

國立臺灣大學電機資訊學院電信工程學研究所
博士論文

Graduate Institute of Communication Engineering
College of Electrical Engineering & Computer Science
National Taiwan University
Doctoral Dissertation

強健及分散式語音辨識系統中的
動態量化技術

Dynamic Quantization for Robust and Distributed
Speech Recognition

萬佳育

Wan, Chia-Yu

指導教授：李琳山 博士

Advisor: Lee, Lin-Shan, Ph.D.

中華民國九十七年六月

June, 2008

Dynamic Quantization for Robust and Distributed Speech Recognition

by

Chia-Yu Wan

A Dissertation

Submitted to Graduate Institute of
Communication Engineering
National Taiwan University

in Partial Fulfillment of the
Requirements for the Degree of
Doctor of Philosophy

in

Communication Engineering

Taipei, Taiwan, Republic of China

June 2008

國立臺灣大學博士學位論文
口試委員會審定書

強健及分散式語音辨識系統中的動態量化技術

Dynamic Quantization for Robust and Distributed
Speech Recognition

本論文係萬佳育君 (D94942018) 在國立臺灣大學電信工程學研究所完成之博士學位論文，於民國 97 年 6 月 13 日承下列考試委員審查通過及口試及格，特此證明

口試委員：

李山

(簽名)

(指導教授)

王中川

陳仁宏

陳良基

陳銘憲

簡仁宗

王暉

系主任、所長

(簽名)

誌謝

首先感謝我的恩師李琳山老師，給予學生豐富的研究資源，讓學生能自由選擇感興趣的主題研究，老師待人處世認真嚴謹的態度和對研究的熱忱，是學生終生學習的榜樣。感謝老師總是不眠不休地和學生討論研究問題，修改學生的論文，並傾囊相授完美演說的技巧，讓學生成長許多，非常感謝老師的指導和栽培。

謝謝口試委員王小川老師、陳信宏老師、陳良基老師、陳銘憲老師、簡仁宗老師，給予學生諸多寶貴的建議，讓學生能從不同的角度思考問題，受益良多。感謝王新民老師和陳柏琳老師，常給我們許多研究的建議，為我們加油打氣。感謝很多的論文審稿委員在我研究過程中，給予我非常多很好的意見，感謝許多在國際會議認識的研究先進，開拓了我的視野，並充分激發了我對研究的熱忱。

感謝朱國華學長總是很熱心地給予我研究上的建議與協助、分享人生的寶貴經驗，為我指點迷津。感謝小皓學長在我剛進實驗室時，很主動積極地指導我，讓我可以很快的學會程式和實驗方法。感謝王瑞璋學長和陳冠廷學長常熱心地傾聽學弟妹的問題，並予以協助。感謝明怡學姐和 Thomas 學長，常在許多不同領域的問題裡，提出很有價值的研究建議，你們是我心目中想成為的博士生典範。感謝哲光學長，我每次都很期待你的報告，即使是困難的主題，也能講得精彩而有趣。感謝孫良哲學長，經常發表獨到的見解，讓我有很大啟發。感謝禹吟學姊常常關心我、鼓勵我，讓我心裡感覺很溫暖，感謝尚年學長，碩一時每天看到學長認真寫程式和做研究的態度，讓我從此立志要好好做研究。感謝 Aaron，常常幫我修改論文、指導我英文寫作和報告的技巧。感謝家興學長、碧娟學姐、陳逸學長、羿帆、小妞、東烜和竣安，盡心盡力的管理工作站，讓我們可以順利地做實驗。感謝永禎和峰森，你們是我一起修課和相互關心研究狀況的好伙伴。感謝佳好，將豐富的研究成果交接給我，讓我學到很多，很佩服妳專注認真的研究精神。感謝馬雅，細心地整理相片資料，讓未來的實驗內容能更豐富，同時，我也很想學習你親切待人的態度，帶給人許多溫暖。感謝運寰，熱心幫助我建立語音檢索的系統。感謝力維和所有學弟妹們，讓實驗室的生活充滿熱情與活力。

很感謝陳逸學長一直陪伴我、鼓勵我、照顧我，和我一起面對所有的問題，讓我可以勇敢克服困難，也讓我學會用更體貼和感恩的心，來關心我的家人。

我最想將我的博士論文成果，和我的爸爸媽媽與姊姊分享，我非常感謝爸媽，很辛苦地把我們從小拉拔長大，為我們犧牲、付出很多，讓我和姊姊有最好的教育和成長環境，您們是我生命中最重要的人，我希望有一天能讓您們覺得，女兒已經長大，可以照顧您們、可以成為您們生活的倚靠。我很感謝您們給我一個這麼好的姊姊，從我進入大學離開家生活以來，姊姊就像媽媽一樣關心我、照顧我，每次我能順利度過難關時，都深深覺得，有姊姊真幸福。謝謝您們！

Dynamic Quantization for Robust and Distributed Speech Recognition

by
Chia-Yu Wan

Doctor of Philosophy in Communication Engineering
National Taiwan University, Taipei, Taiwan

Advisor: Lin-Shan Lee



Abstract

Split Vector Quantization (SVQ) is popularly used in a Distributed Speech Recognition (DSR) framework, in which the speech features are vector quantized and compressed at the client, transmitted via wireless networks, and recognized at the server. However, recognition accuracy is inevitably degraded by environmental noise at the input, quantization distortion and transmission errors; these three sources of disturbances naturally mix up with each other and further complicate the problem. The mismatch between the pre-trained VQ codebook and the constantly changing environmental conditions at the moving client is one of several major problems. In this dissertation, two dynamic quantization methods are proposed for both robust and distributed speech recognition.

The first approach, Histogram-based Quantization (HQ), is a novel approach in which the partition cells of the quantization are dynamically defined by the histogram or order statistics of a segment of the most recent past values of the parameter to be quantized. This dynamic quantization scheme based on local signal order statistics is

shown to be able to solve to a good degree many problems related to the mismatch with a fixed VQ codebook. This concept is extended to Histogram-based Vector Quantization (HVQ). A Joint Uncertainty Decoding (JUD) approach is further developed for it, in which the uncertainty caused by both environmental noise and quantization errors can be jointly considered during Viterbi decoding. A three-stage error concealment (EC) framework based on HQ is also developed to handle transmission errors. The first stage detects the erroneous feature parameters at both the frame and subvector levels. The second stage then reconstructs the detected erroneous subvectors by MAP estimation, considering the prior speech source statistics, the channel transition probability, and the reliability of the received subvectors. The third stage then considers the uncertainty of the estimated vectors during Viterbi decoding. At each stage, the error concealment (EC) techniques properly exploit the inherent robust nature of Histogram-based Quantization (HQ).

The second approach is context-dependent quantization, in which the representative parameter (whether a scalar or a vector) for a quantization partition cell is not fixed, but depends on the signal context on both sides, and the signal context dependencies can be trained with a clean speech corpus or estimated from a noisy speech corpus. This results in a much finer quantization based on local signal characteristics, without using any extra bit rate. The context-dependent quantization could be integrated with HQ proposed above. Both partition cells and representative values are dynamically defined in the integrated dynamic quantization process.

These two dynamic quantization techniques are not only useful for DSR, but are also attractive feature transformation approaches for robust speech recognition outside of a

DSR environment. In the latter case, the feature parameters are simply transformed into their representative parameters after quantization. The robust nature of dynamic quantization is analyzed in detail. HQ performs the transformation by block-based order statistics, small disturbances of the feature parameters can be absorbed by the histograms to a good extent. As a result, the proposed HQ scheme can be useful for both robust and distributed speech recognition. For robust speech recognition, HQ is used as the front-end feature transformation and JUD as the enhancement approach at the back-end recognizer. For context-dependent quantization, exploiting high correlation in speech signals also significantly improves the robustness against transmission errors and environmental noise.

All the above claims about speech recognition have been verified by experiments using the Aurora 2 testing environment, and significant performance improvements for both robust and/or distributed speech recognition over conventional approaches have been achieved. In addition, we also apply the concept of dynamic quantization on image features for photograph retrieval. Quantization with dynamic partition cells reduces the mismatch of pixel value distributions between different cameras; thus photos taken from different cameras are more easily retrieved. Quantization with dynamic representative codewords emphasizes more important color bins and texture features; thus the photo difference in more discriminative feature dimension could be preserved well in the quantization process as well. Experimental results show that dynamic quantization on image features can significantly improve photo retrieval results.

強健及分散式語音辨識系統中的動態量化技術

學生：萬佳育 指導教授：李琳山博士

國立台灣大學電信學工程研究所

摘要

架構於無線網路上的分散式語音辨識系統 (Distributed Speech Recognition, DSR)，將傳統的語音辨識分散在手持設備與伺服器兩端：在手持設備執行語音特徵參數的抽取與壓縮，並將壓縮後的資料經過無線通道傳送至伺服器端，以進行特徵參數的還原與辨識。由於隨身攜帶的手持設備面臨多變且無可預知的環境，環境雜訊與壓縮帶來的信號失真以及傳輸造成的錯誤會互相加成起來，嚴重影響分散式語音辨識的效能。

本論文針對聲學模型與量化碼本的訓練語音和實際進行辨識語音的特性不匹配的問題，提出兩種強健性的動態量化法，第一種方法是「以分佈統計為基礎的強健性量化法」，此方法是根據最接近所要量化係數的前面一段區間的順序統計資訊 (order-statistics) 或分佈統計資訊 (histogram)，動態調整其量化邊界，可使量化碼本自動跟隨輸入語料的分佈而改變，解決了傳統以距離為基礎的量化因固定碼本的限制下，量化碼字無法有效表示帶有不同雜訊的語音的問題，而動態的量化區間也使得量化本身較不受不同語者特性所影響；本論文亦進一步提出一種以量化失真與分佈

偏移為基礎的綜合不確定性解碼法，在完全不需要增加額外資料傳輸量的情況下，能夠估測在「以分佈統計為基礎量化法」中的量化失真和雜訊環境下語音特徵參數的兩種不確定性，在辨識器解碼的過程中一併考慮。在「以分佈統計為基礎量化法」的分散式辨識系統中，本論文進一步發展出一套三階式錯誤補償技術。此技術結合了以分佈統計為基礎量化法的強健特性，並同時考慮了語音輸入端的背景雜訊和無線通道錯誤的問題。第一階段方法可偵測出音框和子向量兩種層級的錯誤，第二階段考慮了語音訊號統計資訊、通道傳輸的轉換機率，以及所接收的語音係數可靠程度，利用最大可能性估測法還原錯誤的語音特徵向量。第三階段則將所估測語音特徵資訊的不確定性，加入維特比解碼的過程，使得較不確定的語音係數對辨識率的影響較小。在每一階段中，錯誤補償技術皆能有效利用以分佈統計為基礎量化法的強健本質，我們在 Aurora 2 語料庫上做了完整的實驗，並包含 GPRS 通訊系統的通道錯誤模擬。實驗的結果顯示我們所提出的方法能有效地克服環境雜訊與傳輸干擾的影響，並顯著地提升語音辨識的正確率。

本論文提出的第二種強健性的動態量化法是「前後資訊相關的量化法」，有別於傳統的量化方法都是以音框為單位，考慮單一音框的參數數值來決定此一參數的量化結果，「前後資訊相關的量化法」的每個分割單元在解碼過程並不是對應到單一的碼字，其代表碼字會根據前後特徵參數不同動態決定，此量化法考慮了語音前後相關的特性，可得到較單一音框的量化更具有代表性的碼字。我們亦進一步建立前後音框碼字的三連模型，將語音參數受到雜訊影響可能的變化在訓練模型時即加以考慮，並以最小均方誤差(MMSE)準則來估計語音特徵參數。此量化法可以直接應用在使用者端任何原有的量化方法上，完全不需更改使用者端的計算複雜度和傳

輸的位元數，而是在伺服器端加入前後音框的資訊，透過一對多的解碼方式增加量化特徵參數的解析度。本論文亦將「前後資訊相關的量化法」與「以分佈統計為基礎量化法」結合，能動態的定義分割邊界和代表碼字，對環境雜訊和傳輸錯誤皆極具強健性。

本論文提出的兩種強健性量化法，亦可應用於強健性語音辨識。將語音特徵參數透過強健性量化法轉換為代表值，可視為一套強健性的特徵參數轉換法，量化法本身具有強健的特性，部分的環境干擾可以被量化法吸收掉，實驗結果顯示對低訊噪比環境與不穩定性的雜訊也可有效處理。

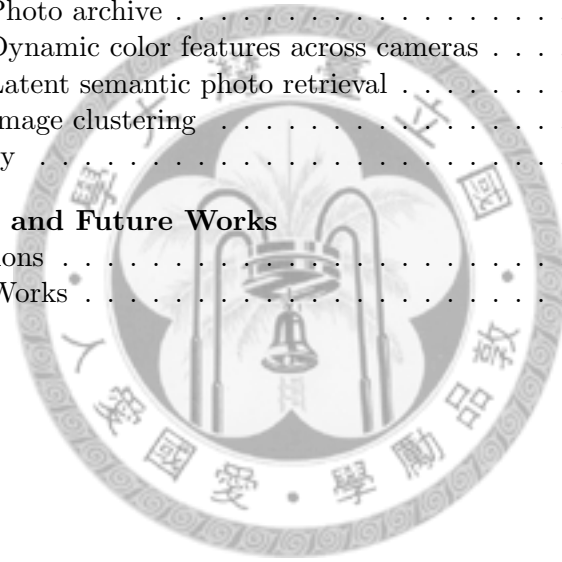
最後，本論文亦將動態量化的觀念應用到圖片特徵參數的量化。由於相片庫中大部分的相片即使有文字註解，亦均為較短的文句，無法充分代表整張相片的語意，若能由相片的圖片特徵抽取出可代表相片特徵的「圖像詞」，利用這些「圖像詞」所建立的潛藏語意模型將有助於相片檢索。本論文將抽取「圖像詞」的過程視為一種量化法，也就是要找出一些能有效表示圖片特徵參數的「圖像詞」。實驗結果顯示使用動態的量化法能有效抽出代表圖片語意的「圖像詞」，大幅改善了相片檢索的效果。

Contents

English Abstract	i
Chinese Abstract	iv
Contents	i
List of Tables	iv
List of Figures	v
1 Introduction	1
1.1 Background	1
1.2 Primary Achievements of this Dissertation	3
2 Preliminaries – Background Review	7
2.1 Introduction	7
2.2 Review of Existing Feature Quantization Approaches	7
2.2.1 Split Vector Quantization (SVQ)	7
2.2.2 Transform Coding	8
2.3 Review of Existing Error Concealment Approaches	10
2.3.1 Error detection and correction	10
2.3.2 Erroneous Feature Estimation	11
2.3.3 Weighted Viterbi Decoding	11
2.4 Experimental environments	12
2.4.1 Speech Corpora	12
2.4.2 Wireless Channel Simulation	12
2.5 Summary	13
3 Dynamic Quantization I - Histogram-Based Quantization	15
3.1 Introduction	15
3.2 General Formulation of HQ	16
3.3 Histogram-Based Vector Quantization (HVQ)	18
3.4 Discussions about Robustness of HQ (and HVQ)	20
3.4.1 The Robust Nature of HQ	20

3.4.2	Comparison with Histogram Equalization (HEQ)	23
3.5	Experimental Results	23
3.5.1	HQ as a Feature Transformation Method	24
3.5.2	HQ as a Feature Quantization Method	27
3.5.3	Further Analysis of Bit Rates vs. SNRs for HQ as a Feature Quanti- zation Method	29
3.6	Summary	30
4	Joint Uncertainty Decoding (JUD) for HQ	33
4.1	Introduction	33
4.2	General Formulation of Uncertainty Decoding	34
4.3	Joint Uncertainty Decoding (JUD) for HQ	35
4.3.1	Quantization Error Uncertainty	35
4.3.2	Environmental Noise Uncertainty	36
4.3.3	Joint Uncertainty Decoding (JUD) for HQ	37
4.4	Histogram-Shift Compensation	38
4.5	Experimental Results	38
4.5.1	HQ and JUD for Robust Speech Recognition	38
4.5.2	HQ and JUD for Distributed Speech Recognition	39
4.6	Summary	43
5	Three-Stage Error Concealment (EC) for HQ-Based DSR Systems	45
5.1	Introduction	45
5.2	Stage 1 - Error Detection	46
5.3	Stage 2 - Erroneous Feature Vector Estimation	48
5.4	Stage 3 - Uncertainty Decoding	51
5.5	Three-Stage EC under the HQ Framework	51
5.6	Experimental Results	52
5.6.1	HQ-Based DSR over Wireless Channels with Transmission Errors, but without Error Concealment (EC)	53
5.6.2	HQ-Based DSR over Wireless Channels with Error Concealment (EC)	54
5.7	Summary	59
6	Dynamic Quantization II - Context-dependent Quantization	61
6.1	Introduction	61
6.2	Proposed Approach	62
6.2.1	Context-dependent Quantization	62
6.2.2	Context-dependent HQ	64
6.3	Experimental Results	66
6.3.1	Context-dependent HQ as a Robust Feature Transformation Method	66
6.3.2	Context-dependent HQ as a Feature Quantization Method for DSR	68
6.4	Summary	71

7	Application of Dynamic Quantization on image features for photograph retrieval	73
7.1	Introduction	73
7.2	Overview of the proposed approach	77
7.3	Probabilistic latent semantic analysis (PLSA)	77
7.4	Low-level image feature extraction	79
7.4.1	Dynamic color features from the images	79
7.4.2	Texture features from images	81
7.5	Document generation for photos	81
7.5.1	Image “terms” extraction and “Cohort Photos” selection from low-level image features	81
7.5.2	Construction of “Documents” with fused features for the photos	83
7.6	Latent semantic photo retrieval with fused image/speech/text features	84
7.7	Image Clustering	85
7.8	Preliminary Experimental Results	85
7.8.1	Photo archive	85
7.8.2	Dynamic color features across cameras	86
7.8.3	Latent semantic photo retrieval	86
7.8.4	Image clustering	89
7.9	Summary	89
8	Conclusions and Future Works	91
8.1	Conclusions	91
8.2	Future Works	92
	Bibliography	95



List of Tables

3.1	The averaged normalized distances between clean and corrupted speech features under different SNR values for HEQ and HQ (1-dim).	26
3.2	Recognition accuracies for feature quantization and compression with clean-condition training, averaged over all SNR values and noise types in sets A, B, and C for different bit rates (4.4 kbps to 2.7 kbps).	28
4.1	Averaged histogram shift for HQ under different SNR conditions.	37
4.2	Accuracies and error rate reductions for HQ alone (one-dimensional, 3.9 kbps) and HQ-s,n,q (with complete JUD) for different testing sets in Fig. 4.1(c).	39
4.3	Accuracies and error rate reductions for HEQ-ECIVQ and HQ-s,n,q (with complete JUD) at 4.4 kbps for different SNR values in Fig. 4.2(b).	42
5.1	Mutual information $I(s_t, s_{t-1})$ for SVQ and HQ.	51
6.1	Comparison of Transform coding (TC) and HQ-mmse, without and with GPRS transmission errors (TCg and HQ-mmseg) for different SNR values.	68
7.1	Average precision and rank for the first few irrelevant photos retrieved (with dynamic codewords)	88
7.2	Average precision and rank for the first few irrelevant photos retrieved (with fixed codewords)	88
7.3	Evaluation for Image clustering	89

List of Figures

2.1	The system of two-dimensional discrete cosine transform coding (2DDCT).	9
3.1	The general formulation of Histogram-based Quantization (HQ).	18
3.2	The concept of Histogram-based Vector Quantization (HVQ) using two dimensions.	19
3.3	Mismatch between the pre-trained fixed VQ codebook and the corrupted testing features.	21
3.4	The robust nature expressed in terms of HVQ.	22
3.5	Accuracies for MFCC baseline and those transformed by MVA filtering, PCA filtering, HEQ and HQ respectively under clean condition training: (a) averaged over all SNR values but separated for different types of noise; (b) averaged over all types of noise but separated for different SNR values; and (c) averaged over all types of noise and all SNR values for different testing sets.	24
3.6	Recognition accuracies for feature quantization and compression with clean-condition training: (a1)-(a4) averaged over all SNR values but separated for different types of noise at bit rates of 4.4 kbps to 2.7 kbps; (b1)-(b6) averaged over all types of noise but separated for different bit rates (4.4 kbps to 2.7 kbps) at different SNR values.	31
4.1	Performance improvements obtained by the various JUD approaches as compared to HQ alone: (a) averaged over all SNR values but separated for different noise types in sets A, B, and C; (b) averaged over all noise types but separated for each SNR value; and (c) averaged over all SNR values and noise types but separated into sets A, B, and C.	40
4.2	Comparison of different approaches discussed in this paper for DSR: (a) averaged over all SNR values but separated for different noise types in sets A, B, and C; (b) averaged over all noise types but separated for different SNR values; and (c) averaged over all SNR values and noise types but separated for sets A, B, and C.	41
4.3	Comparison of different approaches discussed in this paper for DSR (but without transmission errors) under different bit rates and SNR values: (a) clean, (b) 20 dB, (c) 15 dB, (d) 10 dB, (e) 5 dB, and (f) 0 dB.	42

5.1	(a) Recall and (b) Precision rates for error detection using SVQ with the conventional data consistency check and HQ with the HQ-based consistency check proposed here.	47
5.2	The three-stage error concealment (EC) framework.	52
5.3	Comparison of SVQ, HEQ-SVQ and HQ, and those with GPRS transmission errors (SVQg, HEQ-SVQg, HQg), averaged over all types of noise, but separated for each SNR value.	54
5.4	Comparison of SVQ, HEQ-SVQ and HQ with the percentage of words which were correctly recognized if without transmission errors, but incorrectly recognized after transmission.	55
5.5	Comparison of SVQg, TC-SVQg, HEQ-SVQg and HQg (all with GPRS transmission errors), for different bit rates and SNR values: (a) clean, (b) 20 dB, (c) 15 dB, (d) 10 dB, (e) 5 dB, and (f) 0 dB.	56
5.6	Comparison of SVQ under GPRS (SVQg), HEQ-SVQ under GPRS without and with repetition (HEQ-SVQg and HEQ-SVQgr), HQ under GPRS without and with EC techniques (HQg and HQgc): (a) averaged over all SNR values, but separated for different noise types in sets A, B, and C; (b) averaged over all types of noise, but separated for each SNR value; and (c) averaged over all SNR values and noise types but separated for sets A, B, C.	57
5.7	Comparison of SVQgr, TC-SVQgr, HEQ-SVQgr (all under GPRS with repetition), and HQgc (under GPRS with error concealment) for different bit rates and SNR values: (a) clean, (b) 20 dB, (c) 15 dB, (d) 10 dB, (e) 5 dB, and (f) 0 dB.	58
5.8	Comparison of HEQ-SVQ under GPRS without and with repetition, HQ under GPRS without and with EC, at traveling speeds of 3, 50, 100, and 250 km/hr: (a1)/(a2) for car/babble noise at 15 dB SNR; (b1)/(b2) for car/babble noise at 5 dB SNR; and (c1)/(c2) for car/babble noise averaged over all SNR values.	60
6.1	Context-dependent quantization with left and right context codewords m and n	65
6.2	Context-dependent Histogram-based Quantization (HQ).	66
6.3	Word accuracies for HEQ, HQ, HQ-cd and HQ-mmse under clean condition training: (a) averaged over all SNR values (20 dB to 0 dB) but separated for different types of noise; (b) averaged over all types of noise but separated for different SNR values; and (c) averaged over all types of noise and all SNR values (20 dB to 0 dB) for different testing sets.	67
6.4	Comparison of HEQ-SVQ, HQ, and HQ-mmse, and those with GPRS transmission errors (HEQ-SVQg, HQg, and HQ-mmseg): (a) averaged over all SNR values (20 dB to 0 dB) but separated for different types of noise; (b) averaged over all types of noise but separated for different SNR values; and (c) averaged over all types of noise and all SNR values (20 dB to 0 dB) for different testing sets.	69

7.1	The proposed approach: preparation phase includes document construction for each photo and PLSA model training for photo documents, while retrieval phase is based on PLSA.	78
7.2	PLSA-based retrieval model	79
7.3	Dynamic color features defined by histogram-based quantization.	80
7.4	Image feature quantization with fixed and dynamic codewords	82
7.5	Retrieved photos by the text query "Hanauma Bay"	87



This page intentionally left blank.



Chapter 1

Introduction

1.1 Background

With the rapid development of network and wireless technologies, people could access network content from anywhere at anytime via hand-held devices such as personal digital assistants (PDAs) or cellular phones. For these pocket sized devices with smaller screens and little keyboard, speech input would make it easier for users to interact with system in a natural manner. A wide variety of potential applications for automatic speech recognition (ASR) technologies have been highly anticipated [1]. But the recognition accuracy of ASR systems is always the core concern, which is very often seriously degraded by the mismatch between training and testing environments. This mismatch could come from the speaker difference (i.e. vocal tract length, dialect), or acoustic conditions (i.e. background noise, channel effects, room reverberation). Hence robustness for ASR technologies with respect to environmental disturbances is definitely a key issue when considering real world applications .

On the other hand, hand-held devices have limited computation resource, memory, transmission bandwidth, and battery energy. Distributing the speech recognition task across the network could become an attractive alternative [2]. The client-server framework

for Distributed Speech Recognition (DSR) has been widely accepted, in which speech features are extracted and compressed at hand-held clients, transmitted via wireless networks, and recognition is performed at the server [3]. However, recognition accuracy for DSR is inevitably degraded by environmental noise at the input, quantization distortion and transmission errors; these three sources of disturbances naturally mix up with each other and further complicate the problem. The mismatch between the pre-trained VQ codebook and the constantly changing environmental conditions at the moving client increases the quantization distortion. Also, speech features corrupted by noise are more sensitive to transmission errors. Many approaches extract robust feature parameters before quantization at the client to reduce codebook mismatch in DSR and make features less sensitive to transmission errors. However, the meager computational resources available on hand-held devices should be considered for many useful advanced robust approaches [4]. The reduction of floating-point calculation to fixed-point implementation has to be considered for filtering-based robustness approaches [5, 6, 7, 8]. Principle component analysis (PCA), Linear discriminant analysis (LDA) filters are very successful data-driven robustness approaches which aim to derive optimal sets of time filtering coefficients for a specific recognition task and environment according to some optimization criterion [6, 8]. The attractive performance of these data-driven methods may not be anticipated when the environmental noise and acoustic conditions are unknown and changing at the moving client. Feature normalization techniques may not be applied on the client end either under the recommendation of a standardized VQ codebook [9, 10].

In this dissertation, we aim at solving the problems in DSR, including environmental noise, quantization distortion, and transmission errors. The proposed method can also be used as a robust feature transformation method for robust speech recognition outside of a DSR environment. The primary results obtained in this thesis is presented in the following section.

1.2 Primary Achievements of this Dissertation

In this dissertation, we propose a dynamic quantization method for distributed and robust speech recognition. Both partition cells and representative values could be dynamically defined based on local signal statistics. A Joint Uncertainty Decoding (JUD) approach is further developed to consider the uncertainty caused by both environmental noise and quantization errors. A three-stage error concealment (EC) framework is also developed to handle transmission errors. These approaches are presented in Chapters 3 to 6. The concept of dynamic quantization could be easily applied on image features for photograph retrieval as described in Chapter 7.

