PERCEPTUAL MEASURES OF SIGNAL FEATURES

BY

KENNETH FERENS

A Thesis

Presented to the Faculty of Graduate Studies

of the University of Manitoba

in partial fulfilment of the requirements

for the degree of Doctor of Philosophy

in the Department of Electrical and Computer Engineering

University of Manitoba

Winnipeg, Manitoba, CANADA, 1995

Thesis advisor: Witold Kinsner, Ph.D., P.Eng.

ISBN 0-612-13121-1

Canada

Name ___

Dissertation Abstracts International and *Masters Abstracts International* are arranged by broad, general subject categories. Please select the one subject which most nearly describes the content of your dissertation or thesis. Enter the corresponding four-digit code in the spaces provided.

Engineering, General
SUBJECT TERM

0537
SUBJECT CODE

UMI

No category for Computer Engineering

Subject Categories

THE HUMANITIES AND SOCIAL SCIENCES

COMMUNICATIONS AND THE ARTS
Architecture 0729
Art History 0377
Cinema 0900
Dance 0378
Fine Arts 0357
Information Science 0723
Journalism 0391
Library Science 0399
Mass Communications 0708
Music 0413
Speech Communication 0459
Theater 0465

EDUCATION
General 0515
Administration 0514
Adult and Continuing 0516
Agricultural 0517
Art 0273
Bilingual and Multicultural 0282
Business 0688
Community College 0275
Curriculum and Instruction 0727
Early Childhood 0518
Elementary 0524
Finance 0277
Guidance and Counseling 0519
Health 0680
Higher 0745
History of 0520
Home Economics 0278
Industrial 0521
Language and Literature 0279
Mathematics 0280
Music 0522
Philosophy of 0998
Physical 0523
Psychology 0525
Reading 0535
Religious 0527
Sciences 0714
Secondary 0533
Social Sciences 0534
Sociology of 0340
Special 0529
Teacher Training 0530
Technology 0710
Tests and Measurements 0288
Vocational 0747

LANGUAGE, LITERATURE AND LINGUISTICS
Language
- General 0679
- Ancient 0289
- Linguistics 0290
- Modern 0291

Literature
- General 0401
- Classical 0294
- Comparative 0295
- Medieval 0297
- Modern 0298
- African 0316
- American 0591
- Asian 0305
- Canadian (English) 0352
- Canadian (French) 0355
- English 0593
- Germanic 0311
- Latin American 0312
- Middle Eastern 0315
- Romance 0313
- Slavic and East European 0314

PHILOSOPHY, RELIGION AND THEOLOGY
Philosophy 0422
Religion
- General 0318
- Biblical Studies 0321
- Clergy 0319
- History of 0320
- Philosophy of 0322

Theology 0469

SOCIAL SCIENCES
American Studies 0323
Anthropology
- Archaeology 0324
- Cultural 0326
- Physical 0327

Business Administration
- General 0310
- Accounting 0272
- Banking 0770
- Management 0454
- Marketing 0338

Canadian Studies 0385
Economics
- General 0501
- Agricultural 0503
- Commerce-Business 0505
- Finance 0508
- History 0509
- Labor 0510
- Theory 0511

Folklore 0358
Geography 0366
Gerontology 0351
History
- General 0578
- Ancient 0579
- Medieval 0581
- Modern 0582
- Black 0328
- African 0331
- Asia, Australia and Oceania 0332
- Canadian 0334
- European 0335
- Latin American 0336
- Middle Eastern 0333
- United States 0337

History of Science 0585
Law 0398
Political Science
- General 0615
- International Law and Relations 0616
- Public Administration 0617

Recreation 0814
Social Work 0452
Sociology
- General 0626
- Criminology and Penology ... 0627
- Demography 0938
- Ethnic and Racial Studies 0631
- Individual and Family Studies 0628
- Industrial and Labor Relations 0629
- Public and Social Welfare 0630
- Social Structure and Development 0700
- Theory and Methods 0344

Transportation 0709
Urban and Regional Planning 0999
Women's Studies 0453

THE SCIENCES AND ENGINEERING

BIOLOGICAL SCIENCES
Agriculture
- General 0473
- Agronomy 0285
- Animal Culture and Nutrition 0475
- Animal Pathology 0476
- Food Science and Technology 0359
- Forestry and Wildlife 0478
- Plant Culture 0479
- Plant Pathology 0480
- Plant Physiology 0817
- Range Management 0777
- Wood Technology 0746

Biology
- General 0306
- Anatomy 0287
- Biostatistics 0308
- Botany 0309
- Cell 0379
- Ecology 0329
- Entomology 0353
- Genetics 0369
- Limnology 0793
- Microbiology 0410
- Molecular 0307
- Neuroscience 0317
- Oceanography 0416
- Physiology 0433
- Radiation 0821
- Veterinary Science 0778
- Zoology 0472

Biophysics
- General 0786
- Medical 0760

EARTH SCIENCES
Biogeochemistry 0425
Geochemistry 0996
Geodesy 0370
Geology 0372
Geophysics 0373
Hydrology 0388
Mineralogy 0411
Paleobotany 0345
Paleoecology 0426
Paleontology 0418
Paleozoology 0985
Palynology 0427
Physical Geography 0368
Physical Oceanography 0415

HEALTH AND ENVIRONMENTAL SCIENCES
Environmental Sciences 0768
Health Sciences
- General 0566
- Audiology 0300
- Chemotherapy 0992
- Dentistry 0567
- Education 0350
- Hospital Management 0769
- Human Development 0758
- Immunology 0982
- Medicine and Surgery 0564
- Mental Health 0347
- Nursing 0569
- Nutrition 0570
- Obstetrics and Gynecology .. 0380
- Occupational Health and Therapy 0354
- Ophthalmology 0381
- Pathology 0571
- Pharmacology 0419
- Pharmacy 0572
- Physical Therapy 0382
- Public Health 0573
- Radiology 0574
- Recreation 0575
- Speech Pathology 0460
- Toxicology 0383

Home Economics 0386

PHYSICAL SCIENCES
Pure Sciences
Chemistry
- General 0485
- Agricultural 0749
- Analytical 0486
- Biochemistry 0487
- Inorganic 0488
- Nuclear 0738
- Organic 0490
- Pharmaceutical 0491
- Physical 0494
- Polymer 0495
- Radiation 0754

Mathematics 0405
Physics
- General 0605
- Acoustics 0986
- Astronomy and Astrophysics 0606
- Atmospheric Science 0608
- Atomic 0748
- Electronics and Electricity 0607
- Elementary Particles and High Energy 0798
- Fluid and Plasma 0759
- Molecular 0609
- Nuclear 0610
- Optics 0752
- Radiation 0756
- Solid State 0611

Statistics 0463

Applied Sciences
Applied Mechanics 0346
Computer Science 0984
Engineering
- General 0537
- Aerospace 0538
- Agricultural 0539
- Automotive 0540
- Biomedical 0541
- Chemical 0542
- Civil 0543
- Electronics and Electrical 0544
- Heat and Thermodynamics ... 0348
- Hydraulic 0545
- Industrial 0546
- Marine 0547
- Materials Science 0794
- Mechanical 0548
- Metallurgy 0743
- Mining 0551
- Nuclear 0552
- Packaging 0549
- Petroleum 0765
- Sanitary and Municipal 0554
- System Science 0790

Geotechnology 0428
Operations Research 0796
Plastics Technology 0795
Textile Technology 0994

PSYCHOLOGY
General 0621
Behavioral 0384
Clinical 0622
Developmental 0620
Experimental 0623
Industrial 0624
Personality 0625
Physiological 0989
Psychobiology 0349
Psychometrics 0632
Social 0451

Nom _____

Dissertation Abstracts International est organisé en catégories de sujets. Veuillez s.v.p. choisir le sujet qui décrit le mieux votre thèse et inscrivez le code numérique approprié dans l'espace réservé ci-dessous.

_____ ☐☐☐ U·M·I

SUJET — CODE DE SUJET

Catégories par sujets

HUMANITÉS ET SCIENCES SOCIALES

COMMUNICATIONS ET LES ARTS
Architecture 0729
Beaux-arts 0357
Bibliothéconomie 0399
Cinéma 0900
Communication verbale 0459
Communications 0708
Danse 0378
Histoire de l'art 0377
Journalisme 0391
Musique 0413
Sciences de l'information 0723
Théâtre 0465

ÉDUCATION
Généralités 515
Administration 0514
Art 0273
Collèges communautaires 0275
Commerce 0688
Économie domestique 0278
Éducation permanente 0516
Éducation préscolaire 0518
Éducation sanitaire 0680
Enseignement agricole 0517
Enseignement bilingue et multiculturel 0282
Enseignement industriel 0521
Enseignement primaire. 0524
Enseignement professionnel 0747
Enseignement religieux 0527
Enseignement secondaire 0533
Enseignement spécial 0529
Enseignement supérieur 0745
Évaluation 0288
Finances 0277
Formation des enseignants 0530
Histoire de l'éducation 0520
Langues et littérature 0279
Lecture 0535
Mathématiques 0280
Musique 0522
Orientation et consultation 0519
Philosophie de l'éducation 0998
Physique 0523
Programmes d'études et enseignement 0727
Psychologie 0525
Sciences 0714
Sciences sociales 0534
Sociologie de l'éducation 0340
Technologie 0710

LANGUE, LITTÉRATURE ET LINGUISTIQUE
Langues
- Généralités 0679
- Anciennes 0289
- Linguistique 0290
- Modernes 0291

Littérature
- Généralités 0401
- Anciennes 0294
- Comparée 0295
- Médiévale 0297
- Moderne 0298
- Africaine 0316
- Américaine 0591
- Anglaise 0593
- Asiatique 0305
- Canadienne (Anglaise) 0352
- Canadienne (Française) 0355
- Germanique 0311
- Latino-américaine 0312
- Moyen-orientale 0315
- Romane 0313
- Slave et est-européenne 0314

PHILOSOPHIE, RELIGION ET THÉOLOGIE
Philosophie 0422
Religion
- Généralités 0318
- Clergé 0319
- Études bibliques 0321
- Histoire des religions 0320
- Philosophie de la religion 0322

Théologie 0469

SCIENCES SOCIALES
Anthropologie
- Archéologie 0324
- Culturelle 0326
- Physique 0327

Droit 0398
Économie
- Généralités 0501
- Commerce-Affaires 0505
- Économie agricole 0503
- Économie du travail 0510
- Finances 0508
- Histoire 0509
- Théorie 0511

Études américaines 0323
Études canadiennes 0385
Études féministes 0453
Folklore 0358
Géographie 0366
Gérontologie 0351
Gestion des affaires
- Généralités 0310
- Administration 0454
- Banques 0770
- Comptabilité 0272
- Marketing 0338

Histoire
- Histoire générale 0578
- Ancienne 0579
- Médiévale 0581
- Moderne 0582
- Histoire des noirs 0328
- Africaine 0331
- Canadienne 0334
- États-Unis 0337
- Européenne 0335
- Moyen-orientale 0333
- Latino-américaine 0336
- Asie, Australie et Océanie 0332

Histoire des sciences 0585
Loisirs 0814
Planification urbaine et régionale 0999
Science politique
- Généralités 0615
- Administration publique 0617
- Droit et relations internationales 0616

Sociologie
- Généralités 0626
- Aide et bien-être social 0630
- Criminologie et établissements pénitentiaires 0627
- Démographie 0938
- Études de l' individu et de la famille 0628
- Études des relations interethniques et des relations raciales 0631
- Structure et développement social 0700
- Théorie et méthodes. 0344
- Travail et relations industrielles 0629

Transports 0709
Travail social 0452

SCIENCES ET INGÉNIERIE

SCIENCES BIOLOGIQUES
Agriculture
- Généralités 0473
- Agronomie. 0285
- Alimentation et technologie alimentaire 0359
- Culture 0479
- Élevage et alimentation 0475
- Exploitation des pâturages 0777
- Pathologie animale 0476
- Pathologie végétale 0480
- Physiologie végétale 0817
- Sylviculture et faune 0478
- Technologie du bois 0746

Biologie
- Généralités 0306
- Anatomie 0287
- Biologie (Statistiques) 0308
- Biologie moléculaire 0307
- Botanique 0309
- Cellule 0379
- Écologie 0329
- Entomologie 0353
- Génétique 0369
- Limnologie 0793
- Microbiologie 0410
- Neurologie 0317
- Océanographie 0416
- Physiologie 0433
- Radiation 0821
- Science vétérinaire 0778
- Zoologie 0472

Biophysique
- Généralités 0786
- Medicale 0760

SCIENCES DE LA TERRE
Biogéochimie 0425
Géochimie 0996
Géodésie 0370
Géographie physique 0368
Géologie 0372
Géophysique 0373
Hydrologie 0388
Minéralogie 0411
Océanographie physique 0415
Paléobotanique 0345
Paléoécologie 0426
Paléontologie 0418
Paléozoologie 0985
Palynologie 0427

SCIENCES DE LA SANTÉ ET DE L'ENVIRONNEMENT
Économie domestique 0386
Sciences de l'environnement 0768
Sciences de la santé
- Généralités 0566
- Administration des hôpitaux 0769
- Alimentation et nutrition 0570
- Audiologie 0300
- Chimiothérapie 0992
- Dentisterie 0567
- Développement humain 0758
- Enseignement 0350
- Immunologie 0982
- Loisirs 0575
- Médecine du travail et thérapie 0354
- Médecine et chirurgie 0564
- Obstétrique et gynécologie 0380
- Ophtalmologie 0381
- Orthophonie 0460
- Pathologie 0571
- Pharmacie 0572
- Pharmacologie 0419
- Physiothérapie 0382
- Radiologie 0574
- Santé mentale 0347
- Santé publique 0573
- Soins infirmiers 0569
- Toxicologie 0383

SCIENCES PHYSIQUES
Sciences Pures
Chimie
- Généralités 0485
- Biochimie 487
- Chimie agricole 0749
- Chimie analytique 0486
- Chimie minérale 0488
- Chimie nucléaire 0738
- Chimie organique 0490
- Chimie pharmaceutique 0491
- Physique 0494
- PolymÈres 0495
- Radiation 0754

Mathématiques 0405
Physique
- Généralités 0605
- Acoustique 0986
- Astronomie et astrophysique 0606
- Électronique et électricité 0607
- Fluides et plasma 0759
- Météorologie 0608
- Optique 0752
- Particules (Physique nucléaire) 0798
- Physique atomique 0748
- Physique de l'état solide 0611
- Physique moléculaire 0609
- Physique nucléaire 0610
- Radiation 0756

Statistiques 0463

Sciences Appliqués Et Technologie
Informatique 0984
Ingénierie
- Généralités 0537
- Agricole 0539
- Automobile 0540
- Biomédicale 0541
- Chaleur et thermodynamique 0348
- Conditionnement (Emballage) 0549
- Génie aérospatial 0538
- Génie chimique 0542
- Génie civil 0543
- Génie électronique et électrique 0544
- Génie industriel 0546
- Génie mécanique 0548
- Génie nucléaire 0552
- Ingénierie des systèmes 0790
- Mécanique navale 0547
- Métallurgie 0743
- Science des matériaux 0794
- Technique du pétrole 0765
- Technique minière 0551
- Techniques sanitaires et municipales 0554
- Technologie hydraulique 0545

Mécanique appliquée 0346
Géotechnologie 0428
Matières plastiques (Technologie) 0795
Recherche opérationnelle 0796
Textiles et tissus (Technologie) 0794

PSYCHOLOGIE
Généralités 0621
Personnalité 0625
Psychobiologie 0349
Psychologie clinique 0622
Psychologie du comportement 0384
Psychologie du développement 0620
Psychologie expérimentale 0623
Psychologie industrielle 0624
Psychologie physiologique 0989
Psychologie sociale 0451
Psychométrie 0632

PERCEPTUAL MEASURES OF SIGNAL FEATURES

BY

KENNETH FERENS

A Thesis submitted to the Faculty of Graduate Studies of the University of Manitoba in partial fulfillment of the requirements of the degree of

DOCTOR OF PHILOSOPHY

ABSTRACT

This thesis presents perceptual measures of signal features primarily for compression. Classical distortion measures, such as the mean squared difference between samples of the signal and the reconstructed signal, have little meaning in measuring perceived features. In this thesis perceptually relevant features of signals are measured, quantified, and extracted by methods which are based on human perception. These methods are the wavelet transform, artificial neural networks, and multifractals. The result of the processing of the signals is that most of the redundant and irrelevant information is removed, and, in this way, compression is achieved while quality is perceptually optimized.

Compression of music involves the adaptive quantization of the wavelet subband coefficients. The amplitude probability density of the coefficients in each subband is modelled using learned scalar quantization employing frequency sensitive competitive learning. The source data consists of 1-channel sampled data , 44.1 kHz sampling rate, and quantized to 16-bit linear data read from a compact disc containing examples of classical and pop music. Preliminary results show a bit-rate of 150 kbps, rather than 705.6 kbps, with no perceptual loss in quality.

Compression of images involves vector quantization employing various competitive learning algorithms, including hard competitive learning, soft competitive learning, Kohonen's self organizing feature map, and frequency sensitive competitive learning. Grey scale images are compressed by 16:1 and transmitted at 0.5 bits per pixel, while maintaining a peak signal-to-noise ratio of approximately 30 dB.

Finally, this thesis presents a perceptually meaningful and objective measure of the distribution of information in signals. The measure is multifractal based on Rényi's generalized entropy and a generalized correlation dimension. The features of the signal are represented by information subsets. The multifractal entropy distribution function shows a slightly wider breadth for the baboon as compared to Lena, indicating the baboon contains a higher degree of non-uniformity. This thesis reports Hausdorff fractal dimensions for

Lena and the baboon of 2.5958, and 2.6562, respectively.

ACKNOWLEDGEMENTS

This thesis was financially supported in part through a research scholarship from the University of Manitoba, a research scholarship and grant from the Natural Sciences and Engineering Research Council (NSERC) of Canada, and a research scholarship from the Telecommunications Research Laboratory (TR*Labs*), Winnipeg.

I would like to acknowledge my advisor for introducing me to the subject of data and signal compression, and particularly to neural networks, wavelets, fractals, and multi-fractals. Also, I would like to acknowledge his financial support of the many conferences, most notably the EMBS in Philadelphia; *IASTED* in Calgary; WESCANEX in Regina, Saskatoon, and Winnipeg; and ICMCM in Berkeley and Boston. Furthermore, I would like to acknowledge him for the computer equipment in our laboratory, and for arranging the computer equipment at TRLabs. There were some memorable times we experienced together, like at Christmas time, 1994, when he appeared in our laboratory looking like Santa and bearing gifts for all of his graduate students. His research-motivated spirit and support of his students is unprecedented.

I would like to acknowledge Armein Langi for the friendship and the many fruitful discussions on research related topics. I would also like to acknowledge the friendship and help of Dave Blight, Howard Card, Bob McLeod, Dean McNeill, Peter Czezowski (reviewed manuscript), Zaifu Zhang, and the other members of "wcrew". Without their valuable discussions on our regular meetings, this work would be not be possible.

Finally, I would like to acknowledge my family, and especially, my parents for their enormous patience and love.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

ABBREVIATION	PAGE

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

CHAPTER 1

Introduction

In this exponentially progressing technological age, the days of communicating information through text alone are coming to an end, very quickly. People want to express themselves through multiple forms of media (multimedia), such as images, pictures, audio, music, moving pictures, and video - everything and anything that will serve to get their message across more efficiently and effectively. Today, multimedia has become the communication method of choice and demand.

In this also highly digital information age, streams or sequences of digital data (100111011101010111 ...) are used to represent the multimedia, with which we are all familiar. These streams or sequences of digital data represent the state of some system, such as a still picture of a natural scene, a person's vocalization, Bach's Seventh Overture, or a video clip. Hence, these digital streams or sequences are also called signals. The way in which we communicate information is by transmitting or storing these signals in their digital form. Thus, in today's digital information age the physical communication of multimedia demands the ability to transmit and to store digital data.

Today's technology gives us great ability to communicate multimedia, ranging from one personal computer (PC) to the next through local area networks (LAN), from one distant location to the other through wide area networks (WAN), or an interconnection

of LANs and WANs, such as the information super highway, through, for example, the INTERNET. However, today's demand for information is exponentially increasing, and technology must keep up.

One way of keeping up with the demand is by increasing the physical capacity of the transmission and storage systems. Indeed, today we are witnessing the explosive nature of the increase in memory size and the implosive nature of their cost. For example, this author purchased a 20 Mega-byte (MByte) hard disk for approximately $700.00 Cdn during the first year of his graduate program. Today, five years later, a 2 Giga-byte (GByte) hard disk can be purchased for the same number of dollars (~100 % decrease). Furthermore, we are witnessing the expansion of the speed of computer networks to facilitate the ever increasing demand. For example, the Asynchronous Transfer Mode (ATM) standard implemented on a fibre optics channel will replace the once touted Ethernet standard. And, if the ATM standard is not sufficient by itself, then more fibre channels will be added.

However, even with the constant and continual improvement of technology, the currently available bandwidth will always be too small. Full motion video with high quality audio requires enormous bandwidth, exceeding 0.75 Giga-bits-per-second (Gbps) for just one instance. For example, 30 frames-per-second (fps) video with 24-bit color and a resolution of 1024x1024 pixels requires a bit rate of 754,974,720 bits-per-second (bps). The corresponding Compact Disc (CD) quality audio stream requires a bit rate of 1,411,200 bps. Today, the existing bandwidth of 14,400 bps of the basic telephone network, up to 622 Mega-bits-per-second (Mbps) of the OC12 ATM network [TRla95] is not even adequate to facilitate the capacity required for one instance of 30 fps video. Yet, this is not including the eventuality of say millions of homes and offices wishing to receive as

many different video clips at the same time.

The philosophy of "if the highway is too small, then construct more lanes" is not the only solution a telecommunications engineer can address. There is another less expensive and more efficient, practical, and interesting way in which the effective speed of transmitting signals on an information super highway can be increased, or the effective capacity for storing signals in a storage medium can be increased. This approach is called signal compression.

Signal compression may make the dream of communicating vast amounts of information on a super highway a reality without the substantial increase in the cost due to physical channel widening. Signal compression is less expensive, more efficient, more practical, and smarter because it increases the effective speed and storage capacity without increasing the number of 'lanes'. Signal compression is more interesting because it deals with the very nature of human perception and may eventually lead to a better understanding of the perception mechanism. This may be the key step for realizing the ultimate dream of creating machine intelligence comparable with that of the human brain.

The objective of digital signal compression is to minimize the samples of a signal, while minimizing the loss of *perceived* quality. Since the human perception system observes and ultimately assesses the quality of signals, the signal compression coding must be driven by the criterion of minimizing the perceived error. Classical distortion measures, such as the mean squared difference between samples of the signal and the reconstructed signal, have little meaning in measuring perceived error. The amount of perceived error or distortion introduced by compression coding must be quantified using perceptually meaningful measures, and if possible quantification should be done using perceptually meaningful *objective* measures. These measures must produce values, which

are similar to the sense returned by the human perceptual process upon observing a signal.

There is a theoretical limit to which a signal can be compressed. Compressing a signal means removing statistical redundancy and perceptual irrelevancy from the signal. A signal consists of the sum of relevant, irrelevant, and redundant information. Signals cannot be compressed very well by removing redundancy alone. In order to achieve high compression rates, both redundant and irrelevant information should be removed. The limit to which a human signal can be compressed should not be determined by the amount of redundancy, but must be determined through a measure of the relevant information in the signal, or the perceptual entropy of a signal.

Thus, the compression coding of signals must incorporate models of human signal perception. Known models of perception are based on psycho-acoustic and/or physical properties of the human perception mechanism.

The psycho-acoustic model [Hout92] is based on the inability of the human observer to differentiate between two signal components in nearly the same space, time, or scale. Psycho-acoustic compression coding removes the irrelevant information that is masked by the stronger more relevant information in the same spacial, temporal, or scale locality. The physical-acoustic models qualitatively match the acoustic mechanism of human perception. The physical aspects of the human perception mechanism naturally remove redundant and irrelevant information. By modelling the physical aspects of human perception, the physical-acoustic compression algorithms tend to do the same.

In the literature, signal compression research has focused on the use of artificial neural networks, wavelets, and fractals. These compression techniques have the potential of yielding very large compression ratios, while maintaining very good quality. In these areas, many researchers have provided good compression methods and techniques. Much

work has been done on the subject of wideband audio compression [BGSS92], [Jaya92], [Hout92], [Seit91], [Veld92]. The majority of this work deals with either subband coding, transform coding, or a combination thereof. These coders strive to maximize the signal-to-noise ratio (SNR), the fitness function of a human audio perception model, or a hybrid. One problem associated with these techniques is the inadequacy of the classical metrics to convey a measure of signal features that is reflective of human perception. In order to circumvent this problem, some researchers have attempted to develop the so called perceptual metrics [BGSS92]. Others have used an artificial neural network for modelling the similarity metric used by the human perception mechanism [AKCM90]. And other researchers have used wavelets, and other transforms, which are reflective of or model the human perception mechanism [KiLa93]. Finally, other research work involves the use of fractals and multifractals [Gras83], [GrPr83], [HePr83], [Kins94b], [Kins94c], [Mand83].

1.1. Thesis Statement

This thesis proposes perceptual measures of signal features primarily for compression. Since classical distortion measures, such as the mean squared difference between samples of the signal and the reconstructed signal, have little meaning in measuring perceived error, this thesis develops perceptually meaningful measures and perceptually meaningful *objective* measures of signal features. This thesis extends the signal compression work done in the literature in the following three areas: the wavelet transform, artificial neural networks, and multifractals.

1.2. Contributions of the Thesis

This thesis applies the wavelet transform to model the human perception mechanism. In the literature, many types of conventional transforms, such as the Fourier transform, have been used to model the human perception system. This thesis uses the wavelet transform because the wavelet transform offers a more relevant model of perception. Along these lines, this thesis develops a perceptually meaningful measure of the amount of irrelevant information in the audio signal. Wavelet based features of the audio signal are extracted and processed for compression in a manner that is similar to the cochlea.

Next, this thesis applies an artificial neural network to learn vector quantization. Learned vector quantization models the way in which features are learned and stored in the human perception and storage system. A novel graphical user interface for visualization of the dynamic and adaptive learning procedure is developed.

Finally, this thesis applies multifractals for obtaining an objective measure of the subjective nature of the perception of signal features.

1.3. Thesis Organization

In Chapter 2, the first part of the thesis presents adaptive wavelet subband coding of wideband audio signals for low-bit rate and high quality signal compression. The wavelet transform transforms the signal by a process similar to the physical mechanism of human perception. Once a signal is represented by the wavelet transform, perceptually motivated, non-linear methods of data reduction are applied. The coefficients in the wavelet subbands of wideband audio are quantized using a scheme that adapts to the subband

signal by setting the quantization step size for a particular subband to a size that is inversely proportional to the subband energy. Furthermore, within a subband the scheme modifies the energy determined step size as inversely proportional to the amplitude probability density of the coefficient. Thus, these non-linear methods are motivated by the perceptually influential factors of energy and frequency. The amplitude probability density of the coefficients in each subband is modelled using learned scalar quantization employing frequency sensitive competitive learning. The source data consists of 1-channel sampled data, quantized to 16-bit, and sampled at 44.1 kHz from a CD containing examples of classical and pop music. Preliminary results show a bit-rate of 150 kbps, rather than 705.6 kbps, with no perceptual loss in quality.

In Chapter 3, the second part of this thesis presents a study and implementation of still image compression using learned vector quantization. Grey scale, still images are compressed by 16:1 and transmitted at 0.5 bits per pixel, while maintaining a peak signal-to-noise ratio of 30 dB. The vector quantization is learned using a variety of competitive learning artificial neural networks. While not only being representative of the training set, the prototype vectors also serve as a basis for other histogram-similar images. Hence, these codebooks quantize other images not in the training set. Various optimization techniques are investigated. The effects of the uniform, linear, and cubic nested learning rate and neighborhood functions on rate of convergence are studied. Simulated annealing is applied to the self-organizing-feature-map (*SOFM*) network. By inserting impulses of high temperature at increasing time intervals, codebooks learn more quickly. Competitive learning, frequency sensitive competitive learning, and Kohonen's neighborhood learning are studied. An XView interface on the SUN SPARC Station 2 is built to facilitate a user interface, and to graphically illustrate the dynamic learning of the codebooks and vector-

by-vector quantization and reconstruction of images.

In Chapter 4, the third part of this thesis presents a perceptually meaningful measure of the distribution of information in signals. The proposed measure is a multifractal based on Rényi's generalized entropy and a generalized correlation dimension. The features of the signal are represented by information subsets. The multifractal measure of a subset indicates its information content, whether it be redundant, irrelevant, or relevant. The multifractal entropy distribution function $f(\alpha)$ shows a slightly wider breadth for the baboon as compared to Lena, indicating the baboon contains a higher degree of non-uniformity.

Chapter 5 gives the conclusions and recommendations for future work.

CHAPTER 2

Adaptive Wavelet Subband Coding for Wide-band Audio Compression

2.1. Introduction

Wide-band audio signal compression has been gaining much attention, most notably in the field of multimedia communication [KFAG93]. Given the compact disc (CD) technology, people naturally expect similar quality sound in multimedia. The bit-rate of CD quality audio (1.4 Mbps) currently prohibits its use in many multimedia platforms, such as personal computer (PC) systems. One way to allow high quality audio in multimedia systems is through the use of data and signal compression.

Much work has been done on the subject of wide-band audio compression [BGSS92], [Jaya92], [Hout92], [Seit91], [Veld92]. The majority of this work deals with either subband coding, transform coding, or a combination thereof. These coders strive to maximize the signal-to-noise ratio (SNR), the fitness function of a human audio perception model, or a hybrid. On the one hand, the SNR based coder attempts to minimize the distortion at a given bit-rate based on conventional distortion measures, such as the Euclidean metric. On the other hand, the perceptual coder attempts to model human perception of the audio signal based on the psychoacoustic masking phenomenon [Veld92] and on models of the cochlea itself. These models are applied in either the time-domain, the Fourier frequency-domain, or a derivative of the frequency domain, such as the discrete cosine transform (DCT).

