Entropy-Constrained Recursive Vector Quantization and Application to Image Sequence Coding

by

Majid Foodeei

Department of Electrical Engineering
McGill University, Montreal
1995

A Thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements of the degree of
Doctor of Philosophy

ISBN 0-612-12369-3

to my parents

Abstract

In this dissertation, we consider some classical problems in source coding theory in general and image sequence coding in particular. Shannon rate-distortion-function (RDF) provides the unbeatable lower bound for a source coder's performance with an arbitrary complexity. In theory, for most cases of interest, the RDF is only achievable in the limit of infinite delay. Practical communication systems impose constraints on the maximum coding delay and complexity. Hence the objective of source coding is to achieve performance closest to the RDF bound for a given delay and complexity. The fact that real-world input sources are usually nonstationary and have memory may make the achievement of the above objective more difficult.

Vector quantization (VQ) is known to provide performance close to the RDF, given sufficient delay and complexity. Such high VQ performance can become available through the entropy constraint VQ (EC-VQ) strategy. To overcome the problem of impractical delay and complexity of EC-VQ, we propose a new EC design technique which combines the merits of EC design with the benefits of recursive and adaptive VQ. For historical reasons, we will refer to this new scheme as EC code-excited linear predictive (EC-CELP) quantization and will demonstrate that it has an excellent potential in accomplishing the aforementioned source coding objective. In comparison with other schemes, it achieves the closest performance to the rate-distortion bound for a given VQ dimension (delay) and with a relatively low complexity. To quantify the EC-CELP's superior performance we provide new analysis for various EC quantizers which also includes the effects of quantization noise at low bitrates. New formulations for the maximum available entropy-coded quantization gains are developed which incorporate rate-distortion theoretic limitations at low bitrates.

In the video coding portion of this thesis we propose a *recursive* and *multi-frame*

video coding system using EC-CELP quantization in the temporal domain. This new video coding configuration is designed to overcome the performance limitations of the commonly used differential pulse code modulation (DPCM) temporal video coding. Its recursive nature sets this new technique apart among other recent non-recursive multi-frame techniques which can not provide similar high temporal video compression with minimal delay (few frames). A suitable motion estimation and coding configuration is also suggested. Within the scope of this thesis, some of the problems and issues pertinent to the proposed coding system are addressed. Significant bitrate reduction can be obtained by using the proposed temporal quantizer over the conventional scheme.

Résumé

Titre de thèse: ***"Quantification vectorielle récursive avec contrainte d'entropie et application au codage de séquences d'images"***

Dans cette thèse, nous considérons des problèmes classiques de la théorie du codage de source en général et de ses applications au codage de séquences d'images en particulier. La courbe débit distorsion ("RDF") de Shannon fournit la limite supérieure des performances d'un codeur de source de complexité arbitraire. En théorie, cette "RDF" ne peut être réalisée qu'à la condition d'un retard infini. En pratique, les systèmes de communication imposent des limites à la complexité et au retard. L'objectif du codage de source est donc de s'approcher au plus près de la "RDF" au sens débit-distorsion, pour une complexité et un retard donnés. Cet objectif est rendu d'autant plus difficile qu'en pratique les sources sont généralement non stationnaires et à mémoire.

La quantification vectorielle ("VQ") fournit des performances proches de la "RDF", avec le retard et la complexité idoines. De telles performances peuvent être atteintes en utilisant une méthode de "VQ" contrainte en entropie ("EC-VQ"). Pour surmonter les problèmes de retards et de complexité, nous proposons dans cette thèse une nouvelle technique de contrainte en entropie ("EC"): elle permet de bénéficier des avantages de l'EC ainsi que de ceux de la quantification vectorielle récursive et adaptative. Pour des raisons historiques, nous nommerons cette nouvelle méthode "EC code-excited linear predictive (EC-CELF) quantization", et nous montrerons qu'elle fournit une excellente réponse aux problèmes du codage de source mentionnés auparavant. Par rapport à d'autres méthodes, "EC-CELP" atteint des performances plus proches de la limite fournie par la "RDF", tout en gardant une complexité relativement faible. Pour estimer ces performances, nous

fournissons une nouvelle technique d'analyse des quantificateurs à "EC", qui tient compte du bruit de quantification à bas débit. Ont également été développées de nouvelles formulations du gain maximum qui peut être atteint par quantification contrainte en entropie, qui tient compte des limites théoriques au sens débit-distorsion à bas débit.

Dans la partie de cette thèse concernant le codage vidéo, nous proposons un schéma de codage récursif et multi-trame, qui utilise la méthode "EC-CELP" de quantification dans le domaine temporel. Ce nouveau schéma est étudié pour améliorer les performances de la méthode classique "DPCM". Sa nature récursive le distingue parmi les nouvelles techniques multi-trames non récursives qui ne peuvent offrir de taux élevés de compression vidéo avec un retard minimal (quelques trames). Un algorithme d'estimation de mouvement associé est également proposé. Comparé aux méthodes classiques, ce schéma permet une réduction notable du débit.

Acknowledgment

My biggest debt of gratitude goes to my supervisor Professor Eric Dubois for his guidance. Without his vision and understanding of the limitations of the current video coding problems, the motivation for this thesis would have not existed. I am also thankful to Professor Nariman Farvardin of University of Maryland for constructive remarks on the subject of entropy-constrained quantization. The major portion of the research was conducted at Institut National de la Recherche Scientifique (INRS) - Télécommunications. The laboratory and facilities provided by INRS were a great help to my research. The financial support provided by my supervisor from the Canadian Institute for Telecommunications Research (CITR) under the NCE program of the Government of Canada and other research funds was appreciated.

I would like to give my special thanks to my parents, my brother, and sisters for their consistent support as well as to Pol for her love and understanding during the course of this thesis. I am also grateful for the companionship and help provided by my friends Shahrokh Valaee, Rob Dijkerman, Masoud Khansari, Remi Depommier, Ahmad Jalali, by my office-mate Albert Golembiowski, and by many others at INRS, Bell Northern Research (BNR), and McGill university.