We first review some basic feature quantization and error concealment techniques in Chapter 2, including the conventional split vector quantization (SVQ), two-dimensional discrete cosine transform coding (2D-DCT), error detection and correction methods, erroneous feature estimation techniques, and reliability estimation in Viterbi Decoding. We also present the speech/noise corpora used in this dissertation for experiments.

In Chapter 3, Histogram-based Quantization (HQ) is proposed to solve the many related problems mentioned above. HQ is a novel approach in which the partition cells for quantization are dynamically defined by the histogram or order statistics of a segment of recent past samples of the parameter to be quantized. It is actually a dynamic quantization, completely based on the local statistics of the signal, not on any distance measure, nor directly related to any pre-trained codebook. On one hand, in the case of DSR, many of the above-mentioned problems that arise from a fixed pre-trained VQ codebook in conventional DSR framework are shown to be solved to a good extent with this new approach, because the quantization is dynamic and not solely based on a fixed pre-trained codebook at all; therefore the mismatch between the corrupted feature vectors and a fixed pre-trained codebook is reduced. This concept of HQ is then further extended to Histogram-based Vector Quantization (HVQ). On the other hand, HQ is also shown to be useful as a good approach

for robust feature transformation, which can produce more robust features, because most of the noise disturbances can be automatically absorbed by the dynamic histogram. This robust nature of HQ against environmental noise is extensively explored and analyzed, including considering quantization resolution (or required bit rate), noisy environment and transmission conditions.

In Chapter 4, Joint Uncertainty Decoding (JUD) is developed to be applied with HQ for improved recognition accuracy, and the approach was evaluated for both cases of robust speech recognition and DSR. For both robust and/or distributed speech recognition, feature vectors corrupted by environmental and/or quantization errors used at the recognizer can be viewed as random vectors with uncertainty. Uncertainty decoding approaches have been proposed to consider such uncertainty [11, 12, 13, 14, 15, 16], including handling those produced by environmental noise [11, 13, 14] and estimating the uncertainty generated in the quantization process [15, 16]. However, in DSR with environmental noise, it is naturally better to consider environmental noise and quantization errors jointly. But this is difficult because environmental noise is hidden in the quantized codewords, or mixed with quantization errors. In Joint Uncertainty Decoding (JUD), we jointly consider the uncertainty caused by both the environmental noise and the quantization errors in Viterbi decoding under the framework of HQ.

Except for quantization errors and environmental noise, transmission errors caused by communication channel is also a key issue in DSR. In Chapter 5, a three-stage error concealment (EC) framework based on Histogram-based Quantization (HQ) for DSR is proposed, in which noisy input speech is assumed and both the transmission errors and environmental noise are considered jointly. The first stage detects the erroneous feature parameters at both the frame and subvector levels. The second stage then reconstructs the detected erroneous subvectors by MAP estimation, considering the prior speech source statistics, the channel transition probability, and the reliability of the received subvectors.

The third stage then considers the uncertainty of the estimated vectors during Viterbi decoding. At each stage, the error concealment (EC) techniques properly exploit the inherent robust nature of HQ.

In addition to the dynamically defined partition cells in HQ for different environments, in Chapter 6, we propose a new concept of context-dependent quantization, in which the representative parameter (whether a scalar or a vector) for a quantization partition cell is not fixed, but depends on the signal context on both sides. The signal context dependencies can be trained with a clean speech corpus or estimated from a noisy speech corpus. This results in a much finer quantization based on local signal characteristics, without using any extra bit rate. This approach is equally applicable to all (scalar or vector) quantization approaches, and can be used either for signal compression in DSR or for feature transformation in robust speech recognition. In the latter case, each feature parameter is simply transformed into its representative parameter after quantization. This concept is integrated with HQ, and both partition cells and representative values of the context-dependent HQ is dynamic defined based on local statistics.

The concept of dynamic quantization could be used in other applications. In Chapter 7, we apply dynamic quantization on image features for photograph retrieval. The partitions of color space are dynamically defined based on the histogram of photos taken from each camera. Quantization with dynamic partition cells reduces the mismatch of pixel value distributions between different cameras and photos taken from different cameras are more easily retrieved. For the quantization of color histogram features and texture features, we use dynamic representative values to preserve discriminative information. For each photo, different sets of features are considered with difference importance to select representative codeword. In this way, the photo difference in the more discriminative feature dimension could be preserved well in the quantization process.

At last, we conclude this thesis in Chapter 8, by summarizing the works that

we have accomplished. There are still several issues regarding to dynamic quantization techniques that we have not been able to investigate. These issues will be discussed in the future works in Chapter 8.

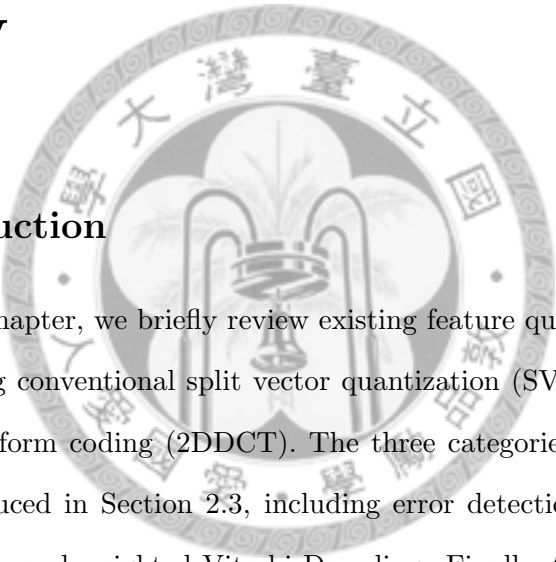


Chapter 2

Preliminaries – Background

Review

2.1 Introduction



In this chapter, we briefly review existing feature quantization approaches in section 2.2, including conventional split vector quantization (SVQ) and two-dimensional discrete cosine transform coding (2DDCT). The three categories of error concealment techniques are introduced in Section 2.3, including error detection and correction, erroneous feature estimation, and weighted Viterbi Decoding. Finally, the speech corpora and wireless channel simulation conditions used in this dissertation for experiments are depicted in Section 2.4.

2.2 Review of Existing Feature Quantization Approaches

2.2.1 Split Vector Quantization (SVQ)

Split Vector Quantization (SVQ) has been recommended by the ETSI-DSR standard[17]. The ETSI-DSR standard defines a feature extraction front-end and an encoding scheme for

compressing speech features. The feature extraction front-end generates a 14-element vector consisting of 13 cepstral coefficients (C1-C12 and C0) and log Energy. The feature vector is directly quantized with a split vector quantizer. The 14 coefficients are grouped into 7 pairs, and each pair is quantized using its own VQ codebook. The VQ codebook is pre-trained and fixed for each pair. The codebook size is 64 for the first 6 pairs and 256 for the pair C0 and logE. The closest VQ centroid is found using a weighted Euclidean distance to determine the index, and the weight matrix is identity for the first 6 pairs (C1C2,..., C11C12). For the pair C0 and logE, two sets of weight matrix are defined for different speech sampling rate.

Each feature vector is quantized to 44 bits via SVQ. Two of the quantized 10 ms mel-cepstral frames are grouped together as a pair. A 4-bit CRC is calculated on the frame-pair and is appended to it, resulting in a 92-bit long frame-pair packet. These packets are concatenated into a bit-stream for transmission via a GSM channel with an overall data rate of 4.8 kbps.

2.2.2 Transform Coding

The above standard compression method, SVQ, is an intra-frame vector quantization. This intra-frame compression is not very effective in bit rate because in the feature extraction front-end the transformation from mel-filter bank output to MFCC lets the MFCC coefficients uncorrelated. On the other hand, there is high correlation in consecutive frame because of the overlap of processing window in the front-end processing. The transmission rate could be further reduced if inter-frame correlation could be properly utilized in the quantization process [18]. Also, vector quantization performed in a transformed domain (obtained with transforms such as Discrete Cosine Transform (DCT) [19, 20, 21] has been shown to be able to efficiently improve the desired robustness for feature vectors under environmental disturbances.

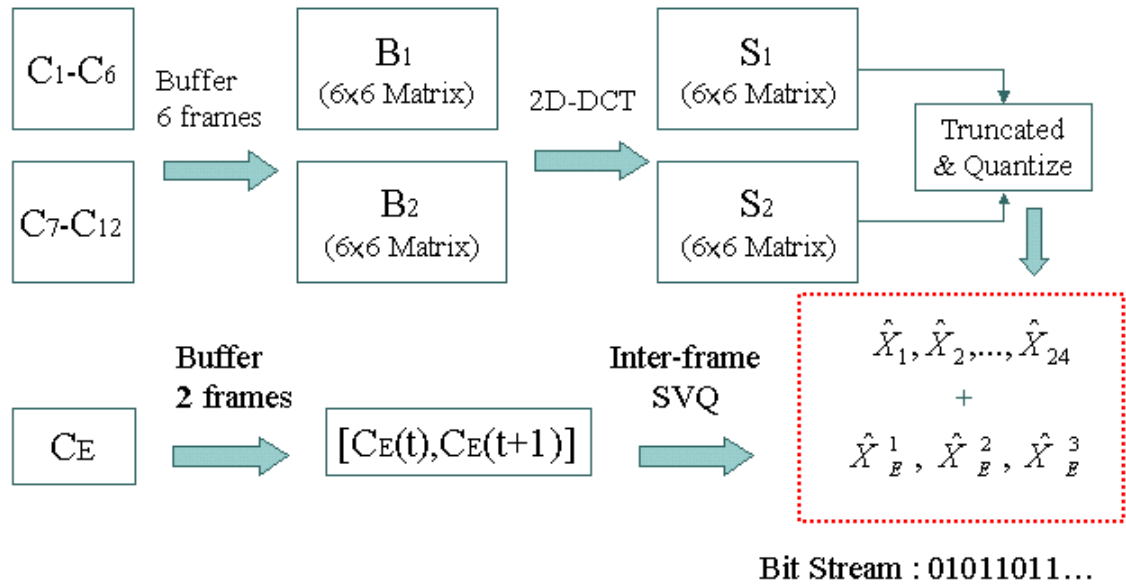


Figure 2.1: The system of two-dimensional discrete cosine transform coding (2DDCT).

The two-dimensional discrete cosine transform coding (2D-DCT) method has been popularly used in image compression. This 2D-DCT scheme was modified for the quantization of speech features [20, 21]. As shown in Fig. 2.1, the input features are MFCC coefficients (i.e., C_1 to C_{12} plus log-energy). First, buffer 6 frames to form 2 ($6 * 6$) MFCC matrix (B_1 and B_2), and then perform 2D-DCT on them to get 2 transformed matrix (S_1 and S_2). Because the coefficients in the first 2 columns are often with high energy, we can truncate the last 4 columns and only reserve the first 2 columns. Then, we perform scalar quantization on the reserved 24 coefficients, and transmit these bit patterns over wireless channel. Finally, the log-energy coefficients are quantized by inter-frame VQ method. Also, an iterative bit allocation algorithm was carefully designed to make use of every bit transmitted in an efficient way. [21]. Graceful degradation of recognition performance is achievable at a bit rate of 3.4 kbps.

2.3 Review of Existing Error Concealment Approaches

Various packet loss compensation schemes have been proposed for audio streaming applications [22] to reconstruct the time-domain signal. However for DSR applications the challenge is to maintain good speech recognition performance. Various error concealment (EC) techniques have been proposed to handle transmission errors problem in DSR. In section 2.3.1, we would describe the first group of techniques, which aims to reduce transmission errors through error detection and correction [17, 23, 24]. The second sets of techniques are presented in 2.3.2, which reconstructs the feature vectors by estimating the erroneous sub-vectors [25, 26, 27, 28, 29]. The third set of methods consider the reliability of the estimated vectors at the decoding stage [30, 31, 32], as will be described in section 2.3.3.

2.3.1 Error detection and correction

The ETSI standard uses a 4-bit cyclic redundancy check (CRC) to a pair of MFCC vectors to detect transmission errors [17]. When a bit error being detected, the pair of MFCC vectors is declared lost. To avoid consecutive frame errors, this CRC-based detection is modified to encode MFCC vectors individually using the 4-bit CRC [24]. This reduces the number of consecutive frame errors but it requires additional CRC bits. Channel coding and interleaving is very helpful for reducing the consecutive bit errors [27, 37]. Reed-Solomon coding is one of the forward error correction methods, and it is very effective in protecting feature vector stream from channel errors [23]. In particular, using unequal amounts of error protection is helpful for minimizing the overall word error rate as channel conditions becoming worsen. A further method investigates that bit errors in feature vectors, which cause incorrect observation probabilities, have more serious influence on recognition results than the loss of feature vector [30]. As such a CRC block code is proposed which varies the level of protection according to the channel conditions in an effort to increase the error

detection ability for bit errors [30].

2.3.2 Erroneous Feature Estimation

The most simple way to deal with lost vectors is to splice together the sequence of received vectors and input these into the speech recognizer [25]. A better alternative is to repeat the vector received immediately before packet loss. This ensures that the timing of the feature vector stream is maintained and adds no delay to the system. A similar scheme is to estimate lost vectors by duplicating the nearest correctly received vector either before or after the loss [26]. There are simple techniques which provide estimates of the static and/or dynamic component of lost feature vectors. Better methods include interpolation of the most recently received vector in the event of loss [27, 28]. Estimation of lost vectors by using speech prior probabilities derived from VQ codebooks has also been shown effective in achieving performance gains [29].

2.3.3 Weighted Viterbi Decoding

The final set of error concealment methods aims to minimize the degradation of recognition performance caused by transmission errors by modifying the decoding process of the recognizer. With recognition on the server side, the channel characteristics influences the reliability of the decoded features. When channel characteristics get worse, one can no longer guarantee the confidence in the decoded feature. The weighted Viterbi recognizer, presented in [38], modifies the Viterbi algorithm to take into account the reliability in the decoded feature. In this way, more reliable feature vectors are emphasized in the decoding process. Scaling the observation probability of a restored feature vector according to its reliability gives increased robustness over just modifying the features. The reliability used in weighted Viterbi decoding could be estimated in terms of its temporal distance from a correctly received vector [30, 31], or based on the soft channel output or the speech

characteristics [32, 33, 34]. There are several systems deal with lost vectors entirely at the decoding stage through missing feature theory [39, 40].

2.4 Experimental environments

2.4.1 Speech Corpora

All the experiments reported in this paper were conducted on the AURORA 2 testing environment [41] based on a corpus of English connected digit strings. Two training conditions (clean-condition and multi-condition) and three testing sets (sets A, B, and C) were defined in AURORA 2. Both clean and noisy speech signals were prepared by filtering the TI database 8 (both training and testing) using a telephone-bandwidth bandpass filter. The testing set A included four types of noise which were used in the multi-condition training (subway, babble, car and exhibition), while the testing set B included another four types of noise not used in the multi-condition training (restaurant, street, airport and train station). The testing set C was filtered with a MIRS (Modified Intermediate Reference System, which simulates the band-pass filtering [300-3400 Hz] behavior of the telephone channels in the public switched telephone networks [PSTN]) characteristic filter [41, 58] before adding two additive noise types (subway in set A and street in set B). In all sets A, B, and C, the signal-to-noise ratio (SNR) tested ranged from 20 to -5 dB. The MFCC extraction follows the WI007 front-end [41] defined in AURORA 2 with frame length 25 ms and frame shift 10 ms, which gives 13 coefficients (C1-C12 and log energy) to be used to obtain the delta and delta-delta features together for recognition.

2.4.2 Wireless Channel Simulation

General Packet Radio Service (GPRS) was chosen in this research as an example for wireless channels in the experiments; GPRS was developed by ETSI based on a packet switching framework to enhance the GSM system. GPRS shares the GSM frequency bands

and uses several properties of the physical layer of the GSM system. It includes four different error control coding schemes, CS1-CS4, each with a different code rate.

The GPRS simulation software used in the tests described here was developed by the Wireless Communication Laboratory of National Taiwan University [32], in which all complicated transmission phenomena have been carefully simulated in detail, such as the propagation model, multi-path fading, Doppler spread, etc. The GPRS simulator considers both large-scale fading (slow fading) and small-scale fading (fast fading) when the client is travelling. Large-scale fading is caused by diffraction and shielding phenomena due to terrain variation (e.g. reflections, refractions and diffractions of the signal from buildings, trees, rocks). The large-scale fading results in relatively slow variations in the mean signal power over distance. The large-scale fading is modeled as a log-normally distributed random variable (with a zero dB mean and a standard deviation of 4 to 10 dB) in our experiments. The experimental results presented below are based on the following simulation configurations: typical urban (TU, an environment more frequently encountered with a more severe fading problem), the client traveling at speeds of 3, 50, 100, 250 km/hr, single antenna, hard decision at the receiver, and CS4 (i.e., without any protection) coding scheme, which corresponds to a transmission bit error rate of 5.3% for a client traveling at a speed of 3 km/hr.

2.5 Summary

In this chapter, we briefly reviewed several existing feature quantization approaches and different kinds of error concealment techniques. These two distance-based quantization method would be implemented to compare with the proposed dynamic quantization in chapter 3. In chapter 5, the three-stage error concealment techniques would integrate the idea of these three categories of concealment method. Finally, the speech corpora and wireless channel characteristics used in this dissertation were also addressed.

This page intentionally left blank.



Chapter 3

Dynamic Quantization I - Histogram-Based Quantization

3.1 Introduction

Various schemes for compression of ASR features have been proposed in recent years. Distance-based vector quantization (VQ) has been found very useful for clean speech and/or matched VQ codebook conditions [16, 42] and Split Vector Quantization (SVQ) has been recommended by the ETSI standard[17]. But environmental noise and quantization distortion naturally tend to jointly degrade recognition performance. The quantization process may increase the distance between clean and noisy features, and environmental noise may also move the feature vectors to a different quantization cell. The quantization distortion is actually related to the bit rates, which is another key parameter in DSR. The higher bit rate required for lower quantization distortion naturally becomes another difficult issue for transmission. Vector quantization or SVQ performed in a transformed domain (obtained with transforms such as Discrete Cosine Transform (DCT) [19, 20, 21] or Histogram Equalization (HEQ) [43, 44, 45]) has been shown to be able to efficiently improve

the desired robustness for feature vectors under environmental disturbances; differential encoding of transformed coefficients was shown to be very helpful as well [18]. However, while all these approaches have proven more robust than the conventional SVQ (i.e. performing SVQ on MFCC directly), they are still based on VQ or SVQ, which are distance- and codebook-based. As long as the quantization is based on a pre-trained codebook and some distance measure with the codebook, the mismatch between VQ codebook and testing feature vectors under lower SNR conditions remains a difficult problem.

In this chapter, Histogram-based Quantization is proposed to solve the above problems. Below in Section 3.2 we introduce the basic idea and formulation of Histogram-based Quantization (HQ). In section 3.3, the one-dimensional HQ is extended to Histogram-based Vector Quantization (HVQ). Then, the robust nature of dynamic quantization is analyzed in detail in section 3.4. Experimental results are offered in Sections 3.5, with the summary finally given in Section 3.6.

3.2 General Formulation of HQ

The concept of HQ is to perform quantization of a feature parameter y_t at time t based on the histogram or order statistics of that feature parameter within a moving segment of the most recent past T samples, $[y_{t-T+1}, \dots, y_{t-1}, y_t] \triangleq Y_{t,T}$, up to the time t being considered [46]. As shown in Fig. 3.1, the values of these T parameters in $Y_{t,T}$ are sorted to produce a time-varying cumulative distribution function $C(v)$, or histogram, which changes for every time instant t , where $C(v_0) = b_0 = 0$ and $C(v_N) = b_N = 1$, v_0 and v_N are respectively the minimum and maximum values within $Y_{t,T}$. Also shown in Fig. 3.1, N partition cells, $\{D_i = [b_{i-1}, b_i], i = 1, 2, \dots, N\}$, together with their corresponding representative values, $\{\bar{z}_i, i = 1, 2, \dots, N\}$, are defined on the vertical scale $[0, 1]$, which are derived from a standard Gaussian $N(0, 1)$ with cumulative distribution $C_0(v)$ via the Lloyd-Max algorithm [47, 48]. Note that the boundaries $\{b_i, i = 0, 1, 2, \dots, N\}$ on the vertical

scale can be either uniformly or non-uniformly distributed [46]. In the case of non-uniform quantization, the Lloyd-Max algorithm can be performed with respect to any distribution, including the distribution of training sets. Since different training sets may have different distributions, we performed the Lloyd-Max algorithm based on uniform, Laplacian and Gaussian distributions in the preliminary experiments. The best performance was obtained with Gaussian distribution under noisy environments, probably because the distribution of feature parameters under noisy environments on the vertical scale is closer to a Gaussian distribution. Using the dynamic histogram $C(v)$ constructed with $Y_{t,T}$, these partition cells on the vertical scale, $\{D_i, i = 1, 2, \dots, N\}$, are then transformed to the horizontal scale to be the N partition cells $[v_{i-1}, v_i], i = 1, 2, \dots, N$ on the horizontal scale for the quantization of y_t , where $C(v_i) = b_i$. In other words, the partition cell $[v_{i-1}, v_i]$ on the horizontal scale is obtained from the partition cell $D_i = [b_{i-1}, b_i]$ on the vertical scale via the dynamic histogram $C(v)$. Thus the partition cell $[v_{i-1}, v_i]$ on the horizontal scale is dynamic. However, the representative values $\{z_i, i = 1, 2, \dots, N\}$ for these partition cells $\{[v_{i-1}, v_i], i = 1, 2, \dots, N\}$ on the horizontal scale are fixed, and are transformed from the representative values $\{\bar{z}_i, i = 1, 2, \dots, N\}$ previously obtained on the vertical scale by the histogram $C_0(v)$ of the standard Gaussian.

The above formulation indicates that HQ is based on a hidden codebook $\{(D_i, \bar{z}_i), i = 1, 2, \dots, N\}$ derived from a standard Gaussian on the vertical scale, which is then transformed by a dynamic histogram $C(v)$ into time-varying partition cells $[v_{i-1}, v_i]$, and by a fixed histogram $C_0(v)$ into the fixed representative values z_i , both on the horizontal scale. The quantization here is then similar to all conventional quantization processes, in that it is a mapping relation which maps the present parameter y_t to a fixed representative value z_i , if y_t is within the partition cell $[v_{i-1}, v_i]$, except that this partition cell is dynamically defined,

$$y_t \rightarrow z_i, \quad \text{if } b_{i-1} < C(y_t) < b_i, \text{ or } v_{i-1} < y_t < v_i,$$

$$C(v_{i-1}) = b_{i-1}, C(v_i) = b_i, i = 1, 2, \dots, N. \quad (3.1)$$

Note that the quantization codebook here includes a set of dynamic partition cells $\{[v_{i-1}, v_i], i = 1, 2, \dots, N\}$ and a set of fixed representative values $\{z_i, i = 1, 2, \dots, N\}$. It will be shown below that many practical problems mentioned previously can be automatically solved to a good extent in this way. Also, although here HQ is a quantization process, it can also be used as a feature transformation process offering the desired robustness as will also be discussed below, in which each parameter y_t is transformed to its representative value z_i for the corresponding partition cell.

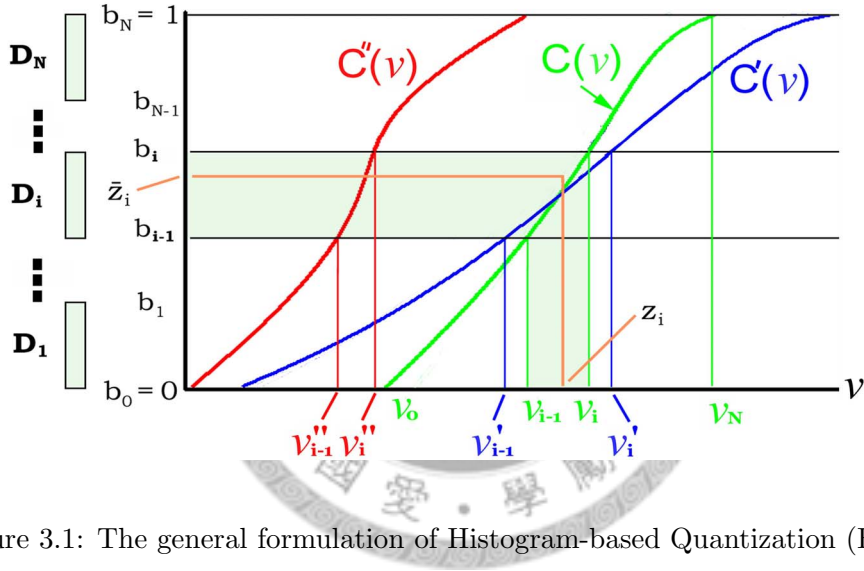


Figure 3.1: The general formulation of Histogram-based Quantization (HQ).