The purpose of this section of the thesis is to develop a model of the hybrid coder of wide-band audio signals in the wavelet domain. The wavelet transform is used because of its ability to model the human perception mechanism. In the model, the time-domain audio samples are transformed into the wavelet-domain. The wavelet transform decomposes the audio signal into wavelet subbands, thus extracting wavelet based features. In the wavelet domain, each subband consists of coefficients which indicate the degree to which the audio signal belongs to that subband, or feature. To achieve signal compression based on the wavelet-based features of the signal, the coefficients in each subband are quantized using a two-step, adaptive, non-uniform, frequency sensitive quantization. First, a quantization step size is assigned to each wavelet subband according the subband's energy content. Second, inside the subband, the energy determined step size is further modified to reflect the frequency of occurrence of the wavelet coefficients in that subband. Highly probable coefficients are given small step sizes, while improbable coefficients are assigned large step sizes. Thus, this allocation of step sizes is done non-uniformly. The probability density of the coefficients in each subband is learned using a frequency sensitive competitive learning (FSCL) neural network, and the assignment of step sizes is done by the learned quantization. Thus, this allocation of step sizes is also done adaptively.

This chapter is organized as follows. The introduction explains what is meant by wide-band audio, its quality, bandwidth, and bit-rate. With signal compression in mind, the sources of redundant and irrelevant information are described. This is followed by a brief description of some of the applications of wide-band audio and a brief outline of the current implementations of the compression of wide-band audio. Next, the coding approach is described, with a focus on what type of compression should be used in wide-band audio. This develops motivation for using the wavelet transform. Having developed

motivation, the next section briefly describes the background theory of the wavelet transform. This is followed by a discussion of the type of quantization that can be performed on the coefficients in the wavelet subbands. This leads to the energy and frequency adaptive, non-uniform quantization of wavelet coefficients used in this thesis. Having discussed the motivation and background theory, discussed next is the software that was developed to test the theory. This is followed by a description and analysis of the experiments and results, followed by the chapter summary.

2.2. What is Wide-band Audio?

Technically speaking, wide-band audio, like all sound, is characterized by the perceived degree of air pressure at a certain frequency. The audio signal is physically manifested as a sound pressure wave, as for example described in [KlKi87]. Figure 1 shows the sound pressure and frequency limits for average human perception of various classes of sounds [Ever89].

The pressure limit is bounded below by the threshold of hearing curve. Sound pressure waves having a level below this threshold cannot be heard by the average human observer. As the pressure level is increased above the threshold of hearing for a given frequency, the sound can be sensed. When the pressure level exceeds the threshold of feeling, pain can be felt. There are also frequency limits. Sounds having frequencies below about 10 Hz and above 20 kHz cannot be heard. As shown in the figure, wide-band audio has a bandwidth of about 20-20 kHz. Note that speech can also be considered a wide-band audio signal, since its bandwidth is a subset of the former.

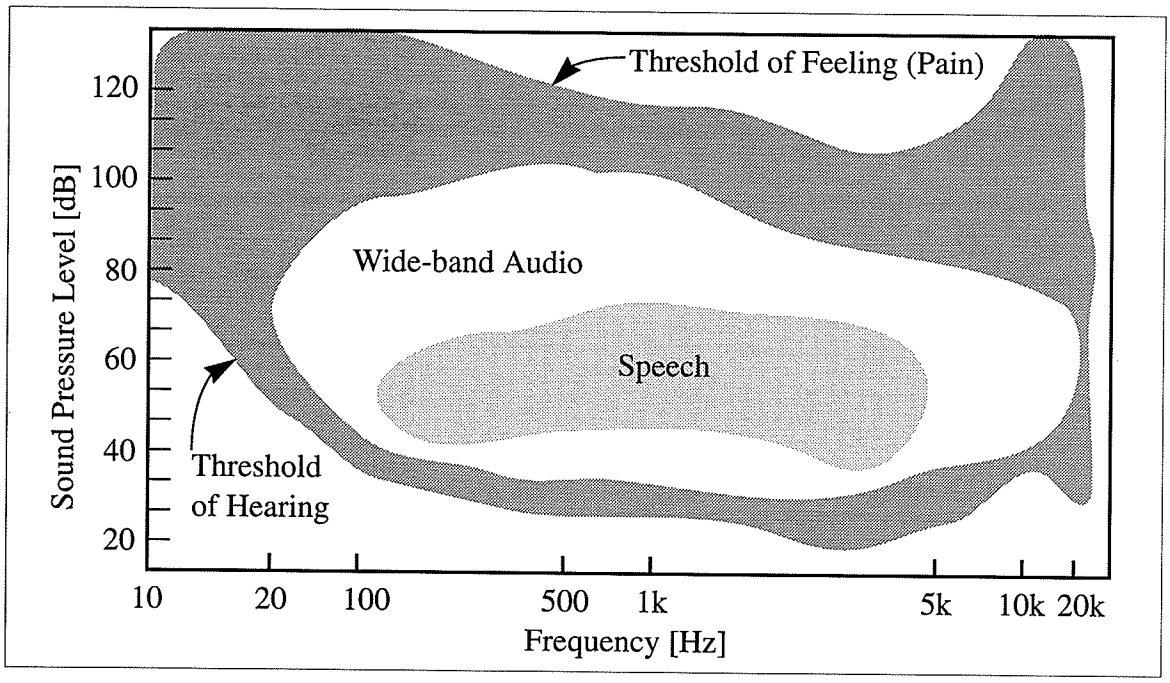

Fig. 1. Limits of aural perception [Ever89].

The above describes a technical nature of wide-band audio, but most of us are familiar with its definition as the quality of sound contained in CDs. But there are also many other applications. These include multimedia documents, multimedia CD ROMs, computer games, video games, educational tools, tutorials, language instruction, music instruction, video telephones, virtual reality, voice and audio email, and desktop audio publishing.

2.3. Problem of Wide-band Audio

The problem associated with implementing wide-band audio for various applications is the large cost of storage and transmission. For example, digital audio having characteristics of a 44.1 kHz sampling rate, 16-bit per sample, and two-channels requires a bit-rate of 1,411,200 bps. Consequently, 50 MBytes of free disk space is required in order to

store a mere five minutes of CD quality music. The problem is even more pronounced for the digital audio tape (DAT) format, because DAT uses a 48 kHz sampling rate.

Much work has been done in order to solve the large storage and transmission rate problem. Figure 2 shows the source bit rate and the target bit rate, along with some of the established compression schemes and standards. This work refers to standards such as the International Standards Organization (ISO) [Musm90], moving picture experts group (MPEG), [LeGa91], digital audio broadcasting (DAB), [Spur90], and DAT. The work also includes implementations such as digital compact cassette (DCC) and precision adaptive subband coding (PASC) [Lokh91], masking-pattern universal subband integrated coding and multiplexing (MUSICAM), and ASPEC [Musm90]. At the low end of the spectrum, code-excited linear predictive coding (CELP), [CaTW90] is used for telephone quality and low bit rate speech [Lang92].

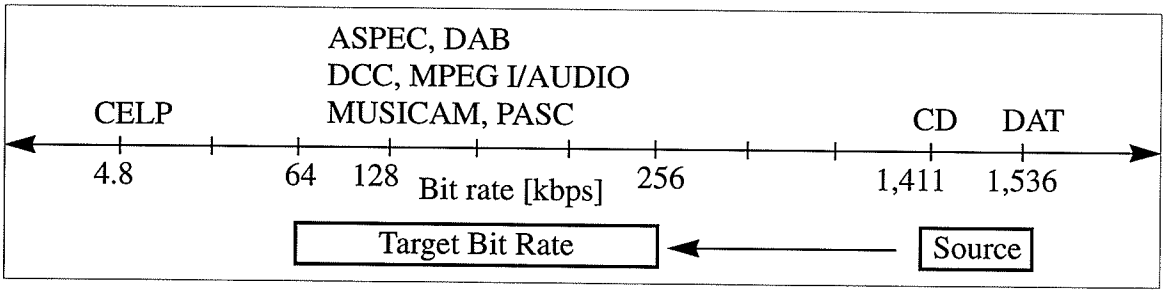

Fig. 2. Bit rates of some standards and implementations of wide-band audio.

2.4. Information Nature of the Wide-band Audio Pressure Wave

As shown in Fig. 2, the target bit rate is 64-256 kbps. In order to achieve the target bit rate, one must consider the sources of redundant and irrelevant information in the signal. Knowing the sources of redundant and irrelevant information, one can devise perceptually optimal compression algorithms that suit or model the signal. Once identified, this information can be removed and thus the representation of the signal reduced.

2.4.1 Redundancy Information in Wide-band Audio

The redundancy in wide-band audio manifests itself in basically two ways. The first source is inherent redundancy in the analog signal itself. Figure 3a shows the voltage-time relationship of typical wide-band audio. As can been seen in Fig. 3a, the statistics change over time, i.e., some portions show pseudo periodic behavior, while others look like (and sometimes sound like) noise. More specifically, the autocorrelation function of typical audio depends on absolute time, not on the time difference, as is the case for stationary signals. However, while the wide-band audio signal is non-stationary, there is still some predictability in the signal, and, therefore, a parametric coding of the signal can theoretically achieve compression.

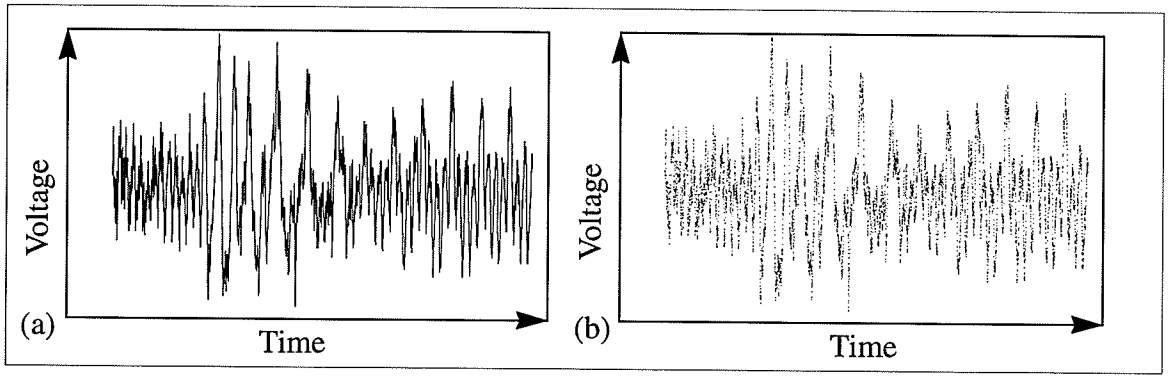

Fig. 3. (a) Analog voltage-time representation of wide-band audio. (b) Digitized version of the analog signal in (a).

The second source of redundancy is embedded in the digital samples of the signal. For the purposes of communication and storage, the wide-band audio signal is captured in the digital domain using an analog-to-digital converter (ADC). The sound is digitized into two channels, at a rate of 44.1 kHz, using 16 bits-per-sample. The 44.1 kHz sampling rate is chosen according to the Nyquist criterion, given the fact that the most of the energy of

wide-band audio is contained in the bandwidth of 20-22,050 Hz. Consequently, because the signal is *uniformly* sampled, some of the digital samples of the slower varying portions of the signal are redundant. For example, for a one second, 11 kHz portion of a signal digitized at 44.1 kHz, there are 44,100 – 22,000 = 22,100 redundant samples. This represents a 50% compression, or a 2:1 compression ratio.

The fact that there is little predictability in the audio signal is the reason for the inability of redundant information based compression schemes (also known as first order entropy encoders), such as Huffman and arithmetic coders [Kins91], to achieve any substantial compression. The average amount of compression one can obtain using the entropy encoders is 50%, or a 2:1 compression ratio, as given by Theorem 1.

Theorem 1: Consider a bandlimited and linearly distributed signal. The average amount of compression that can be obtained through the removal of redundancy due to over-sampling is 50%.

Proof: This follows from the assumption that the frequency of the signal is linearly distributed along its bandwidth 20-22,050 Hz. For a given percentage of the signal, Eq. 1 yields the fraction of the sampling rate required to represent the signal.

$$S\sum pb = S\left(\lim_{n\to\infty}\sum_{i=1}^{n}\frac{1}{n}\frac{i}{n}\right) = S\left(\lim_{n\to\infty}\left(\frac{1}{n^2}\left\lceil\frac{n(n+1)}{2}\right\rceil\right)\right) = \frac{1}{2}S \qquad (1)$$

The S is the sampling rate, p is the percentage of the signal having bandwidth b, and b is the percentage bandwidth. ■

Therefore, the removal of redundant information alone is not sufficient to achieve the target bit rate of Fig. 2. To achieve greater compression, the irrelevant information must be removed as well.

2.4.2 Irrelevant Information in Wide-band Audio

The irrelevant information in the audio signal manifests itself as a result of the human audio-perception processing of the *cochlea*. The cochlea processes the incoming sound pressure wave [YaWS92] by trapping scale based features of the signal in a scalar-topological manner, as shown in Fig. 4.

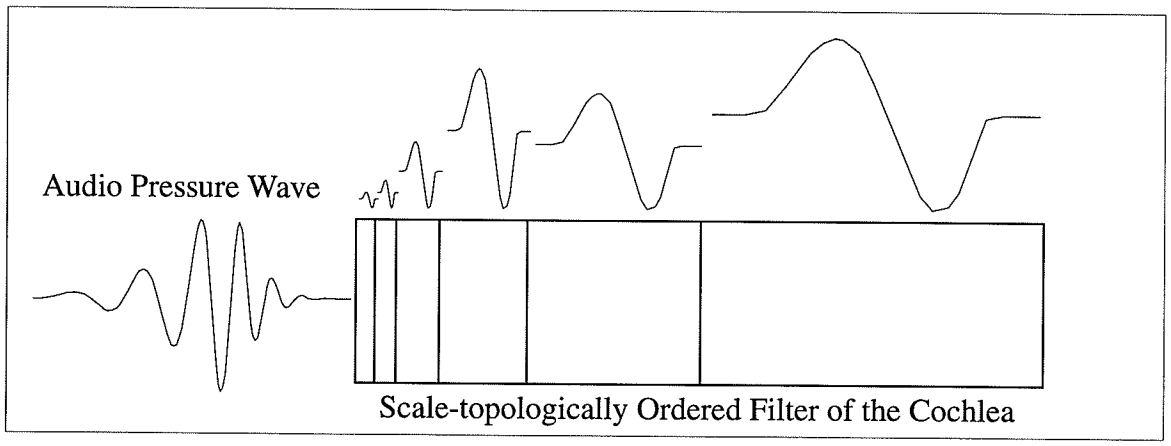

Fig. 4. Simplified model of the scale-topologically ordered filter of the cochlea.

Fine scales or rapidly varying portions of the signal are trapped by short spatial windows along the cochlear length, while slowly varying or course scale portions of the signal fall into the larger window bins. The literature shows that the size of the scale-windows are logarithmically related. (Actually, they are logarithmically related above 800Hz, and linearly related below 800Hz. But, the band 20-800Hz represents a small 4% of the audio bandwidth 20-22,050 Hz [YaWS92].)

Once the cochlea traps the scales of a signal into their appropriate bins, further cochlea and subsequent neuronal units apply a non-linear distortion to them. The amount of distortion applied to a given bin differs from another. The exact amount of distortion in

a particular bin cannot be quantified, since this measure is subjective and varies from one person to the next. However, a fuzzy range can be applied based on Fig. 1, the perceived degree of air pressure at a certain frequency. Compiling these observations, Fig. 5 shows an approximation of this non-linear processing inside the cochlea bins. In Fig. 5, the widths of the bins are illustrative of the degree of distortion.

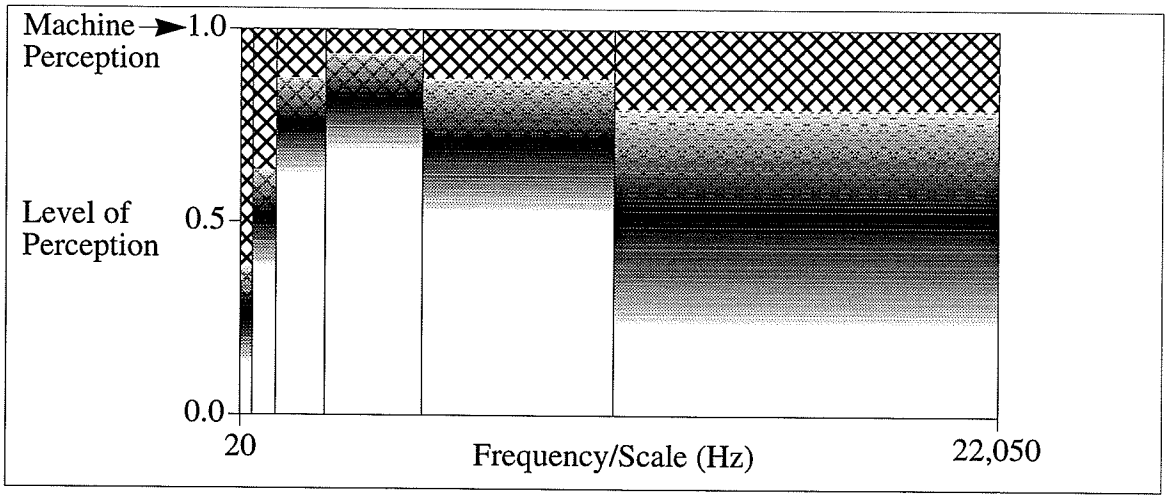

Fig. 5. Logarithmic bin model of perception of the cochlea. The area of machine perception (MP) includes the entire rectangle, while human perception (HP) contains the area underneath the fuzzy bars.

Note that this type of distortion is in direct contrast to machine perception. If a machine were to interpret the audio signal, then all scales would be equally as important. In the human case, not all of the information is relevant. Therefore, from a relevancy point of view, the cochlea can be modelled by a scale filter that splits the signal into *logarithmic bins of perception* within the bandwidth of 20-22,050 Hz, as shown in Fig. 5. The amount of irrelevancy is the difference between the areas MP - HP = ⊠ .

2.5. Coding Approach

As discussed above in Fig. 2, the standard compression rate of wide-band audio is approximately 64-256 kbps [Musm90], [LeGa91], [Spur90], [Lokh91], [Musm90], [CaTW90]. The objective of this section of the thesis is to achieve the same bit rate. The difference between the standard methods and this thesis is the use of the wavelet transform at the heart of the compression coding. The coding of the wide-band audio signal is approached from the point of view of modifying the perceptual characteristics of the signal, much in the same way as the cochlea's model developed above.

Figure 6 shows a block diagram of the audio compression communication system. As shown in Fig. 6, the stream of digital data is first transformed into an appropriate domain. This transformation of digital time domain data into another domain is done mainly to obtain a data stream that is similar to the stream processed by the human perception mechanism. Once in the transform domain, the transform domain coefficients are quantized. Quantization is done in such a way as to remove or reduce the resolution of those coefficients not contributing to the perceptually significant features of the source signal, much in the same way as the filter of the perception mechanism. This represents the irrelevancy removal. Also, this is the lossy portion of the compression, since some information (irrelevant information) is removed. In effect, this irrelevancy removal increases the redundancy space. Thus, the next step represents the redundancy removal. Redundancy removal is called first order entropy encoding, and is usually implemented by Huffman or arithmetic coding [WaFK93]. The coded data is then either stored for future reference, or directly transmitted to the receiver, whereupon the receiver decodes the data in reverse order.

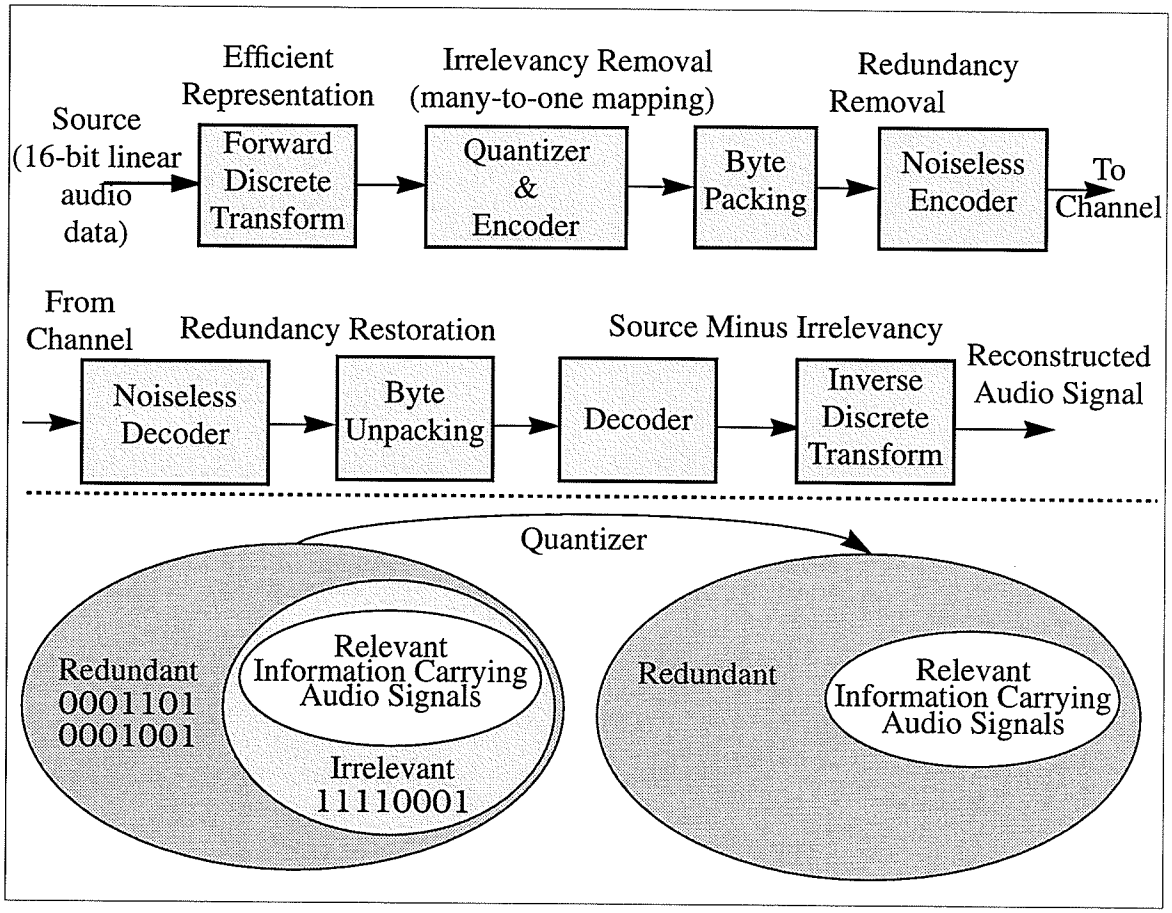

Fig. 6. Coding approach block diagram.

2.5.1 Transform Considerations

As mentioned above, the bandwidth of wide-band audio is approximately 20 kHz. However, the energy distribution in this bandwidth is uneven, typically, as shown in Fig. 7a. This uneven distribution provides motivation for using signal decomposition, or sub-band coding. The basic idea is to divide the signal into subbands, and to treat each sub-band according to its energy. From a signal theory standpoint, it is reasonable to assign more importance to those portions of the signal containing higher energy. From a perceptual standpoint, it is also reasonable that the division of the signal into subbands be done in a manner that is similar to the cochlea filter, as shown in Fig. 7b.

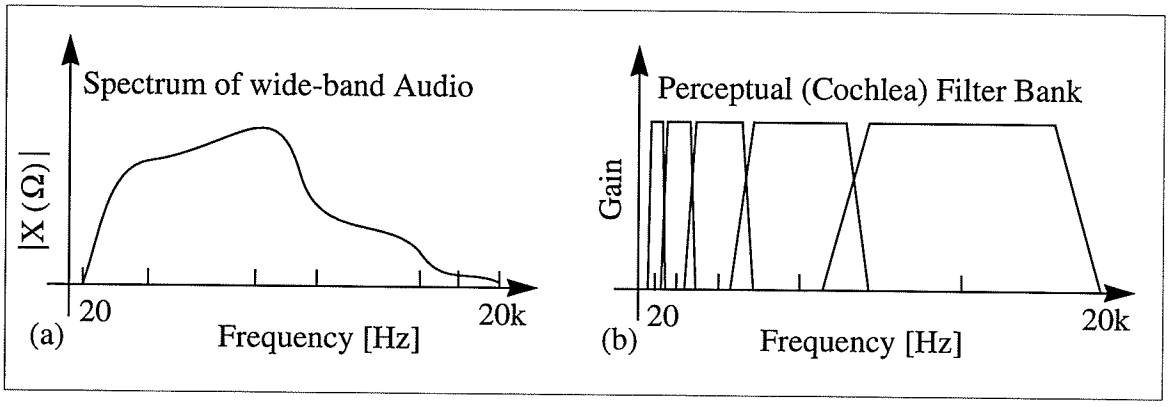

Fig. 7. (a) Energy spectrum of typical wide band audio. (b) Perceptual filter bank of audio.

2.5.2 Transform Comparison

In order to obtain an efficient transform, the transform must match the signal and the cochlea. Intuitively, from a signal compression standpoint, it is clear that a transform whose basis function most closely resembles the input signal would yield better compression ratios, because the transform coefficients would represent the signal more efficiently. The coefficients result from projecting the signal onto the basis function of the transform. Therefore, the coefficients express the degree to which the signal matches the basis functions.

In choosing a transform, the characteristics of the signal should be considered. In order to consider typical audio in the worst case, two extreme features of wide-band audio, a spike and a periodic waveform, are considered. Figure 8 shows an input signal consisting of a sinusoid and an impulse (Fig. 8a) and three transforms (Fig. 8b to Fig. 8d). As shown in the figure, the impulse basis (Fig. 8b) locates the impulse of the signal very well. But, its representation of the sinusoid is very redundant; almost all of the transform coefficients would have to be transmitted with equal energy in order to reproduce the sinusoid.

The Fourier basis (Fig. 8c) locates the sinusoid very well, but its representation of the impulse is likewise very redundant. The impulse, which very well could be irrelevant information, is buried by the Fourier transform. The wavelet basis (Fig. 8d) provides a *compromise* between the two former transforms. The wavelet transform locates both the impulse and sinusoid. Overall, the wavelet basis matches the input signal better than do the impulse and Fourier standard transforms. Notice that the high scale coefficients exhibit little energy (except for the impulse location), and so they may be allocated a small number of bits, while the small scale coefficients may be encoded with higher resolution.

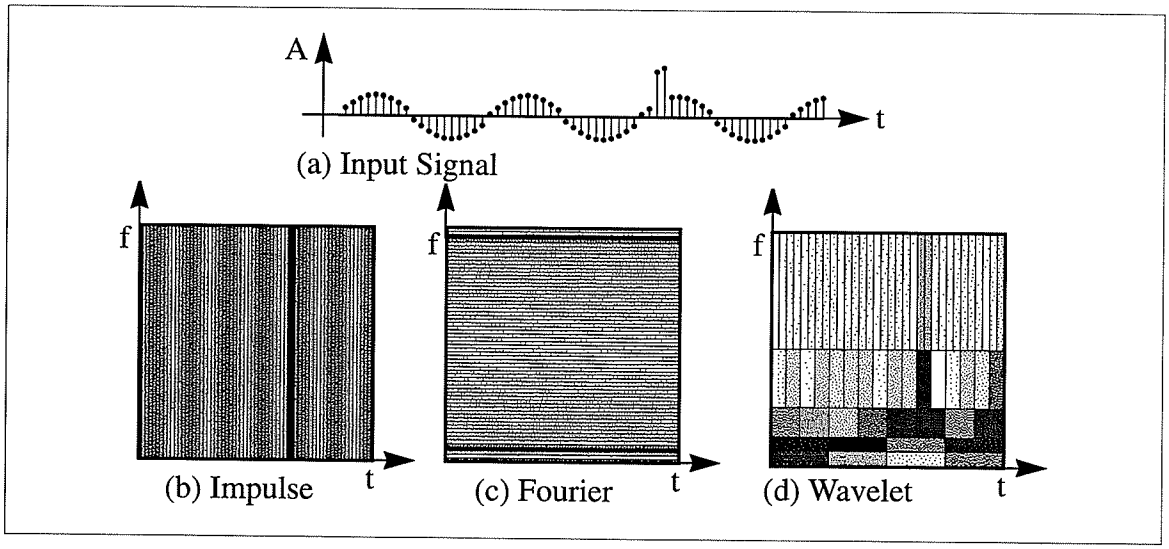

Fig. 8. Comparison of transforms (After [Herl93]).

2.5.3 Discrete Wavelet Transform

The discrete wavelet transform maps the time domain of audio signals into the time-scale domain, an example of which is shown in Fig. 9. As shown in the figure, the signal is segmented into sequential blocks of M samples. (M = 64 is used in this example.) Each block of time domain samples is consecutively transformed into wavelet coefficients. The wavelet coefficients $c_{j,k}$ are obtained by projecting the time-domain samples $x[n]$ onto the basis functions $\psi_{j,k}[n]$, as given by Eq. 2,

$$c_{j,k} = \sum_{n=0}^{N-1} x[n]\,\psi_{j,k}[n] \tag{2}$$

The functions, $\psi_{j,k}[n]$, are called scaled and translated versions of the basic basis function $\psi_{0,0}[n]$. While there are many different kinds of basis functions that can be used in the wavelet transform, such as the Harr, Daubachies, Malat, Meyer, Coiffman, and Spline [Mall89], [ViBL94], [CMQW89], this section of the thesis uses the *Daubachie 4 (Daub4)* functions because they most closely resemble the input, the time domain audio signal [ViBL94]. In other words these particular functions are used because they are *prototypical* of audio signals. The optimal value of M in the frequency content sense has been studied in [LaFK93], where it is shown that for speech, a block size of $M = 32$ yields the most efficient frequency decomposition (a higher value of M would only serve to decompose the frequency axis below 300 Hz, where there is little relevant audio information). For each frame (block of M samples), the time domain audio signal is projected onto each of the scaled wavelet prototype basis, thus forming $\log_2(N)$ subbands. In each subband the wavelet coefficients express the degree to which the signal matches the prototype basis function at that scale. Notice this transformation is similar to the model shown in Fig. 4.

