以增加可調適的轉換矩陣上限數，使有較多分群數的迴歸分類樹的精細分類優點較易顯現出來，但是我們認為，如果調適語料的數量比我們實驗最多的調適語料數量還要多的話，那就表示該名語者的語料數量是非常足夠的，我們需要的就不是快速調適的演算法和技術，而是需要較精確的語者調適方法(如 MAP)或是模型參數重估(parameter re-estimation)的方法，甚至可以直接訓練該語者的語者特定辨識模型。

而從實驗數據來看，可以發現利用我們提出的 Top-down binary splitting 方法建立的迴歸分類樹的模型 TD34，辨識的正確率和精確率都大致和 HTK 所建立的 HTK34 相差不多，並在大多數情形有小幅度的提升，而和較多分群的 HTK64、HTK200 相比，正確率和精確率的表現也不會比較差(事實上 HTK64、HTK200 和 HTK34 的曲線幾乎相同)，這就表示 Top-down binary splitting 方法所決定的 TD34 的 34 個分群數目是合理的數字，不會因為分群過少而使正確率和精確率提早飽和，且和相同分群數的 HTK34 或甚至較多分群數的模型相比，正確率和精

確率也不會比較差，多數情況下有小幅度的提升，因此我們可以由此實驗得出使用我們所提出的 Top-down binary splitting 方法來建立迴歸分類樹是比使用 HTK 的 Centroid splitting 方法來得優秀的，而且其自動決定的迴歸分類樹的分群數目，對於語音辨識方面，也是合適的。

而使用 Bottom-up binary merging 方法所建立的迴歸分類樹 BU34，除了少數的幾種數量語料的情況下外，在大多數的正確率和精確率都有比 TD34 好的表現，表示了用 Top-down binary splitting 方法所產生的結果，在經過 Bottom-up binary merging 方法調整後所建立的迴歸分類樹的架構，在實驗上可以證明也會有較佳的表現。

第五章　系統應用：手持式設備的語音辨識系統

在本章中介紹應用本論文所提之語者調適法，所實作開發出來的手持式設備(Handheld device)的語音辨識系統。

5.1 系統簡介

隨著資訊設備朝輕薄短小發展，手持式設備的愈加普及，如個人數位助理(Personal Digital Assistant)，平板式電腦(Tablet PC)，智慧型手機(Smartphone)等等，且手持式設備的架構也趨向開放化，功能也更加多元化，且因為設備的小型化，傳統的人機介面如鍵盤輸入，滑鼠等都較為不易使用，利用語音輸入作為人機介面就顯得非常便利了。

手持式設備一般來說侷限於運算能力較慢，其上語音輸入系統通常為像聲控撥號、語音標籤指令、或是單字辨認等有限字集的應用，較少大字彙(Large Vocabulary)的語音辨認系統。因此，我們設計了一個分散式運算的架構，利用無線網路將手持式設備和後端伺服器連接，彼此協同運算，由手持式設備負責前端訊號的處理，經無線網路傳送至伺服器作辨認的運算，再將結果回傳給使用者，利用這樣的模式，來完成語音辨認系統。

5.2 系統平台

在系統的平台方面，本系統在伺服器端使用了以 Intel Pentium4 3.0Ghz 為中央處理器的電腦，搭配有 1024 Megabytes 主記憶體，作業系統為 Microsoft Windows XP 專業版。在手持式設備端，使用了 HP iPAQ 5550 型號的個人數位助理(PDA)，使用 Intel PXA255，400Mhz 中央處理器，搭配 128Megabytes 記憶體，作業系統為 Microsoft PocketPC 2003 版，利用內建 802.11b 無線網路介面和伺服器連線。