3.3 Histogram-Based Vector Quantization (HVQ)

The above general formulation of one-dimensional HQ in Fig. 3.1 can be easily extended to HVQ with more than one dimension. Consider SVQ as an example[17], in which two MFCC parameters (e.g. c_1 and c_2) can be quantized jointly by a two-dimensional VQ codebook. Extending from the one-dimensional HQ mentioned above, a moving segment of the most recent past T samples of the first parameter $y_t^{(1)}$ up to time

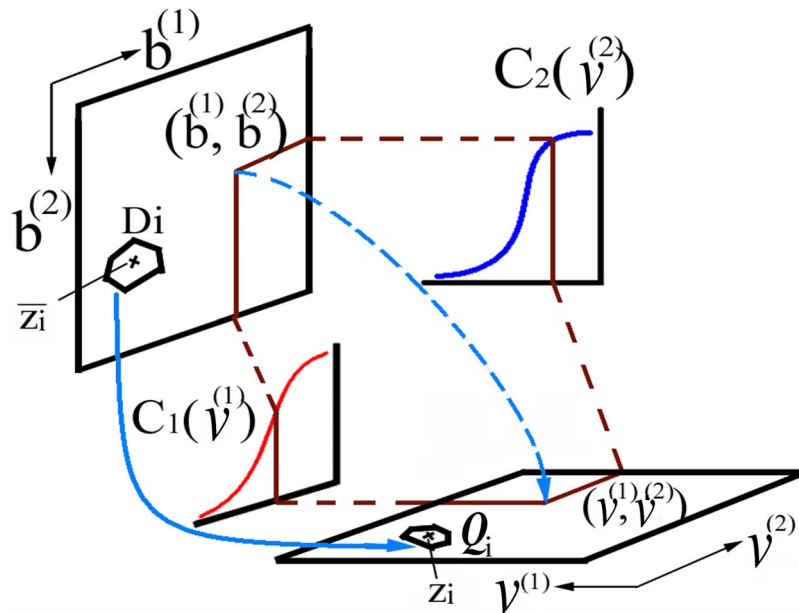


Figure 3.2: The concept of Histogram-based Vector Quantization (HVQ) using two dimensions.

t , $[y_{t-T+1}^{(1)}, \dots, y_{t-1}^{(1)}, y_t^{(1)}] \triangleq Y_{t,T}^{(1)}$, gives a histogram $C_1(v^{(1)})$ for $y_t^{(1)}$, and a similar segment of the past T samples of the second parameter $y_t^{(2)}$ up to time t , $Y_{t,T}^{(2)}$, gives another histogram $C_2(v^{(2)})$ for $y_t^{(2)}$. The formulation below is exactly the same as the one-dimensional HQ in Fig. 3.1, except that here both the vertical and horizontal axes are no longer one-dimensional axes, but are extended to vertical and horizontal two-dimensional planes as shown in Fig. 3.2. On the vertical plane with coordinates $(b^{(1)}, b^{(2)})$, we have a two-dimensional hidden codebook $\{(D_i, \bar{z}_i), i = 1, 2, \dots, N\}$, which is derived from a bi-variate standard Gaussian via the LBG algorithm [60]. Every point $(b^{(1)}, b^{(2)})$ on this plane is then transformed by the above-mentioned dynamic histograms $C_1(v^{(1)}), C_2(v^{(2)})$ back to a point $(v^{(1)}, v^{(2)})$ on the horizontal plane, where $C_1(v^{(1)}) = b^{(1)}, C_2(v^{(2)}) = b^{(2)}$. The set of all these points $(v^{(1)}, v^{(2)})$ on the horizontal plane transformed from those points $(b^{(1)}, b^{(2)})$ on the vertical plane in a certain partition cell D_i then forms the dynamic partition cell Q_i on

the horizontal plane:

$$\begin{aligned} (v^{(1)}, v^{(2)}) \in Q_i, & \text{ if } (b^{(1)}, b^{(2)}) \in D_i, \\ C_1(v^{(1)}) = b^{(1)}, C_2(v^{(2)}) = b^{(2)}, & i = 1, 2, \dots, N. \end{aligned} \quad (3.2)$$

On the other hand, the representative points \bar{z}_i for each partition cell D_i on the vertical plane are similarly transformed back to the fixed representative points z_i on the horizontal plane, except that the transformation is performed by two fixed histograms $C_0(v^{(1)})$, $C_0(v^{(2)})$, both derived from a one-dimensional standard Gaussian. The quantization here is a mapping relation just as one-dimensional HQ in Eq. (3.1), which maps the present parameter set $(y_t^{(1)}, y_t^{(2)})$ to a representative value z_i for the dynamically defined partition cell Q_i ,

$$\begin{aligned} (y_t^{(1)}, y_t^{(2)}) \rightarrow z_i, & \text{ if } (C_1(y_t^{(1)}), C_2(y_t^{(2)})) \in D_i, \\ & \text{ or } (y_t^{(1)}, y_t^{(2)}) \in Q_i, i = 1, 2, \dots, N. \end{aligned} \quad (3.3)$$

Based on the above, the two-dimensional HVQ can be performed dynamically on the $(v^{(1)}, v^{(2)})$ plane. For the present parameter pair $(y_t^{(1)}, y_t^{(2)})$ at time t , the two dynamic histograms $C_1(v^{(1)})$ and $C_2(v^{(2)})$ based on $Y_{t,T}^{(1)}$ and $Y_{t,T}^{(2)}$ give a point $(C_1(y_t^{(1)}), C_2(y_t^{(2)}))$ on the vertical plane. The partition cell D_i on the vertical plane to which this point belongs then determines the partition cell Q_i and representative point z_i on the horizontal plane.

3.4 Discussions about Robustness of HQ (and HVQ)

Conventionally, feature quantization is for data compression and robust features are for handling noise disturbances. The proposed HQ, however, includes the desired robustness in the quantization process.

3.4.1 The Robust Nature of HQ

Consider the conventional SVQ as in Fig. 3.3: the mismatch between the pre-trained fixed VQ codebook and the current corrupted testing features may significantly

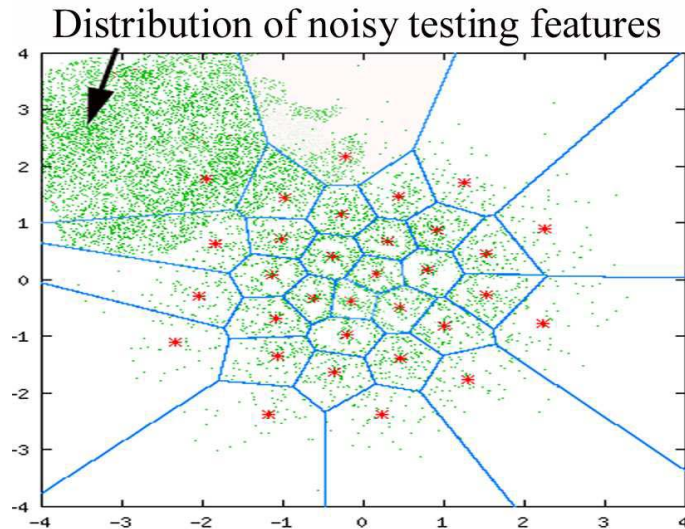


Figure 3.3: Mismatch between the pre-trained fixed VQ codebook and the corrupted testing features.

increase quantization distortions. With the proposed HQ, however, the actual partition cells are dynamically adjusted according to local statistics. For example, as shown in Fig. 3.1, $C(v)$ may be changed to $C'(v)$ when disturbances are encountered. The partition cell on the horizontal scale for the disturbed parameter y'_t may also be changed to $[v'_{i-1}, v'_i]$, where $C'(v'_{i-1}) = b_{i-1}$ and $C'(v'_i) = b_i$, which can be quite different from $[v_{i-1}, v_i]$. Nevertheless, the partition cell D_i and the corresponding representative value z_i for y'_t may remain unchanged as long as $v'_{i-1} < y'_t < v'_i$, since D_i is fixed on the vertical scale, while the disturbances from y_t to y'_t are on the horizontal scale, and z_i is fixed on the horizontal scale. Since the actual partition cells are no longer fixed as in conventional SVQ methods, the codebook mismatch problem mentioned above can thus be avoided to some extent. In other words, HQ is based on the partition cells D_i fixed on the vertical scale and the dynamic histogram $C(v)$, and is therefore less sensitive to disturbances on the horizontal scale: disturbances on the horizontal scale are actually absorbed by the dynamic histogram to a certain degree. When a segment of parameters $Y_{t,T}$ are corrupted by small disturbances, all individual values may be changed ($C(v)$ is disturbed into $C'(v)$), but the order statistics

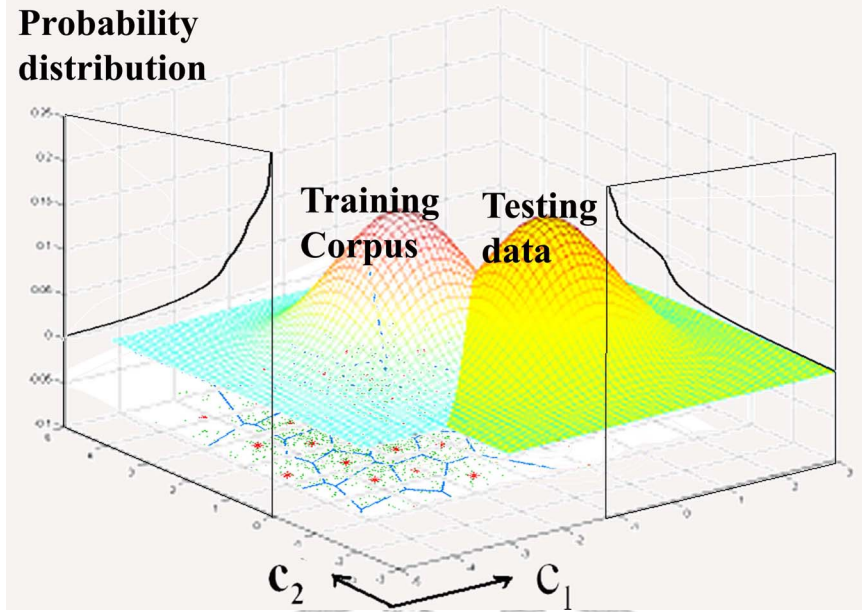


Figure 3.4: The robust nature expressed in terms of HVQ

which produce the partition cells on the horizontal scale may remain similar, and the representative values z_i remain fixed; therefore the changes to the quantization results may be very limited. Such robustness is obtained by local order statistics for the most recent past values of feature parameter. This is why HQ is able to handle various noise conditions as will be shown in the experiments presented below.

The robust nature of HQ can be better visualized for the case of HVQ mentioned above as shown in Fig. 3.4. The distribution of (c_1, c_2) for the testing features may be quite different from that of the VQ training corpus. This mismatch is the source of the primary difficulties in the conventional VQ approaches with fixed codebooks. With the proposed HQ approach, however, we no longer rely on a fixed codebook on the (c_1, c_2) plane, but instead we let the quantization codebook (or look-up table) move with the testing data distributions, because the quantization is now based on the distribution or histogram on the vertical scale. As can be found in Fig. 3.4, the shift of vectors (c_1, c_2) due to disturbances becomes almost irrelevant to the quantization process.

3.4.2 Comparison with Histogram Equalization (HEQ)

The popularly-used HEQ equalizes the cumulative distributions (or histograms) of both the training and testing feature parameters in each temporal span, and has been shown to produce very robust features for recognition [43, 44, 45]. HQ actually borrows the concept from HEQ. The experiments below will show that HQ can be used as an attractive feature transformation approach for robustness purposes as well, and it even performs better than HEQ. It is important to explain why. HEQ actually performs point-to-point feature transformation based on the order statistics, which can absorb the small disturbances to a good degree, although some residual disturbances inevitably remain because the point-based order statistics are in any case more or less disturbed. Quantile-based HEQ [49] performs a piecewise-linear approximation of HEQ. It reduces the computation complexity for histogram estimation, but does not change the point-based nature of the transformation. HQ, on the other hand, performs the transformation block by block; therefore, the small disturbances within each block (D_i in Fig. 3.1) are absorbed by the block-based order statistics. The block-based order statistics certainly introduce uncertainty as well, but with the proper choice of the number of quantization levels N or the block size, this uncertainty may be compensated for by the stochastic nature of the Gaussian mixtures in the HMMs. HEQ can be considered the limiting case of HQ when the number of quantization levels N becomes infinite. As will be shown below, the recognition performance certainly depends on the value of N considering the noise conditions and so on, but N being infinite is not necessarily the best.

3.5 Experimental Results

All the experiments reported here were based on order statistics over segments of most recent past parameter values as mentioned in section 3.2, so there was no time delay. Better results were obtainable if this no-delay condition was removed.

3.5.1 HQ as a Feature Transformation Method

In the first set of experiments, we considered the case of robust speech recognition apart from the DSR environment, in which one-dimensional HQ was used as a feature transformation technique, that is, each feature parameter y_t is transformed to the representative value z_i for the corresponding partition cell as in Eq. (3.1) to be used for recognition.

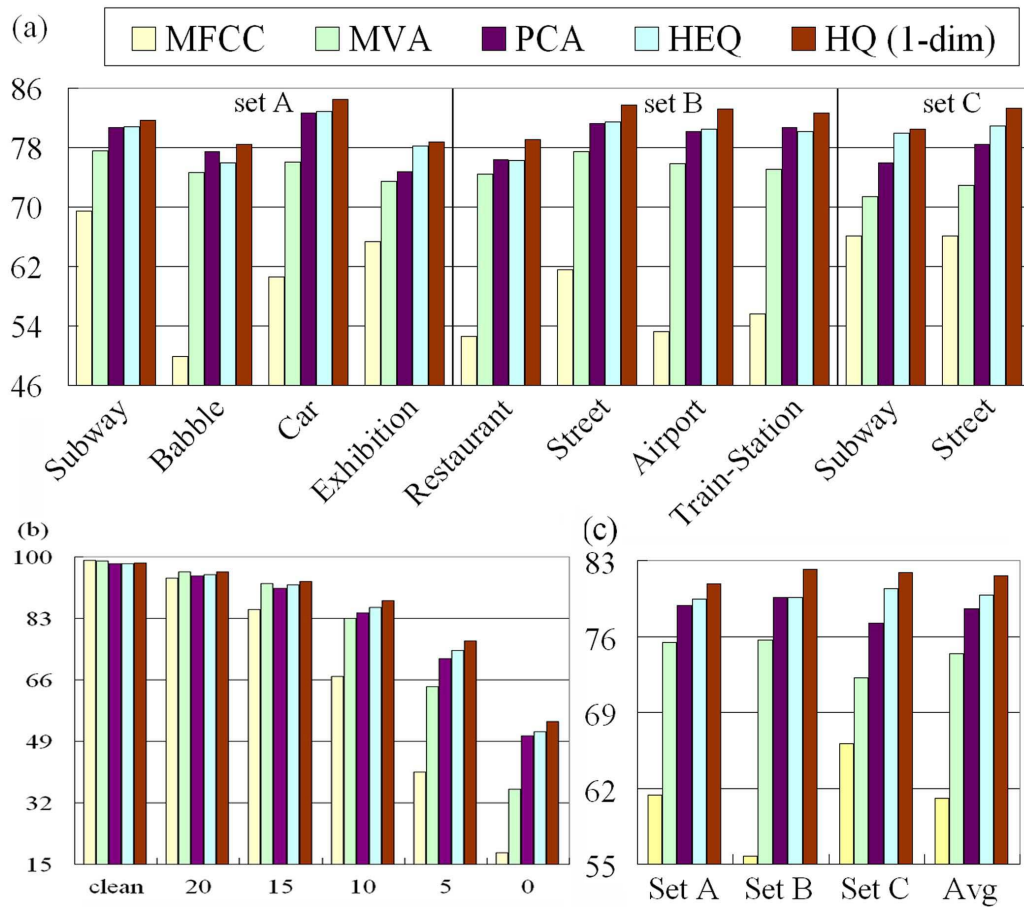


Figure 3.5: Accuracies for MFCC baseline and those transformed by MVA filtering, PCA filtering, HEQ and HQ respectively under clean condition training: (a) averaged over all SNR values but separated for different types of noise; (b) averaged over all types of noise but separated for different SNR values; and (c) averaged over all types of noise and all SNR values for different testing sets.

The results are shown in Fig. 3.5(a), (b), and (c). The recognition accuracies for baseline experiments with original MFCC features, compared to those with MFCC

parameters filtered by the MVA filter (mean and variance normalization followed by Auto-Regression Moving-Average (ARMA) filtering) [56] and the Principal Component Analysis (PCA) filter derived [6, 57], as well as transformed by the well-accepted HEQ [43, 44, 45], and the proposed one-dimensional HQ are respectively shown in Fig. 3.5 under clean-condition training for (a) averaged over all SNR values but separated for different types of noise, (b) averaged over all types of noise but separated for different SNR values, and (c) averaged over all types of noise and all SNR values for testing sets A, B, and C, respectively. Here the order of the MVA filter was $M=2$, the PCA filter was performed with filter length $L=15$, and HEQ was performed in exactly the same way as HQ, based on a moving segment of the most recent T past parameters, and the same value of $T=100$ (or one second) was used for all experiments for both HEQ and HQ. It has been verified that long term features derived from one second time interval carry important speech information [59].

Many observations can be made here. First, it is clear that HQ (the last bar) significantly improved the performance as compared to the baseline MFCC (the first bar) for all testing sets, all SNR values (except for the clean speech case), and all noise types. For example, from Fig. 3.5(a), it can be observed that for speech-like noise such as babble or restaurant noise, the MFCC baseline accuracy (around 50%) was much lower as compared to most other noise types (around 60% or more). HQ was able to absorb the speech-like variation and improved the performance in such a way that the results for different noise types were not only much higher, but also were more similar to each other (around 80%). As another example, in Fig. 3.5(b) the recognition accuracy of HQ was 87.88% as compared to MFCC baseline 66.95% at 10 dB SNR. The improvements became even more significant for lower SNRs. Second, HQ proposed here performed consistently better than MVA, PCA, and HEQ compared here for all testing sets, all noise types, and all SNR conditions (except for clean speech cases). In particular, HEQ and HQ (the 4th and 5th bars) performed better as compared to MVA and PCA (the 2nd and 3rd bars). This is probably because HEQ and HQ

dynamically transform the MFCC features considering the whole distribution locally, while the filters used in MVA and PCA are fixed, and only the first and second moment statistics are taken into consideration. Furthermore, in all Fig. 3.5(a), (b), and (c), HQ performed consistently better than HEQ for all testing sets, all noise types, and all SNR conditions. For example, in Fig. 3.5(a), HQ turned out to be very helpful for babble/restaurant noise (78.41%/79.08%) as compared to HEQ (75.95%/76.28%), probably because in such cases of speech-like noise the order statistics disturbances were better absorbed by HQ's blocks than by HEQ's point-by-point transformation. For subway noise, on the other hand, the improvement of HQ (81.70%) compared to HEQ (80.86%) is relatively less, probably because the impulse-like disturbances may very often exceed beyond the blocks.

Table 3.1: The averaged normalized distances between clean and corrupted speech features under different SNR values for HEQ and HQ (1-dim).

SNR	20 dB	15 dB	10 dB	5 dB	0 dB	-5 dB
HEQ	0.7876	0.8695	0.9516	1.0384	1.1314	1.2276
HQ (1-dim)	0.7172	0.7870	0.8588	0.9362	1.0204	1.1087

We further compared HEQ with HQ (one-dimensional) tested here using a different metric, the averaged normalized distance between the corrupted feature parameters \bar{x}_t and the corresponding clean speech feature parameters x_t ,

$$d = \frac{1}{\sigma T_N} \sum_{t=1}^{T_N} |\bar{x}_t - x_t|, \quad (3.4)$$

where the average in Eq. (3.4) is performed over all feature parameters in all the testing speech in sets A, B, C, T_N is the total number of frames, and σ is the standard deviation for all the clean feature parameters x_t . Both \bar{x}_t and x_t have been processed by either HEQ or HQ, so the difference $(\bar{x}_t - x_t)$ indicates how the mismatch caused by noise disturbance is reduced by either HEQ or HQ for each individual feature parameter. Smaller values of d imply that the features are less influenced by disturbances, although d is not necessary directly related to recognition accuracy. The results are listed in Table 3.1 for different SNR values. We find in the table that the values of d consistently increase as the SNR value

degrades, which makes very good sense, and HQ clearly gives smaller values of d in all cases. This may explain from a different perspective why HQ performed better than HEQ.

3.5.2 HQ as a Feature Quantization Method

The next set of experiments considered HQ as a feature quantization method in a DSR framework. But here we first examined the effect of quantization and compression on recognition accuracy, so we assume that the environmental noise was present with the input speech, but there were no transmission errors. For comparison, recognition accuracies for MFCC features with quantization and compression using the standard SVQ [17], the well-known transform coding [19, 21] (i.e. performing quantization in the transformed domain) followed by SVQ (TC-SVQ), the cascade of the HEQ front-end with SVQ (HEQ-SVQ), and the proposed HQ (actually two-dimensional HVQ) for bit rates 4.4, 3.9, 3.3, and 2.7 kbps are listed respectively in Table 3.2 for clean-condition training, averaged over all ten types of noise and all SNR values in sets A, B, and C. The recognition accuracies for baseline experiments with original MFCC features without quantization is 61.08%. Because all these results are averages over all SNR values from 20 down to 0 dB, the numbers here are not very high. Note that the performance of HQ was consistently and significantly better than SVQ, TC-SVQ, and HEQ-SVQ under all transmission bit rates. For example, at bit rate of 2.7 kbps, the overall accuracy of HQ (82.08%) represented relative error rate reductions of 26.93%, 62.62%, and 64.57% respectively, as compared to those with HEQ-SVQ (75.47%), TC-SVQ (52.06%), and SVQ (49.43%). It is even significantly higher (with an error rate reduction of 53.96%) than the original unquantized MFCC (61.08%). This was clearly due to the robust nature of HQ, as discussed previously. Note that the original uncompressed MFCC degraded seriously under noisy conditions, but HQ held up quite well. Also note that the performance of SVQ, TC-SVQ, and HEQ-SVQ all degraded significantly under lower bit rates, while the performance of HQ remained very stable for different bit rates, or the

Table 3.2: Recognition accuracies for feature quantization and compression with clean-condition training, averaged over all SNR values and noise types in sets A, B, and C for different bit rates (4.4 kbps to 2.7 kbps).

Bit rates (kbps)	4.4	3.9	3.3	2.7
unquantized MFCC	61.08			
SVQ	56.51	55.74	51.13	49.43
TC-SVQ	63.41	62.53	60.33	52.06
HEQ-SVQ	79.79	78.89	78.35	75.47
HQ	81.87	81.95	81.74	82.08

performance of HQ is actually relatively insensitive to the quantization resolution N in Eq. (3.1). These results indicate that, with the conventional distance-based quantization (SVQ), even with the more robust feature transformation front-end (TC or HEQ), the quantization distortion and environmental noise still jointly degraded the performance seriously. The HQ approaches, however, were able to reconstruct the feature parameters based on the order statistics or histogram, which automatically absorbed many of the disturbances, therefore offering a much better recognition accuracy.

The results in Table 3.2 are averaged over all SNR values and all noise types in sets A, B, and C. Further, we see in Fig. 3.6(a1)–(a4) the detailed accuracies obtained in exactly the same experiments, but separated for different noise types and averaged over all SNR values for different bit rates (4.4, 3.9, 3.3, and 2.7 kbps) respectively. From Fig. 3.6(a1)–(a4), we can find that HQ (the last bar in each set) consistently performed much better than the other approaches compared in Table 3.2 (the first 4 bars in each set). HQ can even handle non-stationary disturbances as well to a good degree, clearly because it is based on the dynamic histogram of the most recent past values. For example, in the case of 3.3 kbps in Fig. 5(a3), HQ is actually significantly better than HEQ-SVQ (78.82% vs. 73.69%, 79.40% vs. 73.77%, 83.80% vs. 79.37%, and 83.12% vs. 77.82% for babble, restaurant, airport, and train-station noise cases respectively), and the corresponding numbers for MFCC, SVQ, and TC-SVQ approaches were much lower.

3.5.3 Further Analysis of Bit Rates vs. SNRs for HQ as a Feature Quantization Method

To see how quantization distortion (or bit rate) mixed with the environmental noise (SNR) in the input speech jointly influences the recognition performance of a DSR system (assuming no transmission errors), the respective accuracies for the same experiments mentioned in section 3.5.2 and listed in Table 3.2 are further analyzed respectively for different bit rates and different SNRs as shown in Fig. 3.6(b1)–(b6) for clean to 0 dB SNR. For clean speech, SVQ performed the best (although slightly lower than unquantized MFCC) under higher bit rates (4.4, 3.9, and 3.3 kbps), while for other approaches (TC-SVQ, HEQ-SVQ, and HQ) feature transformation more or less changed the speech characteristics, and therefore inevitably slightly degraded the performance for clean speech. At a lower bit rate such as 2.7 kbps, however, HQ offered better performance than other approaches. This is probably because SVQ is more sensitive to quantization distortion, so the performance of SVQ, TC-SVQ, and HEQ-SVQ all degraded for lower bit rates. On the other hand, the dynamic nature of HQ makes it relatively insensitive to the quantization resolution (or bit rates), as can be verified in the clean speech case in Fig. 3.6(b1). Under noisy environments (SNR from 20 dB all the way down to 0 dB), HQ consistently performed better than other approaches for all SNR values and all bit rates. Under very poor SNR conditions, the noisy disturbances were very serious, but still well absorbed by the HQ histogram. For example, in the case of 5 dB SNR and 2.7 kbps bit rate, HQ offered an accuracy of 77.61% compared to 22.30% for SVQ, 28.31% for TC-SVQ and 69.07% for HEQ-SVQ. HQ offered an accuracy of higher than 50% (55.27%) even at 0 dB SNR and the low bit rate of 2.7 kbps. These results indicate that for SVQ the mismatched codebooks significantly increase the quantization distortion, especially under poorer SNR conditions. The performance of HQ, however, remains relatively high and even very stable for different bit rates for SNR degrading from 20 dB to 0 dB. This verified that HQ is very robust against both quantization

distortion and environmental noise.

3.6 Summary

In this chapter, a new approach of Histogram-based Quantization (HQ) is proposed for robust and distributed speech recognition (DSR). HQ has shown to be robust for all types of noise and all SNR conditions. For future personalized and context-aware DSR environment, the proposed HQ can be adapted to network and terminal capabilities, with recognition performance optimized based on environmental conditions.



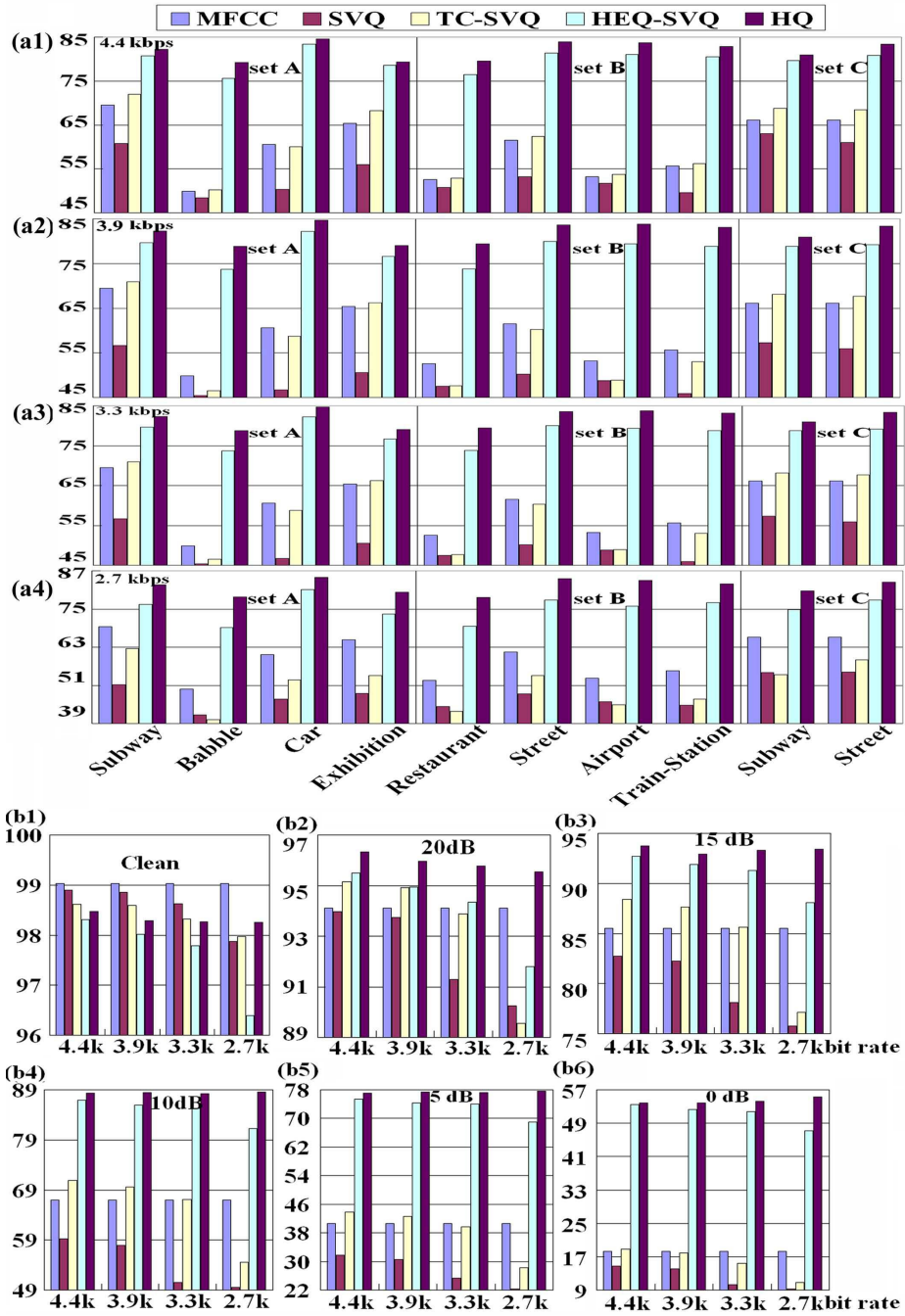


Figure 3.6: Recognition accuracies for feature quantization and compression with clean-condition training: (a1)-(a4) averaged over all SNR values but separated for different types of noise at bit rates of 4.4 kbps to 2.7 kbps; (b1)-(b6) averaged over all types of noise but separated for different bit rates (4.4 kbps to 2.7 kbps) at different SNR values.