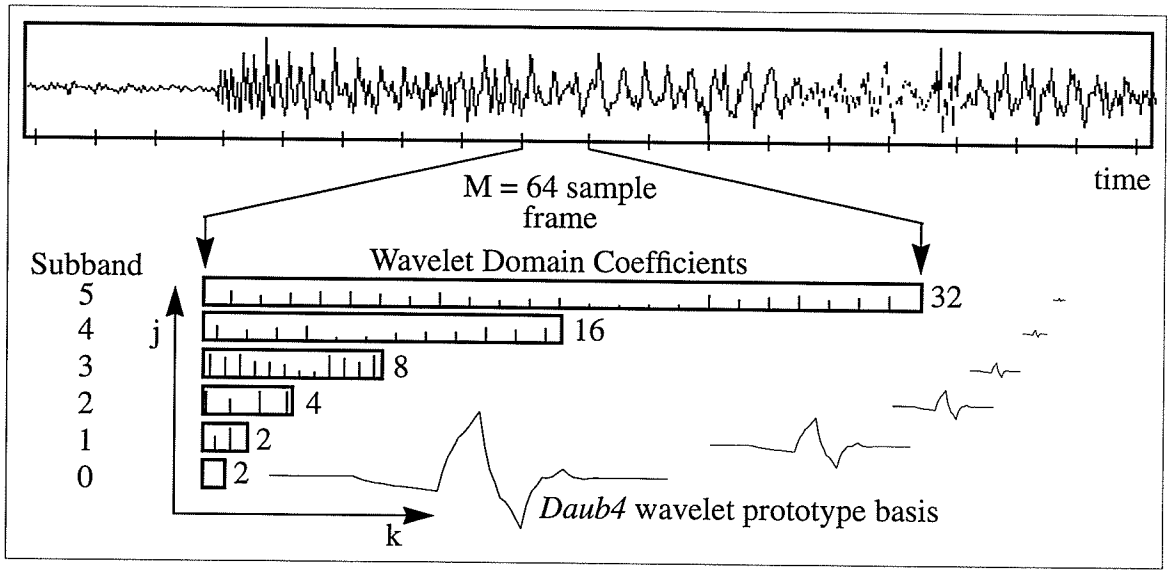

Fig. 9. Example of a discrete wavelet transform of an audio signal. M=64 is used here as an example.

2.5.3.1 Quantization of Transform Coefficients

Having transformed the signal into the wavelet domain, the next step is to quantize the coefficients in the wavelet subbands. There are three main types of quantization, scalar quantization, vector quantization, and overlapping vector quantization. The choice of which type of quantization to use is usually taken on the basis of first determining the granularity of the data. Granularity is defined as the average length of data that constitutes a feature in a signal. In this case the average length of data is the number of sequential wavelet coefficients. In this thesis, the wavelet coefficients of wide-band audio are assumed to be scalars. Therefore, scalar quantization is used.

When quantizing the coefficients, the main idea is to allocate an appropriate number of bits for each subband. Recall that the original audio waveform is sampled using 16-bits at 44.1 kHz throughout the signal's bandwidth. The idea of subband coding of sig-

nals is based on the observation that not all scale subbands need to be sampled at this resolution (16-bits). Some subbands may have more energy than others, and some bands may be more perceptually important than others. Therefore, the more energetic subbands should be allocated more bits. For example, the bit allocation algorithm would allocate 16-bits for the most energetic subband, and a decreasing number of bits for less energetic subbands. This is one method by which non-uniform sampling is implemented.

Once the decision is made on the number of bits to be used to quantize a particular subband, the next step is to consider how each individual coefficient within a subband is to be quantized. This should be done in a manner similar to the cochlea, as shown in Fig. 5.

Quantization within a subband can be done in a straightforward manner, such as uniform quantization. In this case each coefficient in a subband is rounded off to the nearest multiple of quantization step size that is determined by the energy content for that subband. However, uniform quantization treats all coefficients within a subband equally because of the assumption of the uniform distribution of signal features.

2.5.3.2 Non-uniform Quantization Within Wavelet Subbands

When the frequency of occurrence of coefficient amplitudes in a subband is uniform, then uniform quantization is optimal. Since each coefficient occurs as frequently as the others, then each coefficient should be treated equally. However, this is not the case in the wavelet domain of wide-band audio. Coefficients having amplitude near zero, occur much more frequently than large magnitude coefficients. In general, the distribution of features within a wavelet subband is non-uniform. The typical distribution of wavelet coefficients in a subband is as shown at the top of Fig. 12. On the basis of "the more fre-

quent a particular stimulous occurs the higher is the degree of the aroused behavor", it seems reasonable to allocate more bits to the coefficients occurring more frequently. Each coefficient in a subband is rounded off to the nearest multiple of the quantization step size that is determined based on the initial assessment of the energy-based quantization step size, and then the frequency of occurrence of that coefficient. This nested step size is bounded below by the number of bits initially allocated in the energy calculation (best resolution), and bounded above by the maximum amplitude range of the coefficients in the subband (sets all coefficients to zero).

Non-uniform quantization within a subband can be done using the well known μ-Law function, which allocates higher resolution to coefficients having amplitude magnitude near zero, and a log-decreasing resolution to higher magnitude coefficients [Jmax60]. The μ-Law function somewhat suits the distribution of wavelet transform data. But the distribution of wavelet domain coefficients of wide-band audio is only approximately suited to a logarithmic bit allocation. The μ-Law is oblivious to the exact probability density of these coefficients. When the density changes, the μ-Law does not accommodate the change. The μ-Law function does not adapt itself when the variance of the data changes. Therefore, the bit allocation function should not be fixed.

A better non-uniform quantizer is one that not only provides non-uniform step sizes, but adapts these step sizes according the probability density of the coefficients. The problem is this density is not known. However, it can be learned by a neural network. A neural network learns a distribution by training on representative data. The learning algorithm used in this section of the thesis is the frequency sensitive competitive (FSCL) learning, which is described in Section 2.6.3.

2.6. System Description

Software was written to implement the adaptive frequency sensitive wavelet sub-band coder. The main code was written in C, and the graphical user interface was written in XView. The software was run on a Sparc10 workstation. A Sparc10 was used, because it provided 16-bit recording and playback of audio at sampling rates of up to 48 kHz.

The software consists of

(i) Graphical user interface (GUI),

(ii) The forward and inverse wavelet transforms,

(iii) The frequency sensitive competitive learning neural network for learning code-books and for quantization based on the learned codebooks,

(iv) Byte packing, and

(v) Arithmetic and Huffman entropy encoder.

2.6.1 The Graphical User Interface (GUI)

All of the functions, except for the arithmetic and Huffman entropy encoders, are available through the GUI of Fig. 10. This GUI is a modified version of soundtool.c1.31 92/06/23 Copyright 1991 Sun Microsystems [Math93]. The *File* and *Goodies* menus and all of the functions underneath these menus were added by the author. From the *Goodies* menu, the dialog box of Fig. 11 is obtained. From this dialog box, the following functions and controls are available:

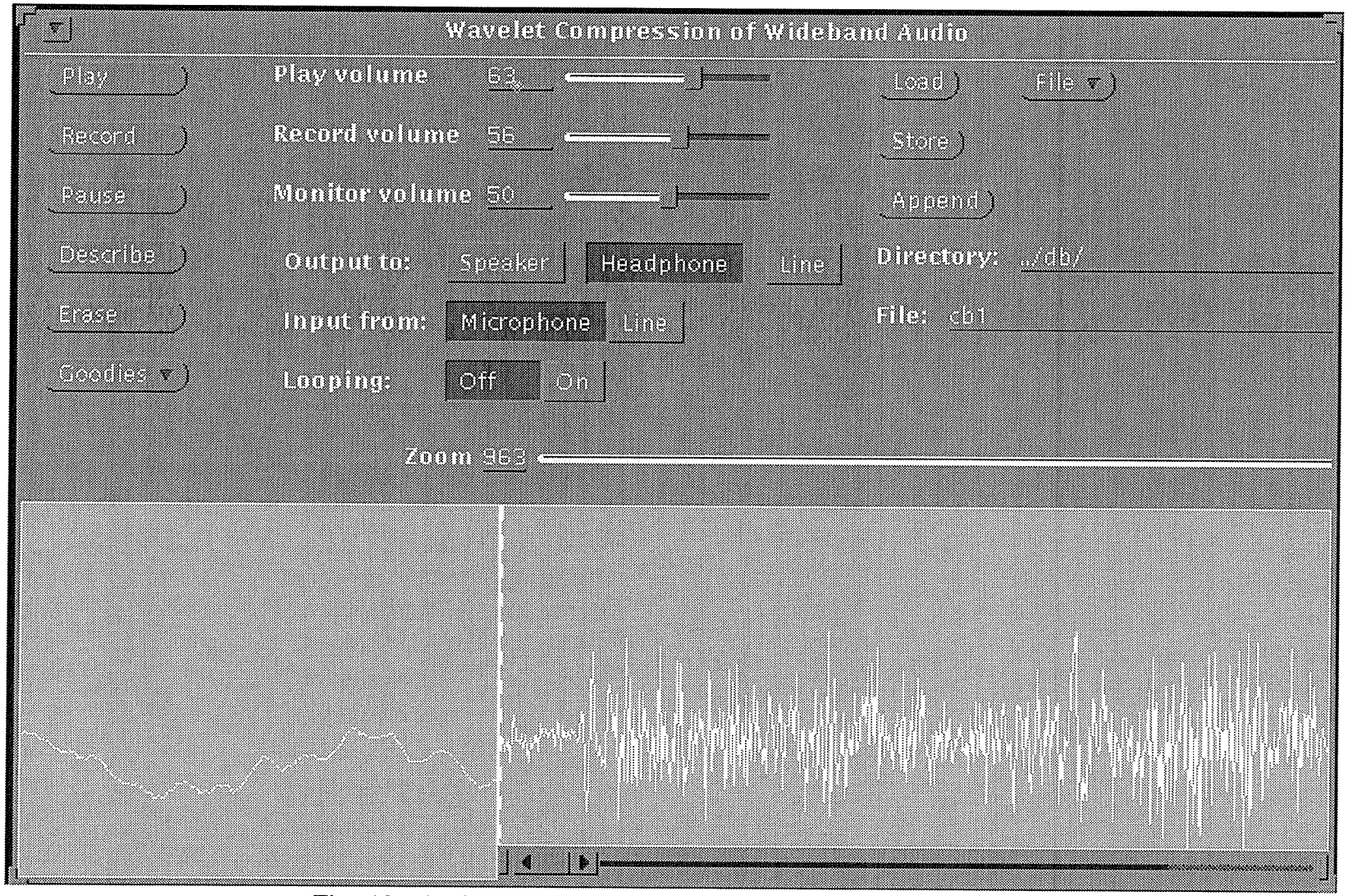

Fig. 10. Main window of GUI. This GUI is based on sound-tool.c1.31, 92/06/23, Copyright 1991 Sun Microsystems.

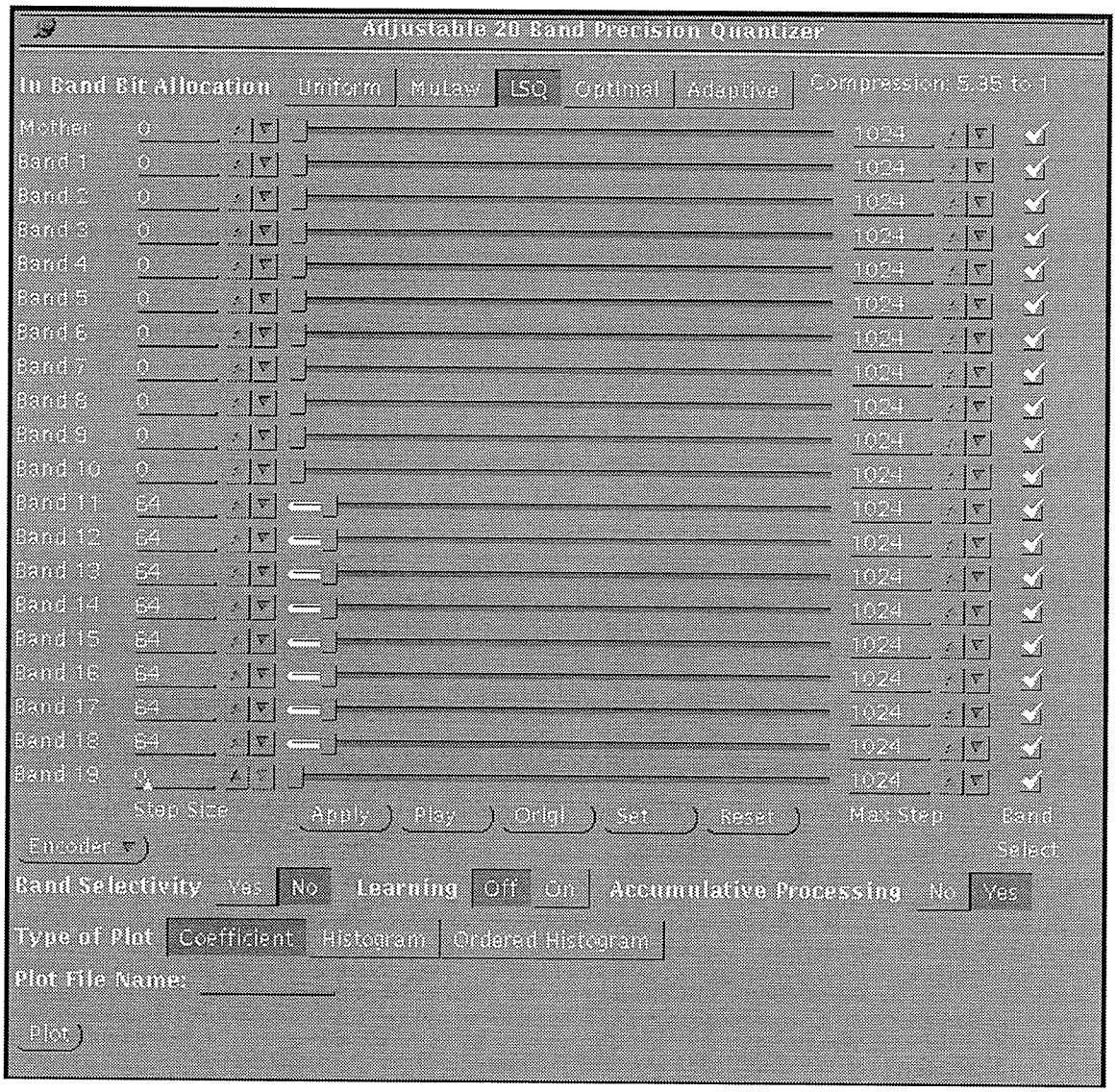

Fig. 11. Controls for the wavelet subband quantizer.

(i) Selection of subband quantization schemes, including the implemented schemes of uniform, μLaw, learning scalar quantization, and the not implemented schemes of optimal, and adaptive.

(ii) Manual selection of quantization step size for each wavelet subband.

(iii) Band select switches for enabling/disabling band processing.

(iv) Frequency sensitive competitive learning (FSCL) switch.

(v) Accumulative processing for iterative quantization.

2.6.2 Forward and Inverse Wavelet Transform

The forward and inverse wavelet transform software used in this section of the thesis is based on [PTVF92]. The wavelet transforms found in this reference were used as functions called by the main program.

2.6.3 Frequency Sensitive Competitive Learning (FSCL) Neural Network

Figure 12 shows the structure of a frequency sensitive competitive neural network [McAR90]. The network learns one codebook for each subband. The training data consists of wavelet coefficients from a particular subband obtained from a representative data base of wide-band audio signals (i.e., similar music). The neural network is forced to extract the most common features exhibited by the training data. As a result, this network provides high resolution to data located near the mean value (most common features), and lower resolution to infrequent data, as shown in the top portion of Fig. 12.

This non-uniform allocation of bits is determined based on the probability density of the data. The neural network learning algorithm is based on a winner take all strategy, where the winner is determined by the Euclidean metric, weighted by a frequency sensitive term. This sensitivity forces the individual neurons to win equally, thus tending to maximize entropy, while minimizing the energy of the error. The error is calculated using the Euclidean metric because it conveys energy, and energy of an audio signal is important to the human listener.

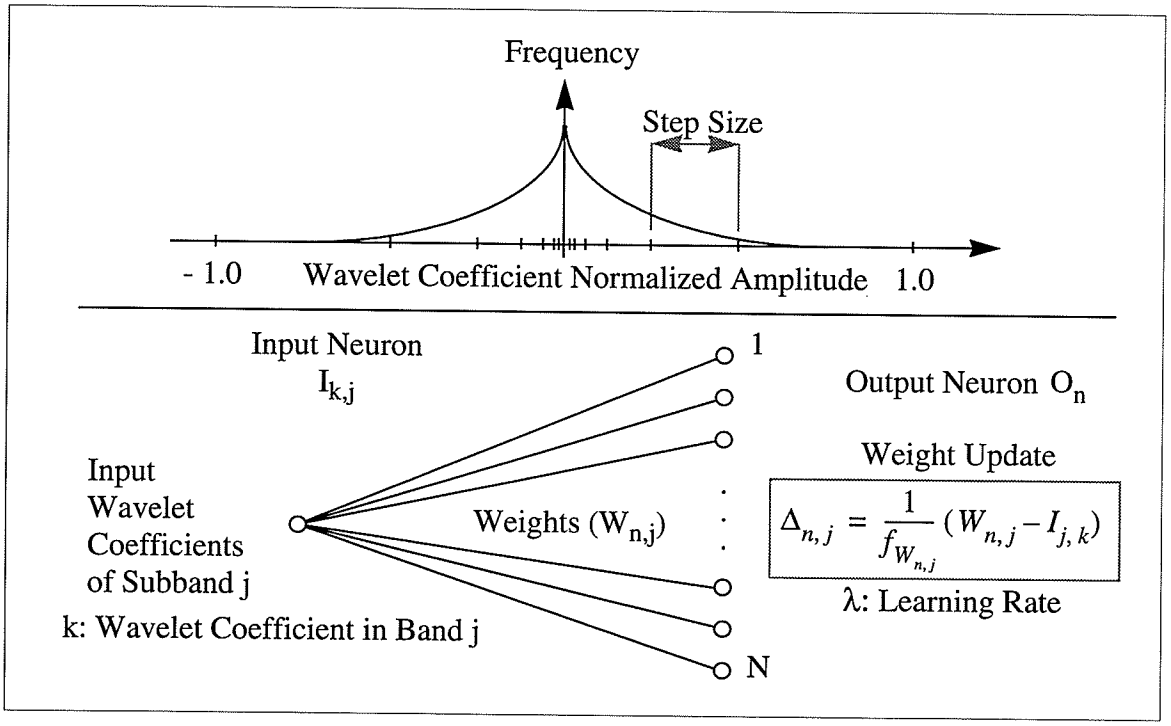

Fig. 12. Neural network structure for learning wavelet coefficient distributions.

2.6.3.1 FSCL algorithm

The following steps give a detailed description of the training procedure of the frequency sensitive competitive learning for scalar quantization of wavelet subband coefficients:

STEP 1: Initialize each weight to the mean value of the coefficients of subband *j* plus some small random perturbation.

STEP 2: Randomly select one of the coefficients of subband *j* and present it to the network.

STEP 3: Compute the distance between each weight and input coefficient, as follows:

$$\Delta_{n,j} = \frac{1}{f_{W_{n,j}}}(W_{n,j} - I_{j,k})$$

where $W_{n,j}$ is the n*th* weight of the codebook for subband *j*, $I_{k,j}$ is the k*th* wavelet coefficient of subband *j*, $\Delta_{n,j}$ is the change in the n*th* weight of the codebook for subband *j*, and $f_{W_{n,j}}$ is proportional to the number of times the n*th* weight of the codebook for subband *j* has won a competition in the past.

STEP 4: Select as the winner that output neuron whose weight $W_{n,j}^{\circ}$ is closest to the input coefficient.

$$W_{n,j}^{\circ} = \min_{1 \le n \le N} \{\frac{1}{f_{W_{n,j}}}(W_{n,j} - I_{j,k})\}$$

STEP 5: Update the weight vector $W_{n,j}^{\circ}$ connected to the winner as follows:

$$W_{n,j}(new) = W_{n,j}(old) + \frac{1}{f_{W_{n,j}}}(W_{n,j} - I_{j,k})$$

STEP 6: If *NTV* = end-of-training (EOT) then stop; otherwise goto STEP 2.

2.6.4 Byte Packing Algorithm

Once the wavelet coefficients in each subband have been quantized and are represented by the FSCL codeword indices, the next step is to transmit the indices to the receiver. Before they are actually transmitted, the codebooks and indices for each subband are passed through the byte packing algorithm. The purpose of the byte packing algorithm is to form a compact representation of the data to be transmitted to the receiver. The data to be transmitted consists of a header followed by the frame data stream, which is shown

in Fig. 13.

The header consists of the frame size, followed by the data of N codebooks. The data representing each of the N codebooks consists of the number of codewords in that codebook, followed by the actual normalized codewords. The number of codewords in a codebook implies the bit-length of each codeword. The frame size indicates to the receiver the value of N, the number of codebooks to expect. The number of subbands is equal to the number of codebooks. Also, the frame size gives the number of coefficient indices per subband.

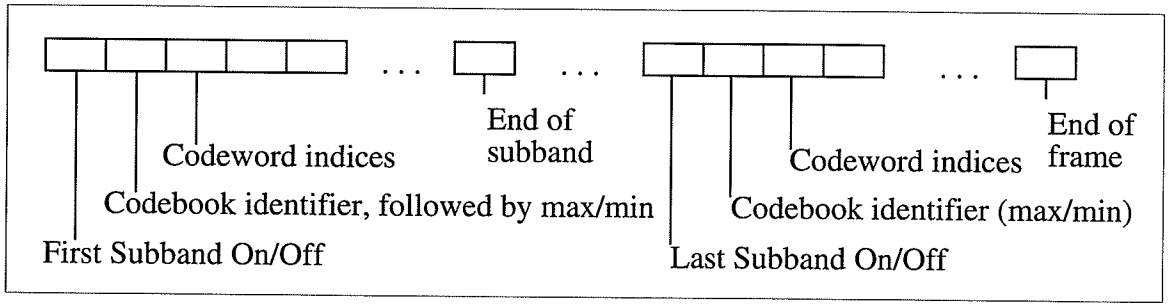

Fig. 13. Frame data stream of the adaptive frequency sensitive wavelet subband coder.

If the subband on/off bit is zero, then no data follows for that subband. This means zero bits were allocated to this subband, due to the lack of energy. If the on/off bit is one, then the next three bits identify which codebook should be used to decode the forthcoming indices. After the three bit identifier, the peak values of the subband are transmitted, using 4-bytes for the positive peak, and 4-bytes for the negative peak. The codewords are normalized amplitudes, so peak values are required. Following this, the codeword indices are transmitted.

2.7. Experiments and Results

Raw audio data contained in selected classical and pop music was copied from the CD to the hard disk.

Experiments were run in order to determine the optimal frame size, where the optimality was measured in terms of the amount of overhead that was required to encode a frame. It was found that the larger the frame size, the smaller the required bit rate. This is because the overhead, e.g., data required to represent the codebooks, decreases for increasing frame size. For different frame sizes, the statistics of the audio samples will vary, and so it is assumed that different codebooks will be required. The trade-off is the resolution. If the statistics of the audio change very drastically within a large frame, then the codebooks would yield larger quantization error. Weighing these factors, a frame size of 1 M was chosen.

Having obtained the digital audio samples, the audio signal was then transformed by the wavelet transform, and decomposed into wavelet subbands. Since the frame size is 1 M, there are $N = 20$ subbands. Since a dyadic scaling is used, there are 2^{19} coefficients in the $19th$ subband, 2^{18} coefficients in the $18th$ subband, ..., 2^1 coefficients in the $1st$ subband, and 2^1 coefficients in the $0th$ subband. Figure 14, Fig. 15, and Fig. 16 show the distribution of coefficients in subbands 19, 18, and 17 of typical pop music.

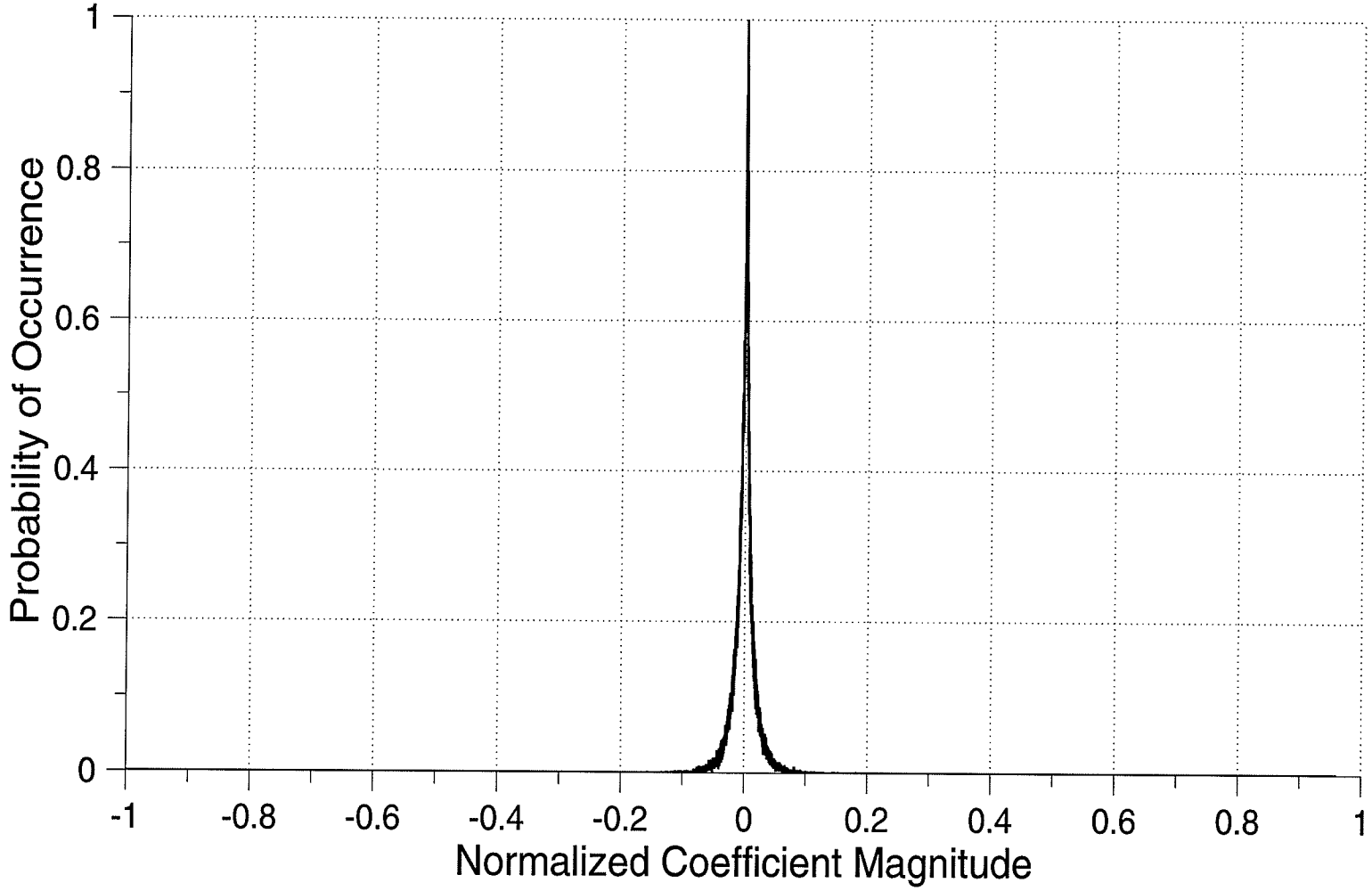

Fig. 14. Utilization of wavelet coefficients in band 19.