Contents

List of Figures

List of Tables

List of Acronyms and Abbreviations

Acr. or Abb.	description
AR(M)	M-th order autoregressive process
bps	bit per symbol
(K)bpsec.	(Kilo) bit per second
BTQ	block transform quantization
CELP	code-excited linear predictive (quantization)
DPCM	differential pulse code modulation
DTC	discrete TC
EC	entropy-constrained
Kbit	Kilo bit
KLT	Karhunen-Loeve optimum transform
LBG	Linde-Buzo-Gray VQ design algorithm
Mbit	Mega bit
MHz	Mega Hertz
MSE	mean-squared-error
MT	motion trajectory
GM(M)	M-th order Gauss-Markov
MSGVQ	mean-gain-shape VQ
MT	motion trajectory

PSNR	peak SNR
PC	predictive coding
PSK	Phase shift keying
PTCQ	predictive TCQ
PVQ	predictive VQ
qpf	quantizer performance factor
RC	(resolution) rate-constrained or alphabet size constrained
RDF	rate-distortion function
RVQ	residual VQ
SBC	subband coding
Sec.	second
SFM	spectral flatness measure
SNR	signal-to-noise-ratio
SQ	scalar quantization
TC	transform coding
TCQ	trellis coded quantization
VLC	variable length code (coding)
VQ	vector quantizer

Chapter 1

Introduction

In today's world, digital signals are almost always preferred over analog ones. Processing flexibility, random access in storage, and higher signal-to-noise-ratio (SNR) for transmission are among the advantages of digital signal processing. The drawback is the increased bandwidth or raw data rate of the signal. In the case of time-varying imagery, if a typical television signal is simply digitized, it amounts to a raw data rate of approximately 188 Mbit/sec.[1] For the HDTV, the data rate may be around 1880 Mbit/sec.[2] High data volume means high storage capacity in the case of storage and high bandwidth or channel capacity in the case of transmission. As another example, for a 4 MHz television signal, sampied at 8 bit per sample, with the required Nyquist rate and a PSK modulation scheme (1 Hz per 2 bits), the required bandwidth is increased by 8 fold (32 MHz). Signal compression reduces the bandwidth cost for the digitized signal. The above examples of the data volume, typical of time-varying imagary, show the essential role of signal compression or source coding for transmission and storage of such signals.

One may think of availability of more capacity both in wideband channel technologies such as fiber optics and high capacity storage media such as optical disks. Also one may consider the lower resolution images which do not necessarily have a very high data volume. This may make the *high data volume* argument seem less convincing. The counter

[1]Assuming $512\times512\times30$ pixel/sec.$\times8$ bit/pixel $\times$ 3 color.

[2]For example using $1125\times1125\times60$ pixel/sec.$\times8$ bit/pixel $\times$ 3 color.

arguments which are not limited to video compression are as follows. First, the number of possible multiplexed signals per channel or media is always a consideration. Second, there are cases of channels and media with limited capacity. Radio and telephony channels are examples of such cases. These factors are particularly important in view of the fact that wireless communication and multimedia are the important technology trends into the next century. Hence the need for compression is not only as strong as ever, but has become crucial for matching the supply with the explosion of the new demands. As it was demonstrated by earlier examples, for high volume signals such as video this need is most crucial.

A general digital communication system model is shown in Fig. 1.1, where encoder and decoder are divided into source and channel encoder and decoder. The purpose of the source encoder is to represent the source output as a sequence of binary digits with a fixed or variable number of bits per unit time for this representation. The role of channel encoder and decoder is to make a reliable reproduction of this binary sequence at the channel decoder output possible. The overall goal of the system is reliable and efficient transmission of source output over the channel, while meeting practical considerations such as end-to-end coding delay and coding complexity.

The objective of source coding is to achieve efficient signal representation by removing the redundancies in the signal. Two important categories which characterize the nature of source coding methods are *lossy* or source coding with *fidelity criterion* and *lossless* or *noiseless* source coding. The latter is also referred to as *entropy coding*. Shannon's *first* and *third coding theorems* provided non-constructive description for *optimum* lossy and lossless coding and *bounds* [103], [104]. Due to compression limitation of lossless coding alone, we have to resort to the use of lossy coding.[3] Lossless coding exploits redundancy due to nonuniformly distributed source symbols and it operates on discrete alphabet sources. Since the output of a lossy quantizer is discrete, we may decrease the work for the lossy coder and increase the efficiency of system (as seen later) by a direct

[3]Full lossless coding techniques applied directly have not provided the required high-compression ratio. For example, only compression of 3:1 is reported with a lossless open-loop DPCM followed by a lossless coding [57] (also see [73]).

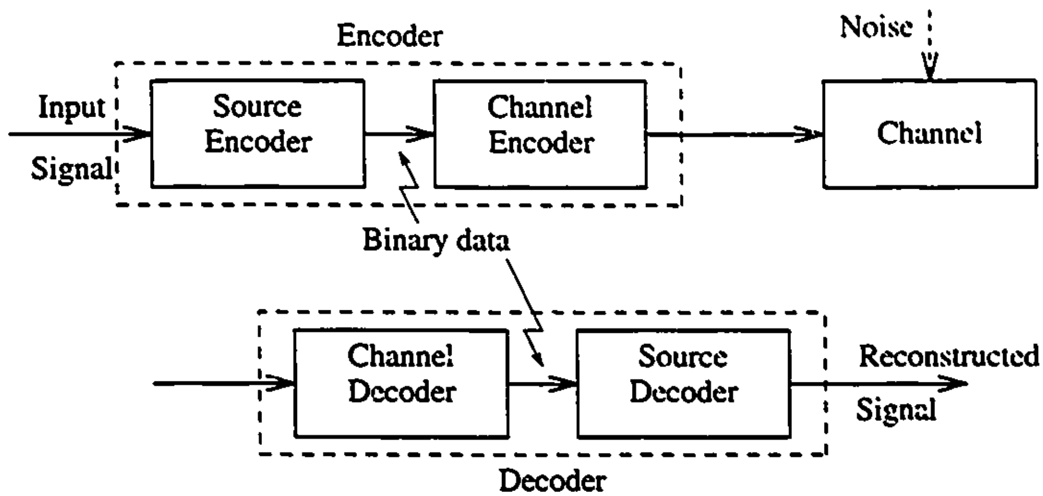


Fig. 1.1 A communication system model.

combination of the two schemes in a sequential lossy+lossless configuration represented in Fig. 1.2. This configuration has advantages over the alternative of lossy coder alone. However, the variable nature of the output rate of most lossless coders (*variable-length-coding* (VLC)) can have some adverse effects in the case of channels which cannot tolerate such fluctuations. There are means to alleviate such adverse effects. As well there are situations where such variations may be tolerated [25], [86].