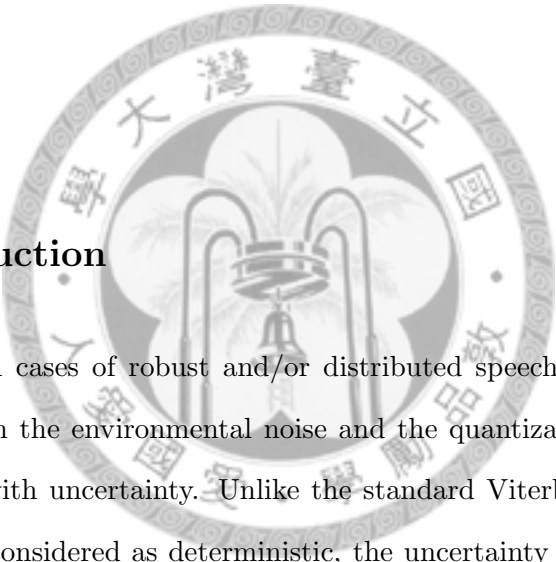
This page intentionally left blank.



Chapter 4

Joint Uncertainty Decoding (JUD) for HQ

4.1 Introduction



For both cases of robust and/or distributed speech recognition, feature vectors corrupted by both the environmental noise and the quantization errors can be viewed as random vectors with uncertainty. Unlike the standard Viterbi decoding process in which such vectors are considered as deterministic, the uncertainty decoding approach considers the uncertainty of these random vectors [11, 13, 14, 15, 16]. Approaches for robust ASR have been modified in the past to estimate such uncertainty produced by the environmental noise [11, 13, 14]. Extended Cluster Information Vector Quantization (ECI-VQ) was also developed to estimate the uncertainty generated in the quantization process [15, 16]. However, for DSR it is actually better to jointly consider the uncertainty for the quantized feature vectors caused by both the environmental noise and the quantization errors.

In this chapter, we consider both cases of robust and/or distributed speech recognition. We jointly estimate the uncertainty caused by both the environmental noise and

the quantization errors in an ASR system using Histogram-based Quantization (HQ), and perform the Joint Uncertainty Decoding (JUD) at the recognizer. Below in Section 4.2 we introduce the basic idea and formulation of uncertainty decoding. The estimation of uncertainty caused by the environmental noise and the quantization errors is described in section 4.3. Histogram-shift compensation is then introduced in section 4.4. Experimental results are offered in Sections 4.5, with the summary finally given in Section 4.6.

4.2 General Formulation of Uncertainty Decoding

In standard HMM decoding, the probability $b_j(w)$ for observing a feature vector w at a state j is

$$b_j(w) = \sum_{m=1}^M c_{jm} N(w; \mu_{jm}, \Sigma_{jm}), \quad (4.1)$$

where m is the mixture index, and $c_{jm}, \mu_{jm}, \Sigma_{jm}$ are respectively the mixture weight, mean, and covariance for the m -th Gaussian mixture in state j . There have been slightly different approaches in formulating the concept of uncertainty decoding [11, 14]. In the approach used here [13, 15, 16], instead of evaluating the observation probability $b_j(w)$ only for a single feature vector w , uncertainty decoding treats the observed feature vector w as being corrupted, and therefore considers the uncorrupted but unobservable feature vector o as a random variable with a distribution $p(o|w)$ during decoding. The probability of observing w , $b_j(w)$, can then be defined as the expected value of $b_j(o)$ with respect to the distribution $p(o|w)$ [13, 15, 16],

$$b_j(w) = E_{o|w}([b_j(o)]) = \int_o p(o|w) b_j(o) do. \quad (4.2)$$

Assuming $p(o|w)$ to be Gaussian with mean $\mu_{o|w}$ and covariance matrix $\Sigma_{o|w}$, $p(o|w) \sim N(o; \mu_{o|w}, \Sigma_{o|w})$, where both $\mu_{o|w}$ and $\Sigma_{o|w}$ can be estimated in various ways, the integration in Eq. (4.2) can be reduced to [13]

$$b_j(w) = \sum_{m=1}^M c_{jm} N(\mu_{o|w}; \mu_{jm}, \Sigma_{jm} + \Sigma_{o|w}). \quad (4.3)$$

Thus the standard HMM decoding using Eq. (4.1) remains unchanged, except that the variance of each Gaussian in the HMMs is increased by $\Sigma_{o|w}$, the uncertainty of the unobservable vector o . In this way, the Viterbi decoding can be based more on reliable parameters with a smaller variance $\Sigma_{o|w}$. The observed feature vector w can be taken as the estimated value of $\mu_{o|w}$ for simplicity, as is done here in this section. But $\mu_{o|w}$ can also be estimated based on previous feature vectors as in the three-stage error concealment approaches as discussed later on. Below, we present the approaches used here to estimate the uncertainty of the unobservable feature vector o , or the covariance matrix $\Sigma_{o|w}$.

4.3 Joint Uncertainty Decoding (JUD) for HQ

There are two sources of uncertainty in HQ-based features: quantization errors and environmental noise. Here we first separately estimate them and then consider them jointly.

4.3.1 Quantization Error Uncertainty

In an HQ partition cell, the representative value z_i is the observed corrupted feature vector w in Eq. (4.2), and all the possible samples in the corresponding i -th partition cell $[v_{i-1}, v_i]$ are these samples for the uncorrupted unquantized feature vectors o in Eq. (4.2) collected at the client, which are unobservable at the server. The variance $\Sigma_o^{q,i}$ for quantization errors in the i -th partition cell to be used to take the place of $\Sigma_{o|w}$ in Eq. (4.3) can thus be estimated using a clean speech training set. Taking the one-dimensional HQ as in Fig. 3.1 as an example,

$$\Sigma_o^{q,i} = \frac{1}{L_i} \sum_{v_{i-1} < y_t < v_i} (C_0^{-1}[C(y_t)] - z_i)^2, \quad (4.4)$$

where the summation is over all L_i feature parameters y_t in the i -th partition cell $[v_{i-1}, v_i]$ in the training set. Eq. (4.4) can be easily extended to HVQ for more dimensions. Because

the representative value z_i was obtained via the Lloyd-Max algorithm (or LBG algorithm [60] in the case of HVQ) based on the histogram $C_0(\bullet)$ for a standard Gaussian distribution, all parameters y_t in the partition cell need to be transformed first by $C(\bullet)$ then transformed back via $C_0^{-1}(\bullet)$ to evaluate $\Sigma_o^{q,i}$. Because the Lloyd-Max algorithm produces tightly quantized levels in high density regions and loosely quantized levels in low density regions to minimize total distortion, uncertainty decoding automatically increases the Gaussian variances for the loosely quantized levels. In this way, $\Sigma_o^{q,i}$ can be trained in advance for all partition cells $[v_{i-1}, v_i]$.

4.3.2 Environmental Noise Uncertainty

Under low SNR conditions, disturbances may be very serious. For example, in Fig. 3.1 v_{i-1} and v_i may be changed to v''_{i-1} and v''_i and $C(v)$ to $C''(v)$, or there may be a histogram shift which cannot be well absorbed by the dynamic histogram. Inevitably, then, HQ's performance deteriorates. Such a histogram shift may be reasonably estimated by $C_t^{-1}(0.5)$, because $C_0^{-1}(0.5) = 0$ for a standard zero-mean Gaussian. For server-side histograms constructed based on the quantized codewords, the average values of $|C_t^{-1}(0.5)|$ under all types of noise for the AURORA 2 testing environments for different SNR values are shown in Table 4.1. Clearly, the histogram shift increases with lower SNR values. This is reasonable because under lower SNR conditions, the order statistics and histograms of the original speech samples collected at the client in the respective moving segments change very rapidly; thus the quantized HQ codewords based on these histograms also change quickly and significantly with time. As a result, the server-side histogram constructed using the quantized HQ codewords also change quickly and significantly with time, introducing a significant and fast fluctuating bias or shift $|C_t^{-1}(0.5)|$ in each short segment, even if the original noise added to the signal samples is zero-mean in the long term. Hence we can take the histogram shift $|C_t^{-1}(0.5)|$ as a simple indicator for the SNR condition: that is, higher

Table 4.1: Averaged histogram shift for HQ under different SNR conditions.

SNR	Clean	20 dB	15 dB	10 dB	5 dB	0 dB
Histogram shift $ C_t^{-1}(0.5) $	0.016	0.038	0.053	0.090	0.109	0.132

such shifts correspond to lower SNR values. Therefore, the variance $\Sigma_o^{n,t}$ for uncertainty caused by environmental noise at time t — used in place of $\Sigma_{o|w}$ in Eq. (4.3) — can be reasonably estimated as

$$\Sigma_o^{n,t} = \alpha(C_t^{-1}(0.5))^2, \quad (4.5)$$

where α is an empirically determined scaling factor, and is fixed for all SNR values and noise conditions in our experiments. In fact, the value of $\Sigma_o^{n,t}$ only indicates the relative importance of feature parameters in Viterbi decoding — we found in preliminary experiments that recognition performance is not very sensitive to the value of α chosen here. $C_t(\bullet)$ is the histogram for the HQ-quantized codewords z_i for all feature parameters y_t in the moving segment $Y_{t,T}$ at frame t . In this way, in the DSR case, $\Sigma_o^{n,t}$ can be estimated at the server easily for each time t without any extra bit rate costs. This allows us to solve the problem where the environmental disturbances are hidden in codewords and cannot be estimated directly.

4.3.3 Joint Uncertainty Decoding (JUD) for HQ

The above two types of uncertainties should be jointly considered [50]. A reasonable assumption is that for higher SNR conditions the quantization error uncertainty $\Sigma_o^{q,i}$ dominates, while for lower SNR conditions, the environmental noise uncertainty $\Sigma_o^{n,t}$ dominates. Therefore the joint uncertainty $\Sigma_o^{i,t}$ for a codeword z_i in the i -th partition cell at time t can be estimated as

$$\Sigma_o^{i,t} = \max(\Sigma_o^{q,i}, \Sigma_o^{n,t}), \quad (4.6)$$

where $\Sigma_o^{q,i}$ is pre-trained for the i -th partition cell using Eq. (4.4), and $\Sigma_o^{n,t}$ is estimated in real time using Eq. (4.5). This value of $\Sigma_o^{i,t}$ can then be used as $\Sigma_{o|w}$ directly in Eq. (4.3).

4.4 Histogram-Shift Compensation

As mentioned previously, histogram shift occurring at lower SNR values inevitably results in seriously degraded HQ performance. As a result, in addition to the uncertainty decoding as mentioned above, we can also shift the histogram horizontally to have

$$C_t^{-1}(0.5) = 0 \quad (4.7)$$

for each time t . A large portion of the serious disturbances can be absorbed by such a shift, as will be verified by the experiments below.

4.5 Experimental Results

4.5.1 HQ and JUD for Robust Speech Recognition

Here we consider a complete HQ-based robust speech recognition system under noisy conditions, outside of the DSR or client-server framework. The input speech features were first transformed by HQ just as was presented in section 3.2. In addition, in this section JUD as discussed in sections 4.2-4.4 was further applied at the decoder, including the histogram shift plus the uncertainty estimated for the environmental noise and quantization errors.

The results are plotted in Fig. 4.1. Note that in Fig. 4.1(b) the plots for 5 and 0dB SNR are shown in different scales so as to make the differences easier to observe. The four bars in each set in Fig. 4.1(a), (b), and (c) are respectively for the accuracies obtained with the proposed HQ feature transformation alone (one-dimensional with bit rate (resolution) 3.9 kbps, exactly the same as the last bar in Fig. 3.5 presented in section 3.5.1),

Table 4.2: Accuracies and error rate reductions for HQ alone (one-dimensional, 3.9 kbps) and HQ-s,n,q (with complete JUD) for different testing sets in Fig. 4.1(c).

Accuracy	Set A	Set B	Set C	Overall
HQ (one-dimensional)	80.85	82.17	81.86	81.58
HQ-s,n,q (Complete JUD)	82.40	83.81	83.11	83.67
Relative error reduction (%)	8.09	9.14	6.89	8.27

HQ plus histogram shift (HQ-s, section 4.4), HQ with histogram shift plus uncertainty for environmental noise (HQ-s,n, sections 4.4 and 4.3.2), and HQ with complete JUD including histogram shift and uncertainty for environmental noise and quantization errors (HQ-s,n,q, sections 4.4 and 4.3). It can be found in Fig. 4.1(a), (b), and (c) that with the various JUD approaches proposed in sections 4.3 and 4.4 performed at the decoder, accuracies can be consistently improved step-by-step in all cases. There was almost no performance degradation for clean speech, and slight improvements at high SNR conditions (Fig. 4.1(b)): this implies uncertainty decoding for HQ is able to preserve the discrimination among HMMs. In other words, it is clear that the quantization process produces quantization errors, but with proper design of the quantizer and the uncertainty decoding, quantization errors and environmental disturbances can in fact be well absorbed and compensated for to a good extent. Accuracies for the first and the last bars in Fig. 4.1(c) (HQ alone and HQ-s,n,q with complete JUD) are also compared in Table 4.2. It can be found that significant error rate reduction was actually achieved in all three testing sets.

4.5.2 HQ and JUD for Distributed Speech Recognition

Here we consider a complete DSR system based on the proposed HQ approaches. HQ was first applied at the client end to quantize and compress the input speech features. The quantized codewords were then transmitted to the server. JUD was then applied at the server to improve accuracies.

Conventionally, in DSR this is done using SVQ [17]. If noise can be properly handled to a good degree by cascading an HEQ process at the front, we can also compen-

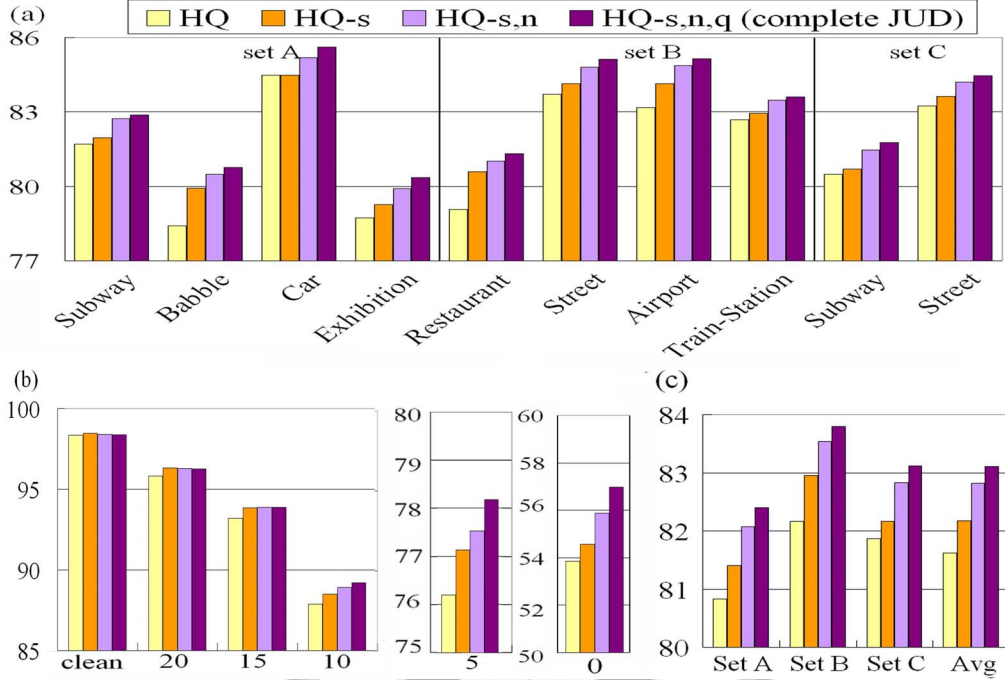


Figure 4.1: Performance improvements obtained by the various JUD approaches as compared to HQ alone: (a) averaged over all SNR values but separated for different noise types in sets A, B, and C; (b) averaged over all noise types but separated for each SNR value; and (c) averaged over all SNR values and noise types but separated into sets A, B, and C.

sate for quantization errors caused by SVQ using some conventional approaches associated with SVQ, for example the well-known Extended Cluster Information Vector Quantization (ECIVQ) [16]. Therefore we need to compare the proposed HQ followed by JUD with such conventional approaches associated with SVQ first. The results are in Fig. 4.2(a), (b), and (c). The six bars in each set in Fig. 4.2 are respectively for SVQ alone, ECIVQ alone, the cascade of HEQ front-end and SVQ (HEQ-SVQ), the cascade of HEQ front-end and ECIVQ (HEQ-ECIVQ), HQ (two-dimensional), and the same HQ with complete JUD including histogram shift (HQ-s,n,q), all with bit rates 4.4kbps. The 1st, 3rd, and 5th bars in Fig. 4.2 are the same as the 2nd, 4th, and 5th bars of the first 4.4kbps group in Fig. 3.6.

We can find from Fig. 4.2 that ECIVQ (2nd bar) performed better than SVQ (1st bar) for sets A and B, but slightly worse for set C, and the same trend can be observed

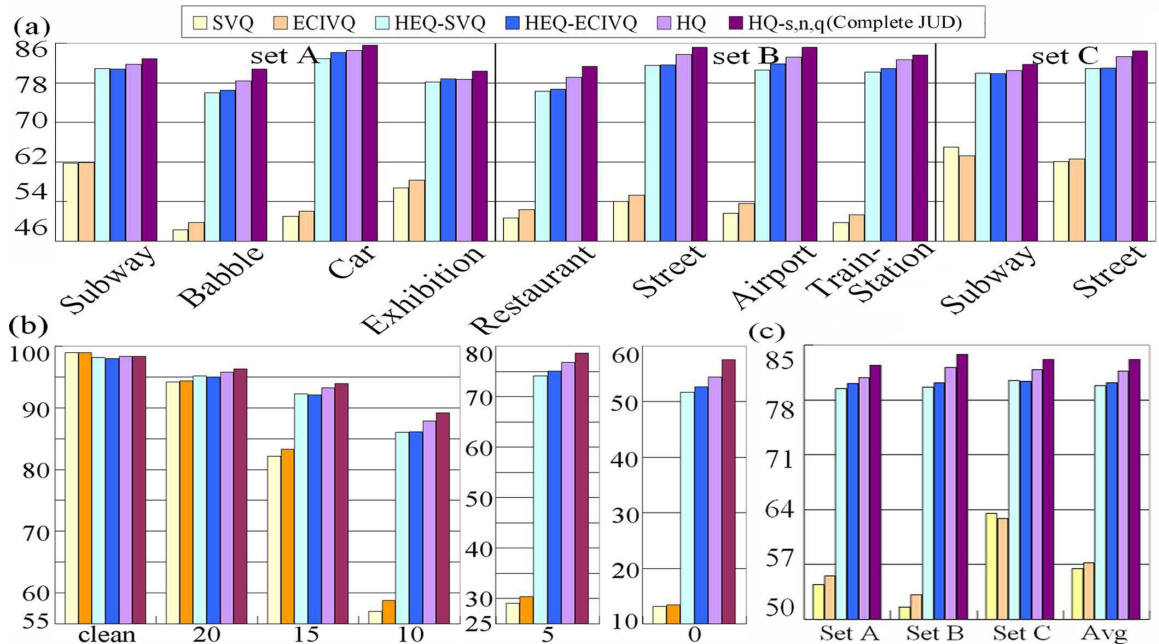


Figure 4.2: Comparison of different approaches discussed in this paper for DSR: (a) averaged over all SNR values but separated for different noise types in sets A, B, and C; (b) averaged over all noise types but separated for different SNR values; and (c) averaged over all SNR values and noise types but separated for sets A, B, and C.

when HEQ is performed as a front-end of SVQ (HEQ-SVQ, 3rd bar v.s HEQ-ECIVQ, 4th bar). This is probably because ECIVQ considers quantization errors only, but the channel mismatch for set C might move the feature vectors to different partition cells, for which the cluster variance used in ECIVQ was not able to help. HEQ offered very significant improvements when cascaded with SVQ or ECIVQ (HEQ-SVQ or HEQ-ECIVQ, 3rd or 4th bar), but the HQ (5th bar) proposed here consistently provided better performance in almost all cases, and the complete JUD proposed here including histogram shift (HQ-s,n,q, 6th bar) offered additional improvements consistently in almost all cases. The accuracies for HEQ cascaded with ECIVQ (HEQ-ECIVQ, 4th bar) and HQ with JUD (HQ-s,n,q, the last bar) are further compared in Table 6.1. The relative error rate reductions shown in the last row are significant and consistent for all SNR values, including the clean and 20 dB cases. The above experimental results are for a 4.4 kbps bit rate. Further analysis was then performed

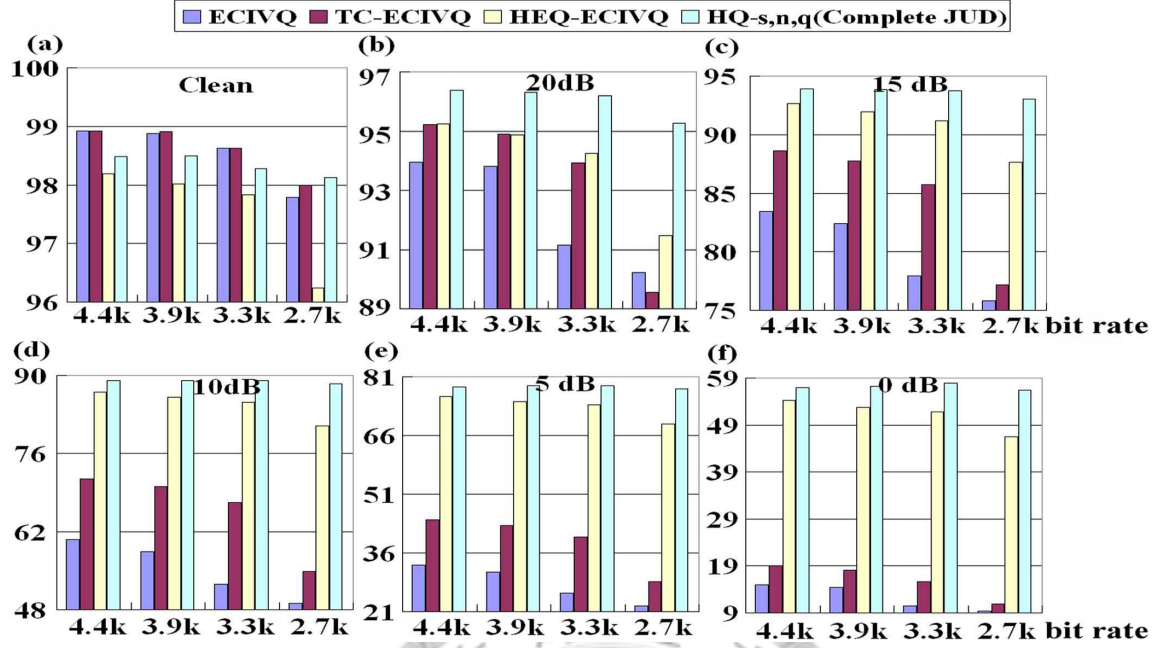


Figure 4.3: Comparison of different approaches discussed in this paper for DSR (but without transmission errors) under different bit rates and SNR values: (a) clean, (b) 20 dB, (c) 15 dB, (d) 10 dB, (e) 5 dB, and (f) 0 dB.

Table 4.3: Accuracies and error rate reductions for HEQ-ECIVQ and HQ-s,n,q (with complete JUD) at 4.4 kbps for different SNR values in Fig. 4.2(b).

SNR	Clean	20 dB	15 dB	10 dB	5 dB	0 dB
HEQ-ECIVQ	98.19	95.25	92.65	86.01	75.96	53.28
HQ-s,n,q(Complete JUD)	98.50	96.38	93.99	89.04	78.34	57.01
Relative error reduction(%)	17.13	23.79	18.23	21.66	9.90	7.98

for several better approaches found above with respect to different bit rates (4.4, 3.9, 3.3, and 2.7 kbps) at all different SNR values. The results are shown in Fig. 4.3(a)–(f) for different SNR from clean to 0 dB, each with different bit rates. The four bars in each set in Fig. 4.3 are respectively for ECIVQ considering quantization error uncertainty for SVQ, the cascade of transform coding (TC) and ECIVQ (TC-ECIVQ), the cascade of HEQ and ECIVQ (HEQ-ECIVQ), and HQ with complete JUD including histogram shift (HQ-s,n,q). Here, except for the clean speech case at higher bit rates, HQ-s,n,q consistently performed better for all SNR values and all bit rates than other combinations of the front-end feature transformation (TC

or HEQ) or back-end compensation considering quantization uncertainty (ECIVQ). Also, the performance of ECIVQ, TC-ECIVQ, and HEQ-ECIVQ are all more sensitive to lower bit rates, while HQ-s,n,q is relatively insensitive to different bit rates at all SNR conditions.

4.6 Summary

In this chapter, Joint Uncertainty Decoding (JUD) under the framework of Histogram-based Quantization (HQ) is proposed here in this paper for robust and/or distributed speech recognition. Improved recognition performance was obtained consistently under all types of noise at all SNR values.



This page intentionally left blank.



Chapter 5

Three-Stage Error Concealment (EC) for HQ-Based DSR Systems

5.1 Introduction

Here we consider the approaches to handling the transmission errors added to the received HQ codewords under the DSR framework [51]. In this chapter, a three-stage EC approach is developed, as presented below. In Section 5.2 we introduce the frame and sub-vector error detection by HQ-consistency check. The estimation of the detected erroneous subvectors are presented in section 5.3, considering the prior speech source statistics, the channel transition probability, and the reliability of the received subvectors. In section 5.4, we introduce the reliability estimation and uncertainty decoding. Section 5.5 gives the overview of the three-stage error concealment (EC) framework. Experimental results are offered in Sections 5.6, with the summary finally given in Section 5.7.

5.2 Stage 1 - Error Detection

In the ETSI DSR standards, every two frames are grouped together and protected with 4-bit CRC[17]. In this way, the entire frame-pair is labeled erroneous even if only a single bit error occurs in the frame-pair packet. Adding check bits at the subvector level is helpful for subvector level error detection, but comes at the cost of additional bandwidth [21]. A more efficient way is to make use of the speech signal characteristics at the subvector level. The data consistency test checks the continuity of the parameters in two neighboring subvectors [35]. When the difference between two consecutive values of a feature parameter in a subvector exceeds a pre-determined threshold obtained from some training corpus, the subvector is classified as inconsistent. However, if the statistics of the testing features are time-varying and different from those of the training corpus, this approach becomes less reliable. With environmental noise, the parameters are likely to be classified as inconsistent even if they are correctly received.