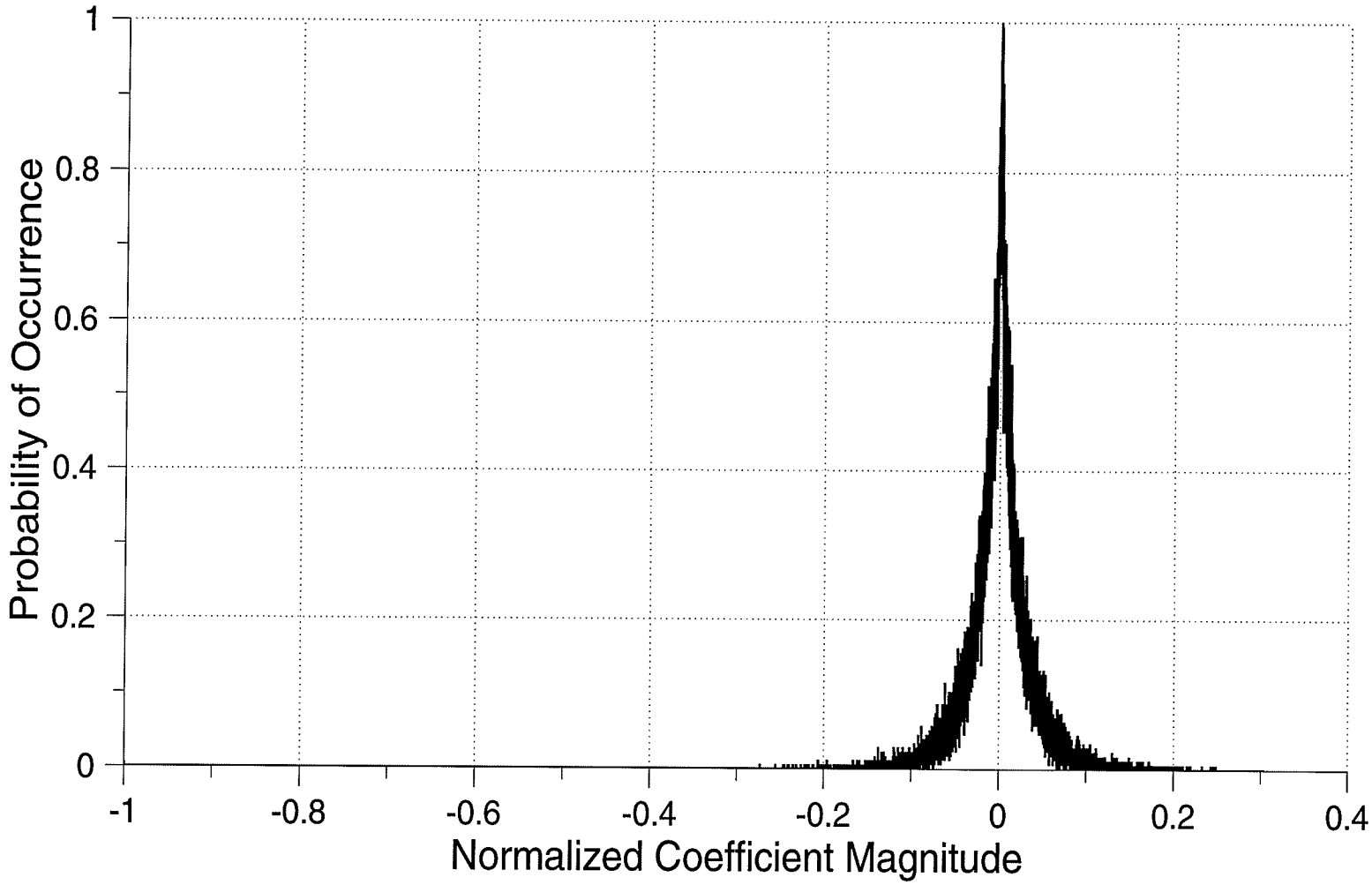

Fig. 15. Utilization of wavelet coefficients in band 18.

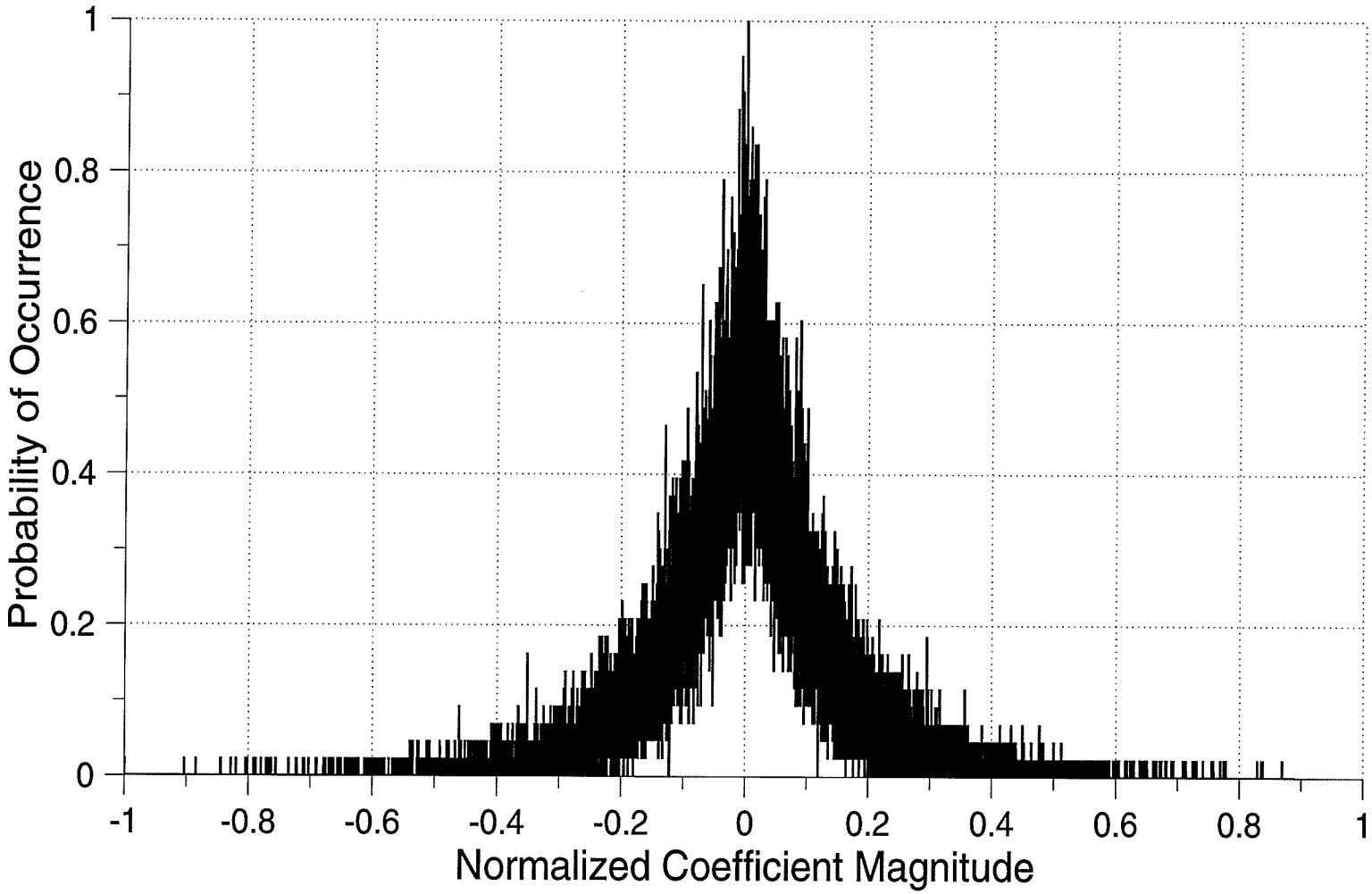

Fig. 16. Utilization of wavelet coefficients in band 17

The FSCL network trained on the coefficients of each subband separately, and formed codebooks of different sizes, ranging from 2, 4, 8, 16, 32, 64, 128, and 256 codewords each. A good codebook size was found to be 64.

Once the codebooks were learned, then the actual quantization took place. A coefficient is quantized by rounding it off to the nearest codeword in the codebook for the subband to which the coefficient belongs. This process is the lossy stage, because of the incurred quantization error.

After the lossy compression stage, the data was packed into bytes using the byte packing algorithm developed for this application.

Finally, the data was passed through a post-processor for further reduction of redundancy, due to the many coefficients being set to zero. An arithmetic entropy encoder [WaFK93] was used to remove the redundancy.

Under this system, a bit rate of 150 kbps was required to achieve no perceptual loss of quality for the selected classical and pop music, as shown in Fig. 17. The subjective evaluation as shown in Fig. 17 and including the "no perceptual loss of quality" statement above is based on the judgement of this author and the participants at the numerous demonstrations of the work to industry, government, other researchers, and the community at large. These judgements are not formally obtained from mean opinion scores (MOS). The objective evaluation was determine by measuring the SNR using the Euclidean metric. As shown in Fig. 17, the SNR is very low. This is because the wavelet/FSCL coder is based on maximizing the fitness of a human perceptual model. The fact that the Euclidean based SNR is low, while the perceived quality is high demonstrates the inappropriateness of conventional distortion metrics (in particular, the Euclidean metric) to measure perceived features (in this case perceived error) of signals.

Method	Compression Ratio	SNR	Quality (of 10)
Uniform	5:1	21.5	6
μ Law	5:1	21.5	7
LSQ	5.4:1	22.4	8
Original	1:1	-	10

Fig. 17. Typical experimental results applied to the selected music. The subjective evaluation is not obtained from mean opinion scores (MOS).

2.8. Summary of Chapter 2

In summary, this chapter presents an adaptive wavelet subband coding of wide-band audio. The coefficients in the wavelet subbands are quantized using a scheme that adapts to the subband signal by setting the quantization step size for a particular subband to a size that is inversely proportional to the subband energy, and then, within a subband, modifies the energy determined step size as inversely proportional to the amplitude probability density of the coefficient. The amplitude probability density of the coefficients in each subband is modelled using learned scalar quantization employing frequency sensitive competitive learning. The source data consists of 1-channel, 16-bit linear data sampled at 44.1 kHz from a CD containing major classical and pop music. Preliminary results show a bit-rate of 150 kbps, rather than 705.6 kbps, with no perceptual loss in quality.

The wavelet transform provides better results for representing multifractal signals, such as wide band audio, than do other standard transforms, such as the Fourier transform. This is because the wavelet transform offers a compromise in the resolution of time and frequency. It provides good time localization of high frequency components of the wide band audio signal, while maintaining good resolution of the global, low frequency components. The wavelet transform provides an automatic signal decomposition that is comparable with that of the cochlea filter. Furthermore, unlike other transforms, the wavelet transform utilizes a basis function that is prototypical of the audio signal. The self-similar nature and energy content of the scaled and translated wavelets in the time-frequency plane resembles that of wide band audio, and it is this feature that makes the wavelet transform a very good transform from the signal compression viewpoint.

CHAPTER 3

Image Compression Using Learned Vector Quantization

3.1. Introduction

The previous section of this thesis dealt with compressing the audio signal. An equally important signal in multimedia is images. Communication via still images plays an important role in the graphics and moving-images of multimedia. The use of images drastically improves the communication of ideas. But, with this improvement comes a very high price, data storage. For example, many universities, schools, and businesses maintain a database in which information on people is stored. This database is typically in the form of text only. If the database were enhanced with still images of head and shoulder pictures of people, the cost of storing this data would be very high. For instance, a given university stores records on approximately 25, 000 students a year. If an 8-bit 256x256 image were taken of each individual, this would require (25,000)(256x256) = 1.6384 GBytes of storage. Not only is the storage requirement large, but the cost of transmitting this data is also high. Transmitting large amounts of data through low bit-rate channels requires long periods of time. Finally, in this application, privacy and security are other problems associated with uncompressed data. Data compression can provide a natural

source of data encryption.

Applications dealing with image identification do not require high quality. Human beings can identify facial patterns very effectively, even in noisy circumstances. Some information in these images can be removed, while still maintaining sufficient quality for identification.

There are many compression schemes available for data and signal compression. A taxonomy of the major data and signal compression techniques is given in [Kins91]. When selecting a particular compression scheme, it is required to consider the type of signal that is to be compressed. Images are inherently imperfect data. That is, minor losses of data (distortion or noise) in images are tolerable to the human eye. For example, it has been established that we cannot distinguish brightness levels less than approximately 2% of 8-bit resolution [Lehn91]. Furthermore, our perception plays a major role in our visualization of the real world. When we observe images and store them in our memory, we are most likely performing some kind of compression. If we did not do such compression, the amount of data we would have to store in order to recall ordinary events would overflow the memory capacity in our brains in a very short time.

Since images are imperfect data, we choose a lossy compression scheme. As a result the decompressed image will have some degree of distortion. However, for the most part, this distortion is irrelevant. Moreover, because of the inherent nature of the problem, we would tend to choose a compression scheme that mimics the compression that is involved in our, i.e., human, image processing system. This leads to a choice hereby called image compression using learned vector quantization incorporating the use of an artificial neural network (to mimic human signal compression).

3.2. Vector Quantization

Vector quantization is a non-uniform and many-to-one mapping, Ψ, of a vector space, V, into another vector space, S. Many elements of V are mapped into a single element of S. In this way, the space V is partitioned into a number of regions equal to the number of elements in S. There is a non-uniform number of elements of V that are assigned to a region. Each region generally contains a different number of elements from V. The mapping is referred to as an equivalence relation. Many vectors are mapped into a single region because they are equivalent in some sense. Information in V is removed in the mapping, and as such, the mapping is a lossy compression technique.

As a consequence of choosing a lossy compression scheme (vector quantization), we must consider the associated measurements of compression and distortion. The grey-scale image is defined as an m rows by n columns array of 8-bit pixels. Thus, the uncompressed image requires mn bytes of storage. The amount of compression of an image can be measured by a compression ratio or a compression rate. The *compression ratio* (CR) is defined as the number of pixels transmitted or stored divided by the actual number of pixels in the original image, as given by Eq. 3.

$$CR = \frac{Pixels\ transmitted}{Original\ pixels} \quad \left(\text{or} \quad CR = \frac{Original\ pixels}{Pixels\ transmitted} : \frac{Pixels\ transmitted}{Pixels\ transmitted} \right) \tag{3}$$

In other words, the compression ratio gives the memory savings in that it tells us the size of the compressed file is CR ($CR <= 1$) times the original file. The *compression rate* (R) is defined as the number of pixels in a vector divided into the number of bits required to represent that vector (*bits/pixel, bpp*), as given by Eq. 4. For example, if each original vector consists of one byte and no compression is applied, i.e., the original file is transmitted

through the communications channel unmodified, then the compression rate, $R = 8$.

$$R = \frac{Bits\ required\ to\ represent\ vector}{Pixels\ in\ a\ vector} \tag{4}$$

The quality of an image can be evaluated through a subjective evaluation and an objective evaluation. A conventional objective measure involves a *peak signal-to-noise ratio* (*PSNR*), as given by Eq. 5, where *MSE* is defined as the point-wise Euclidean distance between original and reconstructed pixel.

$$PSNR = 10\log\left(\frac{255^2}{MSE}\right) \tag{5}$$

Notice that the argument of the log function is the highest signal-to-noise ratio possible, since the original signal is assumed to be at maximum brightness throughout, thus the name *peak* signal-to-noise-ratio (PSNR). The *PSNR* reduces variability between other experimenter's measurements. For example, two pairs of images (each consisting of an original and a decompressed version) may have the same mean squared error *(MSE)*. However, one of the original images may be brighter than the other, and this would give a misleading *SNR*.

The *distortion measure* is the cost of representing an image vector by a prototype vector. While there are many conventional metrics that can be used, perhaps the best choice is the Euclidean metric. When we view images, it is the intensity of the image that conveys the information. Intensity is related to energy, and the Euclidean metric measures energy. But the Euclidean metric can be deceiving sometimes. For these cases other measures can be taken [Kins91].

Having established the compression type and measurements of compression and distortion, the next section discusses how images, image vectors, regions, and prototype vectors apply to vector quantization.

3.2.1 Image Vector Quantization (VQ)

There are three stages of image *VQ*, and they are prototype creation, image compression and transmission, and image reception and decompression.

3.2.1.1 Prototype Creation

In the first step, the prototypes must be developed. Given an image, instead of representing the image in terms of *m* by *n* pixels, group together an array of local pixels, and call each array a vector IV_i. The maximum value of *i* depends on the number of component pixels *q*, chosen to comprise a vector. Call the set of all such vectors, ***V***:

$$V = \{IV_i \mid i = 1, 2, 3, \ldots, mn/q\} \tag{6}$$

where,

$$IV_i = \{p_1, p_2 p_3, \ldots, p_q\} \tag{7}$$

The p_1, p_2, ..., p_q are the 8-bit pixels chosen to comprise a vector.

Some of these vectors, $IV_i \in V$, will be similar in some sense. For example, many vectors may be extracted from a uniform area of an image. A statistically similar feature of these vectors may be very simply a correlation of their brightness levels, or it can be some unknown statistic. This leads to the following idea. From the set, ***V***, that contains all such vectors from a certain image, determine r ($r \ll (mn)/q$) fundamental features exhibited by the set. Then, map each vector IV_i, to a feature set on the basis that it most closely exemplifies that feature.The smaller the number of features (i.e., smaller r), the

greater is the compression. In other words, the set of vectors, V, is partitioned into r distinct sets, S_i. Each subset, S_i, represents a certain fundamental feature, that is exhibited by its member vectors.

The mapping from an image vector to a feature set is actually done by mapping the image vector to a representative vector of the feature set. So, represent each subset, S_i, by a representative, or a prototype vector, call it PV_i. The prototype vector, PV_i, is chosen in such a way as to optimally represent the image vectors, IV_i, that have been assigned to the feature denoted by the feature set, S_i. Note that this prototype vector may or may not be an element of the set, S_i. Due to this mapping, each of the vectors in subset, S_i, is rounded off or quantized to a prototype vector. In other words, the assignment acts as a many-to-one mapping.

The set of prototype vectors constitutes the codebook. For every feature, there is a prototype vector. This codebook contains r vectors, each of which has an address, (1, 2, ..., r). Because the number of selected features is much smaller than the number of vectors in the image, the number of bits required to represent the address of a prototype vector is much smaller than the number of bits required to represent an image vector.

For example, Fig. 18 shows the scheme that is used in this section of the thesis. The image vectors are chosen as square blocks, where each block consists of 4x4 pixels. Theoretically, these vectors are represented by points in an image vector space. This space is then segmented into 256 feature sets, or feature regions. For each feature set, a prototype vector is determined, and this prototype vector represents the feature. All 256 prototype vectors are collected into a prototype table. The prototype vectors are also called codewords, and the prototype table is called a codebook.

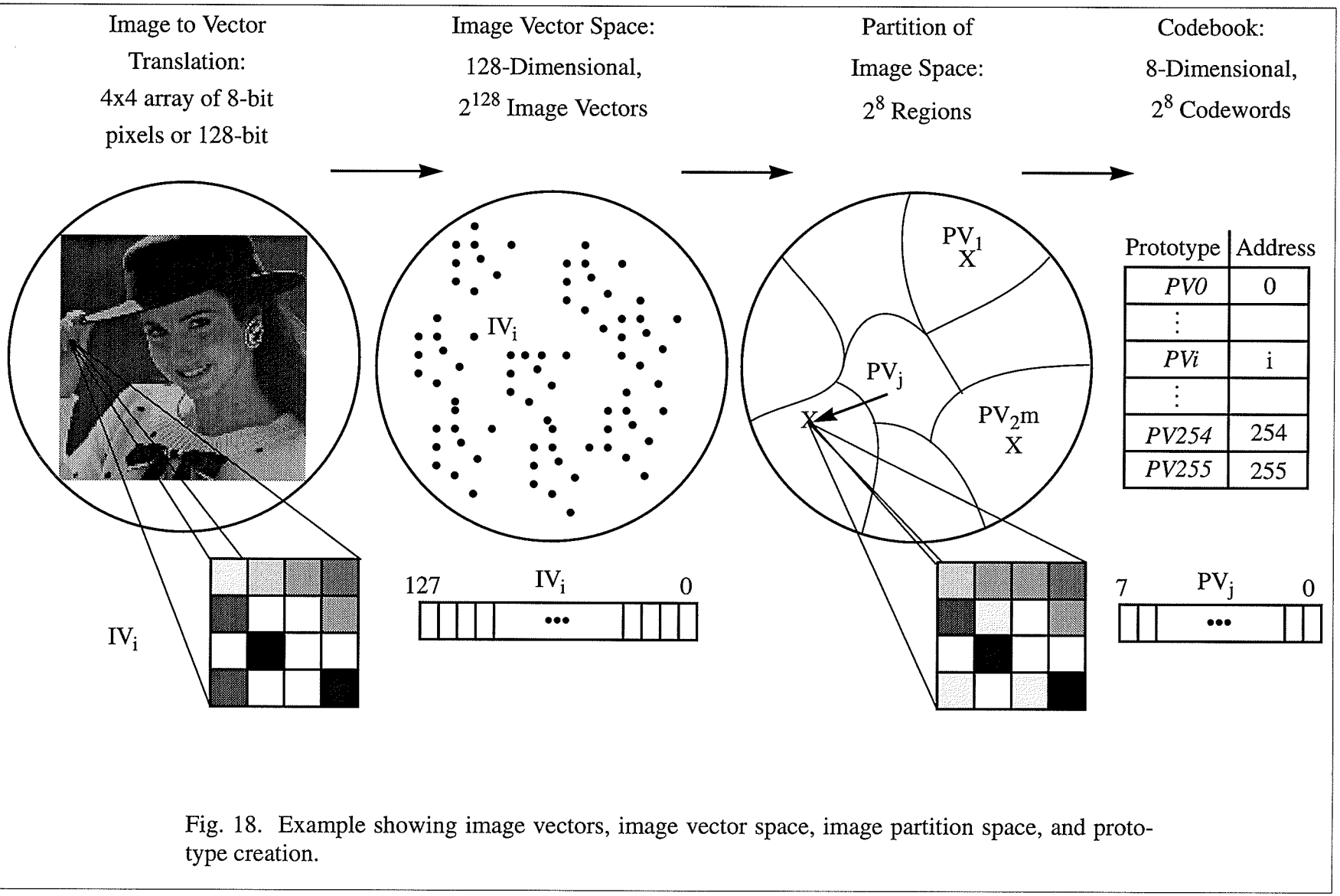

Fig. 18. Example showing image vectors, image vector space, image partition space, and prototype creation.

3.2.1.2 Image Compression

Now that the prototype vectors have been determined, the second step of *VQ* consists of image compression and transmission. Each vector in the original image is compared one-at-a-time to each of the prototype vectors in the codebook. The prototype that most closely resembles the input vector is selected, and its *address* is transmitted through the channel. A sequence of *mn/q* (*8-bit*) addresses is transmitted, because there are *mn/q* vectors in the image. Thus, the original image consisting of *mn* pixels is represented by *8mn/q* bits rather than *8mn* bits. The compression ratio is *q*. The codebook is resident at both the transmitter and receiver.

3.2.1.3 Image Decompression

The final step of *VQ* consists of receiving the compressed image, i.e., a sequence of *mn/q* addresses, and decompressing it. Each of the *mn/q* addresses is used as a pointer to look up the codeword in the codebook. The sequence of these prototype vectors is used to reconstruct the image.

Continuing the example of Fig. 18, Fig. 19 shows the compression, transmission, and decompression of an image. Since each image vector consists of 128-bits (*q=16 implies* ($\Rightarrow$) *16x8 = 128 bits*), and since only 8-bits are required to represent an address of a prototype vector (256 selected features), this represents a compression ratio of 16-to-1 (16:1). Note, that the codebook must be transmitted to the receiver also, but this is relatively small overhead in comparison to the number of indices that are transmitted.

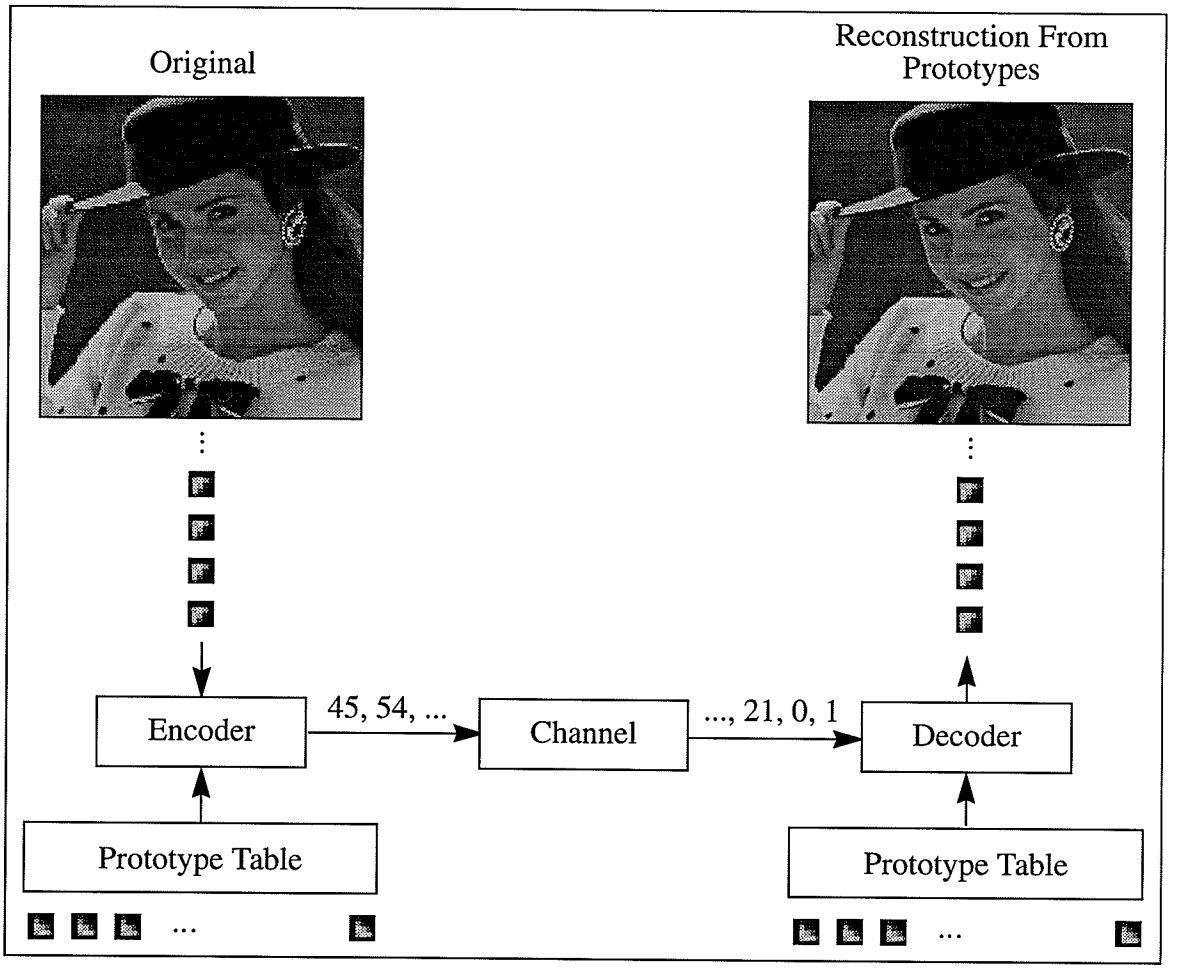

Fig. 19. Example of image VQ compression, transmission, and decompression.

Note that the decompression at the receiving station does not undo exactly the compression performed at the transmitter. This is because of the many-to-one mapping. For each feature represented by set, S_i, there accumulates an error equivalent to the sum of the differences between the exemplar image vectors belonging to set, S_i, and the prototype vector. The total error for the entire image is the sum of all feature set errors. This error is called round off error, reconstruction error, or distortion. One of the main objectives in vector quantization is to use a codebook that minimizes the *relevant* distortion.

This is more easily said than done, because the equivalence relation is unknown. What equivalence relation should be used in partitioning the image vector space and to

create the prototypes? One way of answering this question is by determining the full statistical properties of the input image. However, these statistics are generally not known. This work deals with compressing head and shoulder images of people. We cannot possibly obtain all images of people that have occupied this planet since day one. Therefore, we cannot obtain exact statistics. Even if we were to find a statistical property correlating or clustering the vectors of an image in some sense, we are not guaranteed that this property is similar to one of the, perhaps, many properties we use when we perform image identification and compression. But this is the ultimate goal. We are trying to *discover* the features we use when we try to identify images. In this section of the thesis, an artificial neural network (*ANN*) is applied to the task of prototype determination, as discussed in the next section.

3.2.2 Learning Prototypes using Competitive ANNs

The prototypes in vector quantization can be *learned* by competitive learning (CL) artificial neural networks. There are a few variations of competitive learning networks, each of which has evolved as a result of a basic problem associated with the fundamental competitive learning network [Kins91], [AKCM90], [McAR90], [NaFe88], [TrMe90]. The most basic of these variations is hard Competitive Learning (*CL*).

3.2.2.1 Underutilization of Prototypes

The basic problem associated with *CL* is underutilization of prototypes. This means, once learning is complete, some codebook vectors are used very little by the input

image vectors. Now, if a so-called prototype is used but once in quantizing an image, we would most likely not notice its presence in the image. Furthermore, if a codebook vector is used very little, then it hardly fits the definition of a prototype, that being a vector that is representative of many patterns in an input image space. In this case the network is discriminating too much. The under-utilized prototype is most likely specific to a certain image, not a common feature of many images. This is not very efficient use of the codebook.

3.2.2.2 Solutions to Underutilization

There are solutions to the underutilization problem. These solutions range from soft competitive learning [YaZG92], growing and pruning competitive networks, frequency sensitive competitive learning (*FSCL*) [AKCM90], and Kohonen's self-organizing-feature-map (*SOFM*) [Koho90].

Growing and pruning competitive networks dynamically (while learning) either (i) remove all of the codewords that have a utilization frequency less than some threshold, say one, and/or (ii) add codewords when the current codewords become over-utilized, again with respect to some threshold. When removing units, this means that the size of the codebook would decrease, the compression ratio would increase, and we would, perhaps, not even notice a change in the resulting quality. Over all, the codebook becomes more efficiently utilized.

Another solution is to incorporate a constraint into the network that ensures all vectors in the codebook are being used equally and uniformly. This can be implemented by decreasing the probability that a currently frequent winner neuron wins in the future by

weighting its subsequent metric calculations with a parameter reflecting the number of times it had won a decision in the past. This would tend to ensure that when an image is decompressed, all codebook vectors would be used with about the same frequency, so that the codebook can be said to be efficient in the utilization sense. This type of learning in known as frequency sensitive competitive learning (*FSCL*). *FSCL* uses a frequency of winning parameter in the similarity metric. This decreases the probability that a frequently winning neuron will win a competition in a future presentation of an image vector, and so increases the probability that a not so frequently winning neuron will win a competition.

Yet another solution is Kohonen's self-organizing-feature-map (*SOFM*). SOFM addresses the underutilization problem.The *SOFM* network uses the idea of a neighborhood. One main difference between *FSCL* and *SOFM* is that the neighborhood method induces an organization in the output neuron space, while *FSCL* has no such organization.

Since all of the above network structures are basically equivalent, except for the frequency sensitivity and neighborhood additions, only a detailed description of the *SOFM* is given here.

3.2.3 SOFM Competitive Learning

Figure 20 shows a 2-D structure of the SOFM neural network.