In the system of Fig. 1.2, the signal input symbol[4] $\mathbf{u}(k)$ is analyzed and a set of parameters $\mathbf{s}(k)$ is extracted.[5] This set of parameters is quantized and represented by the vector $\mathbf{i}(k)$. The combination of analyzer and quantizer blocks may be called the lossy encoder block. This means that if the inverse block of lossy decoder at the source decoder was to reconstruct the signal block using $\mathbf{i}(k)$, the resulting error distance between[6] $\mathbf{u}(k)$ and $\widehat{\mathbf{u}}'(k)$ would be the coding distortion or loss. The output of the lossy encoder is encoded by the lossless encoder to obtain the bit stream $\mathbf{c}(k)$ which is transmitted over the channel. In the absence of channel noise, the bit stream $\mathbf{c}'(k)$ received by the decoder

[4]Bold face represents vector signal and k is the time index for the vector.

[5]A less general case does not include the analysis block. In that case we will denote the input and reconstructed signals as $\mathbf{s}(k)$ and $\widehat{\mathbf{s}}(k)$ respectively.

[6]Unless otherwise stated we will be using a squared error distance measure.

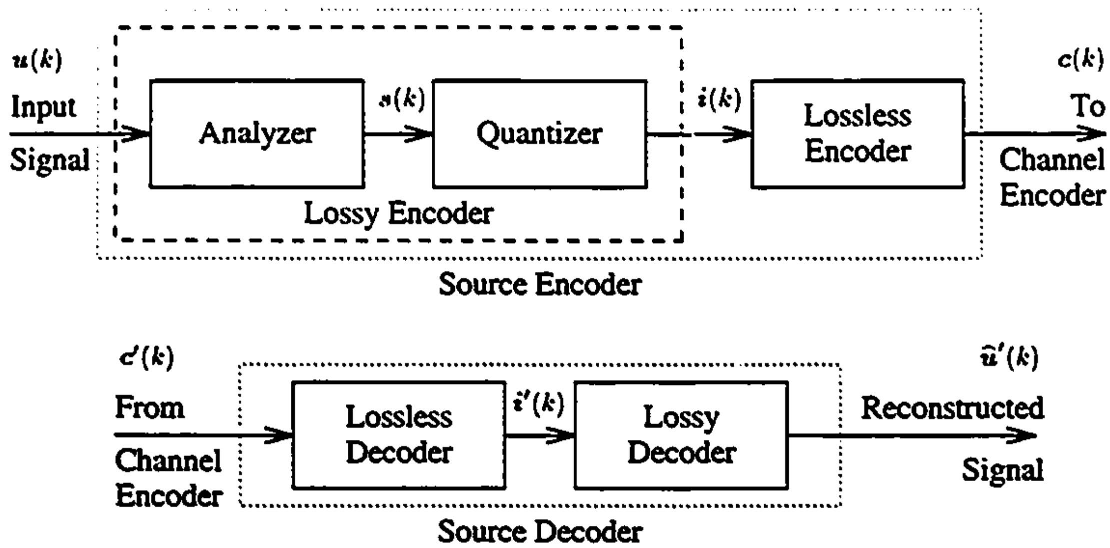


Fig. 1.2 A source coding system combining lossy and lossless coding.

is identical to $c(k)$. At the decoder, the lossy and lossless decoder blocks perform the reverse operation of their encoder counterparts. Parameter set $i'(k)$ is obtained from the bit stream $c'(k)$ and the signal $\hat{u}'(k)$ is reconstructed from $i'(k)$. As seen later, it is possible to combine the lossless and lossy coders in this source coding system and to design them in a joint fashion. Since throughout this thesis we assume an error-free channel, hereafter we may assume $c(k) = c'(k)$ and $i(k) = i'(k)$.

1.1. Source coding techniques

For the general case of coding with fidelity criterion, Shannon provided a theoretical performance bound called the rate-distortion bound. However he did not provide an instrumentable method for obtaining such performance. The ultimate goal of source coding is providing means to accomplish a performance as close as possible to the rate-distortion bound. Also there are always other practical constraints such as coding delay, complexity, and robustness that have to be taken into consideration.

Let us assume that the source coder uses a lossy+lossless configuration as in Fig. 1.2. Huffman coding, Run-Length coding, Arithmetic coding, and many adaptive schemes are among lossless coding techniques. Many variations and combinations of such coders may also be used [107]. Among lossless coders, the VLC uses a smaller length bit stream for the highly probable input blocks, and vice versa. Hence lossless or entropy coding removes redundancies corresponding to the nonuniformly distributed quantized output $i(k)$ in Fig. 1.2. This kind of coding gain is sometimes referred to as *shaping* gain. The goal is to make the average bitrate as close to the entropy of the quantized symbols as possible (hence the name entropy coding).

Predictive coding (PC), transform coding (TC), subband and wavelet coding, and vector quantization (VQ) constitute the main lossy coding techniques. Adaptive variations of the above techniques and a variety of their combinations can also be used. DPCM is the simplest and most common (linear) PC scheme and is frequently used in the temporal component of image sequence coding. PC belongs to a more general class of *recursive* coders. Block coding or VQ coding advantages, with respect to scalar quantization (SQ), are rooted in Shannon's original coding theories. The advantage is due to the conclusion that coding of symbols grouped in vectors or blocks results in a better performance than scalar coding or coding of single symbols [48], [41].

VQ can not only exploit the memory redundancy (*memory gain*) in the signals but will also provide coding gains in the case of independent-identically-distributed (*i.i.d.*) sources. Additional coding gains can be categorized as *dimensionality* or *space filling* and shaping. For the entropy coded configuration in Fig. 1.2, in the absence of shaping gain, the additional gain is limited to space filling.