5.3 系統架構介紹

本系統的架構圖如圖 5-1，其流程及各方塊說明如下：

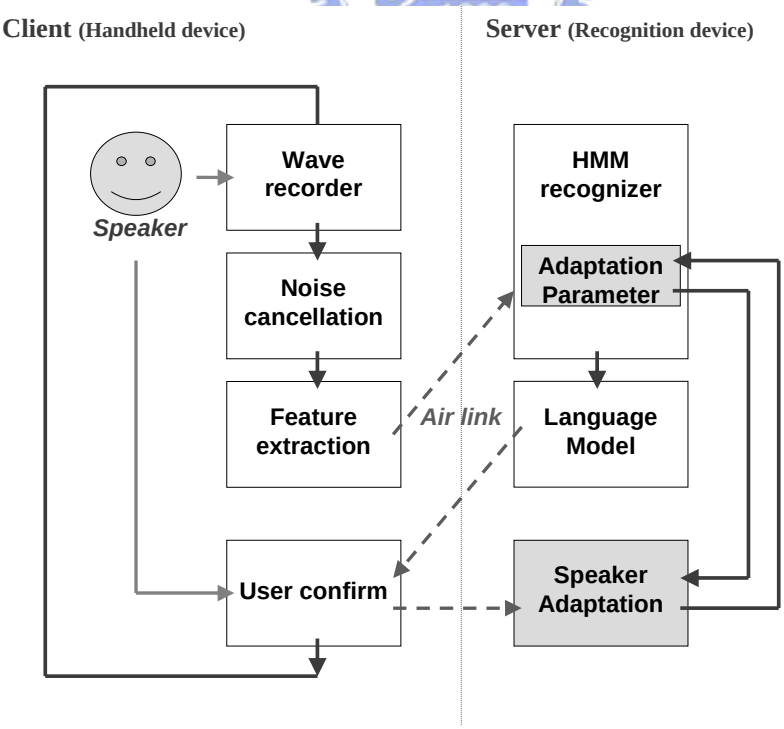

圖 5-1 本論文應用之手持式設備語音辨識系統架構圖

- Wave recorder (波型檔錄製)：使用者(Speaker)透過手持式設備上的程式介面，將輸入語音錄製成為 wave 格式檔案。

- Noise cancellation (雜訊消除)：對輸入之 wave 檔，估計其雜訊部分，加以消除，以強化語音部分，以避免輸入語音時的背景噪音影響辨識結果，在此使用了參考資料【25】的方法。

- Feature extraction (語音特徵參數提取)：對於 wave 檔，提取其特徵參數值，以供後端隱藏式馬可夫模型辨識使用，在此語音特徵參數使用了 12 維的梅爾倒頻譜參數(Mel-Frequency Cepstral Coefficients, MFCC)，加上 1 維的對數能量參數，提取後的參數經無線網路傳送至後端伺服器。在伺服器上，將此 13 維參數，計算其一階差量和二階差量，合成共計 39 維的特徵參數。

- HMM recognizer (隱藏式馬可夫模型辨識器)：對於輸入的 39 維特徵參數，利用隱藏式馬可夫模型加以辨識成此段語音所代表的音節 (Syllable，在國語語音中所使用音節表見附錄 A)，在此使用了和第四章實驗部分相同的隱藏式馬可夫模型，其為使用 TCC300 中 260 人的語料，所訓練出的語者無關語音辨識模型，並使用了兩階段的辨識方法，對於連續語音，先辨識出音節結果和各音節所在此段語音中的出現時間，再以此時間資訊，對特徵參數依各音節作分段，成為各單音節的特徵參數，再作第二階段的辨識，對各單音節作辨識，並輸出機率值前十大的音節結果，作為候選音，輸出此段語音中各音節的前十候選音給語言模型使用。

- Language model (語言模型)：對各音節的候選音，對照詞庫加以構詞，把輸入語音所代表之字句建立出來，在此使用了參考資料【26】所提之

方法。最後將機率前十大之候選字句經無線網路回傳至手持式設備。

- User confirm (使用者確認)：手持式設備接收到此段語音所代表之前十可能候選字句後，透過程式介面給使用者選擇確認，經確認之結果，回傳至伺服器端的語者調適部份。如使用者輸入之語句都不包含在候選字句中，使用者可以輸入正確結果供語者調適使用，或是拒絕(Rejection)，表示此段錄音效果不佳，不需將此語料進行語者調適處理。