HQ performs feature parameter quantization based on the local histogram (or order statistics), so the quantized codewords represent the local order-statistic information of the original parameters. The quantization process does not change the order statistics of the parameters, and if there are no transmission errors, the histogram for the subvector codewords received at the server should be similar to the histogram for the original feature parameters at the client. Thus the partition cell obtained by re-performing HQ on the received subvector codeword, based on the dynamic histogram for these received codewords, should be the original partition cell. If not, it is very possible that the order statistics have been changed and the received subvector codeword may be erroneous. Based on this observation, the consistency test in the HQ framework proposed here is as follows. Taking a two-dimensional HVQ as an example, $z_i = (z_i^{(1)}, z_i^{(2)})$ is a received subvector codeword at some time, and $\text{HQ}\{(z_i^{(1)}, z_i^{(2)})\}$ represents the representative value for the subvector $(z_i^{(1)}, z_i^{(2)})$ assigned by HQ performed at the server based on the histogram for the received

codewords. The subvector $(z_i^{(1)}, z_i^{(2)})$ is then classified as consistent if

$$HQ\{(z_i^{(1)}, z_i^{(2)})\} = (z_i^{(1)}, z_i^{(2)}). \quad (5.1)$$

In other words, if these two parameters are correctly received, their order statistics at the server should be similar to the order statistics for the original values before quantization at the client, and therefore similarly quantized into the same HQ partition cell.

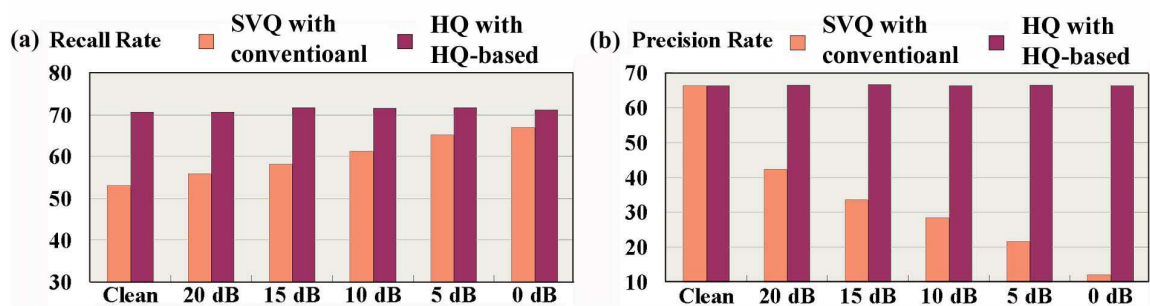


Figure 5.1: (a) Recall and (b) Precision rates for error detection using SVQ with the conventional data consistency check and HQ with the HQ-based consistency check proposed here.

We compared the error detection accuracy of the conventional SVQ scheme with the data consistency check [35] and the proposed HQ with the HQ-based consistency check mentioned above under all different noise conditions for the AURORA 2 testing environment with the transmission errors introduced by the General Packet Radio Service (GPRS) wireless environment. The averaged recall (percentage of detected errors out of all errors) and precision (percentage of correct errors out of all detected errors) rates for error detection are shown in Fig. 5.1(a) and (b). For lower SNR cases, it is clear that the noise seriously affects the SVQ with data consistency check as verified by the precision degradation in Fig. 5.1(b) (from 66% at clean down to 12% at 0 dB). With the proposed HQ-based consistency check approach, however, the precision rate is much more stable at all SNR values, and both recall and precision rates are higher.

Note that when Eq. (5.1) is not satisfied, it is also possible that the present

codeword is actually correctly received, but instead the dynamic histogram, on which the HQ in Eq. (5.1) is based, is disturbed by erroneous received codewords in the past T frames. This is one good reason why the precision rate in Fig. 5.1(b) for HQ with the proposed consistency check is slightly less than 70%, i.e. some detected inconsistencies are actually correctly received codewords. But this precision is much higher than SVQ with conventional approach. In fact, the probability that the inconsistency in Eq. (5.1) is due to the disturbed histogram rather than the considered codeword being erroneous is lower, because the effect of the erroneous codewords in the past T frames is reasonably absorbed by the histogram (the order statistics of a large number of codewords) as well as the partition cells in HQ. In other words, with erroneous codewords in the past T frames, the change of the histogram may not be very serious and the partition cell that the present codeword being considered belongs to may remain unchanged. This is verified in Fig. 5.1(b) where the precision rate, although much less than 100%, remains almost the same from clean speech to 0 dB SNR.

5.3 Stage 2 - Erroneous Feature Vector Estimation

Different techniques for estimating the detected erroneous feature vectors have been proposed. Repetition and interpolation only use the correctly received feature vectors [23], while statistical-based techniques use prior knowledge about speech source in addition, and have been shown to offer better performance [28].

The erroneous subvector estimation proposed here under the HQ framework is based on the maximum a posteriori (MAP) criterion, which determines the estimated value \hat{s}_t of a certain transmitted subvector codeword s_t at time t , which is detected as erroneous (here both \hat{s}_t and s_t are certain codewords z_i mentioned above for some i respectively). This MAP estimation is conditioned on the present and previously received corresponding subvector codewords r_t and r_{t-1} (here both r_t and r_{t-1} are also certain codewords z_i

mentioned above for some i respectively),

$$\hat{s}_t = \operatorname{argmax}_{z_i} \{P(s_t = z_i | r_t, r_{t-1})\}, \quad (5.2)$$

where $s_t = z_i$ denotes that s_t is the i -th HQ codeword out of the N possible codewords. The maximization here is over all of these codewords. If we assume r_t and r_{t-1} are independent,

$$P(s_t | r_t, r_{t-1}) \approx \frac{P(s_t | r_{t-1})P(s_t | r_t)}{P(s_t)} = \frac{P(s_t | r_{t-1})P(r_t | s_t)}{P(r_t)}. \quad (5.3)$$

With the denominator in Eq. (5.3) left out in the maximization in Eq. (5.2), the probability in Eq. (5.2) can be approximated by the codeword bigram $P(s_t = z_i | r_{t-1})$ and the channel transition probability $P(r_t | s_t = z_i)$,

$$\hat{s}_t = \operatorname{arg max}_{z_i} \{P(s_t = z_i | r_{t-1})P(r_t | s_t = z_i)\}. \quad (5.4)$$

In Eq. (5.4), the codeword bigram $P(s_t = z_i | r_{t-1})$ can be estimated by the bigram of the considered subvector codewords $P(s_t = z_i | s_{t-1})$ trained from a clean training set (for example, the clean training set of AURORA 2). Also, the channel transition probability $P(r_t | s_t = z_i)$ in Eq. (5.4) can be estimated from the bit error rate (BER) of the present frame being considered,

$$P(r_t | s_t = z_i) = BER^{d[b(z_i), b(r_t)]} * (1 - BER)^{K - d[b(z_i), b(r_t)]}, \quad (5.5)$$

where BER is estimated as the total number of inconsistent subvectors (in simulation analysis, it was found that in most cases there is only one bit error in an erroneous codeword, and therefore this number can be used to estimate the total number of erroneous bits) detected in the first stage (discussed in section 5.2) in the present frame divided by the total number of bits in the frame, K is the total number of bits in the received subvector codeword r_t , $b(z_i)$ and $b(r_t)$ are respectively the bit patterns for the codewords z_i and r_t , and $d(\bullet, \bullet)$ represents the Hamming distance between two bit patterns. The value of $P(r_t | s_t = z_i)$ in Eq. (5.5) is actually the probability of z_i being changed to r_t if BER can be accurately

estimated. With Eq. (5.5), when r_t is less reliable (or has a larger BER), the values of $P(r_t|s_t = z_i)$ for all possible codewords z_i with different i become closer to each other (i.e., the difference in $P(r_t|s_t = z_i)$ is insignificant for different Hamming distances $d(\bullet, \bullet)$). On the other hand, when r_t is more reliable (or has a smaller BER), $P(r_t|s_t = z_i)$ is larger for only few values of i . In this way, more emphasis can be put on the codeword bigram $P(s_t = z_i|r_{t-1})$ than on the channel transition probability $P(r_t|s_t = z_i)$ in Eq. (5.4) when the channel condition is less reliable.

Because the basic principle here is to exploit the short-time correlation between consecutive frames in speech signals to estimate the lost subvectors, the robustness of HQ as mentioned in section 3.4 is very helpful. If the quantization process is less robust, the environmental noise may move the feature vectors to a different partition cell and the subvector transition relationship in speech signals may be disturbed. This problem is actually lessened by the HQ's robustness, as can be verified by the mutual information $I(s_t, s_{t-1})$ between the present and previous subvector codewords s_t and s_{t-1} ,

$$I(s_t, s_{t-1}) = H(s_t) - H(s_t|s_{t-1}), \quad (5.6)$$

where

$$H(s_t) = \sum_{j=1}^N -P(s_t = z_j) \log[P(s_t = z_j)] \quad (5.7)$$

and

$$H(s_t|s_{t-1}) = \sum_{i=1}^N \sum_{j=1}^N -P(s_t = z_j, s_{t-1} = z_i) \log[P(s_t = z_j|s_{t-1} = z_i)] \quad (5.8)$$

are respectively the degree of uncertainty for the present subvector s_t , and the remaining degree of uncertainty for s_t after the previous subvector s_{t-1} is known. Thus the mutual information $I(s_t, s_{t-1})$ in Eq. (5.6) shows how much the codeword bigram model reduces uncertainty for the subvectors s_t . In other words, a bigram model with higher mutual information implies that predicting the present subvector s_t given the previous subvector s_{t-1} is easier. The mutual information for the conventional SVQ and the proposed HQ

Table 5.1: Mutual information $I(s_t, s_{t-1})$ for SVQ and HQ.

$I(s_t, s_{t-1})$	c_1, c_2	c_3, c_4	c_5, c_6	c_7, c_8	c_9, c_{10}	c_{11}, c_{12}	$c_0, \log E$
SVQ	1.365	0.998	0.791	0.652	0.611	0.568	1.455
HQ	1.473	1.110	0.856	0.722	0.678	0.619	1.541

averaged for different subvectors from the three testing sets of AURORA 2 is listed in Table 5.1. We can see that HQ's mutual information is always higher than that of SVQ, which indicates that the HQ framework allows for more precise estimation of the lost subvectors.

5.4 Stage 3 - Uncertainty Decoding

The uncertainty decoding discussed in section 4.2 can be used here in the final stage. Consider section 4.2: the above received codeword r_t is taken as the observed corrupted feature vector w in Eq. (4.2), and all of the possible transmitted codewords, $s_t = z_i$, $i = 1, 2, \dots, N$, are the possible samples of the uncorrupted but unobservable feature vector o in Eq. (4.2). The distribution of the probability $P(s_t = z_i | r_t, r_{t-1})$ obtained in Eq. (5.2) then characterizes the uncertainty of the observed codeword. With the estimated codeword \hat{s}_t in Eq. (5.2) taken as the mean $\mu_{o|w}$ and the covariance estimated using the probability distribution $P(s_t = z_i | r_t, r_{t-1})$ taken as the covariance $\Sigma_{o|w}$, both used in Eq. (4.3), uncertainty decoding can then be directly performed within the HQ framework as presented previously by increasing the variance of each Gaussian mixture by $\Sigma_{o|w}$ in the HMMs as in Eq. (4.3) [50]. In this way, HMM decoding puts more emphasis on more reliable subvectors, i.e. those with lower covariance $\Sigma_{o|w}$ for the probability distribution $P(s_t = z_i | r_t, r_{t-1})$ in Eq. (5.2).

5.5 Three-Stage EC under the HQ Framework

As shown in Fig. 5.2, the three stages of EC under the HQ framework can be easily integrated. At the first stage, the received frame-pairs are first checked with CRC

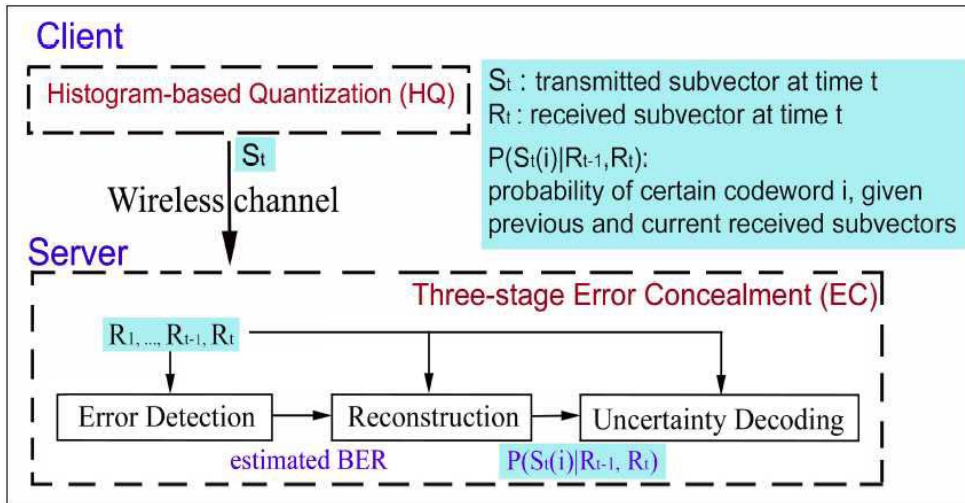


Figure 5.2: The three-stage error concealment (EC) framework.

to detect errors at the frame level. The erroneous frame-pairs are then further checked at the subvector level by the HQ consistency test as mentioned in section 5.2. At the second stage, the erroneous subvectors detected at the first stage are estimated and reconstructed as presented in section 5.3. At the third stage, uncertainty decoding in the Viterbi search process makes the HMMs less discriminative for subvectors with higher uncertainty as presented in section 5.4.

5.6 Experimental Results

Here we finally consider a complete DSR system based on the proposed HQ approaches. HQ was first applied at the client end to quantize and compress the input speech features. The quantized codewords were then transmitted via wireless networks to the server. There were inevitable transmission errors introduced by the wireless channels, and the three-stage error concealment (EC) was applied.

5.6.1 HQ-Based DSR over Wireless Channels with Transmission Errors, but without Error Concealment (EC)

We first compared the robustness of SVQ and HQ against environmental noise at the client end plus the transmission errors at a client traveling speed of 3 km/hr, assuming no Error Concealment (EC) approach was used. Fig. 5.3 is the averaged results over all different types of noise but separated for different SNR values. The first three bars are the results for the standard SVQ, SVQ followed by HEQ front-end (HEQ-SVQ), and HQ (two-dimensional), all at 4.4kbps and without transmission errors, exactly the same as the 1st, 3rd, and 5th bars in Fig. 4.2(b), and the next three bars are those suffering from GPRS transmission errors (SVQg, HEQ-SVQg, HQg: the label "g" indicates GPRS). For SVQ, the performance degradation caused by GPRS (1st bar compared to 4th bar) is larger when SNR is lower, even with HEQ (2nd bar compared to 5th bar, e.g. 98.07% to 87.78% for clean speech, 91.97% to 76.74% for 15 dB SNR, and 85.86% to 68.73% for 10 dB SNR). Clearly, features corrupted by noise are more susceptible to transmission errors. The improvements that HQ offered over HEQ-SVQ when transmission errors were present (6th bar to 5th bar) are consistent and significant at all SNR values. For example, in the case of 10 dB SNR with GPRS, HQ (6th bar) offered an accuracy of 78.69% while the number was 69.84% for HEQ-SVQ (5th bar). This verified that HQ is robust against both environmental noise and transmission errors.

To analyze the degradation of recognition accuracy caused by transmission errors, we examined the percentage of words which were correctly recognized without transmission errors, but incorrectly recognized after transmission. The comparison of this percentage for SVQ, HEQ-SVQ and HQ for exactly the same experiments as reported in Fig. 5.3 are shown in Fig. 5.4. The rapid increase of this percentage for SVQ when input speech SNR is degraded indicated that the noise-corrupted SVQ symbols were very susceptible to transmission errors. HEQ-SVQ was much better, while HQ was the best in all cases.

The above results in Fig. 5.3 are for a 4.4 kbps bit rate. Further analysis was then performed for several better approaches found above with respect to different bit rates (4.4, 3.9, 3.3, and 2.7 kbps) for all SNR values (from clean to 0 dB) as shown in Fig. 5.5(a)–(f). The four bars in each set in Fig. 5.5 are respectively for SVQg, transform coding followed by SVQ (TC-SVQg), the cascade of HEQ and SVQ (HEQ-SVQg), and HQg, all with GPRS transmission errors. Here HQ consistently performed better than different versions of SVQ enhanced by some feature transformation approaches (TC or HEQ) for all SNR values and all bit rates. With SVQ, features with environmental noise and quantization distortion are more sensitive to lower bit rates when transmission errors are present. For example, in the case of 5 dB SNR, the performance of HEQ-SVQ degraded from 56.66% at 4.4 kbps to 51.88% at 2.7 kbps. On the other hand, the performance of HQ is very stable for different bit rates in all cases of SNR, even with the presence of transmission errors. This verified that HQ is robust against not only quantization distortion and environmental noise, but transmission errors as well.

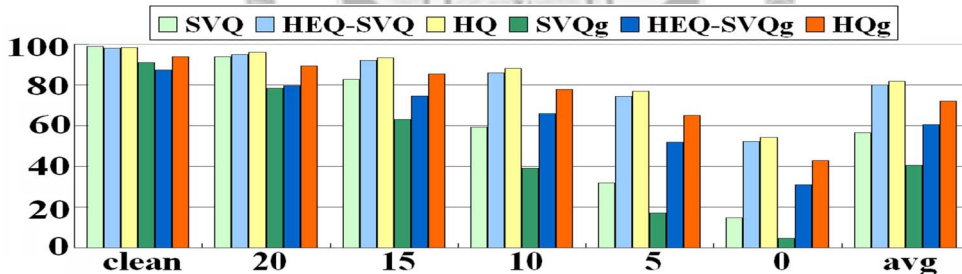


Figure 5.3: Comparison of SVQ, HEQ-SVQ and HQ, and those with GPRS transmission errors (SVQg, HEQ-SVQg, HQg), averaged over all types of noise, but separated for each SNR value.

5.6.2 HQ-Based DSR over Wireless Channels with Error Concealment (EC)

The next set of experiments tried to examine the effectiveness of the three-stage EC techniques for HQ. Fig. 5.6 shows the results with GPRS transmission errors at a speed of 3

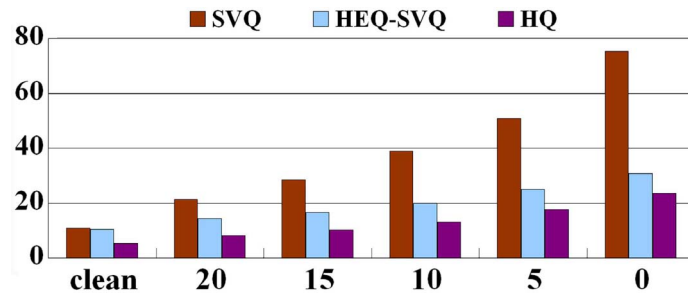


Figure 5.4: Comparison of SVQ, HEQ-SVQ and HQ with the percentage of words which were correctly recognized if without transmission errors, but incorrectly recognized after transmission.

km/hr, without and with the different Error Concealment (EC) approaches. The five bars in each set are respectively for SVQg, HEQ-SVQg, HEQ-SVQ with GPRS and with repetition (HEQ-SVQgr: the label "r" indicates the ETSI-recommended error mitigation strategy by repetition), HQg, and HQ with GPRS and the three-stage EC techniques propose here (HQgc: the label "c" indicates three stage EC), all at bit rate of 4.4 kbps. Fig. 5.6(a) are those averaged over all SNR values but separated for different noise types in sets A, B, and C, (b) are those averaged over all types of noise but separated for different SNR values, and (c) are those averaged over all types of noise and all SNR values but separated for sets A, B, and C. It can be found that the ETSI repetition technique actually degraded the performance of HEQ-SVQg (3rd bar vs. 2nd bar), probably because the whole feature vectors including the correct subvectors are replaced by estimations that are very possibly inaccurate. Under GPRS, HQg without any EC techniques (4th bar) actually outperformed the first three bars for all cases. Applying the proposed three-stage EC techniques (HQgc, 5th bar) then further improved the performance significantly for all cases. This verified that the three-stage EC framework is robust against not only transmission errors, but against environmental noise as well.

The above results in Fig. 5.6 are for a 4.4 kbps bit rate. Further analysis was then performed with respect to different bit rates (4.4, 3.9, 3.3, and 2.7 kbps) for all SNR

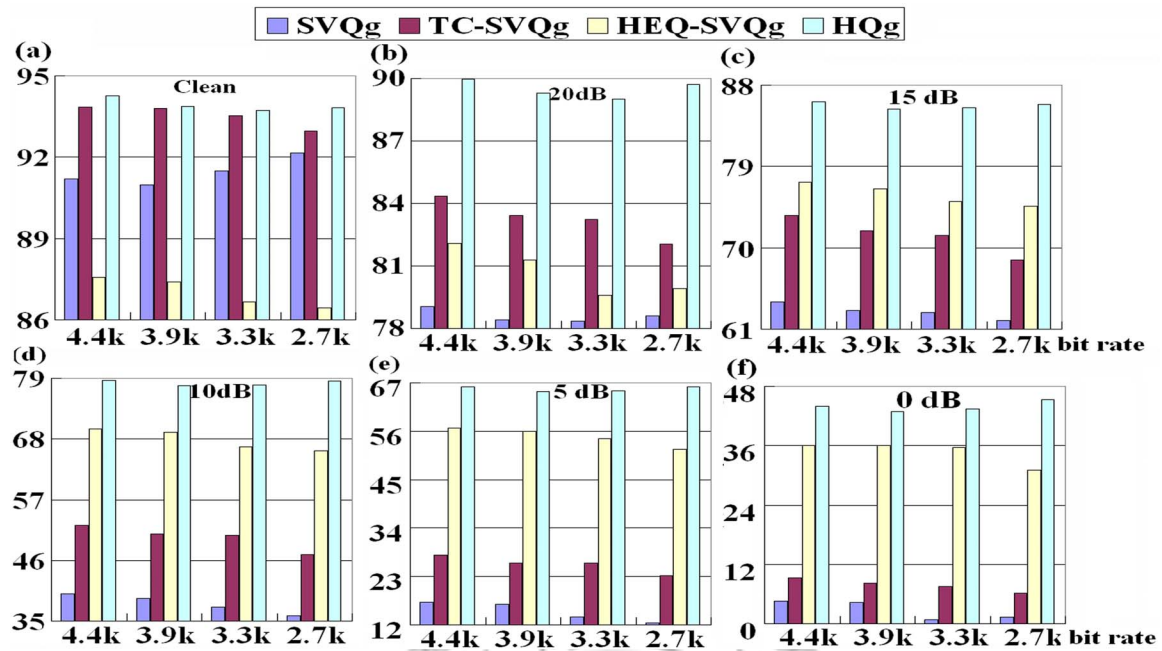


Figure 5.5: Comparison of SVQg, TC-SVQg, HEQ-SVQg and HQg (all with GPRS transmission errors), for different bit rates and SNR values: (a) clean, (b) 20 dB, (c) 15 dB, (d) 10 dB, (e) 5 dB, and (f) 0 dB.

values as shown in Fig. 5.7(a)–(f). The four bars in each set in Fig. 5.7 are respectively for SVQ with GPRS errors and with repetition (SVQgr: the label "r" indicates the ETSI-recommended error mitigation strategy by repetition), TC-SVQ with GPRS errors and with repetition (TC-SVQgr), HEQ-SVQ with GPRS errors and with repetition (HEQ-SVQgr), and HQ with GPRS and the three-stage EC techniques propose here (HQgc). Here HQgc consistently performed better than all other approaches for all SNR values and all bit rates. For example, in the case of 10 dB SNR and a 3.3 kbps bit rate, HQgc offered an accuracy of 81.57% compared to 38.92% for SVQgr, 53.34% for TC-SVQgr and 64.97% for HEQ-SVQgr. HQgc offered an accuracy of higher than 65% (67.42%) even at 5 dB SNR and the low bit rate of 2.7 kbps. These indicate that HQ with the three-stage EC is robust against both environmental noise and transmission errors, and is insensitive to different bit rates.

The above results in Fig. 5.6 and 5.7 are for a client traveling at a speed of 3 km/hr. We then consider other different client traveling speeds at 4.4 kbps in Fig. 5.8.

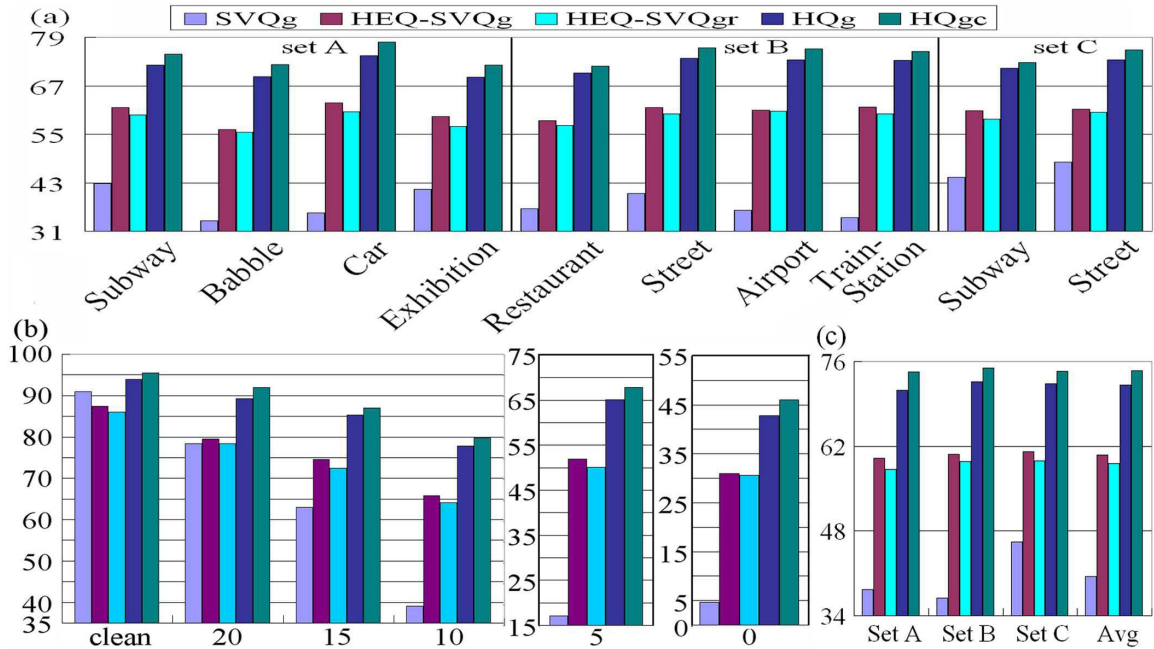


Figure 5.6: Comparison of SVQ under GPRS (SVQg), HEQ-SVQ under GPRS without and with repetition (HEQ-SVQg and HEQ-SVQgr), HQ under GPRS without and with EC techniques (HQg and HQgc): (a) averaged over all SNR values, but separated for different noise types in sets A, B, and C; (b) averaged over all types of noise, but separated for each SNR value; and (c) averaged over all SNR values and noise types but separated for sets A, B, C.