It is well known that VQ can achieve performance near the rate-distortion bound. However the required dimension (delay) and complexity, especially in the case of nonstationary signals, makes direct VQ impractical. We will see later that using hybrid techniques with VQ, one may alleviate this problem. Among other means to address this problem are the classes of *constrained* and *structured* VQ. Structured VQ techniques such as lattice VQ are more suited for high rates and cases where low delay is not of consideration. For low rates, hybrid techniques using structured VQ are more appropriate and have been successful [72], [99].[7]

TC and transform VQ can be considered to belong to the class of constrained VQ [41]. Among the most widely used source coding techniques for spatial image coding is the TC method using the discrete cosine transform (DCT). For nonstationary signals, adaptive techniques are devised. TC achieves decorrelation and what is referred to as *energy compaction*. Subband and wavelet coding are closely related to TC. Wavelet coding which has recently become an active research topic, particularly in image coding, can be advantageous due to its interesting properties. An inherent multi-resolution representation of signal is one advantage. Another potential advantage of wavelet transform in the case of image signals is due to its regularity properties, which can provide good approximations to a diversity of signals with relatively small number of coefficients.

Advantages of VQ and other methods may be combined in hybrid (combinatory) coding configurations. Among hybrid techniques, combining the advantages of recursive and adaptive VQ, is the scheme of code-excited-linear predictive (CELP) quantization [101]. CELP has been the most successful speech coding scheme in the past decade and has a central role in this thesis. However, we emphasize that we use the name CELP both in its generic speech coding sense and in a sense of recursive and adaptive VQ, which may be considered as a good general coding method for nonstationary input sources with memory.

The performance of the source coder of choice also depends on other considerations such as design method and bit allocation where it may be applicable (e.g. bit allocation in subband and TC). There are two classical approaches for designing quantizers. One uses the criterion of minimization of average distortion for a fixed number of quantizer output alphabet. This type of design is referred to as *alphabet size* or *resolution-constrained* (RC). Alternatively, the minimization may be subject to a constraint on the output entropy (constant entropy). For the configuration of Fig. 1.2, which uses an entropy coder, the second alternative of *entropy-constrained* (EC) design is known to be advantageous [26], [14]. EC design as applied to the CELP quantizer (EC-CELP) is one of the key contributions of

[7] Also see [23], [22] for lattice quantization, [70] for *scalar VQ* (fixed rate structured VQ), [62] for alternative fixed-rate structured VQ, and [83] for Trellis coded quantization based on [106].

this dissertation.

1.2. Video coding techniques

The spatial and temporal characteristics of video signals are very different. For the spatial domain, consideration with regard to representation of image edges significantly effects the outcome of the spatial compression performance. Motion and its characteristics play the key role in the temporal domain compression. In this introductory section we will briefly describe the elements of the common video coding techniques. We will postpone the detailed discussion of common video coders, of more advanced techniques, and a more exhaustive survey of relevant publications in high-compression video coding until chapters 2 and 5. We will see that there is a plethora of research works responding to the need for high compression video coding.

In common video coding techniques, the temporal and spatial redundancies are processed independently (although theoretically this is not optimum). As mentioned, the spatial redundancies are often removed using TC schemes. In particular DCT, providing an order of 50:1 compression for still images, has been the method of choice. Other than energy compaction and decorrelation properties of the DCT, the existence of fast algorithms has made the DCT popular. The success of the DPCM configuration, as the method of choice in temporal redundancy removal in most practical video coders, is due to its good performance and its relative simplicity. Combining spatial TC and DPCM results in the *hybrid* DPCM-TC video coding, the most common video coding configuration.

Motion-compensated video coding uses the *a priori* knowledge of motion characteristics to provide a significant gain. The assumption is that, regardless of motion, there is a high correlation among image pixels in successive frames projected from the same scene points. Motion-compensated coding is also almost always used in the DPCM-TC hybrid video coder. For motion estimation, both block (*block matching*) and pixel-based (*dense*) motion estimation can be used. Almost all practical video coders up to now have used block matching for reasons of complexity. Once the motion-compensated residual (difference) image in a DPCM configuration is obtained, it is fed into the spatial block transform quantizer. Before sending them to the channel encoder, both motion parameters and quantized residual image are entropy encoded. Sometimes temporal redundancies are further exploited with motion compensated interpolation.

The above common approaches in video coding and available compression ratios to these methods have limitations. Use of spatial square blocks in spatial coders at higher compression poses problems which manifests itself in "blockiness" of the reconstructed image. Temporally, reduced-frame-rate video coding may save additional bits at the cost of further degradation in quality. Higher performance compression is also obtained with modifications such as subpixel motion accuracy to the common motion-compensated hybrid DPCM configuration (Appendix E of Ref. [52]). However to obtain the magnitude of compressions required by the new video applications, the video research community has already made the conclusion that departures from the assumptions, models, and quantization methods used in the common hybrid coders will be unavoidable. As an example of the new directions in the spatial domain, the region-based approach has the potential benefits of more efficient representation (modeling and compression) of stationary patches (regions) in the still image [76]. Coding techniques such as the wavelet transform have the promise of bringing the benefits of multi-resolution and better time-frequency resolution through time-localization. Also there has been occasional departure from the two frame differential idea of DPCM (entailing SQ in the temporal domain). But the benefits of such *multi-frame schemes* had not been shown until quite recently [94].

The video coding aspect of this dissertation also falls into this group of recent research. But it will be seen, while all of the schemes in this new group are non-recursive in nature, the proposed scheme in this thesis being *recursive* has important advantages. The proposed motion-compensated recursive video coding system is based on the temporal EC-CELP quantization. To our knowledge, prior to this work, no other temporally *recursive* multi-frame system has been suggested.

1.3. Overview of thesis motivation and contributions

The domains of this thesis are source coding theory in general and video coding in particular. Therefore the contributions of this dissertation are also twofold: 1– Design and analysis of a new efficient high compression source coder, entropy-constrained code-excited linear predictive (EC-CELP) quantization, 2– A new high-compression multi-frame recursive video coding configuration based on EC-CELP. The first part can be further divided into two parts. a– Design of EC-CELP and its special cases. b– Entropy-coded quantization theory at low rates and analysis of EC-CELP and its special cases.

As mentioned, the video coding aspect of the thesis is in the context of the current strive for pushing the frontiers of the state of art video coding by providing better quality coded video at lower rates (high-compression). In motivating this work, we first make the observation that the limits of performance of DPCM in temporal video coding plays a key role. We proceed by investigating such limits and a search for more efficient quantization techniques. The input sources of interest are sources with memory in general and highly correlated sources in particular (intensities along motion trajectories in video coding). The above search leads to the new scheme of EC-CELP quantization.