- Speaker adaptation (語者調適)：伺服器端接受到使用者確認，或使用者輸入之語音辨識結果後，建立文稿，進行語者調適，調整隱藏式馬可夫模型中的參數，使模型更匹配使用者的語音特性，增進語音辨識系統的準確度。我們使用了 MLLR 調適法，加上本論文所提方法建立之迴歸分類樹。

5.4 系統效能評估

對於本系統的效能，我們實際讓測試語者來測試系統，並記錄系統輸出的結果。測試的語者有十位，測試的環境為普通的辦公室環境，每位語者使用時，都從使用系統預設之語者不特定模型作為辨識核心開始，輸入測試用的文稿。每個語者都使用相同的測試文稿，文稿中含有 30 句中文的查詢句子，使用者逐一將語音輸入作辨識，並記錄結果。如系統在前十大候選句中含有正確的辨識結果，則回傳正確的句子給系統進行語者調適，調整辨識核心。如系統在候選句中都不含有正確結果，則記錄為辨識失敗，使用者手動輸入正確的結果給系統進行語者調適。

測試完畢後，以字為單位，計算各測試者的正確率及準確率並加以平均，最後結果的正確率(Corr.)為 90.09%，準確率(Acc.)為 87.21%。

第六章　結論及未來研究方向

6.1 結論

本論文的研究重點放在以隱藏式馬可夫模型為核心的中文語音辨識系統的語者調適技術，對於最大相似度線性迴歸法(MLLR)所使用的迴歸分類樹架構，傳統之建立方法需要人為經驗來判斷分類樹的基底類別數量，可能會因不合適的設定，導致辨識率不佳或是系統複雜度的上升。針對於此，我們使用了 BIC 作為準則，提出了由上而下的二元分裂法來建立迴歸分類樹，以此法可以使迴歸分類樹的建立自動完成，不需人為判斷介入，避免了利用經驗法則來決定的風險。且經實驗測試後，我們提出的方法都能有效的建立具代表性的迴歸分類樹，效能評估上也有不錯的表現，並不會因自動化的決定而損失了辨識的準確度。此外，我們基於由上而下的二元分裂法產生的迴歸分類樹結果，提出了由下而上的二元合併法來對迴歸分類樹加以調整，建立出更能代表資料在空間上的分佈情形的迴歸分類樹，也實驗中也能對語音辨識的效能加以提升。

而本論文也成功地將提出之方法應用在手持式設備的語音辨識系統當中，採用了分散式計算的架構，使得手持式設備上的大字彙語音辨識系統可以實現，加上了本論文提出的語者調適技術，可以讓使用者在持續使用時不斷地調適辨識模型，使辨識模型之聲學特性和使用者相符合，不斷增進辨識的準確度。

6.2 未來研究方向

由本論文中的研究和實驗之後，我們發現有數個主題是我們未來希望可以繼續研究的重點，在此說明如下。

本論文提出的方法使用了貝氏資訊基準(Bayesian Information Criterion)來作為自動判斷的標準，未來希望可以使用不同種類之基準來作為判斷的標準，也許可以找出比貝氏資訊基準更適用於語音辨識模型參數的基準，

另由實驗過程中我們了解到語料資料庫之重要性，在訓練隱藏式馬可夫模型以及作語者調適和測試時，都需要大規模，有系統的語料資料。對於本論文用來訓練及測試的語料資料庫，其語料來源和環境較為單一，未來希望可以使用其他不同資料庫之語料，以訓練出更一般化的語者不特定模型和進行更廣泛語者的測試。