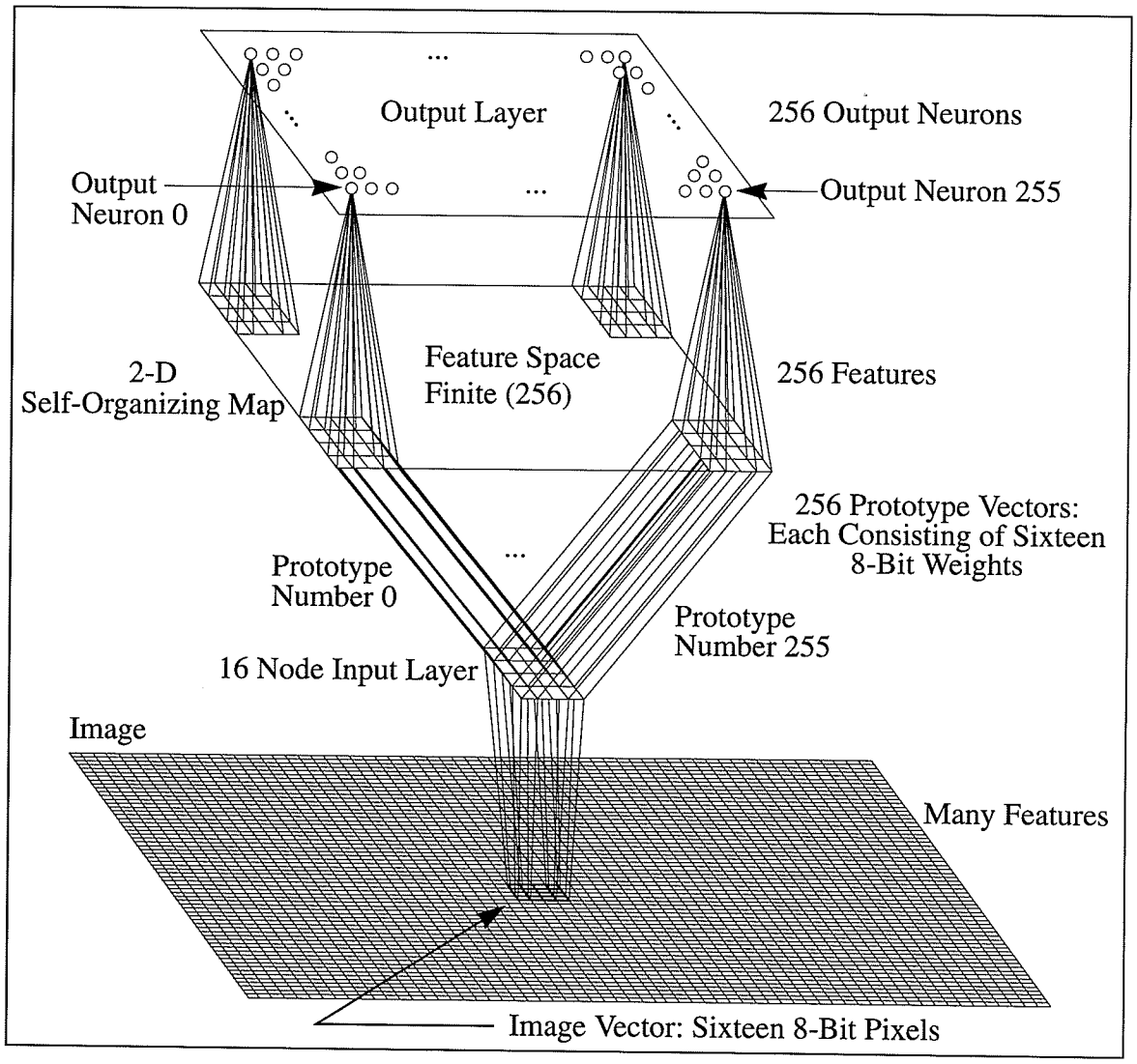

Fig. 20. 2-D structure of the competitive learning network for images.

The input layer consists of 16 neurons, each of which has 8-bit resolution. These are the nodes through which the 8-bit image pixels are input. The input layer is connected to each output neuron through a weight vector. Thus, each output neuron has 16 weights connected to it. Each output neuron has a topological address, ranging from 0 to 255. Each

weight vector will train to become a prototype for a particular feature that is to be extracted from the image training data. The weight vectors are connected to pixels in the feature space. The weight vectors and the feature vectors displaying them have the same dimension as the image vectors. Each of the 256 blocks in the feature layer is connected to an output neuron having an 8-bit address. There can be many images presented as training data, and, therefore, there can be many features in the input image space. There is a finite number (256 in this case) of features in the feature space. Therefore, the many features of the input are quantized to 256 features. These 256 features are learned.

The SOFM is trained using a sequence of image vectors. Each vector consists of sixteen 8-bit pixels extracted from a square block of the image, as shown in the bottom of Fig. 20. During each presentation of an input vector, each weight vector is compared with the input image vector, and one winner is chosen based on a similarity measure, i.e., the winner is the weight vector which most closely resembles the input vector. The winner weight vector along with its currently defined neighborhood become active. The winner's neighborhood is the set of weight vectors within the currently defined neighborhood. In this case the neighborhood would be a larger block encompassing the winner weight vector. The winner weight vector and the neighborhood weight vectors are updated by bringing their 8-bit values closer to the 8-bit value of the input vector. The winner weight vector is updated with a larger learning rate, and the neighborhood weight vectors are updated with a learning rate that is inversely proportional to their topological distance (i.e., their difference in addresses) from the winner weight vector.

Many such input vectors are presented, and a corresponding number of updates are made to the weights. Because there are far more input vectors than weight vectors, the weight vectors cannot satisfy all of the input vectors. Rather, the network is forced to

extract higher order dimensional features of the input image. For each feature that is extracted, for each set of input vectors that are associated with a feature, the network learns a weight vector, i.e., the prototype of that feature.

Notice that the feature extraction is done in an organized way, because neighborhoods of the 2-D map are similar. All input vectors the network deems similar, i.e., belonging to the same feature, will activate a certain localized portion of the map to a smooth degree of belongingness.

3.2.4 SOFM Algorithm

The following steps give a detailed description of the training procedure of the SOFM algorithm:

STEP 1: Initialize each weight in the weight matrix to the mean value of the pixels in the image plus some small random perturbation, *RP*.

STEP 2: Segment the image into 4x4 pixel square arrays and call each a vector of 16 pixels (elements). Randomly select one of the *mn/16* vectors and present it to the network. The number of training vectors *(NTV)* is a count of the number of training vectors that have been currently presented.

Alternatively, select a random number and use this as an index for forming a square vector from the input image. There are *mn* such vectors. Present this vector to the network. (This work implements the former method.)

STEP 3: Compute the distance between each weight vector and input vector, as follows:

$$d_2 = \sum_{i=1}^{16} (IV_i - w_{ij})^2, \quad \text{for i} = 1, \ldots, 256$$

where IV_i is the ith pixel in the input vector, and w_{ij} is the connection strength impinging the j^{th} output neuron and originating from the ith input pixel. w_j is the weight vector connecting the input vector, IV, to the jth output neuron.

STEP 4: Select as the Winner that output neuron whose weight vector w_j^* is closest to the input vector (IV).

$$w_j^* = \min_{1 \le j \le 256} \left\{ \sum_{i=1}^{16} (IV_i - w_{ij})^2 \right\}$$

STEP 5: Update the weight vector w_j^* connected to the Winner and the weight vectors connected to the neighborhood neurons of the Winner as follows:

$$w_{ij}(new) = w_{ij}(old) + \alpha(n, NTV) \bullet (IV_i(t) - w_{ij}(old))$$

The learning rate function $\alpha(n, NTV)$ depends on both the (neighborhood) physical displacement of a neuron (n) in the 2-D surface from the Winner and the number of training vectors (NTV) presented so far. The w_{ij} is the weight vector connected to neuron n.

STEP 6: Update the neighborhood function as shown in Fig. 21. This update should be applied to both horizontal and vertical directions with the winner as the central point. If NTV = end-of-training (EOT) then stop; otherwise goto STEP 2.

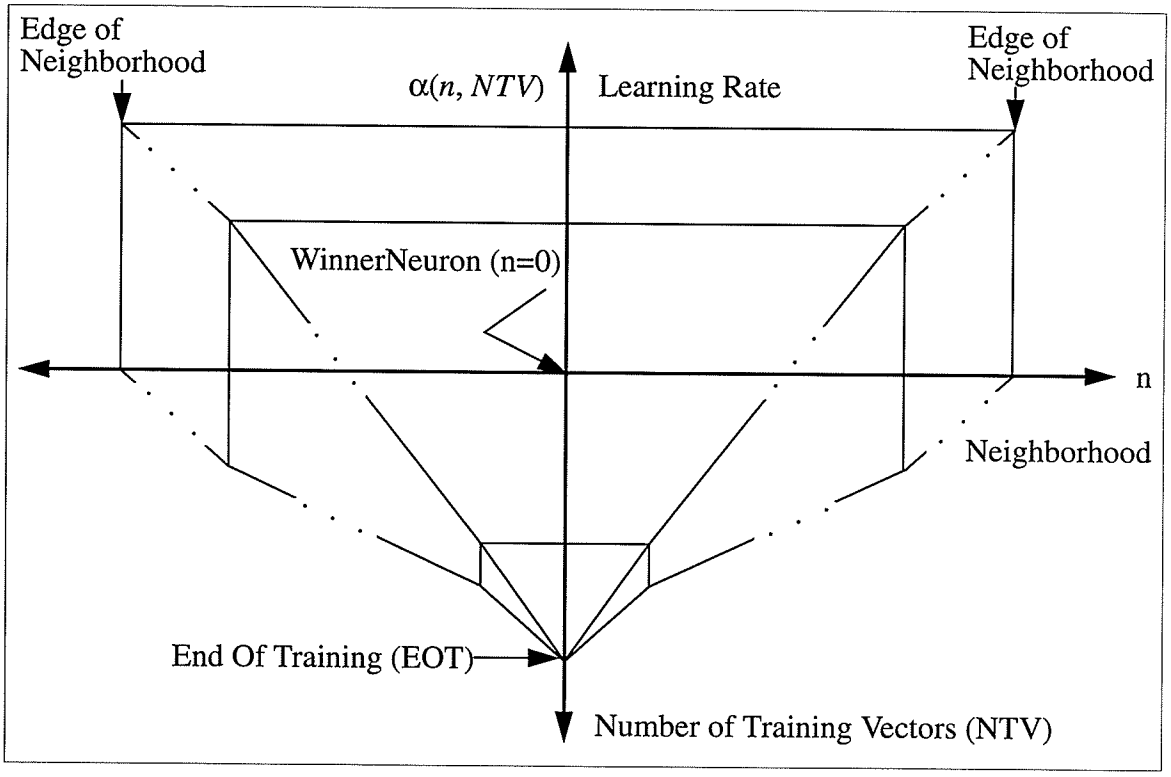

Fig. 21. Learning rate (α) as a function of neighborhood (n) and input vector (NTV).

3.2.5 Termination of Training

Notice that the above algorithm continues iteration until EOT, which is defined as a pre-specified number of presentations of image vectors. Kohonen has shown that the algorithm converges, but parameters governing the rate of convergence are not well understood. A way of determining the EOT could be to temporarily stop training at some point, and measure the total distortion. This would require performing the entire compression of the image and its decompression to measure the total distortion. If the total distortion is above a designed threshold, then training resumes, until at some other point when the calculation of the total distortion is below the threshold. But this would require prohibitively too much time. For example, a 512x512 images requires (512x512/16)x256 com-

pares for a codebook of 256 prototypes. On a SPARC Station 2, this requires about 5 minutes of computing time.

A better way of determining approximately the total distortion is to take advantage of the training cycle. In the training cycle, distortions are already being calculated in order to determine the winner. We can choose a window whose width is equal to the number of vectors in the image. In this window the approximate total distortion is given by the sum of the distortions of a sequence of training vectors. Note that this does not give the exact total distortion, because the training vectors are presented randomly and the weight vectors continue to change.

3.3. System Description

The HCL, SCL, FSCL, and SOFM competitive learning neural networks were implemented in software on a SPARC 2 workstation incorporating a graphical user interface (GUI). The programs for the learning, compressing, decompressing, and other utilities were written in the C language, and the GUI was written using the XVIEW interface. Figure 22 shows the main window of the application. The left hand side displays the original image used for extracting vectors for training, or for quantizing, while the right hand side shows the quantized image. The rectangular image in the middle of the window shows the pixel values of the prototypes. The two small boxes between the original image and the quantized image represent scaled versions of the currently extracted image vector from the original image, and the winner prototype extracted from the prototype rectangle, respectively.

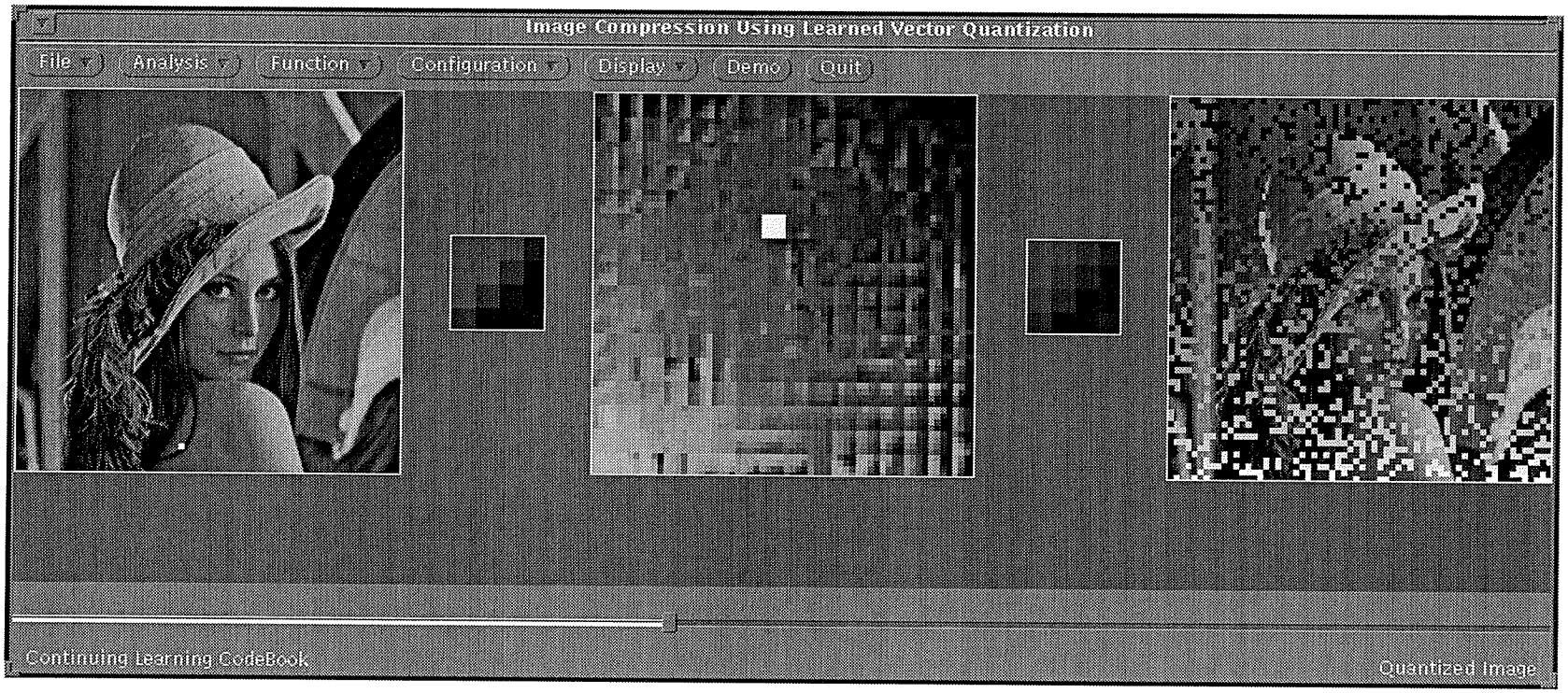

Fig. 22. Main window in the GUI for learning vector quantization of images. Training of the Lena image is in progress. The small white square on the lower shoulder of Lena (left image) represents the location of the randomly extracted vector. This vector is displayed in the 4x4 image on the right hand side of the Lena image. In the center image, the best matching unit is located in the codebook image. This best matching unit is displayed in the 4x4 image on the right hand side of the codebook image. This best matching unit is used to reconstruct the Lena image.

There are basically three modes of operation, learning, quantizing, and decompression. However, decompression is incorporated as the last stage of both learning and quantizing. So, in the application, there are only two modes, learning and quantizing. In the learning mode, a 4x4 pixel square image vector is randomly extracted from the original image, and a scaled version is displayed in the left small box. This vector is then compared against all of the prototypes in the codebook. The winner prototype is determined, and it, as well as its neighborhood, are updated according to the algorithm described above. A scaled version of the winner neuron is displayed in the right small box. The prototype vector then is placed into the quantized image square at the same relative position from which the original image vector was extracted. This procedure is continued until all of the number of training cycles (epochs) are completed. During training the progressive nature of the learning is graphically displayed.

Various network parameters may be selected through the *Configuration* dialog panel. For example, 1-D and 2-D topology, the learning rate, winner threshold, number of training cycles, initial neighborhood can all be specified at the start of or during training.

For educational purposes, the learning speed, the display frequency, and the quantization speed can be controlled through the *Display* control panel. For example, the self organization of the codebook can be visually observed at any speed, from single stepping to full speed. This shows the dynamic behaviour of the network. One can study the effects of different neighborhood reduction schemes, and of different learning rate schedules. The coarse tuning and fine tuning of the prototypes is obviously evident. Also, one can see the difference between the codebooks formed by FSCL and SOFM, the former being random, while the latter being a smooth transition of energy.

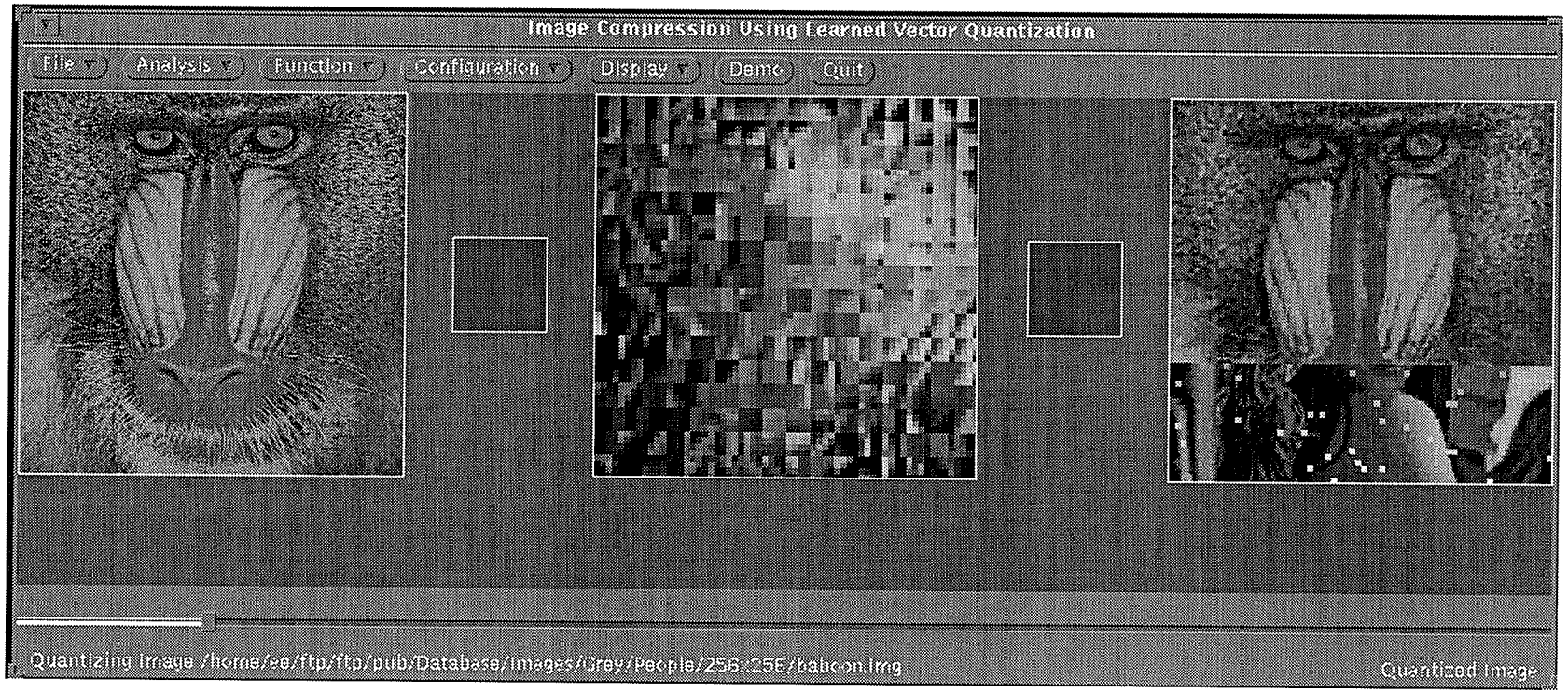

Fig. 23. Main window in the GUI for learning vector quantization of images. Quantization of the baboon image using Lena's codebook is in progress.

Once the network has learned on some training image, then other non-training images may be opened for quantization. For example, Fig. 23, shows the baboon being quantized with Lena's prototypes.

With some theory of competitive learning networks discussed, and an application in which they may be implemented built, the next section discusses the experiments.

3.4. Experiments

The experiments part of this section of the thesis is intended to empirically substantiate and provide motivation for the ideas and methods described in the algorithmic description of the LVQ technique. The results are judged based on both objective and subjective criteria. The first experiment is aimed to establish validity of the procedure. The next experiment establishes a base level quantizer from which other quantizers can be judged. This is followed by discrimination and generalization experiments.

3.4.1 Memorization

If the size of the codebook of prototype vectors is exactly equal to the size of the original image, then it is reasonable to expect any compression algorithm to converge and be capable of reproducing the image exactly, i.e., reproduce the image exactly with no distortion. To show this, a 64x64x8 image (i.e., 64x64 pixel image, each pixel containing one of 256 grey levels) is learned by the codebook with a pixel size of 64x64x8. Each of the $64^2/16$ input image vectors consists of 16 pixels. Each vector is extracted from the image as a square array of 4x4 pixels. Each vector is presented once to the network; i.e., so that

there are exactly $64^2/16$ presentations. The neighborhood is set to one. This type of learning is called hard competitive learning.

Since each winner weight vector is updated with a learning rate of 1.0, it exactly equals the input vector after being updated. The 8-bit address of this weight vector is transmitted. There are $64^2/16$ such addresses. Therefore, the resulting compression ratio is $64^2/16:64^2/16$ or 1:1, and the compression rate is 8/16 = 0.5. The PSNR is infinite, i.e., the mean squared error is zero. The codebook has memorized the original image.

3.4.2 Linear Vector Quantization

When the size of the codebook is smaller than the original image, it is reasonable to assume that distortion will be produced. The smaller the codebook the greater the amount of distortion. This is because the mapping from the original image to the codebook is many-to-one, and the many variable becomes increasingly larger for smaller codebooks. Hence, a main objective of any compression scheme is to minimize the distortion for a given compression ratio. To this end, it is useful to introduce a compression algorithm (i.e., a mapping) that utilizes the most straight forward, simplest, and fastest compression. While this primitive compression scheme does not produce very good results, it represents a starting point from which other techniques can be judged.

Perhaps one of the simplest many-to-one compression algorithms is the linear vector quantizer. The linear vector quantizer maps much like the linear scalar quantizer. In the linear scalar quantizer, each continuous analog sample ranging in value from $-V$ to $+V$ is quantized to a discrete digital number. We may say that the real signal is rounded off to the nearest integer determined by the resolution. In linear vector quantization, an entire vector

consisting of generally different valued components is rounded off to a nearest vector consisting of equal valued components.

In particular, a codebook of size 256x8 is generated by assigning a number between 0 and 255 to the 256 vectors. Each vector consists of 16 pixels, each of which take on the grey level assigned to the vector, as shown in Table 1. This codebook is used to compress a 512x512x8 image of Lena (Fig. 36). Each extracted vector of 16 pixels from the original image is compared in the Euclidean distance sense to all vectors of the codebook. The index of the codebook vector most similar to the input vector is saved to a file. The compressed image is decompressed by a simple look up procedure, and the result is shown in Fig. 37.

The resulting image achieves a PSNR of 25.9db. Observing both images, it is noted that the smallest unit in the original image of Lena is a pixel (i.e., one 8-bit grey level), while the smallest building block of the decompressed image is a 4x4 pixel square. All smooth and uniform areas of the original image are mapped quite well in the decompressed image, while sharp lines and edges are approximated by a staircase function (coarse). This leads to a reasonable suggestion that this method would perhaps be best suited for large images that do not have too many sharp lines or edges. This does not include a wide variety of images. However, while this method has its definite limitations, its usefulness is the establishment of a starting point from which Kohonen's self organizing feature map of learning the codebook for vector quantization may be judged.

Table 1: Linear vector quantizer.

Vector	Values of the 16 pixels
0	(0, 0, 0, ... , 0)
1	(1, 1, 1, ... , 1)
. . .	
255	(255, 255, 255, ... , 255)

3.4.3 Codebook Learning Fine Tuning

From the starting point established above (i.e., PSNR = 25.9db) several experiments applying Kohonen's self organizing map to learn the codebook were performed. These experiments start from the simplest description of the procedure as described by Kohonen [Koho90] and continue by adding complexity in the form of modifying system parameters with the objective of improving the PSNR. The goal of this part of the experiments is to change certain system parameters and study how these changes affect the performance. This thesis examines three such system parameters, the learning rate, neighborhood, and number of training cycles (epochs).

3.4.3.1 Learning Rate

In this thesis the learning rate function of Kohonen's SOFM depends on two variables, namely, the number of vectors presented, k, and the neighborhood, n. The shape of the learning rate as a function of the neighborhood, n, with k held constant, represents short time learning or a nested learning rate. For each vector presented, each weight vector that is

to be updated is done so with this nested learning rate function. The most basic way of doing this is as previously shown in Fig. 21, i.e., by updating all weight vectors connected to the neurons in the neighborhood by an equal amount. Alternatively, it seems reasonable and intuitive to assume that each weight vector associated with the neurons in the currently defined neighborhood should be updated in such a way as to reflect their distance from the winner neuron. In other words, the weight vector connected to the winner neuron should be updated with the maximum learning rate, while the weight vectors connected to neurons that are increasingly farther away from the winner neuron should be updated by proportionally decreasing amounts. The motivation for this idea is in part biological, i.e., in biological neural signals, when a central neuron is active, the weights connected to the neighboring neurons are affected by a characteristic function known as a Mexican hat. This decreasing learning rate as a function of the distance from the winner neuron can be modelled in different ways. In this work, three nested learning rate schedules are compared.

As already mentioned the first learning rate function implemented is the nested step function. In this scheme, after each input vector presentation, the same learning rate is applied to the winner and to all of the neurons in the currently defined neighborhood.

The second variation to the nested learning rate function is the linear decrease. In this scheme, the farther away a neighborhood neuron is located from the winner neuron the lower is its learning rate. In particular, the learning rate of a neighborhood neuron varies as a linear decrease of topological distance from the winner neuron.

Finally, the third variation to the nested learning rate is a Mexican hat (approximated) decrease from the winner neuron. This function is somewhat of a compromise between the step and the linear functions. For example, the learning rate decreases slowly near the winner neuron, very much more so than the linear case. However, near the edge of

the neighborhood, the learning rate decreases very rapidly, very much more so than the linear case.

The above three learning rate functions are implemented in learning the codebook for a 512x512x8 image of Lena, as shown in Fig. 36. The results are tabulated in Table 2. Note the consistent improvement of PSNR down each column (i.e., with increasing number of training vectors) and across each row from left to right (i.e., with the type of learning, namely, step, linear, and Mexican hat).

Table 2: Nested learning rate optimization. Variables: Initial training vectors (ITV), total training vectors (TTV), execution time (XTime). Training consists of 1 epoch and a linearly decreasing neighborhood.

#ITV	#TTV	XTime (min)	Step (dB)	Linear (dB)	Hat (dB)
2048	2048	1/8	25.96	26.27	26.54
4096	4096	1/4	26.04	26.62	26.91
8192	8192	1/2	26.49	27.21	27.94
16384	16384	1	27.14	27.61	27.71
32768	32768	2	27.44	28.10	28.21

3.4.3.2 Neighborhood

In the previous description of the learning rate functions, the neighborhood function decreases linearly with the number of vectors presented. During the early stages of learning, when the number of vectors presented is less than the total number of vectors to be presented and the learning rate is large, the neighborhood should be large, since this is when each of the 256 weight vectors establishes a rough location in the feature map. Later on in training, when the learning rate is small, the neighborhood should decrease, because

this is when fine tuning of the coarse map occurs. Since the fine tuning period is longer than the coarse formation, the neighborhood should decrease faster during fine tuning. Rather than a linear decrease, others, such as quadratic, cubic, and exponential decreases may be evaluated.

Table 3 shows the effect of using a cubic decrease in the neighborhood. The results should be compared with Table 2, since the change from linear to cubic neighborhood decrease is the only difference between the two experiments. Note the consistent improvement in the PSNR, point for point as compared with the results shown in Table 2

Table 3: Nested learning rate optimization. Variables: Initial training vectors (ITV), total training vectors (TTV), execution time (XTime). Training consists of 1 epoch and a cubicly decreasing neighborhood.