As an alternative to DPCM, the new scheme of EC-CELP, not only provides near rate-distortion performance at low bitrates for highly correlated signals but also this is achieved at low delay and relatively low complexity. To fully understand the reasons for such high performance, various quantization theoretic aspects of EC-CELP scheme and other alternatives is analyzed. As another contribution of this dissertation, rate-distortion coding gains at low bit rates are defined and numerical results are obtained. Analytical formulations for low bitrate coding gains of EC-CELP and other coders are derived.

To bring the benefits of the new efficient recursive coding technique to video coding, we then propose a suitable video coding configuration and investigate the performance gains over the conventional and other non-recursive alternatives.

Due to fundamentally different elements in this new video coding configuration, there are many problems and issues to be resolved. Within the scope of the thesis, we suggest methods for resolving some of the main problems. We provide some preliminary simulation results which show the feasibility and success of the proposed temporal EC-CELP motion-compensated video coding configuration. The proposed temporal EC-CELP quantization can be incorporated in an all new motion-compensated video coding system which uses spatio-temporal EC-VQ-CELP quantization. The multi-frame recursive video coding is still in its early phases. Future work should provide a more complete evaluation and comparison of alternative schemes by conducting further simulation of full coders.

1.4. Thesis organization

In chapter 2, we first provide a more formal description of the motivation and the context of the dissertation. This chapter contains both background and a survey of other relevant works for source and video coding aspects of the thesis. For reference, some basic frequently used concepts and formulas are furnished. Throughout this chapter, the notation used in the thesis is also established.

The next two chapters are devoted to the source coding aspect of the thesis. Chapter 3 contains the design of EC-CELP quantizer. It presents the EC-CELP within the more general class of EC recursive and adaptive VQ. It also presents EC-CELP as a general configuration whose special cases include EC-VQ, EC-DPCM, and EC-PVQ. Simulation results for the performance evaluation of EC-CELP and other alternatives are also included in this chapter. Chapter 4 contains the analysis of the various EC predictive quantizers especially EC-CELP and its special cases. It also contains a new classification and formulation for the available coding gains at low bitrate using the entropy-coded quantization theory. Numerical results for the above analysis and classifications are also included in this chapter.

Chapter 5 is devoted to the video coding aspect of the thesis. The new multi-frame recursive video coding configuration based on EC-CELP quantization along with related issues are presented in this chapter. It includes investigation of suitable motion estimation techniques and issues related to the reconstruction of decoded motion-compensated multi-frame images. The summary and conclusions of the dissertation are presented in chapter 6. Some of the techniques developed in this dissertation have been reported in conferences or are to be submitted to journals for publication [30], [32], [31], [33], [35], [34].[8]

[8]The work for this thesis and the first draft of this dissertation were completed in December 1994. Minor modifications and corrections were made to produce this final draft.

Chapter 2

Coding Techniques for Sources with Memory and Video

In this chapter background, notations, a detailed description of motivation, and a survey of past works both for the source coding and the video coding aspects of the thesis are presented. This chapter is meant to facilitate the presentation of other chapters and provide the reader with the tools utilized. In section 2.1, the information-theoretic source coding bounds for input sources with and without memory are reviewed. It also contains some relevant results on high-rate quantization gain classifications which will be used in chapter 4. In section 2.2, a history of the EC design and coding is presented. This will allow the reader to see the place of EC-CELP among other EC coders. In the last two sections, a survey of high-compression video coding activities along with a review of image characteristics and modeling are given. Material in Appendix A, summarizing some basics of information theory concepts, is included for reference and completeness.

It will be seen later that the adaptive feature of the EC-CELP quantizer is one of its important features. For the case of nonstationary input sources with memory (e.g. speech [35]), this feature can be most beneficial. However, for the video coding application proposed in this thesis the adaptive feature is less crucial. To make our studies tractable, throughout this work we often deal with the reference stationary Gauss-Markov (GM) source model. Also it is already mentioned that one feature of EC-CELP is its use of VQ.

In the next paragraph, we use the stationary GM process to introduce the vector notation used throughout the thesis. We use this occasion to demonstrate some useful notions of memory, pertinent to vector representations. We will come back to these notions later in the thesis.

For the discrete asymptotically stationary M-th order GM process, GM(M) $\{s(n)\}_{n=0}^{\infty}$, with regression coefficients $\{a_m : m = 1, \ldots, M\}$, we have

$$s(n) = \sum_{m=1}^{M} a_m s(n-m) + w(n), \quad n = 1, 2, \ldots, \tag{2.1}$$

where $\{w(n)\}_{n=0}^{\infty}$ is the *i.i.d.* (white) Gaussian innovation sequence. For simplicity of examples and notation, we will mostly use the first order ($M = 1$) GM process[1]

$$s(n) = as(n-1) + w(n), \quad n = 1, 2, \ldots, \tag{2.2}$$

where a is the regression coefficient and $s(0) = 0$ is assumed. We sometimes refer to a highly correlated source. By this we mean a ranging from 0.95 to close to 1.0. Note that from the source coding perspective, the memory entails redundancy (higher correlation means higher redundancy). The GM(1), or Gaussian autoregressive process AR(1), can be generated by passing the innovation process $\{w(n)\}_{n=0}^{\infty}$ through the synthesis filter (having the Z-transform $\frac{1}{1-P(Z)}$). The first order prediction filter coefficient is obviously the regression coefficient a. Due to block nature of the flow of signals in the coders

[1]The higher order cases directly extend. However due to block nature of the study, cases of $N > M$ and $N <= M$ have to be considered separately (N is the block size). The inter+intra-block memory description remains valid for both cases.

considered, an equivalent block (vector) representation for the source $\{s(k)\}_{k=0}^{\infty}$ is useful:

$$s(k) = \begin{bmatrix} s_1(k) \\ s_2(k) \\ \cdots \\ s_l(k) \\ \cdots \\ s_N(k) \end{bmatrix} \quad \text{where} \quad s_l(k) = s((k-1)N+l) \quad l = 1,2,\ldots,N; k = 1,2,\ldots.$$