Here the four cases shown in each figure are for HEQ-SVQ under GPRS, without and with ETSI repetition (HEQ-SVQg and HEQ-SVQgr), and HQ under GPRS, without and with the three-stage EC (HQg and HQgc), at traveling speeds of 3, 50, 100, and 250 km/hr. Only two typical types of input speech noise, car for stationary and babble for non-stationary were taken as examples, since for some noise types such as exhibition or restaurant a client traveling speed above 3 km/hr does not make sense. The results for two typical values of SNR, 15 dB and 5 dB plus those results averaged over all SNR values for car/babble noise are shown in Fig. 5.8 (a1)/(a2), (b1)/(b2) and (c1)/(c2), respectively. The superiority of HQ with EC (HQgc) is obvious as verified by the highest curves in all cases. As an example, for 15 dB car noise at 100 km/hr as shown in Fig. 5.8(a1), the performance of HEQ-SVQ degraded seriously (78.74%), applying ETSI repetition on HEQ-SVQ did not help (72.89%),

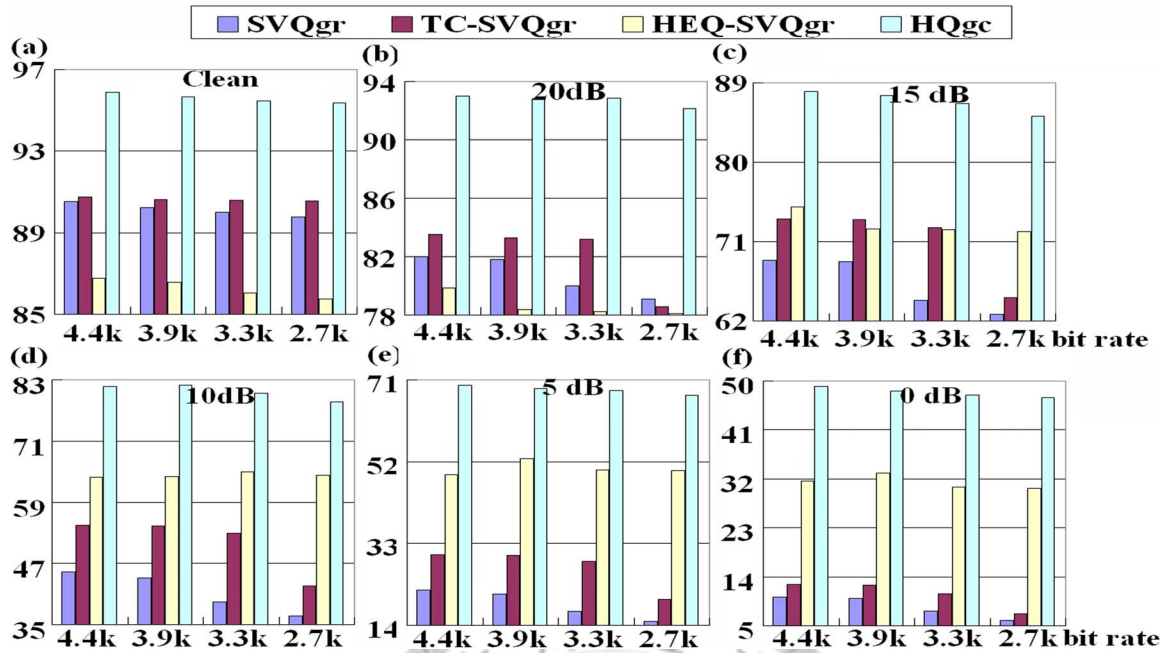


Figure 5.7: Comparison of SVQgr, TC-SVQgr, HEQ-SVQgr (all under GPRS with repetition), and HQgc (under GPRS with error concealment) for different bit rates and SNR values: (a) clean, (b) 20 dB, (c) 15 dB, (d) 10 dB, (e) 5 dB, and (f) 0 dB.

and HQ is much better (86.04%) while the three-stage EC offered very good improvements (92.80%). As another example, for 5 dB car noise as shown in Fig. 5.8(b1), the performance of HEQ-SVQ degraded seriously at high traveling speeds (e.g. 59.20% at 100 km/hr); here HQ was much better (e.g. 66.24% at 100 km/hr), and the three-stage EC further improved the performance significantly (e.g. 78.29% at 100 km/hr). On the other hand, as one more example in Fig. 5.8(a1) the HEQ-SVQ features with noise disturbances were more susceptible to higher transmission errors due to higher client traveling speeds (81.82% at 3 km/hr and 78.74% at 100 km/hr), while HQ features were more robust in this case (87.33% at 3 km/hr and 86.04% at 100 km/hr). This is why the curves for HQg are quite flat in almost all the six figures in Fig. 5.8, while those for HEQ-SVQg and HEQ-SVQgr decline faster as the client traveling speed increases. The curves for HQgc are also quite flat for car noise (Fig. 5.8 (a1)/(b1)/(c1)), but less flat for babble noise (Fig. 5.8 (a2)/(b2)/(c2)); the non-stationary nature of the babble noise is probably more difficult to handle with EC

techniques.

5.7 Summary

In this chapter, a three-stage error concealment (EC) framework based on the Histogram-based Quantization (HQ) for Distributed Speech Recognition (DSR) is proposed. Improved recognition performance was obtained consistently for a wide variety of environmental noise and transmission error conditions.



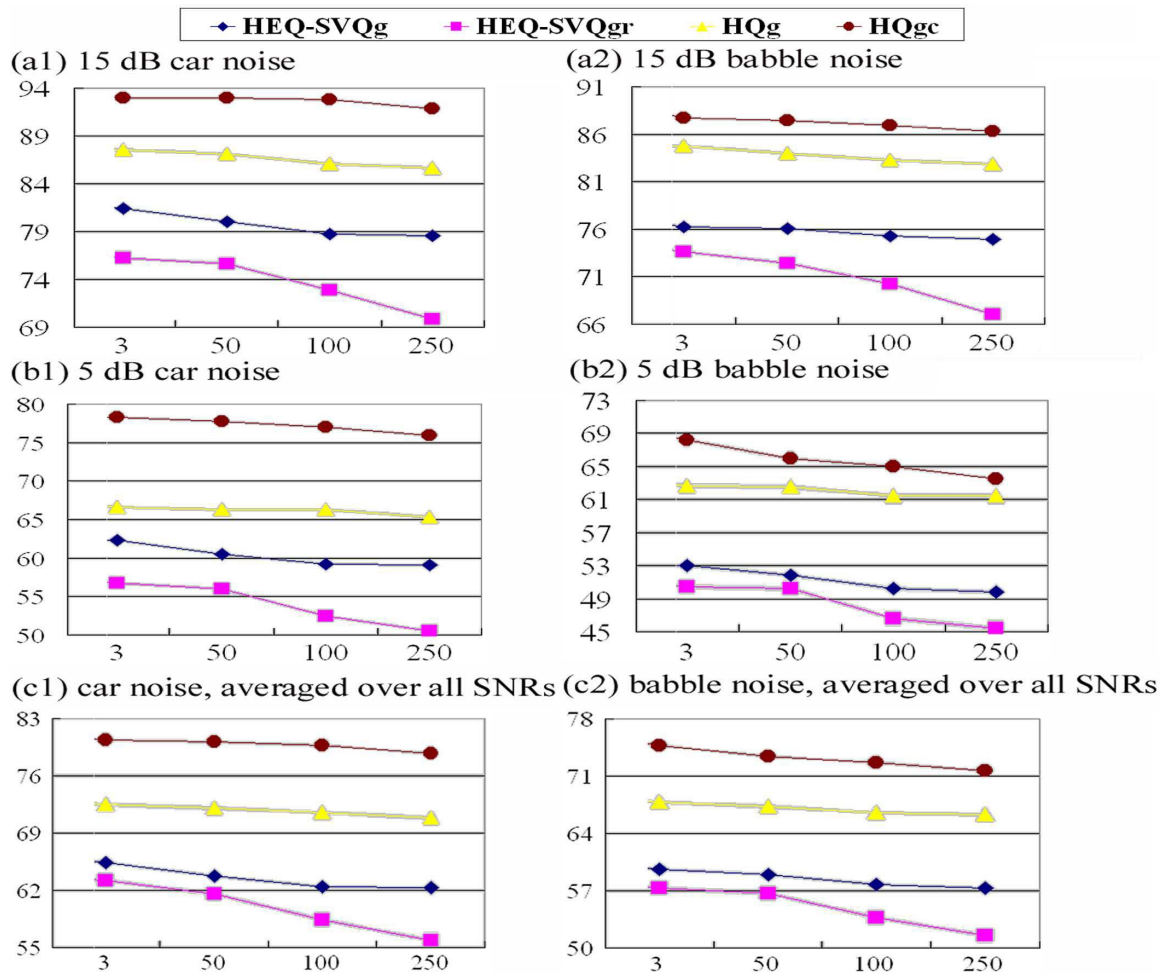


Figure 5.8: Comparison of HEQ-SVQ under GPRS without and with repetition, HQ under GPRS without and with EC, at traveling speeds of 3, 50, 100, and 250 km/hr: (a1)/(a2) for car/babble noise at 15 dB SNR; (b1)/(b2) for car/babble noise at 5 dB SNR; and (c1)/(c2) for car/babble noise averaged over all SNR values.

Chapter 6

Dynamic Quantization II - Context-dependent Quantization

6.1 Introduction

When considering the characteristics of speech signals, it is a well-known fact that the high correlation existing in speech signals is very helpful in various speech processing applications. It is also well-known that for human perception, speech is recognized based on not only the present signal values, but also on the changes in context [20]. Transform coding and differential encoding take context into consideration when performing quantization, and have been widely used for decades [20, 21, 18]. These approaches exploit inter-frame or intra-frame correlations among feature vectors and have been shown to reduce transmission rates significantly. These facts indicated that quantization approaches not using context information are relatively inadequate, because in such approaches, feature parameters with different context are quantized or transformed to the same representative value as long as they are in the same partition cell; thus signal information is not fully utilized. Therefore, properly utilizing context information in quantization to improve robustness against

transmission errors and environmental noise is an important issue.

In this chapter, we propose a new concept of context-dependent quantization, in which the representative parameters for each partition cell are not fixed, but are dependent on the context codewords. Below in Section 6.2.1 we introduce the basic idea and formulation of Context-dependent Quantization. In section 6.2.2, the context-dependent quantization is integrated with Histogram-based Quantization (HQ). Experimental results are offered in Sections 6.3, with the summary finally given in Section 6.4.

6.2 Proposed Approach

6.2.1 Context-dependent Quantization

In conventional (scalar or vector) quantization, a parameter y_t at time t (either a scalar or a vector) is mapped to a representative parameter z_i (either a scalar or a vector), which is in turn represented by a codeword or bit pattern w_t , if y_t is within a certain partition cell Q_i ,

$$y_t \rightarrow Q(y_t) = z_i, w_t = b(Q(y_t)) = b(z_i), \text{ if } y_t \in Q_i, \quad (6.1)$$

where $Q(\cdot)$ is the quantization process and $b(\cdot)$ represents the index of codeword or bit pattern. The concept of context-dependent quantization is very simple. It keeps all the original partition cells unchanged, except now the representative parameters z_i are not fixed, but are dependent on the left and right context [61]. Assume in addition the parameter y_t has a left context parameter y_{t-1} with codeword m and a right context parameter y_{t+1} with codeword n , $y_{t-1} \rightarrow Q(y_{t-1}), b(Q(y_{t-1})) = m$, $y_{t+1} \rightarrow Q(y_{t+1}), b(Q(y_{t+1})) = n$. The representative parameter for the middle frame y_t in the partition cell Q_i is then the average of all such parameters y_t within the partition Q_i with the left and right context m and n

respectively,

$$z_i^{mn} = \frac{1}{L_i^{mn}} \sum_{\substack{y_t \in Q_i \\ b(Q(y_{t-1}))=m \\ b(Q(y_{t+1}))=n}} y_t, \quad (6.2)$$

which is dependent on the context m and n , where L_i^{mn} is the total number of such parameters y_t in the training set. Thus z_i^{mn} is the average of the parameters with the same context codewords. This representative parameter z_i^{mn} can be trained with a clean speech corpus. In this way, context dependency among speech signals is automatically included in the quantization process. Note that assuming there are N partition cells, for each partition cell there are now N^2 different representative parameters because there are N^2 context conditions ($m, n \in \{1, 2, \dots, N\}$). Therefore using the left and right contexts allow for much finer representation of the parameters, although the number of bits needed remains the same. Also, the computational complexity and memory requirement on the client side are the same as those for conventional quantization because the number of partition cells is still N . This is shown in Fig. 6.1, in which a partition cell has many representative parameters $z_i^{m,n}$ for different contexts m and n , as compared to conventional quantization, in which a partition cell has only a single representative parameter z_i . Also, in this scheme for a received codeword sequence, every codeword is decoded considering its context codeword on both sides, and there is no problem regarding the order of decoding. For example, for the received codeword sequence, $\{w_1, w_2, w_3, \dots\}$, $w_1 w_2 w_3$ are used to decode w_2 , $w_2 w_3 w_4$ are used to decode w_3 , and so on.

The above context-dependent quantization can actually be extended to decode speech signals corrupted by noise as well. Assume a noisy speech codeword sequence $[b(Q(y_{t-1})) = m, b(Q(y_t)) = i, b(Q(y_{t+1})) = n]$ is observed, where y_{t-1}, y_t, y_{t+1} are all noisy parameters, and assume that the correct codeword for the corresponding clean speech parameter \hat{y}_t in the middle is $b(Q(\hat{y}_t)) = k$, where \hat{y}_t is the clean speech version of y_t , and the N possible values of the codeword k has a distribution $\{P_i^{mn}(k), k = 1, 2, \dots, N\}$. In other

words, $P_i^{mn}(k)$ is the probability of the correct codeword being k (that is, $b(Q(\hat{y}_t)) = k$) when the observed noisy speech codeword sequence is $[b(Q(y_{t-1})) = m, b(Q(y_t)) = i, b(Q(y_{t+1})) = n]$. These probabilities $\{P_i^{mn}(k), k = 1, 2, \dots, N\}$ can be easily estimated based on the frequency counts of such codeword sequences $[b(Q(y_{t-1})) = m, b(Q(y_t)) = i, b(Q(y_{t+1})) = n]$ and $[b(Q(y_{t-1})) = m, b(Q(\hat{y}_t)) = k, b(Q(y_{t+1})) = n]$ in a corpus including corresponding noisy and clean speech for some noisy conditions. With these probabilities, minimum mean squared error (MMSE) estimation for the codewords for clean feature parameters can be obtained as the conditional expectation values,

$$\begin{aligned} \hat{z}_i^{mn} &= E[z_i^{mn} \mid b(Q(y_{t-1})) = m, b(Q(y_t)) = i, b(Q(y_{t+1})) = n] \\ &= \sum_k P_i^{mn}(k) z_k^{mn}, \end{aligned} \quad (6.3)$$

where z_i^{mn} is the context-dependent representative parameter obtained in Eq. 6.2, and \hat{z}_i^{mn} is the MMSE estimate of the representative parameter from noisy codewords considering context dependency.

Note that the above formulation is for quantization under the DSR framework, but it applies equally to feature transformation for robust speech recognition apart from DSR, in which each original feature parameter y_t is transformed into z_i^{mn} for recognition purposes based on the quantization and its context.

All the above applies equally to all different quantization schemes. Below we apply it to Histogram-based Quantization (HQ).

6.2.2 Context-dependent HQ

In Eq. 6.2 the representative parameter z_i^{mn} is determined given a set of partition cells. However, for HQ the partition cells are dynamic and varying for every time t ; that is, every y_t in Eq. 6.2 is associated with a different set of partition cells. Fortunately, as in Fig. 6.2, we see that even if the partition cells Q_i for HQ are dynamic on the horizontal scale, there are another set of partition cells D_i on the vertical scale which are fixed. The

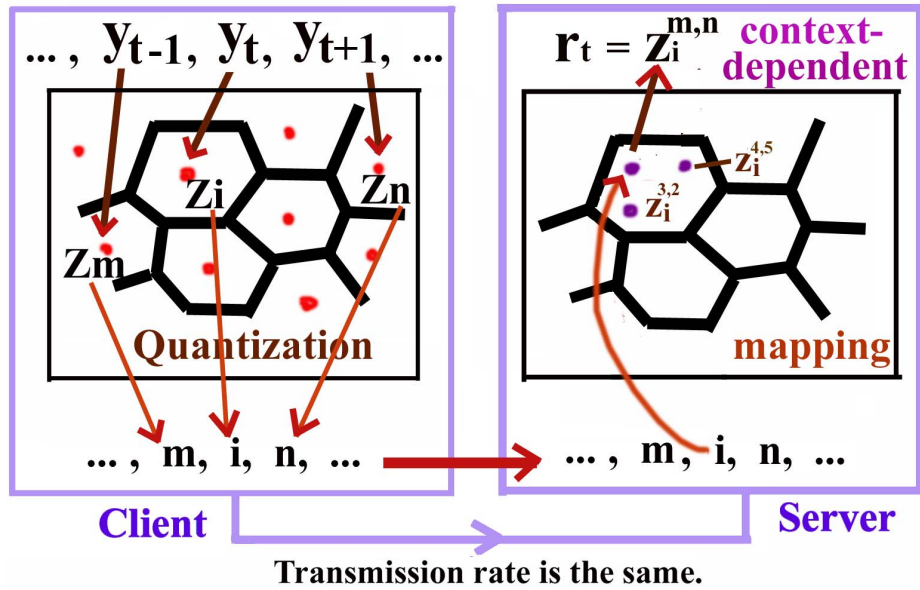


Figure 6.1: Context-dependent quantization with left and right context codewords m and n .

dynamic histogram $C(v)$ defines the relationship between the two sets of partition cells Q_i and D_i . As a result, context-dependent HQ is easily achieved by performing Eq. 6.2 on the vertical scale, and then transforming it back to the horizontal scale using the standard Gaussian histogram $C_0(v)$. In other words, for context-dependent HQ we can have

$$\bar{z}_i^{mn} = \frac{1}{L_i^{mn}} \sum_{\substack{y_t \in Q_i \\ b(Q(y_{t-1}))=m \\ b(Q(y_{t+1}))=n}} C(y_t) \quad (6.4)$$

and

$$z_i^{mn} = C_0^{-1}(\bar{z}_i^{mn}). \quad (6.5)$$

Thus the contextual information represented by z_i^{mn} as obtained from Equations 6.4 and 6.5 is very similar to that of Eq. 6.2.

The context dependency relationships for HQ as analyzed above can then be similarly extended as in Eq. 6.3 to estimate the representative parameters \hat{z}_i^{mn} from noisy codewords. Here, z_i^{mn} obtained from Eq. 6.5 can be used with the probabilities

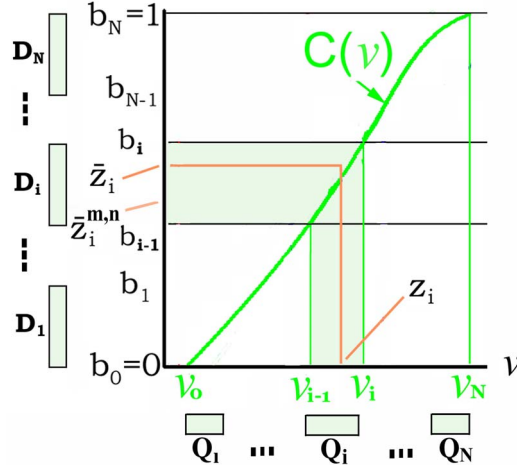


Figure 6.2: Context-dependent Histogram-based Quantization (HQ).

$\{P_i^{mn}(k), k = 1, 2, \dots, N\}$ estimated from corresponding clean/noisy corpus for MMSE estimation as in Eq. 6.3.

6.3 Experimental Results

6.3.1 Context-dependent HQ as a Robust Feature Transformation Method

In the first set of experiments, we considered the case of robust speech recognition apart from the DSR environment, in which context-dependent HQ was used as a feature transformation technique, that is, each feature parameter y_t , either clean or disturbed by noise, is transformed to the representative parameter z_i^{mn} in Eq. 6.5 or \hat{z}_i^{mn} in Eq. 6.3, for the corresponding partition cell considering the context codewords m, n , to be used for recognition. Note that the multi-condition training set and the corresponding clean speech training set in AURORA 2 were used to estimate the probabilities $\{P_i^{mn}(k)\}$ used in Eq. 6.3.

The results in Fig. 6.3 were all under clean-condition training, organized in three parts: (a) averaged over all SNR values (20 dB to 0 dB) but separated for different types

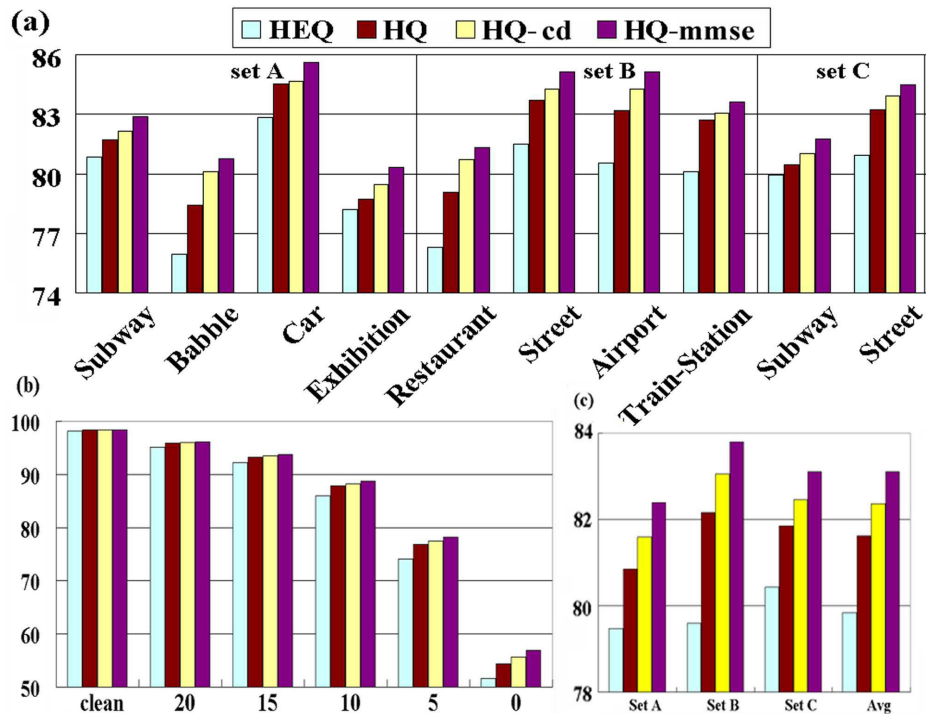


Figure 6.3: Word accuracies for HEQ, HQ, HQ-cd and HQ-mmse under clean condition training: (a) averaged over all SNR values (20 dB to 0 dB) but separated for different types of noise; (b) averaged over all types of noise but separated for different SNR values; and (c) averaged over all types of noise and all SNR values (20 dB to 0 dB) for different testing sets.

of noise, (b) averaged over all types of noise but separated for different SNR values, and (c) averaged over all types of noise and all SNR values (20 dB to 0 dB) for testing sets A, B, and C, respectively. The first two bars in each set in Fig. 6.3 are respectively the recognition word accuracies for the well-known histogram equalization (HEQ) alone [43, 44], and the original HQ [46, 50, 51, 52], which transforms each feature parameter y_t to the HQ representative value z_i without considering the context codewords. The next two bars are then those for context-dependent HQ, using context-dependency trained from a clean speech corpus with Eq. 6.5 for the third bar (HQ-cd) and using MMSE estimates trained with a multi-condition training corpus with Eq. 6.3 for the last bar (HQ-mmse). All the experiments reported here for HQ were based on order-statistics over segments of

the most recent parameter values as mentioned in section 6.2.1, so there was no time delay. Although better results were obtainable if the no-delay condition was removed, they are not shown here due to space limitations. Here HEQ was performed in exactly the same way as HQ, based on a moving segment of the most recent T parameters, and the same value of $T = 100$ was used.

It can be found that HQ (2nd bar) consistently outperformed HEQ (1st bar), while context-dependent HQ (both HQ-cd and HQ-mmse in the 3rd and 4th bars) consistently and significantly outperformed HEQ: in particular MMSE estimation trained with a noisy corpus (4th bar) resulted in much more robust features for recognition. Increasing improvements are apparent in Fig. 6.3 in all cases. In addition, context-dependent HQ trained with clean speech (HQ-cd, 3rd bar) offered greater improvement than original HQ (HQ, 2nd bar) for speech-like noise such as babble, restaurant, and airport, probably because the context-dependent characteristics for these types of noise have been more or less included in the transformation. Furthermore, HQ-mmse (4th bar) consistently outperforms HQ-cd (3rd bar) (Fig. 6.3(c)), which verifies that the context dependency trained from noisy corpora is useful even for unseen noisy environments (e.g. sets B and C).

SNR	Clean	20 dB	15 dB	10 dB	5 dB	0 dB
TC	98.31	95.16	89.55	70.94	43.79	18.75
HQ-mmse	98.37	96.05	93.66	88.71	78.24	56.80
TCg	93.84	84.35	73.55	52.38	27.81	9.29
HQ-mmseg	97.20	93.99	91.09	84.77	72.51	49.60

Table 6.1: Comparison of Transform coding (TC) and HQ-mmse, without and with GPRS transmission errors (TCg and HQ-mmseg) for different SNR values.

6.3.2 Context-dependent HQ as a Feature Quantization Method for DSR

We next considered context-dependent HQ as a feature quantization method in DSR. In Fig. 6.4 in each set the first three bars are respectively the word accuracies for the well-known HEQ followed by the conventional SVQ (HEQ-SVQ), original HQ (the same as

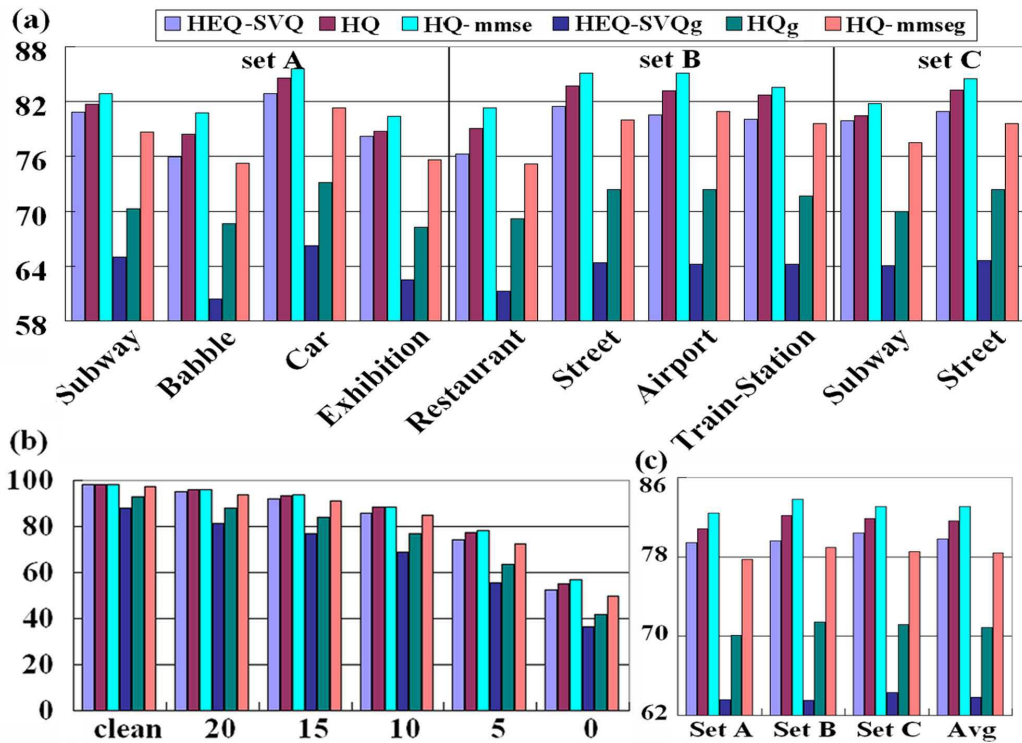


Figure 6.4: Comparison of HEQ-SVQ, HQ, and HQ-mmse, and those with GPRS transmission errors (HEQ-SVQg, HQg, and HQ-mmseg): (a) averaged over all SNR values (20 dB to 0 dB) but separated for different types of noise; (b) averaged over all types of noise but separated for different SNR values; and (c) averaged over all types of noise and all SNR values (20 dB to 0 dB) for different testing sets.

the 2nd bar in Fig. 6.3), and context-dependent HQ with MMSE estimation (HQ-mmse, the same as the 4th bar in Fig. 6.3), all at 4.4 kbps without transmission errors, and the next three bars (HEQ-SVQg, HQ-g, HQ-mmseg: the label "g" indicates GPRS) are those suffering from GPRS transmission errors for a client traveling at 3 km/hr. Fig. 6.4 (a) is averaged over all SNR values (20 dB to 0 dB) but separated for different types of noise, (b) is averaged over all types of noise but separated for different SNR values, and (c) is averaged over all types of noise and all SNR values (20 dB to 0 dB) for testing sets A, B, and C, respectively.