# ITV	# TTV	XTime (min)	Step (dB)	Linear (dB)	Hat (dB)
2048	2048	1/8	27.50	27.80	27.89
4096	4096	1/4	28.10	28.24	28.35
8192	8192	1/2	28.41	28.59	28.68
16384	16384	1	28.77	28.83	28.95
32768	32768	2	28.80	28.92	29.05

3.4.3.3 Epochs

Another method can be used to increase the convergence rate of learning the codebook and to improve the PSNR. In the previous experiments, the VQ algorithm is implemented once, i.e., learning is complete when the predefined number of vectors have been presented to the network. The predefined number of vectors constitutes a training cycle or epoch. Note that learning begins with the initial learning rate, final learning, and neighbor-

hood parameters set as follows: *initlr* = 1.0, *finlr* = 0.5, *initnei* = 128, and *finnei* = 0, respectively. More than one epoch can be applied with the learning rate and neighborhood parameters scaled down at the beginning of each subsequent epoch. In these experiments, *initlr* and *finlr* are scaled down by a factor of 1.25 and *initnei* and *finnei* are scaled down by a factor of 2.0. Recall that at the start of learning, the network establishes a coarse map, and near the end of a training cycle, the network fine tunes the feature map. By introducing another training cycle, this forces the network to perform coarse map formation after the fine tuning process of the previous epoch, and this is followed by another fine tuning, both done with scaled down parameters. For example, Fig. 24 shows a learning rate function over 7 epochs. The *initlr* at the beginning of the 6th epoch is 0.13 or 1/(6x1.25).

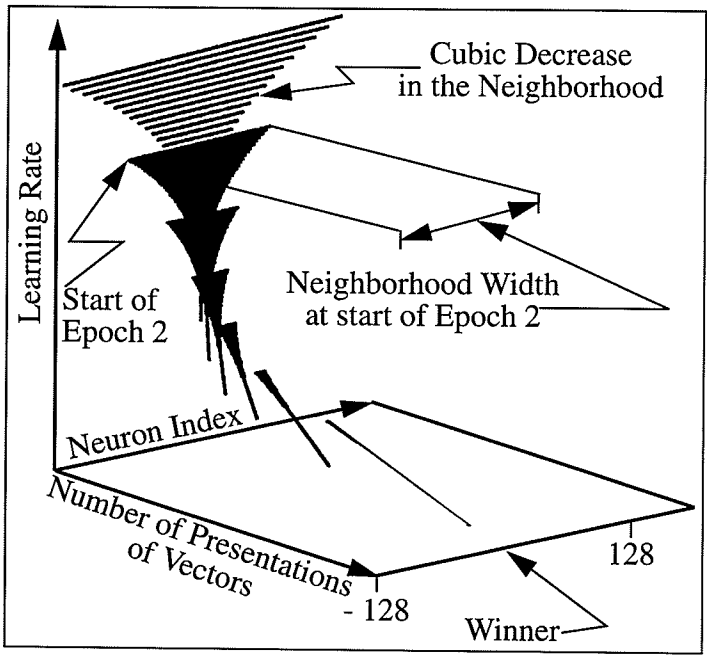

Fig. 24. Seven epoch learning. (7 epoch; 4064 TTVs; N^3; 28.82db.)

Note that at the start of each epoch the error increases, since coarse tuning is done after fine tuning. This is shown in Fig. 25, which shows a sliding error window over every

NIV vector presentations. Increases in error can be seen at the beginning of each epoch, when the number of vectors presented is 1024, 3072, 7168, 15360, 31744, and 64512.

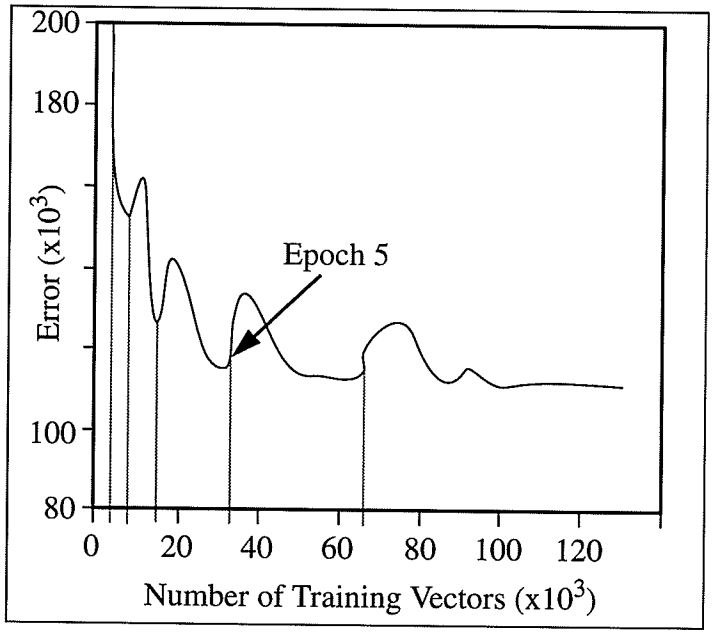

Fig. 25. Error window. (7 epoch; 32512 TTVs; N^3; MexHat; 29.74db.

Learning over multiple epochs, forces the network to reach a lower local minimum faster. Consider observing sugar in a sugar bowl. We see sugar particles at rest in some position that is not the lowest possible energy state. They have attained this position as a result of a compromise between energy and entropy of the system. Entropy may be characterized as a state of a system where a large population of elements are located, even if this location does not represent the lowest possible energy. This location represents one of the many local minima of the system. When we tap the sugar bowl, we inject energy into the system, and this allows the sugar particles to escape the current local minimum and reach another minimum which is lower than the previous. Much in the same way, the minimum achieved by the VQ network after completion of a training cycle may be a local minimum. In order to move the system out of this local minimum, one possible method is to force it to perform coarse tuning of the feature map again, by suddenly increasing the

neighborhood and learning rate. This injects energy into the system, and gives the system momentum to escape the current local minimum. Once escaped, the network performs fine tuning of the map once again and another local minimum is reached. This repeats for each subsequent epoch.

Table 4 shows the results after training the codebook using 7 epochs. This table should be compared with Table 2, since the number of epochs is the only difference between the two experiments. Note the consistent improvement in the PSNR as compared with Table 2. However, note also in Table 3 that the type of nested learning rate function is PSNR insignificant (read across each row) as the number of epochs increases. In fact, the linear learning rate function yields the best results for epoch number 7. The reason for this is that all three functions tend to approximate impulse functions centred at the winner neuron as the number of epochs increases and the initial neighborhood and initial learning rate diminish. While these functions start out with different character, they end up looking the same. Thus, different learning rate functions give different convergence rates for small number of epochs, but these differences vanish for large number of epochs.

Table 4: Neighborhood function optimization. Variables: Initial training vectors (ITV), total training vectors (TTV), execution time (Time).

# ITV	# TTV	XTime (min)	Step (dB)	Linear (dB)	Hat (dB)
16	2032	1/8	27.50	27.61	28.99
32	4064	1/4	28.28	28.51	28.48
64	8128	1/2	28.41	29.91	28.99
128	16256	1	29.36	29.65	29.51
256	32512	2	29.75	29.84	29.74
512	65024	4	29.96	30.10	29.93
1024	130048	8	30.11	30.16	30.13
2048	260096	16	30.17	30.23	30.18

The 7-epoch version can also be compared with the single epoch version by observing Fig. 26, which shows the error window for both systems under similar conditions. Note that the coarse formation of the map is done much faster (greater slope) and reaches a lower error in the 7 epoch version. As a result, fine tuning begins at a lower error than in the single epoch version and, therefore, the system reaches an overall lower local minimum.

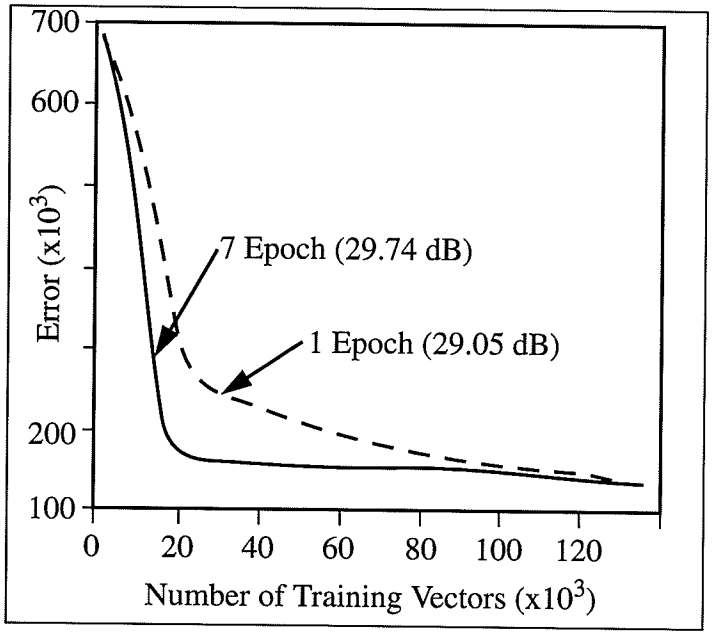

Fig. 26. 7-versus 1-epoch error window. (32512 TTVs; N^3; MexHat; 29.74db.)

3.4.4 Codebook Analysis

Having optimized learning of the codebook, a next step is to analyze how the codebook maps the features of the image. Figure 27 displays an image of the codebook along with the energy distribution.

As mentioned earlier, the output neurons can be organized onto a one-dimensional

topology (ring). The image in Fig. 27 shows the brightness distribution of the weight vectors connected to the output neurons in the ring. There are 256 weight vectors each consisting of a 4x4 square array of pixels. Note that the two ends of the image should be connected to each other in order to form the circular ring. Note also that the envelope of the energy is a smooth function of the weight vector index, i.e., the location of the output neuron in the ring. Weight vectors with similar energy, i.e., similar mean brightness levels, are clustered in sets whose boundaries are fuzzy and continuous. The weight vectors are mapped in this way onto the ring because of the Euclidean metric used in determining the winner neuron and because of the chosen one-dimension topology (ring) of the output. Other methods of organizing the features onto a map are possible, and the visual display of a such a codebook would be different than the one displayed in Fig. 27. For example, we could apply the Euclidean distance metric in determining the winner, and we could use the a two-dimensional grid to define the organization of the feature map. In this case, we would expect the mean brightness levels to be mapped in a smooth two-dimensional function.

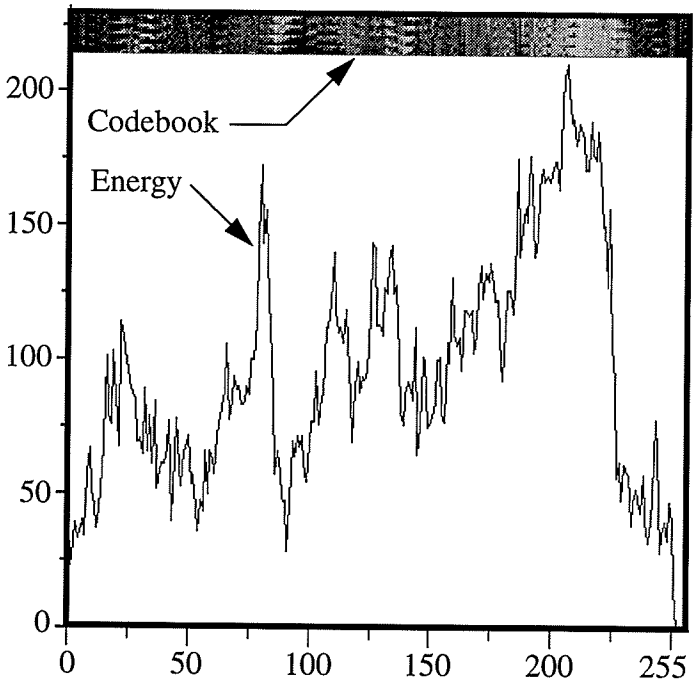

Fig. 27. Grey scale representation of the codebook (top), and energy distribution (bottom).

3.4.5 Generalization

Having optimized learning of the codebook for a specific image (i.e., Lena) and studied how the codebook maps features, the next question that should be asked is: How well does the codebook perform in quantizing other similar images? This question relates to the concept of generalization in artificial neural networks. The initial intention in learning a codebook for vector quantization is that the codebook vectors learn to represent higher order statistics of the original image. Our assumption is that these higher order statistics are not only characteristic of the original image, but they also characterize other similar images. The term similar is a fuzzy concept. However, with respect to this work, similar images mean a collection of head and shoulder images of people.

There are two types of generalization in learned vector quantization, autoassociative and heteroassociative. In autoassociative generalization, there is a great degree of resemblance between the test data and the training data. For example, in the application of interest, the test data consists of very similar head and shoulder pictures of people. In heteroassociative generalization, the resemblance may be very weak, or none.

3.4.5.1 Autoassociative Vector Quantization

The measurement of the degree of generalization is not straight forward since generalization is also a fuzzy concept. However, it can be judged by both a subjective and objective evaluation. We tend to characterize a codebook's ability to quantize images that it has not been trained on by visually comparing the original image with the decompressed image while noting the PSNR of the decompressed image. There are other factors that can be taken into consideration. For example, we can determine the significance of the correlation of the histograms between the trained image and the test images.

Figure 28 shows the concept of learning higher order features from a representative database of images. The neural network is presented with vectors from a large data base of similar type head and shoulder images of people. As such there are very many features in the data base. However, the neural network is forced to extract 256 of the most important features that are in common with all of the images in the training data base. After training on this data base, the neural network is presented with images it has not seen during training. If the new image is similar to the type of images in the data base, then the neural network generalizes well on the new image because our assumption is that the network knows the most important features from the representative data base.

3.4.5.1.1 The Competitive Learning Neural Network Painter

One can think of the neural network in Fig. 28 as a painter of head and shoulder images of people. The images at the top of the figure are its training images. The box in the centre is its palette. The painter learns how to paint the images by training over and over on a representative database. During training, the painter determines an optimal and finite (256) palette of 4x4 blocks of pixels of which to use to paint (reconstruct) the images. If the painter had an unlimited palette, then all of the shapes in all of the images from the data base could be reconstructed perfectly. But, the painter is forced to learn to paint the images in the database with a finite palette, consisting of 256 4x4 blocks of pixels. The important point is the driving force that tells the painter what 4x4 blocks are the important ones. This driving force should be similar to the one we use when we perform the reconstruction of images of people in our minds.

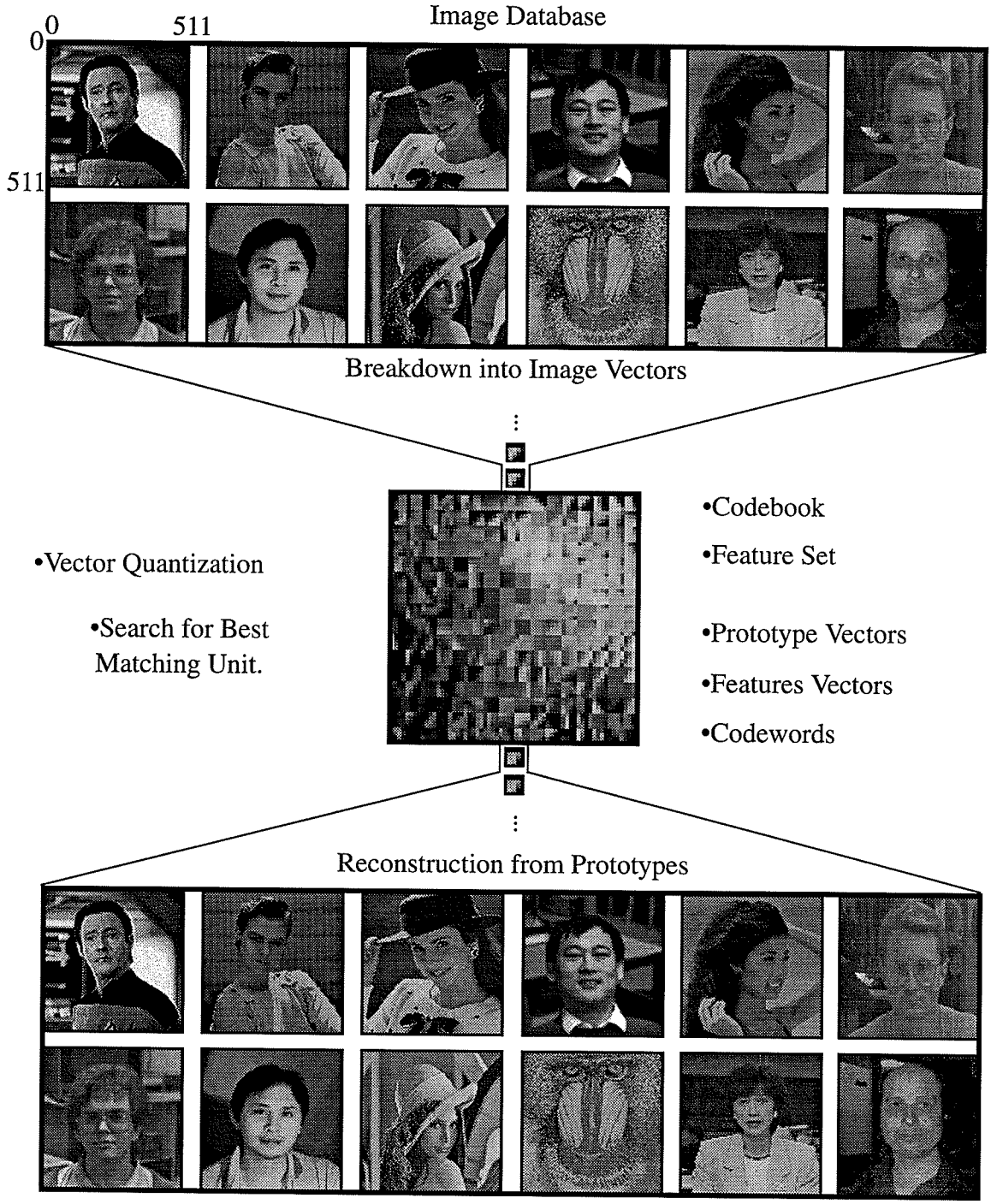

Fig. 28. Conceptual block diagram showing learned vector quantization.

To demonstrate the effectiveness of the artificial neural network painter, a worst case situation is chosen. Instead of learning on a large data base, the neural network is

trained on one representative image. The artificial painter is then tested on other similar type images it has not seen. Figures 39-49 show originals and quantized versions using the 30.35db codebook trained using the 512x512x8 Lena original (Fig. 36). Overall, Lena's codebook performed well on these images, but there are some interesting characteristics that can be pointed out.

Lori (Fig. 39, Fig. 40) achieves a PSNR of 31.6db, which is slightly higher than that achieved by the training image (Lena: 30.35db). We would not expect this unless the test image is a very close subset of the training image and contains more of the same type of redundancy that is contained in the original image. Indeed, in the image of Lori, we see that there exists a greater amount of the same type of redundancy as in the image of Lena. For example, Lori's hat is mostly dark black. This dark black shape can be constructed exactly using a prototype vector extracted from the curved frame of the mirror in Lena's image. Similarly, the background in Lori's image appears to consist of one uniform grey level, whose prototype can be extracted from Lena's image. Also note, that Lori's earring decompresses quite well considering that Lena contains no similar macro object. Finally, note that the nail polish on Lori's finger nails has lost its original grey level.

3.4.5.2 Heteroassociative Vector Quantization

Lena's 30.35db codebook can be tested on different type images. For example, using Lena's 30.35db codebook, the Renoit painting (Fig. 45, Fig. 46) achieved a PSNR of 24.7db. While the image is recognizable to some extent, there are many details lost. The detailed design on the child's dress (neck, breast, and sleeve) has been lost; only faint remnants of the articulate design remain. The artist must have intended to rely on our percep-

tion of such shapes in order to communicate the image to us. As can be seen, such perception influenced images do not compress well using a compression scheme that does not take human perception into consideration. Finally, notice that the white boundary of the painting could not be reconstructed. Examining the original image, Lena, we cannot find a 4x4 pixel square, all of whose pixels have a 255 grey level.

3.4.5.3 Beauty and the Beast

In studying image compression algorithms, it is interesting to determine the limit of generalization. Figure 48 shows the decompressed image of the Baboon image, quantized using Lena's 31.74db codebook. Lena's 31.74db codebook is the codebook learned from the Lena 512x512x8 image.

Figure 49 shows the decompressed image of the Lena image, quantized using the Baboon's 26db codebook. The Baboon's 26db codebook is the codebook learned from the Baboon 512x512x8 image. Note that the Baboon's reconstruction of Lena produced about the same PSNR as that achieved by the linear vector quantizer, our starting point.

The reader should judge the resulting quality in these decompressed images, and, furthermore, determine the implications of such intermixing of prototypes.

3.4.6 Histogram Indicator in Generalization Performance

The degree of generalization of images can be ascertained through an objective and subjective examination, as done above. However, an interesting question could be asked: Given a codebook trained on a certain image, can the suitability of quantizing other

images be determined through a correlation between the histograms of the original image and the image to be compressed?

In general, the answer is no. To show this, a new image can be formed by randomizing the pixels of the 512x512x8 image of Lena. This new image still consists of exactly the same grey level distribution of Lena. When quantized, the decompressed image achieves a PSNR of only 17db. This shows that the VQ method extracts the features that are characteristic of the training image, and the codebook does not learn grey level distributions.

However, the histogram can be used to a limited extent in predicting the performance of a codebook in quantizing an image, provided that it is known a priori that the trained image and the test image are similar to some extent, as with the images experimented with above. In the images discussed above, with some exceptions, the histogram of each test images has a distribution that is a subset of the histogram of Lena, the training image. In other words, with some exceptions, all grey level and all frequencies of grey levels are contained in the histogram of Lena. If we are given an image that is subjectively similar to the training image, and if the histogram of the test image is a subset of the histogram of the training image, then we can say with a certain high probability that the quantization of the test image will produce results comparable with the PSNR achieved by the training image.

3.5. Summary

In this section of the thesis, several competitive learning algorithms (HCL, SCL, SOFM, and FSCL) have been used to compress several grey level images by a compression ratio of 16:1 at a compression rate of 0.5, while maintaining a PSNR of about 30dB. C programs were written to implement learning, compressing, decompressing, analyzing error, and others for the VQ method. These programs were run on the SUN SPARC Station 2. Methods for optimizing learning have been presented. Given an image that is subjectively similar to the training image, and if the histogram of the test image is a subset of the histogram of the training image, then quantization of the test image will produce results comparable with the PSNR achieved by the training image.

CHAPTER 4

A Multifractal Entropy Measure of Signals

4.1. Introduction

This chapter presents a *multifractal measure* of signals. A multifractal function is used to measure the distribution of generalized information in signals. In this thesis, the multifractal measure extracts features of the signal in terms of the signal's generalized information content. The extracted features are mapped into generalized information subsets. Each subset is a collection of points of the signal that has a distinct information-fractal dimension. Thus, features of signals are described based on their multifractal measure. Multifractal measures not only provide an objective description of the information in signals, but also, the measures they yield are perceptually meaningful. Applied to signal compression, the generalized information fractal dimensions of a signal indicate the amount of relevant, irrelevant, or redundant information in the signal.

The measure used in this chapter is a multifractal based on Rényi's generalized entropy [Rény55] and Grassberger and Procaccia's generalized correlation dimension [GrPr83], as described in [Kins94b] and [Kins94c]. However, unique to this thesis, is that the derivation of the generalized dimensions uses the information measure at the core of the multifractal measure specifically for extracting generalized information based features for signal compression.

4.2. Fractal Nature of Signals

Natural signals are perceived like fractals [Pent84]. There are many naturally occurring signals whose nature is perceptually similar to the synthesized signals based on fractal theory. A fractal model of signals captures the perceptually significant features of natural signals. Simple synthesized signals consist of predictable objects, such as spheres, cylinders, and cubes. Complex signals are formed from rough, jagged, and meandering surfaces. Integer models of simple signals do not capture the complexity of natural signals.

In order to better describe signals, a fractal model is used. A fractal model not only provides an objective measure of naturally occurring signals, but the literature [Man83] and [Pent84] has shown that the measures derived from a fractal model are perceptually meaningful. One such measure is the fractal dimension.

4.3. Background Theory of Fractal Dimension and Multifractal Measures

4.3.1 Fractal Dimension

Given the perceptually agreeing quality of the fractal model of signals, the next step is to apply fractal based tools to measure features of signals and/or to compare different signals. The basic tool under the fractal model is the fractal dimension, D. The fractal dimension D provides an objective measure, which is reflective of the *perceptual* degree of smoothness, uniformity, meandering, coarseness, roughness, and irregularity in a signal. There is a substantial correlation between the fractal dimension measure of a feature of a signal and the corresponding perceived "feeling" [Pent84].

Considering the notion of *dimension*, one thinks of several different aspects, depending on the application. Topological dimension is the usual idea of dimension; a point has topological dimension zero, a line (or curve) has topological dimension one, a surface has topological dimension two, etc.

A similar idea of dimension is the embedding dimension. The embedding dimension is the smallest number of Cartesian coordinates required to place the object in a Cartesian space. For example, a point requires one coordinate, a line or curve requires two coordinates, a surface requires three coordinates, etc.

However, the traditional notion of dimension fails to accurately describe complexity of many naturally occurring objects. For instance, there exists different objects that traditionally have held the same topological (or embedding) dimension, but are known to exhibit different levels of complexity. For example, while the Koch curve has topological dimension of one, it is much more complex than a standard curve, such as a parabola, which also has topological dimension of one. Another example showing contradiction in the traditional notion of dimension is the *space filling* curves.

To better characterize complex objects, the concept of dimension was generalized to not only include integers, but also to include fractions, by several key mathematicians, including Brouwer, Hausdorff, Besicovitch, Kolmogorov, and Mandelbrot. Mandelbrot defines fractal dimension as "a set for which the Hausdorff-Besicovitch dimension strictly exceeds the topological dimension" [Mand83]. Based on these earlier works, Kinsner [Kins94b] has compiled, unified, and introduced several different definitions of fractal dimension. A common thread linking these definitions is a power law, together with a measure, a limit procedure, and a critical exponent.

4.3.1.1 Basic Definition of Fractal Dimension

Let the object of interest be defined as a set of points F embedded in a space of dimension E.

Definition 1. Fractal dimension D characterizes the rate at which a measure $M(r)$ of an object F changes as the scale at which the measure is taken decreases to zero.

The basic fractal dimension, D, can be written as,

$$D = \lim_{r \to 0} \frac{\log (M_r)}{\log (1/r)} \tag{8}$$

Equivalently, Eq. 8 can be written as,

$$M_r \underset{r \to 0}{\sim} (1/r)^D \tag{9}$$

It is seen that the *critical* exponent D prevents the product of $M_r \bullet r^D$ from diverging to infinity or from becoming zero.

Figure 29 shows the basic idea of calculating fractal dimension. In the process a sequence of calculations is performed. Each step in the sequence is assigned a successively smaller value of r. At a particular step, the fractal is segmented into volume elements (*vels*), whose size is related to the scale r. Within each *vel*, a measure is applied. All such measurements within the vels are accumulated, and a sum $M(r)$ is formed. This is repeated for each step in the sequence or for each different value of r. The fractal dimension is the rate at which the measure $M(r)$ changes as r decreases to zero.

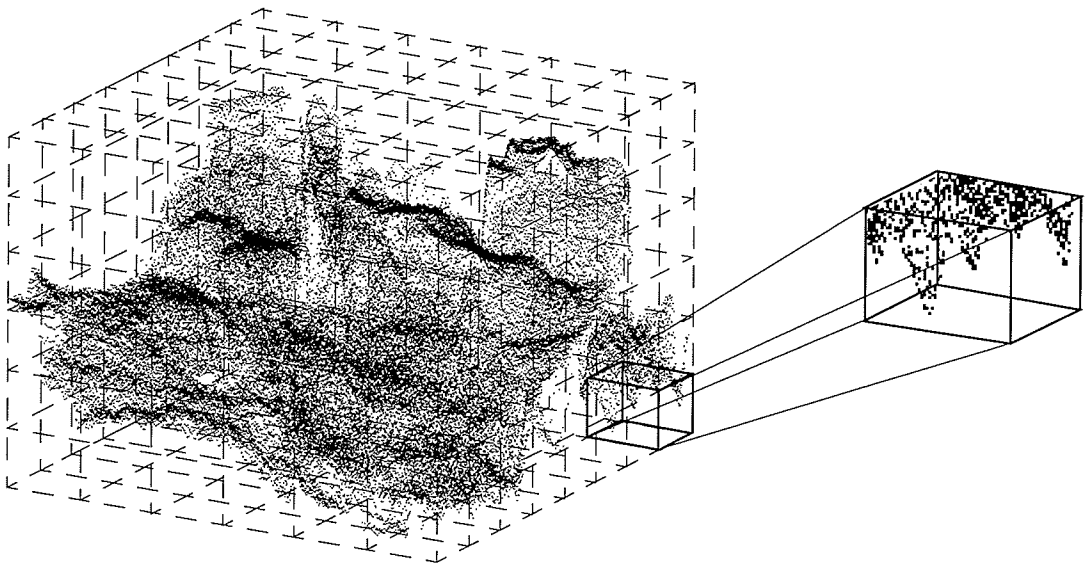

Fig. 29. Illustration of the basic method of calculating fractal dimension of practical fractals.

From this basic definition, we can form several different measure-motivated classes of fractal dimension [Kins94b], as shown in Fig. 30.

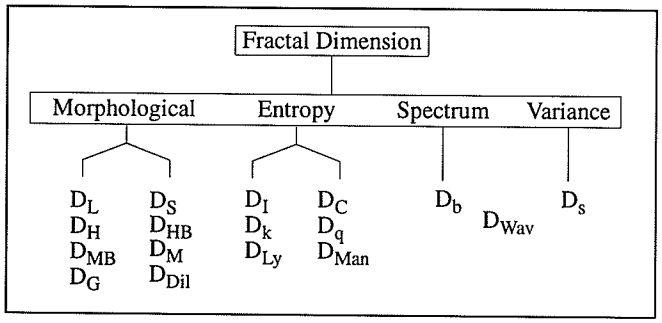

Fig. 30. The different fractal dimensions, including length dimension (D_L), self-similarity dimension (D_S), Hausdorff dimension (D_H), Hausdorff-Besicovitch dimension (D_{HB}), Minkowski dimension (D_{MB}), mass dimension (D_M), gyration dimension (D_G), dilation dimension (D_{Dil}), information dimension (D_I), correlation dimension (D_C), k-th nearest neighbor dimension (D_k), Rényi dimension (D_q), Luyapunov dimension (D_{Ly}), Mandelbrot dimension (D_{Man}), spectral dimension (D_b), variance dimension (D_s), and, wavelet dimension (D_{wav}).