參考文獻

【1】 Eric Chang, Frank Seide, Helen M. Meng, Zhuoran Chen, Yu Shi and Yuk-Chi Li, “A System for Spoken Query Information Retrieval on Mobile Devices”, IEEE Trans. On Speech and Audio Processing, Vol. 10, No.8, November 2002

【2】 Chin-Hui Lee and Biing-Hwang Juang, “A Survey on Automatic Speech Recognition with an Illustrative Example on Continuous Speech Recognition of Mandarin”, Computational Linguistics and Chinese Language Processing, Vol.1, No.1, August 1996

【3】 A. Acero and X. Huang, “Speaker and Gender Normalization for Continuous-Density Hidden Markov Models”, Proc. ICASSP, Vol. 1, pp342-345, Atlanta, GA, USA, 1996

【4】 D. Giuliani, M. Gerosa and F. Brugnara, “Speaker Normalization through Constrained MLLR Based Transforms”, ICSLP, 2004

【5】 Jean-Luc Gauvain and Chin-Hui Lee, “Maximum a Posteriori Estimation for Multivariate Gaussian Mixture Observations of Markov Chains”, IEEE Trans. On Speech and Audio Processing, Vol.2, No. 2, April 1994

【6】 R. Chengalvarayan and Li Deng, “A Maximum A Posteriori Approach to Speaker Adaptation using the Trended Hidden Markov Model”, IEEE Trans. On Speech and Audio Processing, Vol. 9, No. 5, July 2001

【7】 C.J. Leggetter and P.C. Woodland, “Maximum Likelihood Linear Regression for Speaker Adaptation of Continuous Density HMMs”, Computer Speech

and Language, Vol 9, pp171-185, 1995

【8】 C.J. Leggetter and P.C. Woodland, “Flexible Speaker Adaptation using Maximum Likelihood Linear Regression”, Proc. ARPA Spoken Language Technology Workshop, 1995

【9】 Heidi Christensen, “Speaker Adaptation of Hidden Markov Models using Maximum Likelihood Linear Regression”, Thesis of Aalborg University Denmark, 1996

【10】 R. Kuhn, et al, “EigenVoices for Speaker Adaptation”, Proc. ICSLP, Sydney, Australia, November, 1998

【11】 Robert Westwood, Speaker Adaptation Using Eigenvoices, Cambridge University England, 1999

【12】 Xuedong Huang, Alex Acero and Hsiao-Wuen Hon, Spoken Language Processing, Prentice Hall, 2001

【13】 王小川, 語音訊號處理, 全華科技圖書, 2004

【14】 Xuedong Huang, ”A Study on Speaker-Adaptive Speech Recognition”, DARPA Speech and Language Workshop, 1991.

【15】 曹昱,「國語音節及聲調辨識之少量語料語者調適」, 國立台灣大學, 電信工程系碩士論文, 2001

【16】 M.J.F. Gales and P.C. Woodland, “Mean and Variance Adaptation within the MLLR Framework”, Computer Speech & Language, Vol. 10, pp249-264, 1996

【17】 Jeff. Bilmes, “A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Midden Markov Models”, Technical

Report ICSI-TR-97-021, International Computer Science Institute, University of Berkeley, 1998

【18】 周樂生,「以最大似然機率線性回歸法建立線上層級體系語者調適語音辨認」,國立交通大學,電信工程系碩士論文, 2001

【19】 M.J.F Gales, “The Generation and Use of Regression Class Tree for MLLR Adaptation”, Cambridge University England, 1996

【20】 Bowen Zhou, John H.L. Hansen, “Improve Structural Maximum Likelihood Eigenspace Mapping for Rapid Speaker Adaptation”, ICSLP, Vol. 2, pp1433-1436, Denver, USA, Sept. 2002

【21】 C. Fraley and A. E. Raftery, “How Many Clusters? Which Clustering Method? Answers Via Model-Based Cluster Analysis”, The Computer Journal, Vol.41, No.8, pp578-588, 1998

【22】 Trevor Hastie, Robert Tibshirani and Jerome Friedman, The Elements of Statistical Learning, Springer, 2001