Using the above notation, we can describe the following useful equivalent block representation for the equivalent GM(1) source $\{s(k)\}_{k=0}^{\infty}$:

$$s(k) = \begin{bmatrix} s_1(k) \\ s_2(k) \\ \cdots \\ s_l(k) \\ \cdots \\ s_N(k) \end{bmatrix} = a \begin{bmatrix} s_N(k-1) \\ s_1(k) \\ \cdots \\ s_{l-1}(k) \\ \cdots \\ s_{N-1}(k) \end{bmatrix} + \begin{bmatrix} w_1(k) \\ w_2(k) \\ \cdots \\ w_l(k) \\ \cdots \\ w_N(k) \end{bmatrix}$$

$$= \underbrace{\overbrace{\begin{bmatrix} a \\ a^2 \\ \cdots \\ a^N \end{bmatrix} s_N(k-1)}^{\text{inter-block memory}}}_{s_{\mathrm{ZIR}}} + \underbrace{\overbrace{\begin{bmatrix} 0 & \cdots & 0 & 0 \\ a & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots \\ a^{N-1} & \cdots & a & 0 \end{bmatrix} \begin{bmatrix} w_1(k) \\ w_2(k) \\ \cdots \\ w_{N-1}(k) \end{bmatrix}}^{\text{intra-Block memory}} + \overbrace{\begin{bmatrix} w_1(k) \\ w_2(k) \\ \cdots \\ w_N(k) \end{bmatrix}}^{\text{innovation}}}_{s_{\mathrm{ZSR}}} \quad (2.3)$$

for $k = 1, 2, \ldots$ where $k = \lceil \frac{n}{N} \rceil$ or $n = (k-1)N + l$ with $l = 1, \ldots, N$. In the later discussions we will see that the identification of the inter+intra block memory and the separation of the zero-state-response (ZSR) and zero-input-response (ZIR) of the synthesis filter provides means for better comparative analysis of EC-CELP and its special cases. In addition, separated ZSR and ZIR are commonly used in reducing the coding complexity in CELP. Eqn. 2.3 demonstrates that for a simple synthesis filter $\frac{1}{1-P(Z)} = \frac{1}{1-aZ^{-1}}$ and its current state at vector time instant k (filter memory from previous ZSR filtering in this case we have a scalar state $s_N(k-1)$), the k-th ZIR is generated by passing a *zero* vector of dimension N through this given filter. It also demonstrates that k-th ZSR is generated by setting the state (memory) of the synthesis filter ($\frac{1}{1-aZ^{-1}}$) to zero and then passing the k-th innovation vector $w(k)$ through this zero-state filter. The resulting memory will be used for the next ZIR filtering. We will give further details of ZIR and ZSR in section 3.

2.1. Information theoretic source coding bounds

Digital compression in general (but not necessarily) entails the introduction of some kind of coding distortion. The fundamental goal is to achieve the minimum possible distortion for a given coding rate, or equivalently, to achieve a given acceptable level of distortion with the least possible coding rate. There are usually factors of maximum allowable encoding delay and coding complexity constraining this goal. Furthermore the following practical considerations are of importance and make the achievement of the above constrained fundamental goal more difficult. First, practical source coding is usually at low bitrates which entails the effects of high quantization noise. The second factor is that the real world signals are rarely stationary.

The maximum achievable lossless compression [2] is set by the *Shannon noiseless source coding theorem*. It states that for a given source with entropy[3] H, using a sufficiently complex encoding scheme, it is possible to encode losslessly with an average bitrate arbitrarily (ϵ) close to H,

$$R_{\text{lossless}} = H + \epsilon. \tag{2.4}$$

From the definition of entropy one can easily see that for a given discrete symbol set $\mathcal{V}_I$ (I being the number of symbols) and the corresponding index set $\mathcal{I}$,

$$\mathcal{V}_I = \{v^{(i)}; i \in \mathcal{I}\}, \quad \mathcal{I} = \{1, 2, \ldots, I\},$$

[2] Ratio of average bitrate of raw data B to average bitrate of encoded data R in bits per sample (bps).
[3] For definition of entropy of other information notions see Appendix A.

the uniformly distributed set has the highest entropy rate $H_{\text{uniform}} = \log_2 I$. As a result, for the nonuniform set there is an associated redundancy which is exploited by the lossless coders. Lossless coding (variable rate) methods include Huffman, arithmetic, and Lempel-Ziv coding [6], [107], [6]. For applications which usually cannot accommodate variable rate fluctuations, buffering is required. There is a tradeoff between the associated cost and delay against performance [25], [86], [65]. We may describe the set of output prefix-free codes C_I and the corresponding lossless coding mapping Γ as

$$C_I = \{c^{(i)}; i \in \mathcal{I}\}, \tag{2.5}$$

$$\Gamma : \mathcal{I} \to C_I. \tag{2.6}$$

According to the more general fundamental rate-distortion theory, when we are only willing to spend an average bitrate less than H, we have to tolerate some average distortion D. For a given source, there exists the theoretical performance bound of rate-distortion function $R(D)$ (RDF), which may be computed. $R(D)$ sets the limit for coding efficiency, in a sense that for a given D, no coding scheme can do better than the RDF limit,

$$R \geq R(D). \tag{2.7}$$

For a sufficient delay and complexity and a given D, the average rate R can be made arbitrarily close to $R(D)$.

2.1.1. Theories for sources with memory

Eqn. 2.7 expresses the fundamental theoretical performance limit set by the $R(D)$ bound. For sources with memory, being more amenable to compression due to inherent statistical dependencies, based on N-tuple (rather than $N = 1$ in the memoryless case) source symbol mapping and corresponding average mutual information, the useful notion of N-th order rate-distortion function ${}^{N}R(D)$ is defined (see Eqn. A.33). For large N and stationary sources, ${}^{N}R(D)$ approaches $R(D)$,

$$R(D) = \lim_{N\to\infty} {}^{N}R(D).$$

In practical non-recursive quantizers, ${}^{N}R(D)$ is actually more meaningful than $R(D)$ (the N-tuple dependency is directly available). Note that if the source is memoryless, the rate ${}^{N}R(D)$ is independent of N. As seen from the preceding discussions, for a source $\{s(n)\}_{n=0}^{\infty}$ with RDF $R(D)$, we may unambiguously speak of corresponding memoryless source $\{s^{*}(n)\}_{n=0}^{\infty}$ with rate $R^{*}(D)$ (from here on subscript $*$ indicates whiteness or lack of memory) [108].