We first examined the effect of quantization and compression on recognition accuracy, assuming there were no transmission errors. The performance of original HQ (2nd

bar) consistently outperformed HEQ-SVQ, while HQ-mmse (3rd bar) was consistently and significantly better than original HQ, as shown in Fig. 6.4(a)-(c). This verifies the effectiveness of context-dependency. Improvements were even more significant for lower SNR cases (Fig. 6.4(b)), and for several types of non-stationary noise (Fig. 6.4(a)), which indicates where context-dependency is more helpful. We then examined the effect of transmission errors in the last three bars in Fig. 6.4. For HEQ-SVQ, the performance degradation caused by GPRS (4th bar compared to 1st bar) is more serious for lower SNRs. Clearly, features corrupted by noise are more susceptible to transmission errors. The improvements that HQ and context-dependent HQ offered over HEQ-SVQ when transmission errors were present (5th, 6th bars to 4th bar) are consistent and very significant. For example, in the case of 10 dB SNR with GPRS, HQ-mmseg (6th bar) offered an accuracy of 84.77% compared to 69.84% for HEQ-SVQg (4th bar). In addition, it is interesting that the improvements offered by HQ-mmse over HQ when transmission errors were present (6th bar to 5th bar) are much more significant as compared to those comparison without transmission errors (3rd bar to 2nd bar). This indicates that context-dependency among speech codewords is actually very strong, and remains helpful even after heavy disturbance due to environmental noise and transmission errors, and the error propagation problem is not serious here. This is probably because even if there are erroneous context codewords, they may only change the representative parameter z_i^{mn} of the current frame within the same partition cell Q_i in Fig. 6.2, which is actually very limited. Also, the decoding here used only local context codewords, i.e., based on the two neighboring undecoded codewords only; thus erroneous codewords actually do not propagate. It is clear from Fig. 6.4 that HQ-mmse is robust against both environmental noise and transmission errors.

Also shown in Table 6.1 are the detailed word accuracies of transform coding (TC) [21] compared with HQ-mmse, either without or with GPRS transmission errors for all SNR values, average over all noise types. The performance of TC seriously degrades

when transmission errors are present (3rd row vs. 1st row), probably because exploiting speech correlation by grouping several consecutive frames into one block and quantizing them together may be sensitive to transmission errors. In contrast, error propagation is not a serious problem here for HQ-mmseg (the performance degradation is much smaller for the comparison of 4th and 2nd rows).

6.4 Summary

In this chapter, We have proposed context-dependent quantization, a new concept for distributed and/or robust speech recognition. Improved recognition performance was obtained consistently across a wide range of environmental noise and transmission error conditions.

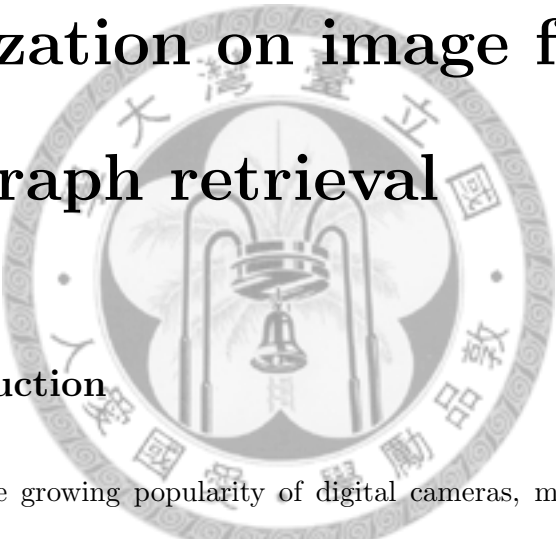


This page intentionally left blank.



Chapter 7

Application of Dynamic Quantization on image features for photograph retrieval



7.1 Introduction

With the growing popularity of digital cameras, many people have saved huge collections of digital images. A resulting challenge is how to exactly to find a desired photo, because it is simply impossible to browse through the entire collection. This calls for an efficient photo retrieval approach.

Content-based image retrieval has been an active research area for years, many successful approaches of which are based on low-level image features, implemented using “query by example” [62, 63]. However, this is not very attractive in practice, because it requires that the user provide an example photo as the query. In fact, most users prefer high-level semantic descriptions of photos that use words as queries, such as who, what, when, where (objects/events) and so on, but again, this is not an attractive solution if it

requires manual annotation of each individual photo. This observation has led to the idea of annotating photos with speech [64, 65]. When such a spoken photo annotation is taken as a spoken document, the problem becomes one of spoken document retrieval.

Many spoken document retrieval approaches have been successful in spotting the query term in the spoken documents, but these approaches usually suffer from the problem of word usage diversity, i.e., the query and its relevant documents may use different sets of words. This problem is especially serious for photo retrieval as considered here, because the annotation may describe location (where), but the query may ask for a person (who), i.e., both annotation and query are typically free-form and vary significantly. In spoken document retrieval, semantic matching strategies have been developed to solve the word usage diversity problem by discovering latent topics inherent in the query and documents. Latent semantic indexing (LSI) and probabilistic latent semantic analysis (PLSA) are two typical examples [66, 67]. In both cases the relevance score between a query term and the spoken documents can be obtained via a set of latent topics, and relevant documents can be retrieved even using query terms that are completely different from those used in the documents. This is because common topics are usually found in sets of documents that each include a set of similar terms, or in sets of terms that each appear in a set of similar documents, and such topical information is used in retrieval.

The above semantic matching methods have not solved the photo retrieval problem described here either. Assume that photo annotation can be formulated into six categories: *who*, what (*object* and *event*), *when*, *where*, and *others*. When labeling a photo, users typically select only one or two categories. As such, related photos may not be labeled using similar terms (e.g., some may be labeled by *where* and some by *who*), and the relationships among terms in different categories cannot be trained using latent topics. For example, given a *where* query, many photos taken at that location may not be retrieved if they are annotated with words in other categories. Also, users generally annotate far too few photos

to train such topic models. Moreover, it is even difficult to define what a “topic” should be for photos. For example, should photos of different people taken at the same location belong to the same topic, or should photos of the same people but taken at different locations belong to the same topic? In other words, the above six categories of labels are orthogonal, but user annotations are usually very sparse. Thus the photo retrieval problem is quite different from the well-investigated spoken document retrieval problem, even if photos have spoken annotations.

Considering all the above, user annotations could not provide enough information to build the semantic relationships among photos. If we could extract some similar “terms” from image features for photos of the same topic, the semantic link among photos with sparse annotations would become stronger through the extracted image “terms.” Note that the terms used in semantic analysis are discrete, while low-level image features are continuous. Therefore, how to quantize these image features to “terms” is a key issue before semantic analysis.

The image feature quantization considered here aims to extract common “terms” from photos having the same topic and distinguished “terms” from photos with different topic. This is because common terms could build stronger semantic relationship for photos with the same topics, and distinguish terms could discriminate photos with different topics. Conventional quantization with fixed and pre-trained codebook cannot well represent image features. On one hand, if the partition cells for defining a color bin are fixed, the same scene taken from different cameras may have very different color histogram features. In this situation, the same scene taken from different cameras could not be retrieved because their image “terms” would be quite different with fixed quantization codebook. Therefore, it is important to apply the concept of dynamic quantization to define dynamic partition cells for photos taken from different cameras. On the other hand, if the representative codewords for the color histogram features and Gabor texture features are fixed, photos with different

topics may locate on close positions in some feature dimensions, and they would be quantized to the same codewords in these dimensions. Extracting common terms from photos with different topics is harmful for semantic analysis because the topical information for photos would become less clear. Therefore, it is important to dynamically define the representative codewords to preserve the discriminative information in the quantization process.

Considering all the above, in this chapter we propose a user friendly semantic-based photo retrieval approach using Fused image/speech/text features. We use low level image features to derive the basic links among photos, since these features are really the universal language describing photos. But we train semantic models to analyze the topics of the photos using PLSA. Because the "terms" in PLSA has to be discrete, while the low level image features have continuous real values, for each given photo we use low level image features to select a group of "cohort photos" from the photo archive with similar image characteristics as the "terms" describing the image characteristic of the photo, which is then fused with speech/text features if some annotation is added by the user. The speech/text annotation can be very "sparse," i.e., only very few words regarding the semantics (e.g. where or who) are needed for only a small portion of photos. In this way, the image/speech/text features are fused with PLSA topic analysis, to be used in PLSA semantic-based retrieval. The sparse text/speech annotation serve as the interface for the user to access the whole photo archive, since the other photos not annotated are actually linked by the semantics of the image features based on PLSA.

The rest of this chapter is organized as follows. Section 7.2 introduces the overall photograph retrieval system. Section 7.3 describes the basic formula of PLSA. Color feature extraction with dynamic partition cells and texture features are introduced in Sections 7.4. In section 7.5, we introduce how to extract image "terms" from low-level image features by using dynamically defined representative codewords. In section 7.6, we construct document for each photo based on photo annotations and the image "terms" and use PLSA to analyze

the topics of photos for photo retrieval. In section 7.7, we perform image clustering based on PLSA model. Experimental settings and results are offered in Section 7.8. Conclusions are given in the last section.

7.2 Overview of the proposed approach

As shown in Fig. 7.1, the proposed approach includes a preparation phase (left part) and a retrieval phase (right part). In the preparation phase, the low level image features are first extracted and used to select the “cohort photos” (Block (B) and (C), middle of the figure) for each photo in the photo archive (Block (A), upper left corner). The cohort photos, used as “terms”, together with the text/speech annotation by the user, if available, are then fused to construct a “document” for each photo (Block (D), lower right corner). These “documents” and their “terms” are then used to train the PLSA topic model (Block (E) and (F), upper right corner). The user query can then include only very few words, in either speech or text form. Semantic-based retrieval by PLSA gives the desired photo in the retrieval phase on the right.

7.3 Probabilistic latent semantic analysis (PLSA)

Probabilistic latent semantic analysis (PLSA) is a probabilistic framework for semantic-based retrieval that uses a set of latent topic variables, $z_k, k = 1, 2, \dots, K$, to characterize the term-document co-occurrence relationships as shown in Fig. 7.2 [67]. A query Q is treated as a sequence of n observed terms, $Q = t_1 t_2 \cdots t_j \cdots t_n$, while document d_i and term t_j are both assumed to be independently conditioned on an associated latent topic z_k . The conditional probability of observing term t_j in document d_i thus is

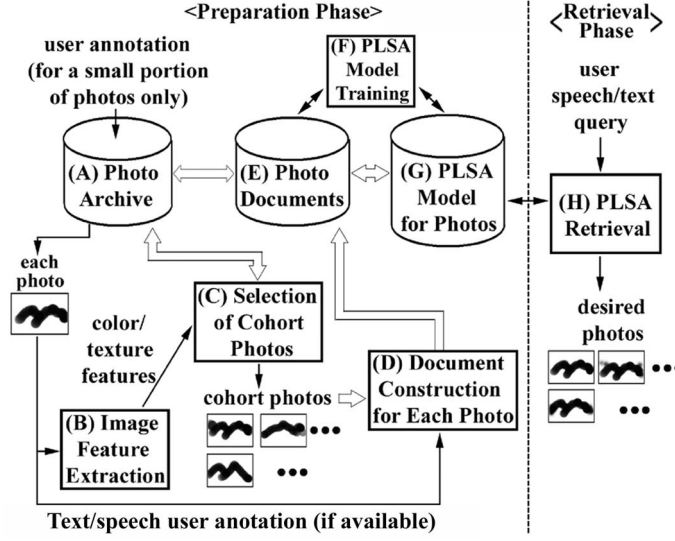


Figure 7.1: The proposed approach: preparation phase includes document construction for each photo and PLSA model training for photo documents, while retrieval phase is based on PLSA.

parameterized as

$$P(t_j|d_i) = \sum_{k=1}^K P(t_j|z_k)P(z_k|d_i), \quad (7.1)$$

where the probabilities $P(t_j|z_k)$ and $P(z_k|d_i)$ are obtained from the PLSA model, which is trained using the EM algorithm by maximizing a total likelihood function. When the terms in the query Q are further assumed to be independent given the document, the relevance score between the query and document is then expressed as

$$P(Q|d_i) = \prod_{j=1}^n \left[\sum_{k=1}^K P(t_j|z_k)P(z_k|d_i) \right]. \quad (7.2)$$

In this way, retrieval is based on topics rather than on terms, i.e., topically relevant documents can be retrieved even using a different set of terms. Such a latent semantic concept of retrieval is highly desired in the photo retrieval problem here, but there are obvious limitations when using it as-is. For photos, topics clearly have to do with scene and image features such as colors and textures, since these—rather than the few words in the annotation—are the universal language that describes all photos. However, these image

features are represented using real numbers, while terms in the PLSA model are discrete. That is why we use such image features to select cohort photos with similar image characteristics, and use these cohort photos as the discrete terms in PLSA document construction, as we explain below in section 7.6.

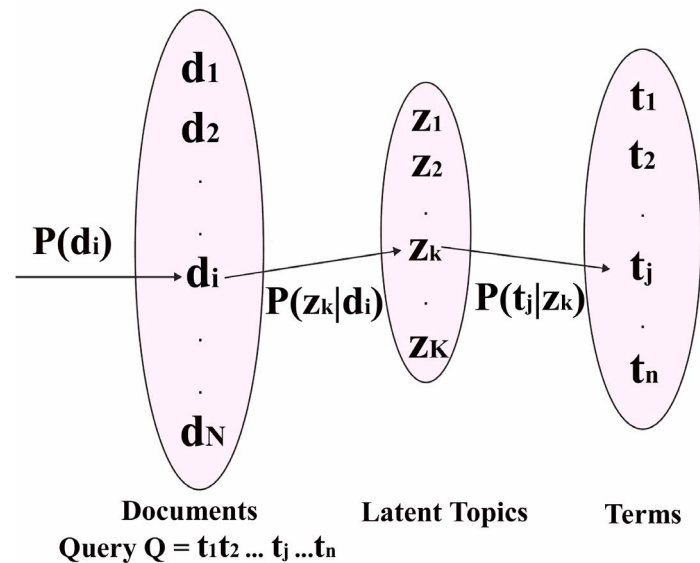


Figure 7.2: PLSA-based retrieval model

7.4 Low-level image feature extraction

7.4.1 Dynamic color features from the images

Color histogram popularly used in image retrieval is adapted here [68]. Each photo k can be represented by a color histogram H_k , in which each entry $H_k(i)$ is the number of pixels belonging to the color bin i . The HSV color space is quantized into 166 colors, including 18 levels of hues (H) * 3 levels of saturation (S) * 3 levels of values (V) + 4 levels of grays [68]. The distance $d_{k,l}$ between two photos k and l is then defined by the L2 distance

measure,

$$d_{k,l} = \sum_{i=0}^{i=N-1} (H_k(i) - H_l(i))^2, \quad (7.3)$$

where $N=166$ here. If we use fixed quantization for the color space (H,S,V plus grays) for all photos, the same scene from different cameras may have very different color histograms. This is why we developed dynamic quantization schemes to derive dynamic color features in order to handle photos taken by different cameras.

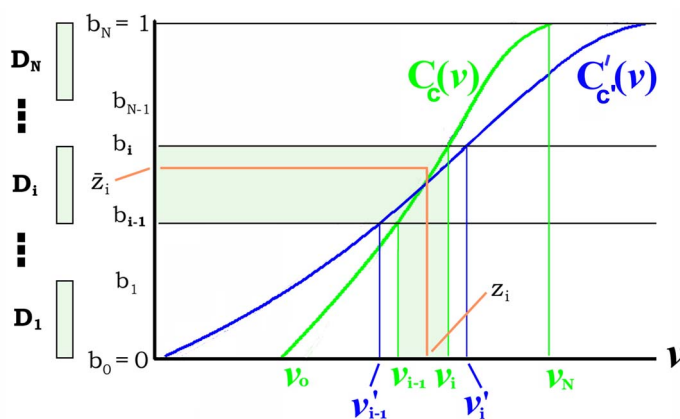


Figure 7.3: Dynamic color features defined by histogram-based quantization.

The dynamic quantization for color space uses the histogram-based quantization (HQ) in section 3, previously developed for distributed and/or robust speech recognition. In this scheme as shown in Fig. 7.3, the partition boundary v_i of a color bin i , whether for H, S, V or gray, for a camera c is based on the histogram of the pixel values for photos taken by the camera c . The pixel values of photos taken by camera c are first sorted to produce a cumulative distribution function $C_c(v)$, or histogram, for H, S, V or gray, which is different for different camera, where $C_c(v_0) = b_0 = 0$ and $C_c(v_N) = b_N = 1$, v_0 and v_N are respectively the minimum and maximum pixel values. On the other hand, N partition cells, $\{D_i = [b_{i-1}, b_i], i = 1, 2, \dots, N\}$ are uniformly defined on the vertical scale

$[0, 1]$. They are transformed to the horizontal scale by the dynamic histogram $C_c(v)$, to be the N partition cells $\{[v_{i-1}, v_i], i = 1, 2, \dots, N\}$ on the horizontal scale for the quantization of the pixel values, where $C_c(v_i) = b_i$. Thus the partition cell $[v_{i-1}, v_i]$ on the horizontal scale is defined differently for different camera c . As shown in Fig. 7.3, when a different histogram $C_c(v)$ is used for a different camera c , the partition cell on the horizontal scale is changed to $[v'_{i-1}, v'_i]$, where $C'_c(v'_{i-1}) = b_{i-1}$ and $C'_c(v'_i) = b_i$. It has been shown that the quantization defined in this way is more robust because the different statistical behaviors of photos from different cameras are absorbed by the histograms [46, 50].

7.4.2 Texture features from images

The Gabor texture features previously proposed and frequently used for image retrieval, produced by a bank of Gabor filters at multiple scales and orientation [69] are adapted here, including four scales and six orientations.

7.5 Document generation for photos

7.5.1 Image “terms” extraction and “Cohort Photos” selection from low-level image features

Two photos with different topic have different color histograms, but the difference may be significant only on certain color bins. If these two photos both have few pixels on many color bins, the quantized results on these bins would be the same, and there would be many common “terms” for these photos with different topics. To solve the above problems, the representative codewords should be dynamically defined to distinguish the difference in main color bins and ignore other color bins in the photo quantization process.

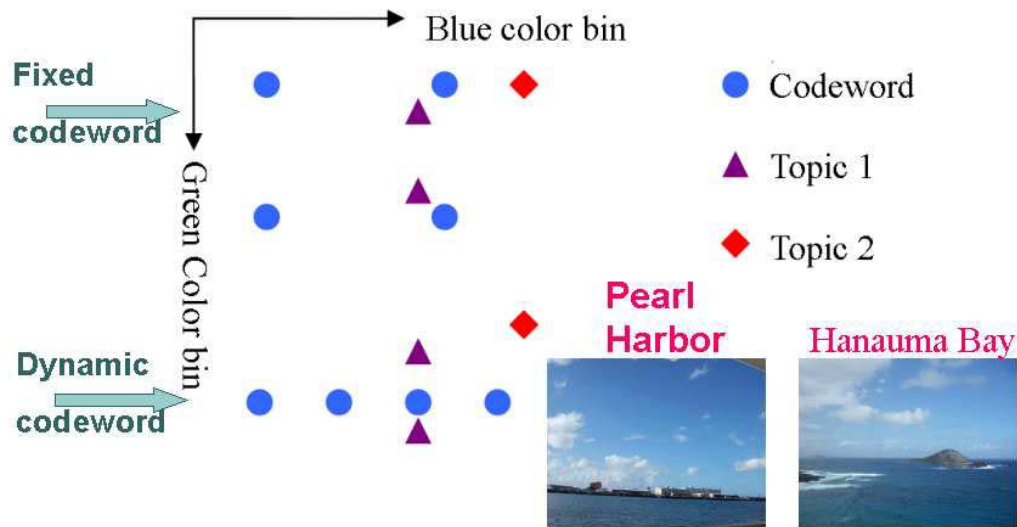


Figure 7.4: Image feature quantization with fixed and dynamic codewords

As shown in Fig. 7.4, to distinguish photos with different topics (Pearl Harbor and Hanauma Bay), the difference in blue color bins should be more important than other bins. With fixed representative codewords, the codewords may be uniformly distributed on each color bin, and the quantization resolution is the same for all bins. Two photos with different topics may be quantized to the same codeword and two photos with the same topic may be quantized to different codewords. These image “terms” extracted by the quantization process with fixed codeword cannot well represent the topical information of the photo, and they may cause ambiguity in latent semantic analysis. By contrast, with dynamic representative codewords, the color bins with rich information would be quantized finer and other color bins with less information would be quantized looser. Because the difference in the distinguished color bin is emphasized in the quantization process, two photos with the same topic can be quantized to the same codeword and two photos with different topics can be quantized to different codewords.

To make the representative codewords more discriminative on the color bins with rich information, a simple way is to use each photo feature vector as a codeword in the

feature space. For example, in the album of Hawaii traveling each photo in the photo album is taken as a codeword and these codewords could provide more discriminative information on blue color bins than other color bins. With all photos in the album as the representative codewords, each photo could be represented as a group of “cohort photos” from the photo archive with similar image characteristics. We use a total of three methods to select cohort photos. The L2 distance in Eq. 7.3 is used as the distance measure for both color features in Section 7.4.1 and texture features in Section 7.4.2. The first method is based simply on the combination of ranks (i.e., the closest top 15 photos) with respect to color and texture features. In the second and third methods, we use one set of features (color or texture) to select the top 30% photos as the candidates, and then use the other set of features to re-score (or re-rank) the selected photos. These three methods are actually complementary to each other, so they are respectively used to generate a total of three sets of cohort photos for each given photo, to be used to construct the photo documents as presented below.

7.5.2 Construction of “Documents” with fused features for the photos

Each photo in the archive must be represented as a document described by the discrete terms used in PLSA modeling. We first define as a term every photo in the archive (thus we have discrete terms), and then we further represent each photo as a document composed of the terms for all of its cohort photos, which are formed as described above using color and texture features. These terms jointly describe the image and scene characteristics of each photo.

As described above in Section 7.5.1, we use three methods to extract cohort photos based on image similarity. For each method, the terms for the top 15 most similar photos are included in the document for a given photo. When the same photo appears in more than one of the three top-15 lists, the corresponding term frequency in the photo document is raised to emphasize its salience.

On the other hand, the speech/text annotation for a given photo is also included in its document. This is straightforward: we simply define word, character, and syllable as terms (the annotation is in Mandarin Chinese) for word- and subword-level indexing as in conventional spoken document retrieval. The subword units (character and syllable) are used to handle OOV words as usual. For speech annotation, utterances are represented in word- and subword-based lattices and all arcs of the lattices with posterior probabilities are included as the terms. These word and subword terms in the lattices are given less weight in PLSA training, in order to reduce interference from noisy word/subwords, but still add indexing functionality if that term appears with greater weight in the lattices. In this way, we construct photo documents with fused image/speech/text features.

7.6 Latent semantic photo retrieval with fused image/speech/text features

The PLSA model is then trained with the constructed documents with terms based on fused image/speech/text features. Because few photos are annotated, the obtained topics are based primarily on image semantics, i.e., photos of the same topic look similar. The input query can be in either speech or text form, represented as a sequence of observed word- or subword-based terms, and the relevance score with respect to each photo (i.e., the document with fused image/speech/text features as discussed in Section 7.5) is then calculated based on PLSA as in Eq. 7.2. Note that there are four types of terms in each photo document: image terms, word terms, character terms, and syllable terms. For unannotated photos, the latter three types of terms are simply blank. The central idea of PLSA-based latent semantic retrieval is that a query and a document may have a high relevance score even if they do not share any terms in common, as long as they share the same topic. In this way, unannotated photos that have no terms in common with the text/speech query (since

the query contains only word/character/syllable terms) can also be retrieved, because the matching is not based on term co-occurrences but on latent topics.

7.7 Image Clustering

Given the above PLSA model, the likelihood that d_i addresses the latent topic T_k (i.e. $P(T_k|d_i)$), we could classify each image (document i) to the topic (or cluster k) with the highest likelihood.

$$c_i = \arg \max_k P(T_k|d_i). \quad (7.4)$$

On the other hand, the representative images R_k of each cluster (i.e. topic k) could be selected as the word (i.e. term t_j) which maximize the term frequency in the latent topic T_k (i.e. $P(t_j|T_k)$).

$$R_k = \arg \max_j P(t_j|T_k). \quad (7.5)$$

7.8 Preliminary Experimental Results

7.8.1 Photo archive

In the preliminary experiments, an archive of 347 photos for a trip to Hawaii was used. They were taken by two different cameras, a Fujitsu and a Canon. Only 12% of these photos were annotated by the users with text labels, in which each photo was annotated by only one of the six categories: *who*, *what* (*object* and *event*), *when*, *where*, and *others*. Each annotation includes 1 to 5 Chinese words or 2 to 6 Chinese characters. Speech annotation was not done for lack of time.

7.8.2 Dynamic color features across cameras

The first experiment measured how the dynamic color features using histogram-based quantization presented in Section 7.4 can help users easily sharing photos taken by different cameras. We arbitrarily took 18 scenes, each with a photo taken by the two different cameras respectively, Fujitsu and Canon (F_1, F_2, \dots, F_{18} from Fujitsu, and C_1, C_2, \dots, C_{18} from Canon, (F_k, C_k) are pictures taken from the same scene k). For each given photo from one of the camera (i.e., F_k), the distance measure in Eq. 7.3 using the color features described in Section 7.4 was used to select the closest photos from all the other 346 photos in the archive. The rank of the corresponding photo for the given scene k taken by the other camera (i.e., C_k) was then obtained. The average rank for these 36 images is 6.5 using the fixed color features, and 4.8 using the proposed dynamic features with histogram-based quantization. This verified that the proposed dynamic color features can reduce the mismatch of pixel value distributions between different cameras.