4.3.1.2 Morphological Dimension

The morphological dimension uses a geometrical covering set as the basic measure. For example, in Fig. 29, the Hausdorff-Besicovitch D_{HB} dimension calculates the minimum number of *vels* required to cover the fractal at a scale *r*.

4.3.1.2.1 Problem with Morphological Dimension

Note that the geometrical-covering-set type of measurement neglects any non-uniform property of the signal, such as the very dense regions near the center of Fig. 29. At a certain scale *r*, the morphological dimension does not place any more/less importance on areas of the signal having different probabilities. The morphological dimension is based on geometrical measurements. They do not take into consideration any higher-order measures, such as spacial, temporal, and scalar variance. Consequently, morphological dimensions cannot accurately describe non-stationary signals. While there may exist some geometrically significant points of the fractal, these same points may be perceptually irrelevant. For example, out-lying, localized, and very dense noise in a signal would demand the same number of geometrical covering subsets as would other sparsely populated points of equivalent areas.

In signal compression, the objective is to extract information based features from signals. The information measure is more suited for signal compression because the information measure does not generally assume a non-uniform probability distribution. Thus, the information measure is more appropriate for the core measure in the fractal dimension.

4.3.1.3 Information Fractal Dimension

Motivated by the information extracting fractal dimension, Shannon's first order entropy (Eq. 10) is used.

$$H_r = -\sum_{j=1}^{N_r} p_j (\log p_j) \tag{10}$$

Assume the following power law,

$$C^{H_r}_{r \to 0} \sim (1/r)^{D_I} \tag{11}$$

(Note the similarity and difference between Eq. 9 and Eq. 11.) This yields the information fractal dimension D_I,

$$D_I = \lim_{r \to 0} \frac{H_r}{\log 1/r} = \lim_{r \to 0} \frac{-\sum_{j=1}^{N_r} p_j \log p_j}{\log 1/r} \tag{12}$$

To calculate D_I, we apply the measure to each of the N_r *vels* at scale r and accumulate a sum H_r. This associates an H_r with an r. D_I is calculated in the limit as r→0.

4.3.1.3.1 Problem with Single Valued Dimensions

A problem with single-valued fractal dimensions is that they fail to accurately describe those types of signals that exhibit multiple fractal dimensions. The single valued dimensions yield one number, which, in the case of multifractal objects, is a global or average dimension of the object. This may conceal very important features from detection, and prevent their extraction.

4.3.1.4 Generalized Information Measure

Obtaining objective measures of the amount of relevant information in signals is the objective of this section of the thesis. Shannon's entropy was considered to be the only feasible basis for describing the amount of information in a random variable or random process, whether this information was irrelevant, redundant, or relevant. However, Rényi realized that Shannon's basic equation was inadequate for describing the variable spatial, temporal, or scale complexity of dynamical systems. Human based signals, such as speech, ecg, eeg, music, images, and moving images are not simple random variables or random processes whose information content can be described by the first-order Shannon entropy. These signals have variable scale, temporal, and spatial complexity whose information content requires a generalized version of Shannon's entropy. Rényi [Rény55] suggested entropy of order q:

$$H_q(X) = \frac{1}{1-q} \log \sum_{x \in X} p^q(x) \tag{13}$$

Rényi's entropy extracts different probability subsets, and for each subset assigns an entropy. For $q \to \infty$, Eq. 13 suppresses all of the small probabilities and sifts out the maximum probability. And for $q \to -\infty$, Eq. 13 suppresses all of the large probabilities and sifts out the minimum probability. In the interstitial space (moments of q), for each successively positive q, the H_q has the effect of extracting fuzzy subsets representing points of the object with successively higher probability. For each successively negative q, the H_q has the effect of extracting fuzzy subsets representing points of the object with successively lower probability. In this way, the object is partitioned into different information bearing subsets, thus providing a generalized order and objective information measure.

4.3.1.4.1 Generalized Information Fractal Dimensions

Motivated by the generalized information measure, we substitute Eq. 13 into Eq. 12 and rearrange to obtain the generalized information fractal dimensions,

$$D_q = \lim_{r \to 0} \frac{1}{q-1} \frac{\log \sum_{j=1}^{N} p_j^q}{\log (r)} \tag{14}$$

Notice that for $q = 1$ (using l'Hospital's rule), H_1 is equivalent to the information dimension, or the Shannon entropy [Schr91].

4.3.1.5 Interpretation of q-th Order Probability

The q-th order probability in Eq. 14 is the probability that q points of the fractal fall in a *vel* of size r. To obtain a practical interpretation of this, consider simplifying the situation, and begin with the squared probability, p^2. Now, this *2nd* order probability is proportional to the relative probability. In other words, given a point j in a current *vel*, the *2nd* order probability is the probability that another point is also within that vel. This is known as the *2nd*-order correlation.

Figure 31 shows the idea of counting the number of points of the fractal that are located within the current *vel* size with respect to pixel j. The relative number of point pairs with respect to pixel p_j can be counted using Eq. 15.

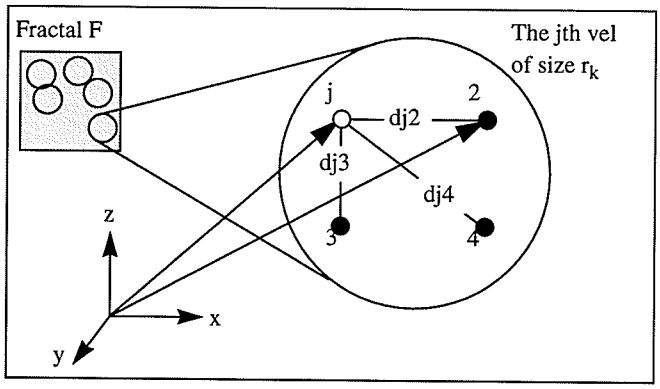

Fig. 31. Example calculation of N_P. There are three pairs of points with respect to pixel j having a distance less than r.

$$\frac{1}{N_T}\sum_{i=1}^{N_T}\theta(r_k - d\langle p_i, p_j\rangle) \tag{15}$$

N_T is the total number of points in the fractal, r_k is the current scale, $d\langle p_i, p_j\rangle$ is the distance of pixel p_i from p_j, and $\theta(\bullet)$ is the Heaviside function of Eq. 16.

$$\theta(x) = \begin{cases} 1 \ \textit{if} \ (x \geq 0) \\ 0 \ \textit{otherwise} \end{cases} \tag{16}$$

This counting procedure is done for all points in the fractal, as shown in Expression 17.

$$\frac{1}{N_T}\left[\sum_{j=1}^{N_T}\left[\frac{1}{N_T}\sum_{i=1}^{N_T}\theta(r_k - d\langle p_i, p_j\rangle)\right]\right] \tag{17}$$

Expression 17 represents the *2nd*-order correlation, which is proportional to the *2nd*-order probability. To obtain the q-th order probability of Eq. 14, we raise Expression 17 to the power of q - *1*.

$$\sum_{j=1}^{N_r} p_j^q = \left[\frac{1}{N_T}\left[\sum_{j=1}^{N_T}\left[\frac{1}{N_T}\sum_{i=1}^{N_T}\theta(r_k - d\langle p_i, p_j\rangle)\right]\right]\right]^{q-1} \tag{18}$$

Substituting Eq. 18 into Eq. 14 and rearranging yields the generalized information fractal dimensions implemented by the generalized correlation integral,

$$D_q = \lim_{r \to 0} \frac{log\left[\left[\frac{1}{N_T}\left[\sum_{j=1}^{N_T}\left[\frac{1}{N_T}\sum_{i=1}^{N_T}\theta(r - d\langle p_i, p_j\rangle)\right]\right]\right]^{q-1}\right]^{\frac{1}{q-1}}}{log\,(r)} \tag{19}$$

4.3.2 Fractal Dimensions of Measures

While Eq. 19 is sufficient for describing the multifractal dimensions of objects, the literature usually goes on further to describe the fractality of the measures themselves. The idea as applied to other measures of objects was first introduced by Mandelbrot [Mand83], and then later described by many others [HePr83], [Grass83], [GrPr83]. We apply this idea to the generalized information based measure of images and other naturally occurring signals.

For a uniform fractal with homogeneous measures, the distribution of the measures, such as probability p, at a given vel of size r, satisfies the single-valued power law,

$$p(r) \sim r^{D} \tag{20}$$

For a non-uniform fractal with inhomogeneous measures, the multi-valued power law is more appropriate,

$$p_j(r_j) \sim r_j^{\alpha_j} \tag{21}$$

In Eq. 21, at each different scale as r approaches zero, one observes that the non-uniform distribution of the measure p_j converges toward a limit curve. The exponent α_j corresponds to the strength of the local singularity of the measure, and is called the Hölder exponent. For each value of α_j, we can associate a subset of points from the fractal F.

Points in a subset have a probability p_j. This is how different subsets of points of different information content (features) within a signal are extracted. To describe the information based strength of each different feature, we then determine the fractal dimension $f(\alpha)$ of the measure as,

$$N_{\alpha(r)} \sim (1/r)^{f(\alpha)} \tag{22}$$

The $N_{\alpha(r)}$ is the number of alphas $\alpha(r)$ that have a particular value.

4.3.2.1 Mandelbrot Dimension

The $f(\alpha)$ is called the Mandelbrot dimension. It can be shown [Schr91] that α and $f(\alpha)$ are related to the Rényi dimension of Eq. 19. The α can be obtained by taking the derivative with respect to the Rényi exponent q.

$$\alpha_q = \frac{d}{dq}[(q-1)D_q] \tag{23}$$

Then, $f(\alpha)$ can be obtained from,

$$f(\alpha) = q\alpha_q - (q-1)D_q \tag{24}$$

Note that Eq. 24 is a Legendre transformation [Schr91] of $(q - 1)D_q$. The literature interprets the relation between $f(\alpha)$ and α as an entropy versus energy description of the object [Kins94b].

4.3.3 Properties of Multifractals

Considering the generalized fractal dimensions D_q of Eq. 14, it is seen that for $q \rightarrow \infty$, Eq. 14 suppresses all of the small probabilities and sifts out the maximum probability. And for $q \rightarrow -\infty$, Eq. 14 suppresses all of the large probabilities and sifts out the minimum probability. In the interstitial space, for each successively positive q, the D_q has the effect of extracting fuzzy subsets representing points of the fractal with successively higher probability. For each successively negative q, the D_q has the effect of extracting fuzzy subsets representing points of the fractal with successively lower probability. Because we are using Rényi's generalized entropy, the above mentioned probability subsets can be referred to as fuzzy information subsets.

Note that for a valid probability distribution function,

$$p_{min} \leq p_{max} \leq 1 \tag{25}$$

Therefore, Dq is a monotonic non-increasing function of q,

$$(D_{q2} \leq D_{q1}) \quad for \ \ (q_1 \leq q_2) \tag{26}$$

If a fractal is strictly self similar, i.e., if all measures on the fractal are uniform and homogeneous, this implies that $f(\alpha) = D_q = D$ = constant, independent of q. Further, the range of variation in D_q with q can indicate the degree of complexity of the multifractal. Figure 32 shows theoretical curves for the generalized dimension D_q, the Hölder exponent $\alpha(q)$, and the multifractal distribution function $f(\alpha)$ [Visc92]. Note that $\alpha(q)$ and D_q intersect at D_1. Also, the peak value of $f(\alpha)$ is the Hausdorff dimension, D_0.

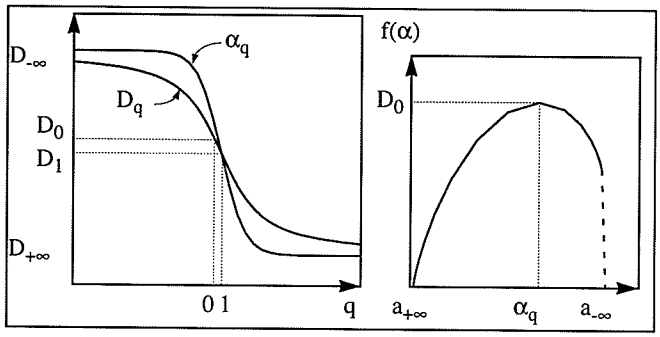

Fig. 32. Theoretical multifractal curves (after [Visc92]).

4.4. Experiments and Discussion

The multifractal measures derived in Section 4.3 are used to measure perceptually relevant features of signals. Although these measures can be applied to all types of signals considered in this thesis, such as images, speech, music, and moving pictures, this section focuses on still images. Work on the application of these multifractal measures to other signals is currently being done in Kinsner's research group [Kins94b].

4.4.1 Application of Multifractal Measures on Images

In the applications part of this chapter, multifractal functions are computed for grey scale images. The software is listed in Appendix B. First, an input image is obtained and embedded in an appropriate space. Usually, the image is obtained as an array of pixels, each having three parameters: x and y coordinates, and an intensity value, as shown in Fig. 33a.

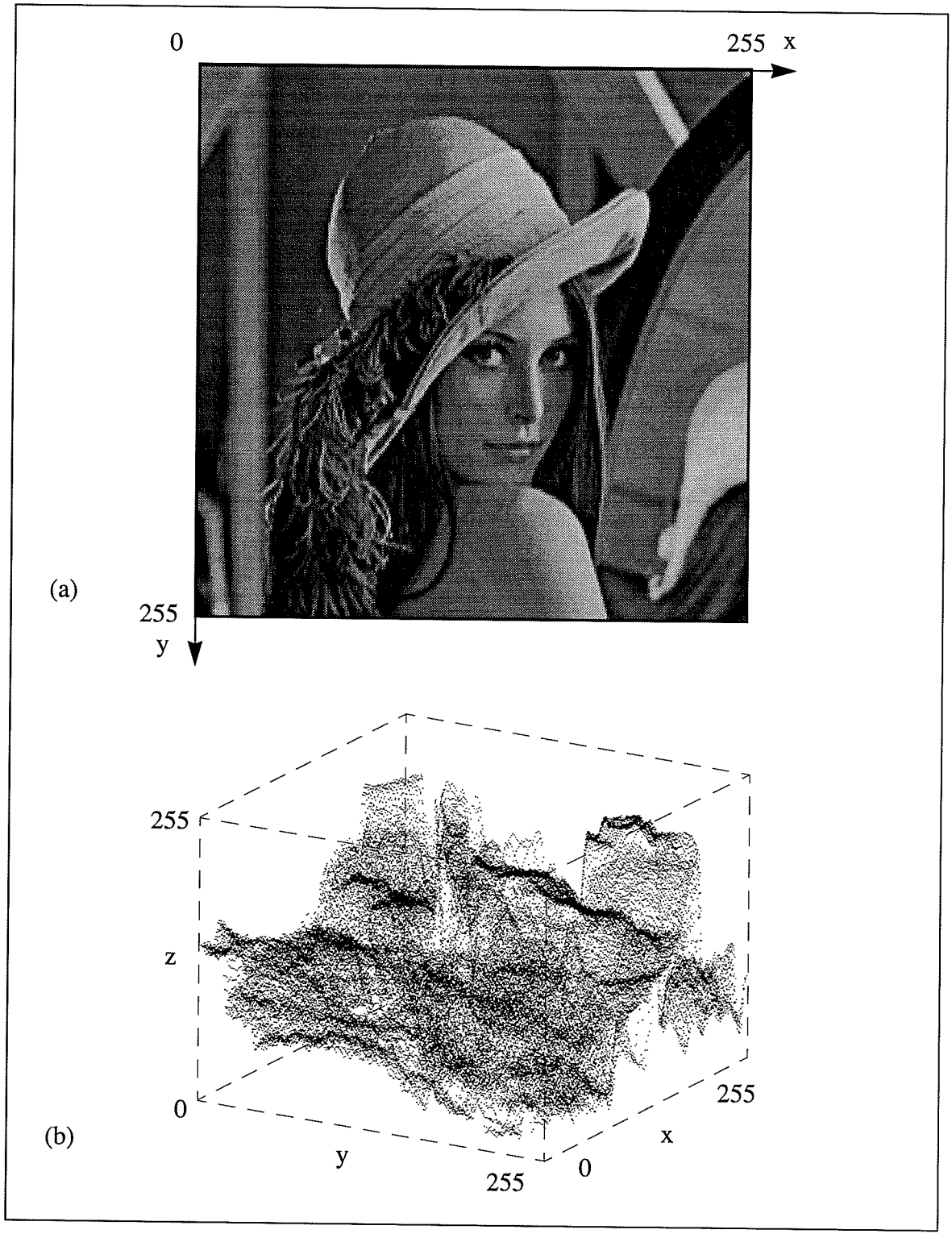

Fig. 33. Representations of image Lena.

The image is embedded into the 3-dimensional Euclidean space, with the intensity

being used as the z-coordinate, as shown in Fig. 33b. Note that the 3-D representation of Lena is not truly 3-D, since each (x, y) pair has only one z value. This should not disturb the experimental procedure, since we are interested in extracting surface feature properties.

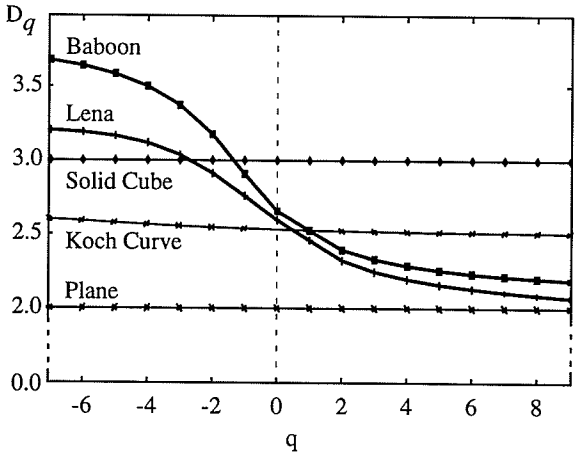

Table 5: Correlation dimension statistics.

Image	D_2	% Error
Plane	2.0006	0.03
Solid Cube	2.9989	0.04
Koch Curve	2.5292	0.2
Lena	2.3218	N/A
Baboon	2.3937	N/A

Fig. 34. Generalized dimensions of selected images.

The experimentally determined generalized dimension curves are shown in Fig. 34. To verify the implementation of the algorithm, the correlation dimension D_2 of a 3-dimensional Koch curve, a solid cube, and a plane were calculated. The 3-D Koch curve was created by first multiplying the 1-D Koch curve by a straight line, giving a fractal dimension of $1 + log_3(4/3)$. Then the resulting 2-D Koch curve is multiplied by itself, giving a 3-D Koch curve of dimension: $2(1 + log_3(4/3)) = 2.5237$. The value of D_2 obtained for the Koch curve agrees with the theoretical value. As expected, for the objects containing strict self similarity, including the Koch curve, solid cube, and plane, the D_q curves are flat. For the non-fractal objects, the dimensions are integer values. Table 5 shows the correlation dimension D_2, along with percentage error with respect to known dimensions, of some of the images that were tested.

The experimentally determined entropy versus energy description is shown in Fig. 35b. As can be seen, the general appearance of $f(\alpha)$ for both the Lena and Baboon images, agrees with the theoretical curve of Fig. 32, except for the domain of large α, where $f(\alpha)$ "doubles back". The technical reason for this is because the D_q plot saturates too quickly for negative q, resulting in a rising hump in the $\alpha(q)$ plot, as shown in Fig. 35a.

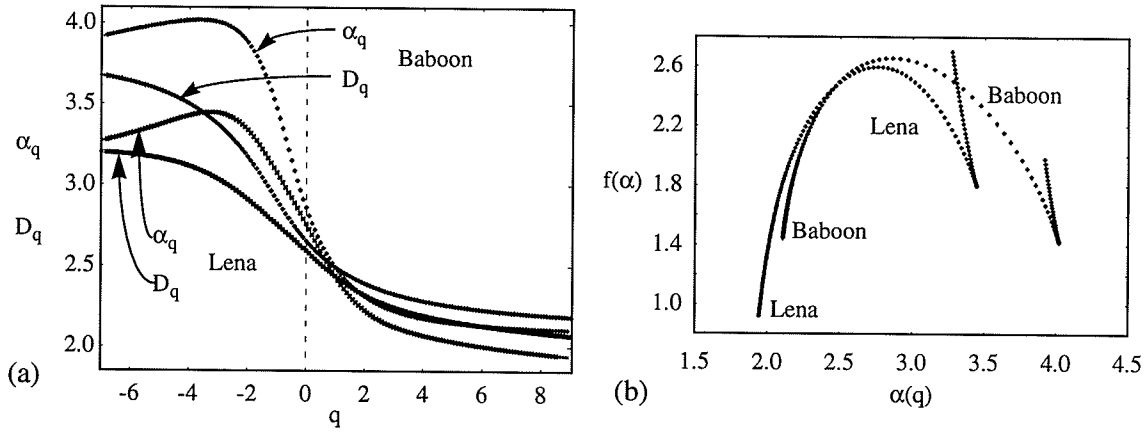

Fig. 35. (a) Generalized dimensions of selected images. (b) Multifractal distribution of Lena and baboon

A reason why D_q saturates early for negative q, may be because of the absence of low probability subsets in the data. Recall that for each successively negative q, the D_q has the effect of extracting subsets of points representing successively lower probability. A reason explaining the inability of the D_q to extract the low probability subsets in the data may be because of insufficient population [AtSV88]. In order to obtain statistically significant results for highly improbable data, there must be a large population from which to extract the statistics. A supporting argument of this is the insufficient resolution of the data, i.e., the pixel density at which the image was projected onto the 2-D space. Or, it could be that there are just no more lower probable data subsets present. For example, p_{min} has been extracted from the Lena image by about $q = -5$.

4.5. Summary of Chapter 4

In this section of the thesis, a multifractal entropy measure for signals is presented. The multifractal measure is based on using the generalized correlation function to determine the generalized dimensions. From the generalized dimensions we obtain a multifractal distribution function that partitions the signal into subsets, each of which has a different fractal dimension. Thus, the signal is represented in terms of a partition, that describes the signal in terms of the information content in each subset. While the idea of obtaining a generalized entropy of a natural image, such as a head and shoulder image of a person, has always been sought for in the literature, this information content has never been described in terms of a multifractal distribution function, as it has been done in this thesis.

In this thesis, we report Hausdorff fractal dimensions for Lena and the baboon of 2.5958, and 2.6562, respectively. The multifractal entropy distribution function $f(\alpha)$ shows a slightly wider breadth for the baboon as compared to Lena, indicating the baboon contains a higher degree of non-uniformity.

CHAPTER 5

Conclusions and Recommendations

5.1. Conclusions

This thesis develops perceptual measures of signal features primarily for compression. Because classical distortion measures, such as the mean squared difference between samples of the signal and the reconstructed signal, have little meaning in measuring perceived error, this thesis develops perceptually meaningful measures and perceptually meaningful *objective* measures of signal features. This thesis extends the signal compression work done in the literature in the following three areas: the wavelet transform, artificial neural networks, and multifractals.

More specifically, this thesis develops a model of the cochlea in terms of processing the scales of signals, and in terms of the non-uniform filtering of the scales within the logarithmic bins of perception. This thesis shows how the wavelet transform coupled with the frequency sensitive competitive learning neural network can be used to implement the cochlea model for compressing music.

The wavelet transform provides better results for representing multifractal signals, such as wide band audio, than do other standard transforms, such as the Fourier transform. This is because the wavelet transform offers a compromise in the resolution of time and frequency. It provides good time localization of high frequency components of the wide band audio signal, while maintaining good resolution of the global, low frequency components. The wavelet transform provides an automatic signal decomposition that is compara-

ble with that of the cochlea filter. Furthermore, unlike other transforms, the wavelet transform utilizes a basis function that is prototypical of the audio signal. The self-similar nature and energy content of the scaled and translated wavelets in the time-frequency plane resembles that of wide band audio, and it is this feature that makes the wavelet transform a very good transform from the signal compression viewpoint. The preliminary results show a bit-rate of 150 kbps, rather than 705.6 kbps, with no perceptual loss in quality.

This thesis develops a model of human compression of images. Compression of images involves vector quantization employing various competitive learning algorithms, including hard competitive learning, soft competitive learning, Kohonen's self organizing feature map, and frequency sensitive competitive learning. Because the codebook contains prototypes of a particular type of image (e.g., head and shoulder images), this demonstrates the self-similar nature of these types of signals. These prototypes can be thought of as a basis through which other similar type images can be reconstructed. The results show that grey scale images are compressed by 16:1 and transmitted at 0.5 bits per pixel, while maintaining a peak signal-to-noise ratio of approximately 30 dB.

Finally, this thesis presents a perceptually meaningful and objective measure of the distribution of information in signals. The measure is a multifractal based on Rényi's generalized entropy and a generalized correlation dimension. The features of the signal are represented by information subsets. The multifractal entropy distribution function shows a slightly wider breadth for the baboon as compared to Lena, indicating the baboon contains a higher degree of non-uniformity. This thesis reports Hausdorff fractal dimensions for Lena and the baboon of 2.5958, and 2.6562, respectively.

The above observations show that the main objectives of this thesis have been achieved.

5.2. Contributions of the Thesis

The contributions of this thesis are listed as follows:

- Development of a model of the cochlea in terms of processing the scales of signals, and in terms of the non-uniform filtering of the scales within the logarithmic bins of perception.
  - Implementation of the cochlea model using the wavelet transform coupled with the frequency sensitive competitive learning neural network.
  - Development of a novel Graphical User Interface (GUI) for auditory visualization of the quantization and filtering of wavelet coefficients in each subband.
  - Demonstration of the self-similar nature of wide-band audio.

- Development of a model of human compression of images based on vector quantization.
  - Implementation of the human compression model using learned vector quantization employing various competitive learning algorithms, including hard competitive learning, soft competitive learning, Kohonen's self organizing feature map, and frequency sensitive competitive learning.
  - Development of a novel Graphical User Interface (GUI) for visualization of the dynamic and adaptive learning procedure.
  - Demonstration of the self-similar nature of images.

- Development of a perceptually meaningful and objective measure (a multifractal) of the distribution of information in signals.
  - A derivation of an expression for the generalized dimensions that uses the information measure at the core of the multifractal measure specifically for extracting generalized information based features for signal compression (Section 4.3.1).
  - A unified definition of fractal dimension (Definition 1, page 84).

- Numerous demonstrations of the work to industry, government, other researchers, and the community at large.

- Work used to assist the government for identification purposes.

- Work used in a commercial implementation of a weather identification system [KLSW94].

- Lectures on learning vector quantization of images given in a graduate and undergraduate courses on Computer Vision and Digital Image Processing [Lehn91].

- The work in this thesis has been published in journals and conferences, including [FeKi94], [LFKK94], [KLSW94], [LaFK93], [FeLK93a], [[FeLK93b]], [WaFK93], [KLIA91], [FeKi91], [AMFK91], [FeKi90], [FeKi89].

5.3. Recommendations for Future Work

The suggested future work includes applying the multifractal measure for signal segmentation, enhancement, and compression, as it is already being done in Dr. Kinsner's research group. In particular, the singularity strength α can be used to extract the different information bearing subsets of the image. These subsets will then identify the segmentation of the image, provide a means for edge detection, and provide a pointer in a look up table for signal quantization.

5.3.1 Multifractal Measure SOFM Codebooks

A fractal-based description of the dynamical process of learning the codebook of a self-organizing feature map (SOFM) can be developed. It can be shown that a measure of the *dynamics of learning* an SOFM codebook can be described by a multifractal. The entropy of a codeword of an SOFM codebook is the number of times the codeword is used to represent an input vector. For a natural scene, such as a facial image, the entropy of the codebook is a non-uniform function. That is, different codewords are used a different number of times. A scale is defined here as the point in the learning process at which temperature is artificially introduced into the system in order to nudge the system out of a local minima. The entropy function when calculated just prior to these insurgence points approaches a curve for scale points much greater than one. In the limit of large scale, this curve represents a probability curve from which the generalized dimensions Dq can be calculated. This Dq curve describes the fractality of the SOFM codebook.

These results can be used several ways in improving image (or any other signal) compression. A stopping point in the learning process can be defined as the point at which the entropy of the codebook approaches a prescribed Dq curve. The optimal stopping point would be the point at which each prototype becomes an exemplar of many image vectors having the same fractal dimension. The prototype vectors would partition the image into fractal dimension subsets. This kind of segmentation can lead to improved compression because the complexity of multifractality is removed. A prototype vector must now only be an exemplar of a set of image vectors having the same fractal dimension.

5.3.2 The Self-Similarly Organized Fractal Feature map

In this thesis it was stated that "The important point [of vector quantization] is the driving force that tells the [neural network which features are] the important ones. This driving force should be similar to the one we use when we perform the reconstruction of [signals] in our minds". Since the fractal dimension is reflective of our perceptual sense of signal features, the multifractal measure can be used as the equivalence relation in learning vector quantization. In this thesis, the Euclidean metric was used to determine the similarity between the input image vector and a codeword in the codebook. Instead of using the Euclidean metric, one can incorporate the multifractal measure to determine the degree of similarity between input signal vectors and codewords.

The learned codebook in this case would represent a partition that is based on self-similar fractal sets. Each region in the partition would contain exemplar signal vectors along with the prototype vector, and they would all have the same fractal dimension. In

other words, a codebook of 256 features implies a partition containing 256 different sets, each of which has an associated fractal dimension.

Furthermore, using the approach of Kohonen's self-organizing feature map, the codebook that uses the multifractal equivalence relation would be a self-similarly organized fractal codebook. Each prototype vector in this codebook would be locally surrounded by other prototypes having similar fractal dimensions. This codebook would yield a multifractal topological map of signals.

This approach would be called the self-similarly organized fractal feature map (SSOFFM).

REFERENCES

[AKCM90] S.C. Ahalt, A.K. Krishnamurthy, P.Chen, and D.E. Melton, Competitive learning algorithms for vector quantization, *Neural Networks*, vol. 3, no. 3, pp. 277-290, 1990.