【23】 S. Young, et. al., The HTK Book (For HTK Version 3.2.1), Cambridge University Engineering Department, 2003

【24】 中華民國計算機語言學學會, TCC-300 國語語音資料庫, http://rocling.iis.sinica.edu.tw/ROCLING/

【25】 沈揚智,「語音強化技術在相加性雜訊環境下的語音辨識之研究」,國立交通大學,資訊工程系碩士論文, 2005

【26】 呂宜玲,「中文語音辨識中語言模型的強化之研究」,國立交通大學,資訊工程系碩士論文, 2005

附錄 A　中文語音基本單位表

序號	拚音	序號	拚音	序號	拚音	序號	拚音
1	Sil	41	g_o	81	l_u	121	sic_a
2	A	42	g_u	82	m_a	122	sic_e
3	Ai	43	h_a	83	m_e	123	sic_i
4	An	44	h_e	84	m_ee	124	sic_iu
5	Ang	45	h_ee	85	m_i	125	sic_o
6	Au	46	h_o	86	m_o	126	sic_u
7	b_a	47	h_u	87	m_u	127	t_a
8	b_e	48	i	88	n_a	128	t_e
9	b_ee	49	ia	89	n_e	129	t_i
10	b_i	50	iai	90	n_ee	130	t_o
11	b_o	51	ian	91	n_i	131	t_u
12	b_u	52	iang	92	n_iu	132	ts_a
13	ch_a	53	iau	93	n_o	133	ts_e
14	ch_e	54	ie	94	n_u	134	ts_empt
15	ch_empt	55	in	95	o	135	ts_o
16	ch_o	56	ing	96	ou	136	ts_u
17	ch_u	57	iou	97	p_a	137	tz_a
18	chi_i	58	iu	98	p_e	138	tz_e
19	chi_iu	59	iuan	99	p_ee	139	tz_ee
20	d_a	60	iue	100	p_i	140	tz_empt
21	d_e	61	iun	101	p_o	141	tz_o
22	d_ee	62	iung	102	p_u	142	tz_u
23	d_i	63	j_a	103	r_a	143	u
24	d_o	64	j_e	104	r_e	144	ua
25	d_u	65	j_ee	105	r_empt	145	uai
26	E	66	j_empt	106	r_o	146	uan
27	Ei	67	j_o	107	r_u	147	uang
28	empt1	68	j_u	108	s_a	148	uei
29	empt2	69	ji_i	109	s_e	149	uen
30	En	70	ji_iu	110	s_empt	150	ueng
31	Eng	71	k_a	111	s_o	151	uo
32	Er	72	k_e	112	s_u		
33	f_a	73	k_o	113	sh_a		
34	f_e	74	k_u	114	sh_e		
35	f_ee	75	l_a	115	sh_ee		
36	f_o	76	l_e	116	sh_empt		
37	f_u	77	l_ee	117	sh_o		
38	g_a	78	l_i	118	sh_u		
39	g_e	79	l_iu	119	shi_i		
40	g_ee	80	l_o	120	shi_iu		

附錄 B　中文發音分類表

Fricative 摩擦音	Unvoiced 清	ㄏ	h
		ㄒ	shi (x)
		ㄕ	sh
		ㄙ	s
		ㄈ	f
	Voiced 濁	ㄖ	r
Affricate 爆破音（塞擦音）	Unvoiced 清	ㄑ	chi (q)
		ㄔ	ch
		ㄘ	thi
	Voiced 濁	ㄓ	ji (zh)
		ㄐ	j
		ㄗ	tz (z)
Stop（爆破音）	Unvoiced 清	ㄎ	k
		ㄊ	t
		ㄆ	p
	Voiced 濁	ㄍ	g
		ㄉ	d
		ㄅ	b
Nasal 鼻音		ㄋ	n
		ㄇ	m
Liguid 邊音		ㄌ	l