7.8.3 Latent semantic photo retrieval

Fig. 7.5 shows one example of the first 9 photos retrieved by the text query “Hanauma Bay (in Chinese)”. In fact only one photo in the archive was annotated with “Beautiful Hanauma Bay (in Chinese),” but many related photos were actually correctly retrieved because of the fused image/speech/text features and semantic approach of PLSA. However, in Fig. 7.5, photos of ranks (5) and (8) were actually taken at “Pearl Harbor,” and are therefore irrelevant, but probably only the user can identify such difference. This is why the performance of semantic-based photo retrieval is difficult to evaluate, because very often only the users themselves can determine whether a photo is relevant or not. As another example, the query “sun rise” may retrieve many photos of “sun set,” while only the user knows which one is which. This is different from the task of “query by example”

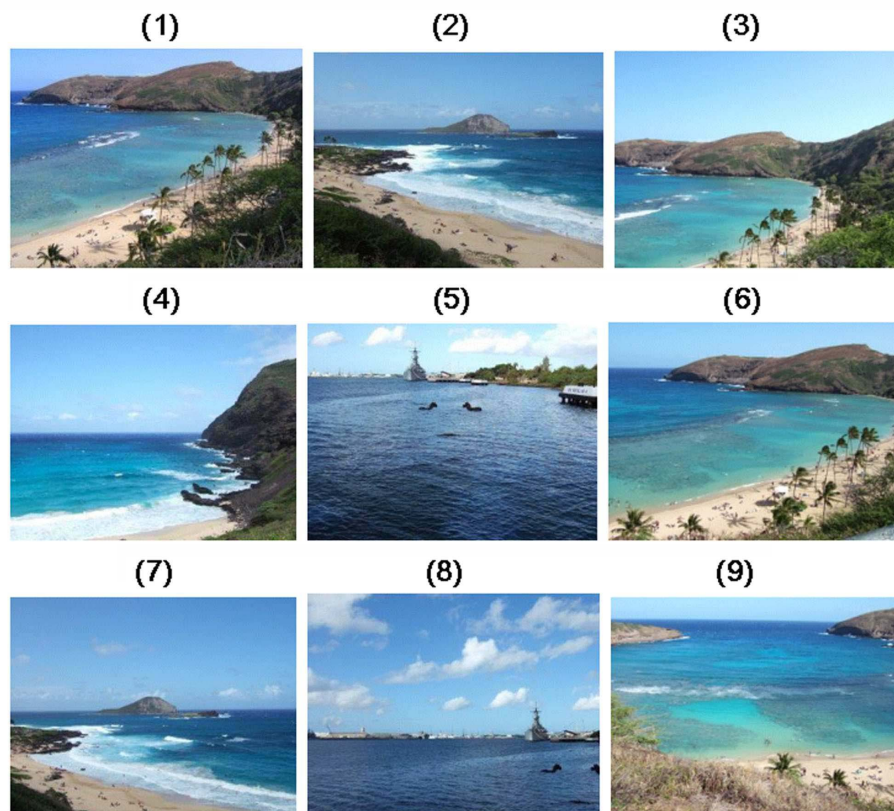


Figure 7.5: Retrieved photos by the text query "Hanauma Bay"

retrieval system, in which the relevant images are simply those close to the query example. This is why here we didn't evaluate the overall precision/recall rates.

In our preliminary experiment, two users participated in the test, each giving 40 text queries and 40 speech queries. Each query includes 1 to 5 Chinese words, or 2 to 6 Chinese characters (or syllables). 30% of the speech queries include OOV words. For each query, the system displayed a ranked list of the retrieved photos. The users were asked to identify the first 5 photos along the given list he or she recognizes as irrelevant, from which 5 precision rates were calculated from rank 1 photo to each of the 5 irrelevant photos. For example, if the second irrelevant photo was of rank 8, the corresponding precision rate is 0.75. Table 7.1 summarizes the results. For example, for text query the second irrelevant

Table 7.1: Average precision and rank for the first few irrelevant photos retrieved (with dynamic codewords)

First few irrelevant photos	Average precision (avg. rank for the irrelevant photo)	
	Text query	Speech query
1st	0.776 (4.5)	0.437 (1.8)
2nd	0.764 (8.5)	0.422 (3.5)
3rd	0.744 (11.7)	0.417 (5.1)
4-th	0.715 (14.0)	0.403 (6.7)
5-th	0.678 (15.5)	0.380 (8.1)
Average of five	0.735	0.412

Table 7.2: Average precision and rank for the first few irrelevant photos retrieved (with fixed codewords)

First few irrelevant photos	Average precision (avg. rank for the irrelevant photo)	
	Text query	Speech query
1st	0.241 (1.3)	0.132 (1.2)
2nd	0.319 (2.9)	0.174 (2.4)
3rd	0.233 (3.9)	0.147 (3.5)
4-th	0.220 (5.2)	0.134 (4.6)
5-th	0.234 (6.5)	0.157 (5.9)
Average of five	0.249	0.149

photos have an average rank of 8.5 and a corresponding precision of 0.764, and the speech queries gave relatively lower performance clearly due to the OOV words and very high word error rates. Table 7.2 shows the results for photo retrieval with fixed representative codeword. There are many irrelevant photos retrieved in the top rank, for example for text query the second irrelevant photos have an average rank of 2.9 and a corresponding precision of 0.319, and the speech queries gave worse retrieved results. These results shows the “image terms” extracted from image features quantized with fixed codebook are very harmful for latent semantic retrieval. This is because many irrelevant photos may share many common image “terms,” and these terms would cause confusing in PLSA. In contrast, the “image terms” extracted with dynamic codebook indeed help for latent semantic analysis as shown in Table 7.1.

Table 7.3: Evaluation for Image clustering

1	2	3	4	5
67%	19%	8%	4%	2%

7.8.4 Image clustering

The goal of the image clustering is to help the user to reduce the number of images to be browsed before getting the desired image. So we design the experiment to evaluate the performance. The system clusters all the photos into 21 clusters (i.e. about 17 photos in one cluster in average), and then displays four representative photos for each cluster. Giving the user a photo as the desired image, and count the number of browsed clusters before the user find the given image. There are three users, 40 queries per user, and 120 queries are performed totally.

It can be observed from Table 7.3 that the about 67% of the desired images are found in the first cluster. The result is pretty good because the system could help the user find the image in an efficient way. The number of clusters user have browsed before finding the desired image is almost below 3, only 6% exceed 3 as shown in Table 7.3.

7.9 Summary

In this chapter, we apply the concept of dynamic quantization on image features for photograph retrieval. The PLSA model, based on image terms extracted through dynamic quantization, significantly improves the photo retrieval results.

This page intentionally left blank.



Chapter 8

Conclusions and Future Works

8.1 Conclusions

Conventionally, quantization and robustness techniques are considered as two separate problems to solve. Feature quantization is for data compression and robustness techniques for handling noise disturbances. Most of all quantization methods obtain good results for clean speech and/or matched vector quantization (VQ) codebook conditions. However, the problems for environmental noise and transmission errors are not considered in the quantization process, because these issues are usually left out and are taken care of by robust front-end/back-end and error concealment techniques.

In this dissertation, a novel approach of dynamic quantization is proposed, automatically includes the desired robustness in the quantization process for robust and distributed speech recognition (DSR). The dynamic codebook could well represent noisy speech features and absorb the noise disturbance in the quantization block. These two dynamic quantization methods, Histogram-based Quantization (HQ) and context-dependent quantization, have been shown to be robust for all types of noise and all SNR conditions for either conventional speech recognitions systems, or DSR at all bit rates. In particular, context-dependent HQ utilizing strong speech correlation offered significant improvements and is

very robust against transmission errors. The configuration of HQ or context-dependent HQ could be easily scalable based on bandwidth or noise conditions. For future personalized and context-aware DSR environments, HQ or context-dependent HQ can be adapted to network and terminal capabilities, with recognition performance optimized based on environmental conditions. In addition, dynamic quantization applied on image features could extract image “terms,” and these terms well represent the semantics of photos for PLSA to build the semantic link among photos. In the experiments, while only 12% photos have very “spares” annotations, the retrieval results are very encouraging. This verified that dynamic quantization provides very distinguished image terms for PLSA training.

8.2 Future Works

Although many issues of environmental noise and transmission errors have been investigated in the dynamic quantization, there are still several important topics opened for further research. Each of our proposed approaches in the above five major chapters in this thesis may be further studied to determine some possible contributions. Following list is just to depict some issues of the dynamic quantization framework:

1. Extend the definition of quantization distortion measure to discriminate representative codewords for speech recognition,
2. Better integration of uncertainty source in Distributed speech recognition framework,
3. Jointly optimization of dynamic quantization (source coding) and channel coding,
4. Combination of various front-end feature processing approaches for improving the accuracy of the speech recognition system.

Based on the results and techniques that we have investigated and built-up, there are several topics that we could extend our current work for further research in dynamic quantization.

In Chapter 3, we successfully jointly consider the issues of compression and robustness, and the integration could be applied for both robust and distributed speech recognition. Another interesting idea is to jointly consider compression and discrimination issues. In Chapter 3, the hidden codebook on the vertical scale is derived based on uniform, Laplacian and Gaussian distribution via Lloyd-Max algorithm, which aims to minimize the overall quantization distortion. Every data point is treated with the same importance in the quantization process. However, there may be some regions in the feature space more critical than other regions. The critical region has smaller margin among HMM models and small distortion for samples in these critical regions could cause recognition errors. Therefore, the samples in the critical region should be carefully considered to enlarge the margin among HMM models. On the other hand, quantization distortion in some features may be more important than distortions in others. The quantization distortion sensitivity for different feature parameters should be integrated in the quantization distortion measure to optimize the recognition performance.

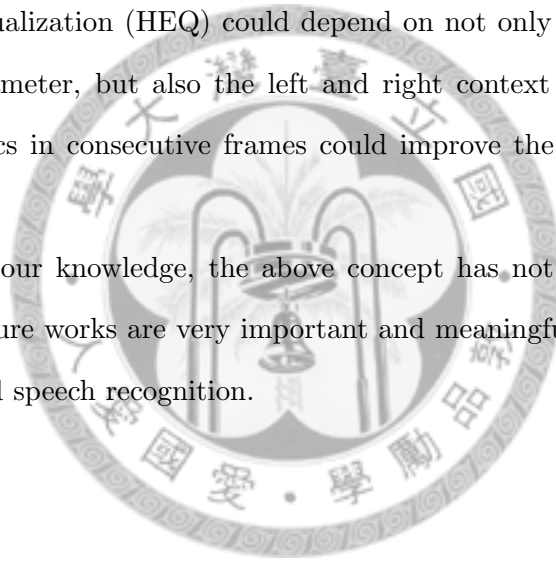
In Chapter 4, we jointly consider the uncertainty caused by both environmental noise and quantization errors. In Chapter 5, the reliability of received feature vectors is considered in Viterbi decoding in the third stage of error concealment. For distributed speech recognition, it would be better to jointly consider these three source of uncertainties: quantization distortion, environmental noise and transmissions. The above uncertainty estimation is derived from feature perspective. On the other hand, the reliability could be estimated based an entropy-based measure to determine the discriminating ability of a feature parameter in identifying the correct acoustic models [70, 72, 71]. The uncertainty or reliability estimated from feature or model perspective could be further integrated in Viterbi decoding to improve the recognition performance.

In the three-stage error concealment(EC) framework in Chapter 5, the error detection is based on the characteristics of HQ features. There is no channel coding scheme

applied on the encoded HQ symbols. If the source coding and channel coding are considered jointly, the recall and precision rates of error detection could be further improved. Also, with channel coding, the soft decision decoding at receiver could offer channel reliability information for weighted Viterbi decoding.

In Chapter 6, the context-dependent quantization exploiting speech correlation in the quantization process improves the robustness against environmental noise and transmission errors. This is probably because the speech context change could provide additional information for human perception and speech recognition. The concept of context-dependency could be also applied to other feature transformation methods. For example, the transformation of Histogram equalization (HEQ) could depend on not only the order-statistics of the current feature parameter, but also the left and right context parameter. The correlation of order-statistics in consecutive frames could improve the robustness of feature parameters.

To the best of our knowledge, the above concept has not been reported in the literature yet. These future works are very important and meaningful in the research area of robust and distributed speech recognition.



Bibliography

- [1] Special section on “Speech Technology in Human-Machine Communication,” *IEEE Signal Processing Magazine*, vol. 22, no. 5, Sep. 2005.
- [2] D. Pearce, “Enabling new speech driven services for mobile devices: An overview of the ETSI standards activities for distributed speech recognition front-ends,” *Proc. Applied Voice Input/Output Soc. Conf.*, May 2000.
- [3] V. V. Digalakis and L. G. Neumeyer and M. Perakakis, “Quantization of cepstral parameters for speech recognition over the world wide web,” *IEEE Select. Areas Commun.*, vol. 17, no. 1, pp 82-90, Jan. 1999.
- [4] J. -Y. Li, Bo Liu, R. -H. Wang and Li. -R. Dai, “A complexity reduction of ETSI advanced front-end for DSR,” *Proc. IEEE Int. Conf. Acoust. Speech, Signal Processing*, Apr. 2004.
- [5] A. Agarwal, and Y. M. Cheng, “Two-Stage Mel-Warped Wiener Filter for Robust Speech Recognition,” *Proc. ASRU99*, 1999.
- [6] J. -W. Hung and L. -S. Lee, “Comparative Analysis for Data-Driven Temporal Filters Obtained Via Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) In Speech Recognition,” *Proc. Eurospeech*, pp 1959-1962, 2001.
- [7] S. Vuren and H. Hermansky, “Data-Driven Design of RASTA-Like Filters,” *Proc. ICSLP*, 1996.
- [8] Ni-chun Wang, Jeih-weih Hung and Lin-shan Lee, “Data-driven temporal filters based

- on multi-eigenvectors for robust features in speech recognition,” *Proc. IEEE Int. Conf. Acoust. Speech, Signal Processing*, 2003.
- [9] S. Furui, “Cepstral Analysis Technique for Automatic Speaker Verification,” *IEEE Trans. Acoust. Speech Signal Processing*, 1981.
- [10] O. Viikki and K. Laurila, “Noise Robust HMM-based Speech Recognition Using Segmental Cepstral Feature Vector Normalization,” in *ESCA NATO Workshop Robust Speech Recognition Unknown Communication Channels*, pp 107-110, 1997.
- [11] J. Droppo and A. Acero and L. Deng, “Uncertainty Decoding with SPLICE for Noise Robust Speech Recognition,” *Proc. IEEE Int. Conf. Acoust. Speech, Signal Processing*, pp 57-60, 2002.
- [12] J. Droppo, L. Deng, and A. Acero, “Evaluation of the SPLICE algorithm on the Aurora2 database (web update),” *Proc. Eurospeech 2001*, pp 217-220, Sep. 2001.
- [13] J. A. Arrowood and M. A. Clements, “Using Observation Uncertainty In HMM Decoding,” *Proc. ICSLP*, 2002.
- [14] H. Liao and M. J. F. Gales, “Joint Uncertainty Decoding for Noise Robust Speech Recognition,” *Proc. Eurospeech*, pp 3129-3132, 2005.
- [15] N. B. Yoma and C. Molina and J. Silva and C. Busso, “Modeling, Estimating, and Compensating Low-Bit Rate Coding Distortion in Speech Recognition,” *IEEE Trans. Speech, Audio Processing*, vol. 14, no. 1, pp 246-255, Jan. 2006.
- [16] J. A. Arrowood and M. Clements, “Extended Cluster Information Vector Quantization (ECI-VQ) for Robust Classification,” *Proc. IEEE Int. Conf. Acoust. Speech, Signal Processing*, pp 889-892, May 2004.
- [17] ETSI, “Speech Processing, Transmission and Quality Aspects (STQ), Distributed speech recognition; Extended advanced front-end feature extraction algorithm; Compression algorithms; Back-end speech reconstruction algorithm,” *ES 202 212 V1.1.1 Recommendation*, Nov. 2003.

- [18] I. Kiss and P. Kapanen, "Robust feature vector compression algorithm for distributed speech recognition," *Proc. Eurospeech*, pp 2183-2186, 1999.
- [19] B. Milner and X. Shao, "Low Bit-rate Feature Vector Compression Using Transform Coding and Non-uniform Bit Allocation," *Proc. IEEE Int. Conf. Acoust. Speech, Signal Processing*, pp 129-132, Apr. 2003.
- [20] Q. Zhu and A. Alwan, "An efficient and scalable 2D-DCT based feature coding scheme for remote speech recognition," *Proc. IEEE Int. Conf. Acoust. Speech, Signal Processing*, pp 113-116, 2001.
- [21] W. -H. Hsu and L. -S. Lee, "Efficient and Robust Distributed Speech Recognition (DSR) over Wireless Fading Channels: 2D-DCT Compression, Iterative Bit Allocation, Short BCH Code and Interleaving," *Proc. IEEE Int. Conf. Acoust. Speech, Signal Processing*, pp 69-72, 2004.
- [22] C. Perkins, O. Hodson, and V. Hardman, "A survey of packet loss recovery techniques for streaming audio," *IEEE Network Mag.*, pp 40-48, 1998.
- [23] C. Boullis and M. Ostendorf and E. A. Riskin and S. Otterson, "Graceful Degradation of Speech Recognition Performance over Packet-Erasure Networks," *IEEE Trans. Speech, Audio Processing*, vol. 10, no. 8, pp 580-590, Nov. 2002.
- [24] Z. -H. Tan and P. Dalsgaard, "Channel error protection scheme for distributed speech recognition," *Proc. ICSLP 02*, 2002.
- [25] B. P. Milner and S. Semnani, "Robust speech recognition over IP networks," *Proc. ICASSP*, 2000.
- [26] L. Docio-Ferandez and C. Garcia-Mateo, "Distributed speech recognition over IP networks on the Aurora 3 database," *Proc. ICSLP*, 2002.
- [27] B. P. Milner and A. B. James, "Analysis and compensation of packet loss in distributed speech recognition using interleaving," *Proc. Eurospeech*, 2003.

- [28] B. Milner and A. James, "Robust Speech Recognition over Mobile and IP Networks in Burst-Like Packet Loss," *IEEE Trans. Speech Audio Processing*, vol. 14, no. 1, pp 223-231, Jan. 2006.
- [29] A. Gomez, A. M. Peinado, V. Sanchez, and A. J. Rubio, "A source model mitigation technique for DSR over lossy packet channels," *Proc. Eurospeech*, 2003.
- [30] A. Bernard and A. Alwan, "Low-bitrate distributed speech recognition for packet-based and wireless communication," *IEEE Trans. Speech, Audio Processing*, vol. 10, no. 8, pp 570-579, Nov. 2002.
- [31] A. Cardenal-Lopez and L. Docio-Fernandez and C. Garcia-Mateo, "Soft decoding strategies for distributed speech recognition over IP networks," *Proc. IEEE Int. Conf. Acoust. Speech, Signal Processing*, pp 49-52, May 2004.
- [32] V. Ion and R. Haeb-Umbach, "A Unified Probabilistic Approach to Error Concealment for Distributed Speech Recognition," *Proc. Interspeech*, pp 2853-2856, Sep. 2005.
- [33] A. M. Gomez, A. Peinado, V. Sanchez, A. Rubio, "An integrated scheme for robust distributed speech recognition over lossy packet networks," *Proc. IEEE Int. Conf. Acoust. Speech, Signal Processing*, pp 857-860, Apr. 2007.
- [34] V. Ion and R. Haeb-Umbach, "Multi-resolution soft features for channel-robust distributed speech recognition," *Proc. Interspeech*, pp 594-597, Sep. 2007.
- [35] Z. -H. Tan and P. Dalsgaard and B. Lindberg, "A subvector based error concealment algorithm for speech recognition over mobile networks," *Proc. IEEE Int. Conf. Acoust. Speech, Signal Processing*, May 2004.
- [36] A. M. Peinado and V. Sanchez and J. L. Perez-Cordoba and A. J. Rubio, "Efficient MMSE-Based Channel Error Mitigation Techniques Application to Distributed Speech Recognition Over Wireless Channels," *IEEE Trans. Wireless Communication*, vol. 4, no. 1, pp 14-19, Jan. 2005.

- [37] B. Delaney, "Increased robustness against bit errors for distributed speech recognition in wireless environments," *Proc. IEEE Int. Conf. Acoust. Speech, Signal Processing*, 2005.
- [38] A. Bernard and A. Alwan, "Channel noise robustness for low-bitrate remote speech recognition," *Proc. ICSLP*, 2002.
- [39] T. Endo, S. Kuroiwa, and S. Nakamura, "Missing feature theory applied to robust speech recognition on IP networks," *Proc. Eurospeech*, 2003.
- [40] H. K. Kim and R. V. Cox, "A bitstream-based front-end for wireless speech recognition on IS-136 communication systems," *IEEE Trans. Speech Audio Processing*, vol 9, no 5, pp 558-568, 2001.
- [41] H. G. Hirsch and D. Pearce, "The AURORA Experimental Framework for the Performance Evaluations of Speech Recognition Systems under Noisy Conditions," *ISCA ITRW ASR2000*, Year Sep. 2000.
- [42] K. K. Paliwal and S. So, "Scalable Distributed Speech Recognition Using Multi-Frame GMM-based Block Quantization," *Proc. ICSLP*, 2004.
- [43] S. Molau and M. Pitz and H. Ney, "Histogram based normalization in the acoustic feature space," *Proc. ASRU*, 2001.
- [44] A. de la Torre and A. M. Peinado and J. C. Segura and J. L. Perez-Cordoba and M. C. Benitez and A. J. Rubio, "Histogram equalization of speech representation for robust speech recognition," *IEEE Trans. Speech Audio Processing*, vol. 13, no. 3, pp 355-366, May 2005.
- [45] S. Chen and R. Gopinath, "Gaussianization," *Proc. Neural Information Processing Systems*, pp 423-429, 2000.
- [46] C. -Y. Wan and L. -S. Lee, "Histogram-based Quantization (HQ) for Robust and Scalable Distributed Speech Recognition," *Proc. Interspeech*, pp 957-960, Sep. 2005.

- [47] S. P. Lloyd, "Least Squares Quantization in PCM," *IEEE Trans. Information Theory*, vol. 28, pp 129-137, Mar. 1982.
- [48] J. Max, "Quantizing for Minimum Distortion," *IEEE Trans. Speech Audio Processing*, vol. 6, no. 1, pp 7-12, Mar. 1960.
- [49] F. Hilger and H. Ney, "Quantile-based histogram equalization for noise robust speech recognition," *Proc. Eurospeech*, pp 1135-1138, 2001.
- [50] C. -Y. Wan and L. -S. Lee, "Joint Uncertainty Decoding (JUD) with Histogram-based Quantization (HQ) for Robust and/or Distributed Speech Recognition," *Proc. IEEE Int. Conf. Acoust. Speech, Signal Processing*, pp 125-128, May 2006.
- [51] C. -Y. Wan and Yi Chen and L. -S. Lee, "Three-Stage Error Concealment for Distributed Speech Recognition (DSR) with Histogram-Based Quantization (HQ) under Noisy Environment," *Proc. IEEE Int. Conf. Acoust. Speech, Signal Processing*, pp 877-880, Apr. 2007.
- [52] C. -Y. Wan and L. -S. Lee, "Histogram-based Quantization (HQ) for Robust and Distributed Speech Recognition," *IEEE Trans. Audio Speech and Language Processing*, vol. 16, no. 4, pp 859-873, May 2008.
- [53] L. Bahl and J. Cocke and F. Jelinek and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inf. Theory*, vol. 20, no. 2, pp 284-287, Mar. 1974.
- [54] V. Sanchez and A. M. Peinado and J. L. Perez-Cordoba, "Low Complexity Channel Error Mitigation for Distributed Speech Recognition over Wireless Channels," *Proc. IEEE Int. Conf. Communications*, pp 3619-3623, May 2003.
- [55] J. -H. Chen, "Receiver design and simulation analysis of GPRS physical layer," *Master Thesis, National Taiwan University* Jun. 2001.
- [56] C. -P. Chen and J. A. Bilmes, "MVA Processing of Speech Features," *IEEE Trans. Speech Audio Processing*, vol. 15, no. 1, pp 257-270, Jan. 2007.

- [57] J. -W. Hung and L. -S. Lee, "Optimization of Temporal Filters for Constructing Robust Features in Speech Recognition," *IEEE Trans. Speech Audio Processing*, vol. 14, no. 3, pp 808-832, May 2006.
- [58] ITU-T (Telecommunication Standardization Sector, International Telecommunication Union), "Subjective Performance Assessment of Telephone-band and Wideband Digital Codecs, Annex D: Modified IRS Send and Receive Characteristics," ITU-T Recommendation P.830 Feb. 1996.
- [59] H. Hermansky, "TRAP-TANDEM: data-driven extraction of temporal features from speech," *Proc. ASRU*, pp 255- 260, 2003.
- [60] Y. Linde and A. Buzo and R. Gray, "An Algorithm for Vector Quantizer Design," *IEEE Trans. Speech Audio Processing*, vol. 28, no. 1, pp 84-95, Jan. 1980.
- [61] C. -Y. Wan and Yi Chen and L. -S. Lee, "Context-dependent Quantization for Robust and/or Distributed Speech Recognition," *Proc. IEEE Int. Conf. Acoust. Speech, Signal Processing*, pp 4413-4416, Mar. 2008.
- [62] M. Flickner and H. Sawhney and W. Niblack and J. Ashley and Q. Huang, and B. Dom, "Query by image and video content: the QBIC system," *IEEE Computer*, Sep. 1995.
- [63] John R. Smith and Shin-Fu Chang, "VisualSEEk: a fully automated content-based image query system," *ACM Multimedia*, 1996.
- [64] J. Chen and T. Tan and P. Mulhem and M. Kankanhalli, "An improved method for image retrieval using speech annotation," *Proceedings of the 9th International Conference on Multi-Media Modeling*, 2003.
- [65] Timothy J. Hazen and Brennan Sherry and Mark Adler, "Speech-based annotation and retrieval of digital photographs," *Proc. Interspeech*, 2007.
- [66] G. W. Furnas and S. Deerwester and S. T. Dumais and T. K. Landauer and R. Harshman and L.A. Streeter and K.E. Lochbaum, "Information retrieval using a singular

- value decomposition model of latent semantic structure,” *Proc. ACM SIGIR Conf. R&D in Information Retrieval*, 1988.
- [67] T. Hofmann, “Probabilistic latent semantic indexing,” *Proc. ACM SIGIR Conf. R&D in Informational Retrieval*, 1999.
- [68] M. J. Swain and D. H. Ballard, “Color indexing,” *Int. Journal of Computer Vision*, 1991.
- [69] B. S. Manjunath and W. Y. Ma, “Texture features for browsing and retrieval of image data,” *IEEE T-PAMI*, Aug. 1996.
- [70] Y. Chen and C. -Y. Wan and L. -S. Lee, “Entropy-Based Feature Parameter Weighting for Robust Speech Recognition,” *Proc. IEEE Int. Conf. Acoust. Speech, Signal Processing*, 2006.
- [71] Y. Chen and C. -Y. Wan and L. -S. Lee, “Confusion-Based Entropy-Weighted Decoding for Robust Speech Recognition” *Proc. Interspeech*, 2008.
- [72] Y. Chen and C. -Y. Wan and L. -S. Lee, “Robust Speech Recognition By Properly Utilizing Reliable Frames And Segments In Corrupted Signals” pp 99-104, *Proc. ASRU*, 2007.