[AMFK91] C. Anderson, D. Mcneill, K. Ferens, and W. Kinsner, "A speech processing system using ADPCM and xilinx," Int. Conf. Engineering, Medicine, and Biology Society (EMBS), 1990.

[AtSV88] H. Atmanspacher, H. Scheingraber, and W. Voges, "Global scaling properties of a chaotic attractor reconstructed from experimental data," *Phys. Rev.*, vol. A37, no. 4, pp 1314-1322, Feb. 15, 1988.

[BGSS92] K. Brandenburg, H. Gerhser, D. Seitzer, and T. Sporer, "Transform coding of high quality digital audio at low bit rates-algorithms and implementation," *IEEE Int. Conf. Comm.* ICC, IEEE Cat. No. 90CH2829-0, vol. 3, pp. 932-936, April, 1992.

[CaTW90] J. P. Cambell, Jr., T. E. Tremain, and V. C. Welch, "The proposed Federal Standard 1016 4800 bps voice coder: CELP." *Speech Technology*, pp. 58-64 Apr./ May 1990.

[CMQW89] R. R. Coifman, Y. Meyer, S. Quake, and M. V. Wickerhauser, "Signal processing and compression with wave packets," *(preprint)*, 15p., 1989.

[DaMW92] I. Daubechies, S. Mallat, and A. S. Willsky, "Introduction to the special issue on wavelet transforms and multiresolution signal analysis," *IEEE Trans. Information Theory*, 38, pp. 529-53, 1992.

[Ever89] F. A. Everest. *The Master Handbook of Acoustics*. Blue Ridge Summit, PA:TAB Books, 1989, 366 pp.

[FeKi89] C. Love, K. Ferens and W. Kinsner, "A speech recorder and synthesizer using ADPCM,"Int. Conf. Engineering, Medicine, and Biology Society (EMBS), IEEE cat. no. 89 CH2770-6, pp 659-660, 1989.

[FeKi90] K. Ferens and W. Kinsner, "A speech processing system for a speech splicer," The Canadian Medical and Biological Engineering Society (CMBES), 1990.

[FeKi91] K. Ferens and W. Kinsner, "Word synthesis using fuzzy splicing," Proc. IEEE WESCANEX '91, (SK, Canada), IEEE 91CH2927-2, pp 210-215, 1991.

[FeKi94] K. Ferens and W. Kinsner, "A multifractal entropy measure for feature extraction of images," in *Proceeding of 1994 Canadian Conference on Electrical and Computer Eng.*, (ISSN 0-7803-2416-1, IEEE CAT 94TH8023), Appendix, Sep. 1994.

[FeLK93a] K. Ferens, W. Lehn, and W. Kinsner, "Image compression using learned vector quantization," *Proc. IEEE WESCANEX '93*, (SK, Canada), IEEE 93CH3317-5, pp. 299-312, 1993.

[FeLK93b] K. Ferens, A. Langi, , and W. Kinsner, "Energy and frequency adaptive wavelet subband coding for wideband audio compression," *J. Mathematical Modelling and Scientific Computing* (special issue), (ISSN: 1067-0688), (in print, accepted for publication: 1993).

[Fere95] K. Ferens, Perceptual measures of signal features: software, *Technical Report*, UMECE TR-95-008, Dec. 1995, 278 pp.

[FLIK91] K. Ferens, C. Love, A. Indrayanto, A. Langi and W. Kinsner, "A neural network Hamming encoder and decoder", *IASTED International Conference on Computer, Electronics, Control, and Communication*, Calgary, 1991.
(This paper was awarded first prize in the best student paper competition).

[Gras83] P. Grassberger, "Generalized dimensions of strange attractors," *Phys. Lett.*, vol. 97A, no. 6, pp. 227-230, 1983.

[GrPr83] P. Grassberger and I. Procaccia, "Measuring the strangeness of attractors," *Physica*, vol. 9D, pp. 189-208, 1983.

[HePr83] H.G.E. Hentschel and I. Procaccia, "The infinite number of generalized dimensions of fractals and strange attractors," *Physica*, vol. 8D, pp. 435-444, 1983.

[Herl93] C. Herley, "Wavelets and filter banks," *Ph.D. Thesis*, The University of Columbia, 212 pp., 1993.

[Hout92] A. J. M. Houtsma, "Psychoacoustics and modern digital audio technology," *Philips Journal of Research*, Vol. 47, R1263, pp. 3-14, 1992.

[Jaya92] N. Jayant, "Signal compression: Technology targets and research directions," *IEEE Trans. Selected Area in Comm.*, IEEE 0733-8716/92, vol. 10, no. 5, pp. 796-818, June 1992.

[Jmax60] J. Max, "Quantizing for minimum distortion," *IRE Trans. Information Theory.*, 1960.

[KFAG93] W. Kinsner, K. Ferens, A. Langi, W. Grieder, L. Wall, G. Stacey, T. Tessier, and M. Kinsner, "Demos of data and signal compression for multimedia," InfoMedia '93 Symposium and Exhibition; Winnipeg, MB, March 9-10, 1993.

[KiLa93] W. Kinsner and A. Langi, "Speech and image signal compression with wavelets," *Proc. IEEE WESCANEX '93*, (SK, Canada), IEEE 93CH3317-5, pp. 368-375, 1993.

[Kins91] W. Kinsner, Review of data compression methods, including Shannon-Fano, Huffman, arithmetic, Storer, Lempel-Ziv-Welch, fractal, neural network, and wavelet algorithms, *Technical Report*, DEL91-1, Jan. 1991, 157 pp.

[Kins94a] W. Kinsner, "Microprocessor and microcomputer interfacing for real-time systems," *Lecture Notes*, Dept. Electrical and Computer Eng., University of Manitoba, 175 pp., 1994 (6.1a ed).

[Kins94b] W. Kinsner, "A unified approach to fractal and multifractal dimensions," *Technical Report*, DEL94-4. Department of Electrical and Computer Engineering, University of Manitoba, Winnipeg, Manitoba, Canada, 140 pp, May 1994.

[Kins94c] W. Kinsner, "Fractals and Chaos Engineering," *Lecture Notes*, Dept. Electrical and Computer Eng., University of Manitoba, 1994.

[KlKi87] G. A. Klimenko and W. Kinsner, "A study of ADPCM, CVSD, and phoneme speech coding techniques," M. Sc. thesis, Department of Electrical and Computer Engineering, University of Manitoba, Winnipeg, Manitoba, Canada, 228pp, 1987.

[KLSW94] A. Keck, L. Legal, G. Sawatzky, D. Westmore, R. Vivanco, W. Kinsner, A. Langi and K. Ferens, "Automated recognition of severe summer weather features in radar data based on neural networks and expert systems," in *Proceeding of 1994 Conference on Decision Support System 2001*, Toronto, pp. , Sep. 1994.

[Koho90] T. Kohonen, The Self-Organizing Map, *Proc. IEEE*, vol. 78, no. 9, pp. 1464-80, Jan. 1990.

[LaFK93] A. Langi, K. Ferens, and W. Kinsner, "A wavelet model of LPC excitation for speech signal compression," *Proc. 9thg Int. Conf. on Math. and Comp. Modelling,* 1993.

[Lang92] A. Langi, "Code-excited linear predictive coding for high quality and low bit rate speech," *M.Sc. Thesis*, University of Manitoba, Winnipeg, MB, Canada, 138 pp., 1992.

[LaFK93] A. Langi, K. Ferens, and W. Kinsner, "A wavelet model of LPC excitation for speech signal compression," *J. Mathematical Modelling and Scientific Computing* (special issue), (ISSN: 1067-0688), (in print, accepted for publication: 1993).

[LeGa91] D. LeGal, "MPEG: A video compression standard for multimedia applications," *Commun. of ACM*, pp. 47-58, April, 1991.

[Lehn91] W. Lehn, Digital Image Processing and Computer Vision, *Lecture Notes.* Winnipeg, MB; Department of Electrical and Computer Engineering, University of Manitoba, 1991.

[Lokh91] G. C. P Lokhoff, "DCC-digital compact cassette," *IEEE Trans. on Consumer Electronics*, Vol. 37, Iss. 3, pp. 702-706, 1991.

[LFKK94] A. Langi, K. Ferens, W. Kinsner, T. Kect, and G. Sawatzky, "Intelligent storm identification system using a hierarchiecal neural network," in *Proceeding of 1994 Canadian Conference on Electrical and Computer Eng.*, (ISSN 0-7803-2416-1, IEEE CAT 94TH8023), pp. 501-504, Sep. 1994.

[Mall89] S. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Trans. Patt. Anal. Machine Intell.,* vol. 11, no 7, pp. 84-95, July 1989.

[Mand83] B. Mandelbrot, *The Fractal Geometry of Nature*. New York, NY: W.H. Freeman, 1983, 468 pp.

[Math93] The original 8-bit soundtool.c1.31 was modified by Adolf Mathias to process 16-bit audio. The modified version was obtained by this author via email from Adolf Mathias (mathias@ira.uka.de) at about January of 1993.

[McAR90] J. McAuliffe, L. Atlas, and C. Rivera, A comparison of the LBG algorithm and Kohonen neural network paradigm for image vector quantization, *Proc. IEEE Intern .Conf. Acoustics, Speech & Sign. Processing,* ICASSP90 (Albuquerque, NM; Apr. 3-6, 1990), IEEE Cat. no. 90CH2847-2, vol. 4, pp. 2293-96, 1990.

[Musm90] H. G. Musmann, "The ISO coding standard," *IEEE GLOBECOM '90*, IEEE Cat. No. 90CH2827-4, pp. 511-517, 1990.

[NaFe88] N. Nasrabadi and R. Feng, Vector quantization of images based on Kohonens self-organizing feature map, *IEEE Intern .Conf. Neural Networks,* vol. 1, pp 101-8, June 1988.

[PTVF92] W. H. Press, W. H., S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical recipes in C: the art of scientific computing*. New York, NY: Cambridge University Press, 994p, 1992.

[HJKP86] T. C. Halsey, M. H. Jensen, L. P. Kadanoff, I. Procaccia, and B. I. Shraiman, "Fractal measures and their singularities," *Physical Review A,* vol. 33, no. 2, pp. 1141-1151, Feb. 1986.

[Pent84] A.P. Pentland, "Fractal-based description of natural scenes," *IEEE Trans. Pattern Anal. Machine Intell.*, IEEE 0162-8828/84, vol. PAMI-6, no. 6, pp. 661-674, November 1984.

[Rény55] A. Rényi, "On a new axiomatic theory of probability," *Acta Mathematica Hungarica* 6, pp. 285-335, 1955.

[Schr91] M. Schroeder, *Fractals, Chaos, Power Laws*. New York, NY: W.H. Freeman, 1991, 429 pp.

[Seit91] D. Seitzer, T. Sporer, K. Brandenburg, H. Gerhauser, B. Grill, and J. Herre. "Digital coding of high quality audio," *IEEE Advanced Computer Technology, Reliable Systems and Applications*, IEEE Cat. No. 91CH3001-5, pp. 148-154, May, 1991.

[Spur90] W. J. Spurlin, "Rate-reduced digital audio in the broadcast environment," *Proc. of the NAB engineering conference*, Atlanta, 1990.

[TrMe90] K. Truong and R. Mersereau, "Structural image codebooks and the self-organizing feature map algorithm", *Proc. IEEE Intern .Conf. Acoustics, Speech & Sign. Processing*; ICASSP90 (Albuquerque, NM; Apr. 3-6, 1990), IEEE Cat. no. 90CH2847-2, vol. 4, pp. 2289-92, 1990.

[TRla95] Y. Bourlais, etal., Telecommunications Research Laboratories (TRLabs), Winnipeg, Manitoba, Canada, 1995.

[Veld92] R. N. J. Velduis, "Bit rates in audio source coding," *IEEE Journal on Selected Area in Comm.*, IEEE 0733-8716/92, vol. 10, no. 1, pp. 86-96, January 1992.

[ViBL94] J. D. Villasenor, B. Belzer, and J. Liao, "Filter evaluation and selection in wavelet image compression", *Proc. IEEE 1994 Data Compression Conference*, Utah, pp. 351-360.

[Visc92] T. Vicsek, *Fractal Growth Phenomena.* Singapore: World Scientific, 1992, 488 pp.

[WaFK93] L. Wall, K. Ferens, and W. Kinsner, "Real-time dynamic arithmetic coding for low bit rate channels," *Proc. IEEE WESCANEX '93*, (SK, Canada), IEEE 93CH3317-5, pp. 381-391, 1993.

[Wall90] G.K. Wallace, Overview of the (ISO/CCITT) JPEG still image compression standard, Image Processing Algorithms and Techniques. In *Proc. SPIE*, vol. 1244, pp. 220-233, Feb. 1990.

[YaWS92] X. Yang, K. Wang, and S.A. Shamma, "Auditory representations of acoustic signals," *IEEE Trans. on Information Theory.*, IEEE 0018-9448/92, vol. 38, no. 2, pp. 824-839, March 1992.

[YaZG92] E. Yair, K. Zeger, and A. Gersho, "Competitive learning and soft competition for vector quantizer design," *IEEE Trans. on Signal Processing.*, IEEE 1053-587X/92, vol. 40, no. 2, pp. 294-309, February 1992.

APPENDIX A

Images of Chapter 3

This appendix contains the images used in the experimental part of the section of the thesis.

Fig. 36. Lena original (512x512 by 8 bpp).

Fig. 37. Lena (512x512 by 0.5 bpp) quantized using linear vector quantization. The PSNR is 26db. Note the smallest building block is a 4× 4 array of pixels, each pixel having the same grey level

Fig. 38. Lena (512x12 by 0.5 bpp) quantized using Lena's 30.35 db codebook. The PSNR is 30.35 db.

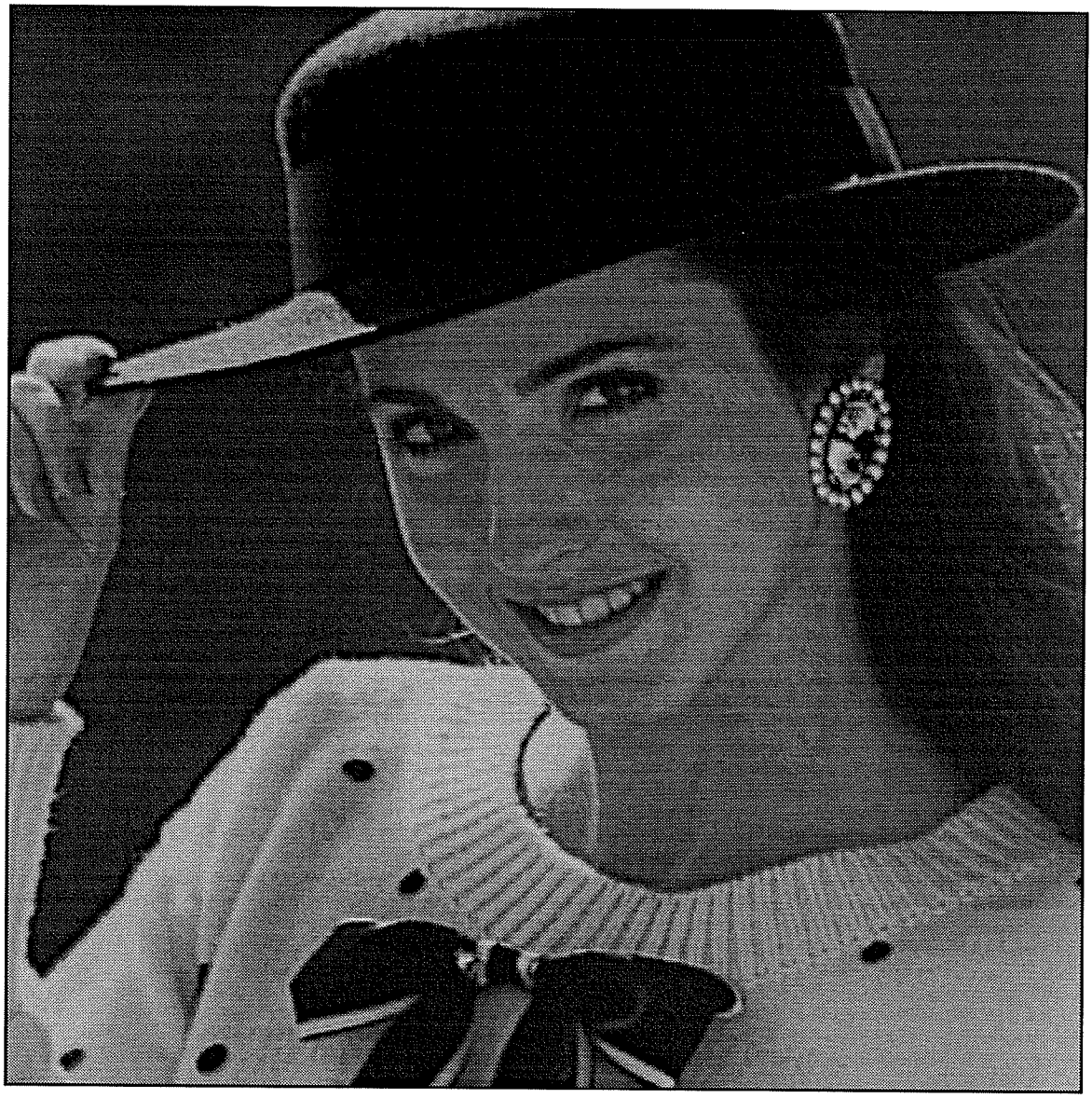

Fig. 39. Lori original (512x512 by 8 bpp).

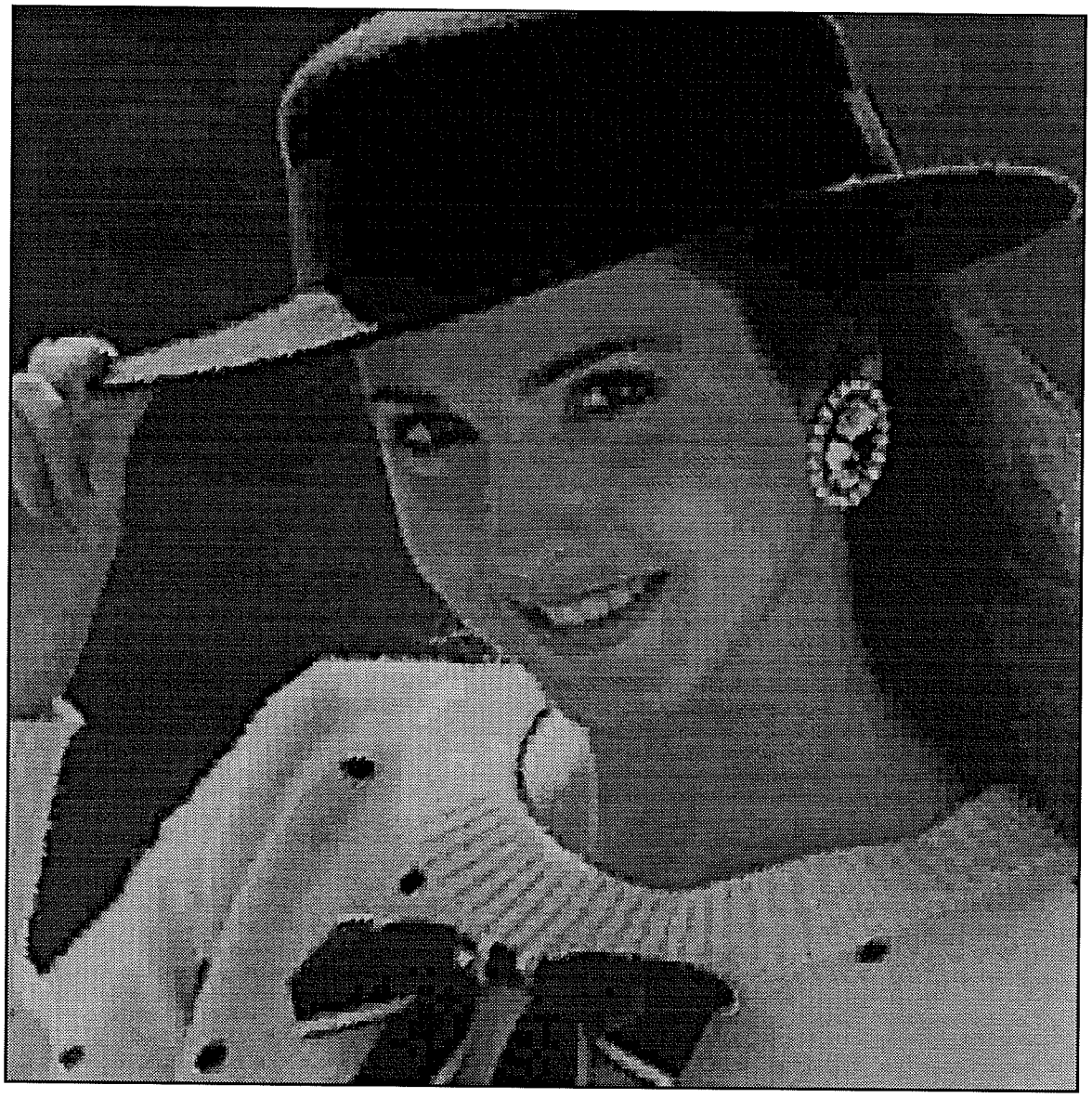

Fig. 40. Lori (512x512 by 0.5 bpp) quantized using Lena's 30.35 db codebook. The PSNR is 31.6 db.

Fig. 41. Andrea original (512x512 by 8 bpp).

Fig. 42. Andrea (512x512 by 0.5 bpp) quantized using Lena's 30.35 db codebook. The PSNR is 33.45 db.

Fig. 43. Tanya original(512x512 by 8 bpp).

Fig. 44. Tanya (512x512 by 0.5 bpp) quantized using Lena's 30.35 db codebook. The PSNR is 34.0 db.

Fig. 45. Renoit painting (512x512 by 8bpp).

Fig. 46. Renoit painting (512x12 by 0.5 bpp) quantized using Lena's 30.35 db codebook. The PSNR is 24.7 db.

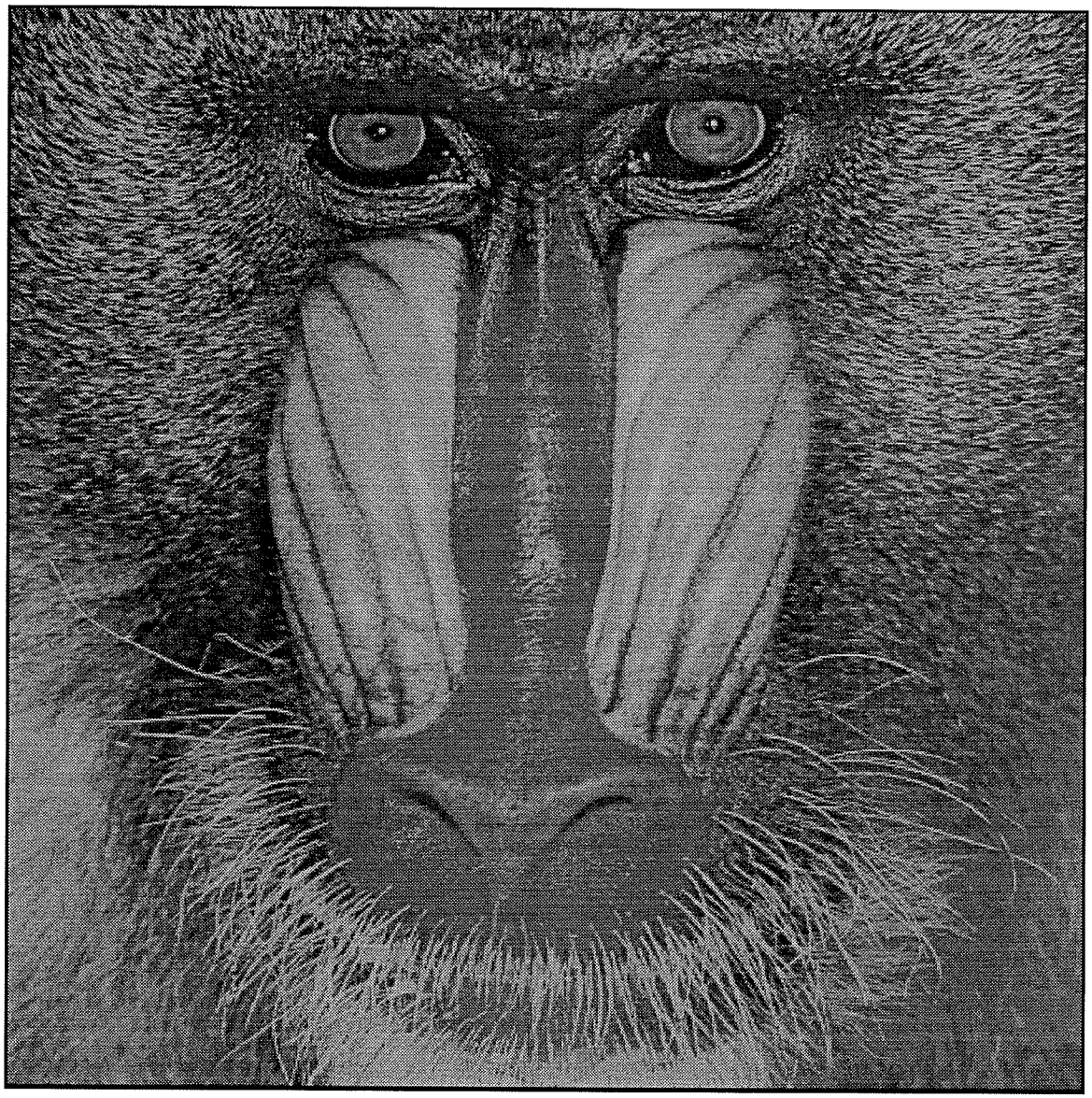

Fig. 47. Baboon original (512x512 by 0.5 bpp).

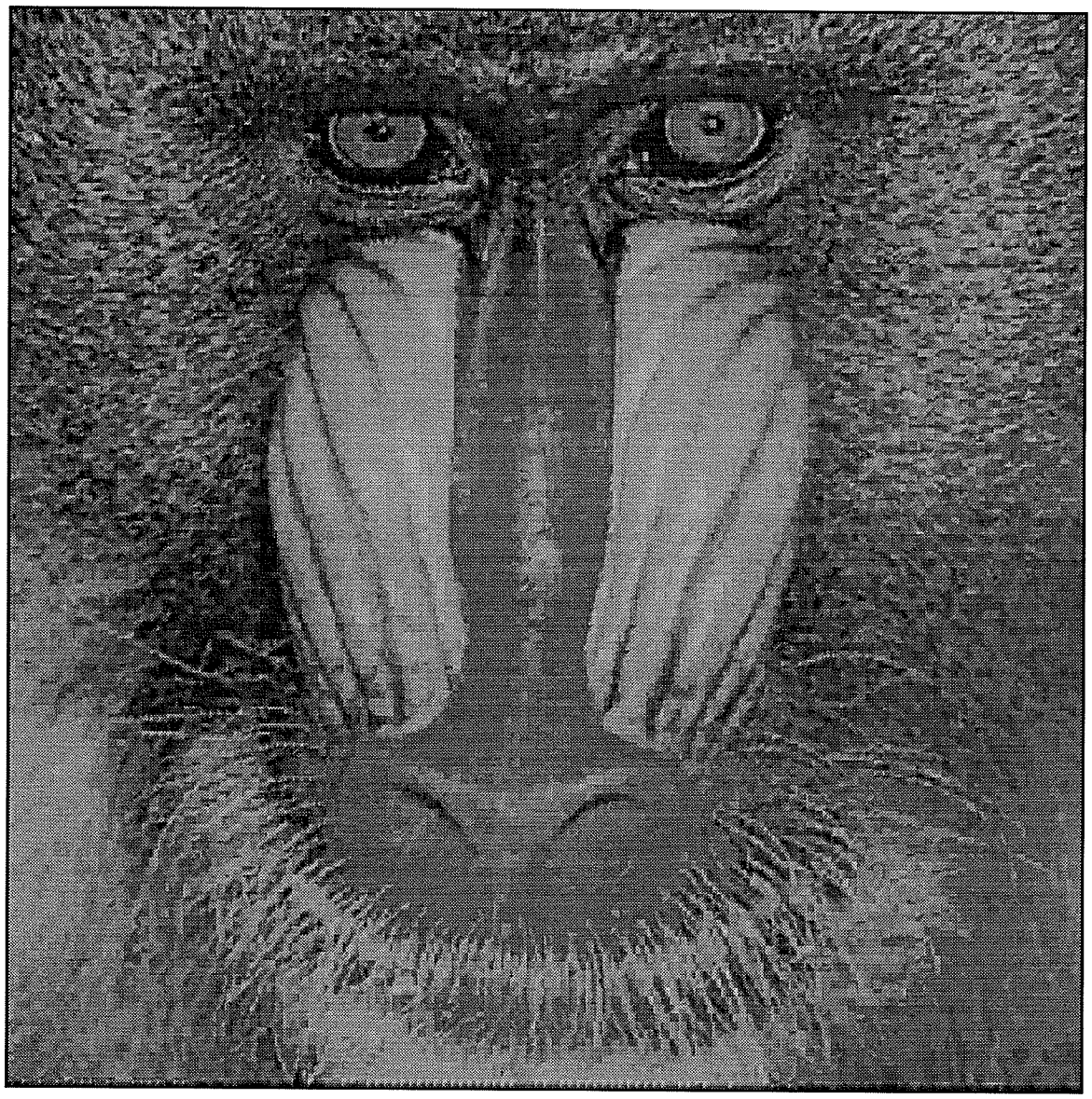

Fig. 48. Baboon (512x512 by 0.5 bpp) quantized using Lena's 30.35 db codebook. The PSNR is 24 db, 2db lower than if Baboon's 26 db codebook is used.

Fig. 49. Lena (512x512 by 0.5 bpp) quantized using Baboon's 26 db codebook. The PSNR is 26.7 db, 3.65db lower than if Lena's 30.35 db codebook is used.

APPENDIX B

Software Listing

The software developed in this thesis can be found in technical report [Fere95].