For the Gaussian sources, the N-th order RDF and its limit $R(D)$ are known and are given by the following equation pairs,

$$\text{White}\begin{cases} R^{*}(D) = \max\{0, \frac{1}{2}\log_2 \frac{\sigma_{s^*}^2}{D}\} = \begin{cases} \frac{1}{2}\log_2 \frac{\sigma_{s^*}^2}{D} & 0 \le D \le \sigma_{s^*}^2 \\ 0 & D \ge \sigma_{s^*}^2 \end{cases} \\ D^{*}(R) = 2^{-2R}\sigma_{s^*}^2, \end{cases}$$
$$\text{Correlated}\begin{cases} {}^{N}R(\phi) = \frac{1}{N}\sum_{k=1}^{N} \max\{0, \frac{1}{2}\log_2 \frac{\lambda_k}{\phi}\} \\ {}^{N}D(\phi) = \frac{1}{N}\sum_{k=1}^{N} \min\{\phi, \lambda_k\}, \end{cases}$$
$$\text{Correlated}\begin{cases} R(\phi) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \max\{0, \frac{1}{2}\log_2 \frac{S_{ss}(e^{j\omega})}{\phi}\}d\omega \\ D(\phi) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \min\{\phi, S_{ss}(e^{j\omega})\}d\omega, \end{cases} \tag{2.8}$$

where λ_k is the k-th eigenvalue of the correlation matrix ${}^{N}R_{ss}$ of the process S with power spectral density (PSD) function $S_{ss}(e^{j\omega})$. More details about the derivation of the above formulas are provided in Appendix A. The RDF results for the low bitrate region can be computed parametrically by varying $\phi > 0$. For GM(1) source with regression coefficient a, $\frac{1-a}{1+a} \le \phi \le \frac{1+a}{1-a}$. The high rate or small distortion cases of $D_\phi = \phi < \min\{\lambda_k\}$ and $\phi < \min_\omega\{S_{ss}(e^{j\omega})\}$, simplifies to the special cases of

$$\text{Correlated}\begin{cases} {}^{N}R(D) = \frac{1}{2}\log_2 \frac{\left[\prod_{k=1}^{N}\lambda_k\right]^{1/N}}{D} = \frac{1}{2}\log_2 \frac{{}^{N}Q}{D} \\ {}^{N}D(R) = 2^{-2R}\ {}^{N}Q, \end{cases}$$

$$\text{Correlated}\begin{cases} R(D) = \frac{1}{2}\log_2 \frac{\gamma_s^2\sigma_s^2}{D} \\ D(R) = \gamma_s^2 2^{-2R}\sigma_s^2. \end{cases} \tag{2.9}$$

${}^{N}Q \triangleq \mid {}^{N}R_{ss} \mid^{1/N}$ is the N-th order entropy power ($|\ .\ |$ indicating determinant), and γ_s^2 is the *spectral flatness measure* (SFM) for the process $\{s(n)\}_{n=0}^{n=\infty}$. SFM is defined as

$$\gamma_s^2 = \frac{\exp\left[\frac{1}{2\pi}\int_{-\pi}^{+\pi}\log_e S_{ss}(e^{j\omega})d\omega\right]}{\frac{1}{2\pi}\int_{-\pi}^{+\pi}S_{ss}(e^{j\omega})d\omega} = \frac{\eta_s^2}{\sigma_s^2}, \tag{2.10}$$

with $0 \leq \gamma_s^2 \leq 1$ and η_s^2 as the minimum prediction error variance. For white Gaussian process for which $S_{ss}(e^{j\omega}) = \sigma_s^2$, we get $\gamma_s^2 = 1$. Process non-whiteness (color), quantified by γ_s^2, represents its memory redundancy. The related factor of spectral redundancy (ideal memory) gain is the inverse of the SFM, $(\gamma_s^2)^{-1}$. As an example, for the GM(1) $\gamma_s^2 = 1-a^2$ [58], [7].

Comparing the correlated source and white source expressions, it is easy to verify the widely held belief that memory decreases the required bitrate when comparing two sources with the same given variance. For the high rate region, it is easy to verify that the RDF of GM(1) process S or $\{s(n)\}_{n=0}^{\infty}$, generated by innovation process $\{w(n)\}_{n=0}^{\infty}$, is the same as the non-redundant innovation process RDF (since we have $\gamma_s^2\sigma_s^2 = \sigma_w^2$). Also as seen next, the entropy and RDF of a source with memory is bounded by the entropy and RDF of related memoryless source. For GM(1) source, this can be demonstrated by the fact that, the higher the correlation, the lower the entropy or RDF curve. Note that for other sources, explicit formulations of RDF as for the GM(1) source are not available. However using the numerical algorithm of Blahut [8], algorithmic estimation of RDF may be obtained for many sources.

In [108], [46], [47], the following more general bounds for rate-distortion of sources with memory are provided. These bounds generalize the above conclusions (sometimes even for nonstationary sources). The above GM source case is a special case of these studies. The proposed bound is

$$R^*(D) - \Delta R_{\text{memory}}(N) \leq {}^{N}R(D) \leq R^*(D), \tag{2.11}$$

where

$$\text{High rate:} \quad \Delta R_{\text{memory}}(N) = \frac{1}{N} E \log \left[\frac{{}^N P_s(s)}{{}^N P_{s^*}(s)} \right] = h(S) - \frac{1}{N} {}^N h(S),$$

and where $h(\cdot)$ is the continuous or discrete entropy and $E(\cdot)$ is the expectation operator. For stationary memoryless (independent) N-dimensional vector, we have the following definition for the distribution ${}^N P_{s^*}(s) = \prod_{i=1}^{N} {}^1 P(s_i)$. The limit of the above as $N \to \infty$ ($\Delta R_{\text{memory}}(\infty)$) is also defined. For this limit-bound for some cases, including the GM(1) source, equality holds for lower bound (assuming high-resolution). For such case, the following relation with SFM can be easily verified

$$\text{High rate:} \quad \lim_{N \to \infty} \Delta R_{\text{memory}}(N) = \frac{1}{2} \log_2 \gamma_s^2.$$

Note that for the GM(1) source (with memory) the associated memoryless source (*) is the Gaussian memoryless source with the same variance.

A somewhat more general classification by Lookabaugh and Gray, formulated to quantify VQ advantages, includes other coding gains [80][4]. The overall gain of the N-dimensional VQ over SQ, using the relationship of distortion due to repeated SQ over distortion due to VQ, is denoted by $\Delta(N)$. In order to use the units of dB [5] we define $\Delta(N) = 10 * \log_{10}(\frac{D_{SQ}}{D_{VQ}(N)})$. The gain is subdivided to *memory, space filling,* and *shaping*. Assuming squared-error distortion [6] and units of dB for the RC quantization[7] we have

$$\text{High rate RC:} \quad \Delta(N) = \Delta_{\text{memory}}(N) + \Delta_{\text{filling}}(N) + \Delta_{\text{shaping}}(N). \tag{2.12}$$

Memory gain is directly related to the spectral memory gain, $(\gamma_s^2)^{-1}$, defined earlier. Lookabaugh and Gray used the Zador [110] equation for the expected distortion for the N-dimensional VQ and the Gersho's conjecture [40] for the coefficient of quantization for

[4] Also see the alternative classification of granular and overload distortion as summarized in [71].
[5] We will use the operator dB($\cdot$)=$10 * \log_{10}(\cdot)$.
[6] Throughout this work we use squared-error distortion.
[7] As mentioned in chapter 1, for the resolution-constrained (RC) quantizer design, the entropy coding in Fig. 1.2 is not assumed.

high rate VQ to show that the high rate RC and EC memory gain would be given by

$$\text{High rate RC:}\ \ \Delta_{\text{memory}}(N) = \text{dB}\left(\frac{\| {}^{N}P_{s^*}(s)\|_{N/(N+2)}}{\| {}^{N}P_{s}(s)\|_{N/(N+2)}}\right),$$
$$\text{High rate EC:}\ \ \Delta_{\text{memory}}(N) = \text{dB}\left(2^{2[H({}^{N}P_{s^*}(s)) - (1/N)H({}^{N}P_{s}(s))]}\right), \tag{2.13}$$

where $\|f(x)\|_\nu = (\int f(x)^\nu)^{1/\nu}$. For the Gaussian sources such as GM(M) source, the high rate gain for both RC and EC is shown to be [80]

$$\text{High rate Gaussian:}\ \ \Delta^{\text{Gaussian}}_{\text{memory}}(N) = \text{dB}\left(\frac{\sigma_s^2}{NQ}\right). \tag{2.14}$$

Comparing the above with Gaussian ^{N}RDF in Eqn. 2.9, the consistency of the above with RDF results is seen. The high rate memory gain for the GM(1) process is shown in Fig. 2.1.

Space filling gain is related to the dimensionality of VQ and is given by

$$\Delta_{\text{filling}}(N) = \text{dB}\left(\frac{C(1)}{C(N)}\right), \tag{2.15}$$

where $C(N)$ is the coefficient of quantization, as defined by Zador [110]. For high resolution, Gersho [40] has conjectured that $C(N)$ is determined by the optimum regular cell (Voronoi) shape for VQ matched to a uniform probability function. Based on Polytope N-dimensional centroid and volume, bounds and approximation for the filling gains of dimension N are provided in [80]. They suggest the following useful bounds (the detail of which can be found in [80]): the spherical bound (approximation based on a partitioning of input space using spheres), the Conway and Sloane conjectured bound [15], the known lattice lower bounds, and Zador lower bound. We reproduce these high rate bounds for filling gain in Fig. 2.2 [80]. Comparing the maximum filling gain with memory gain in Fig. 2.1, it is obvious that for the highly correlated GM(1) source, the memory gain is the more significant of the two.

Finally, as mentioned in chapter 1, shaping gain is related to the possible coding gain due to non-uniformly distributed source symbols (entropy or lossless coding provides this

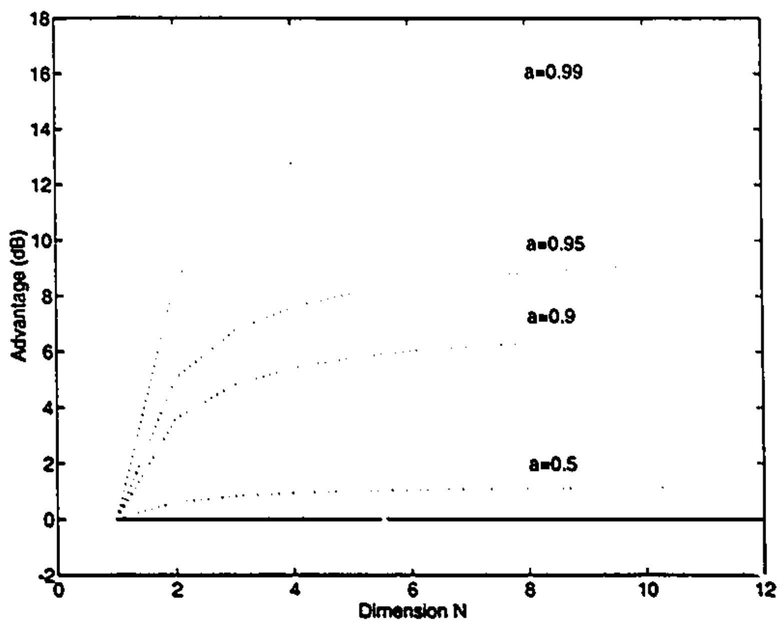


Fig. 2.1 High rate memory gain (VQ advantage over SQ) for GM(1) source with coefficient a.

gain). For the EC system analysis we need not to be concerned with the shaping gain (shown to be zero) due to the use of entropy coding. Hence we will have

$$\text{High rate EC:}\;\; \Delta(N) = \Delta_{\text{memory}}(N) + \Delta_{\text{filling}}(N). \tag{2.16}$$

The rigorous analytic expressions for the above gains at high rates, although informative, are not suited for low bitrates. This is mainly due to the limits induced by RDF "water-filling" concept at low bitrates. Since the above categorizations are powerful tools and extremely useful as coding performance reference, in chapter 4 we develop alternative formulation for these gains at low bitrates. The formulation incorporates dependence on bitrate as well as dimension N. Such low bitrate categorization provides a reliable reference for low bitrate coding and a better understanding of the advantages of EC quantization at low bitrates.

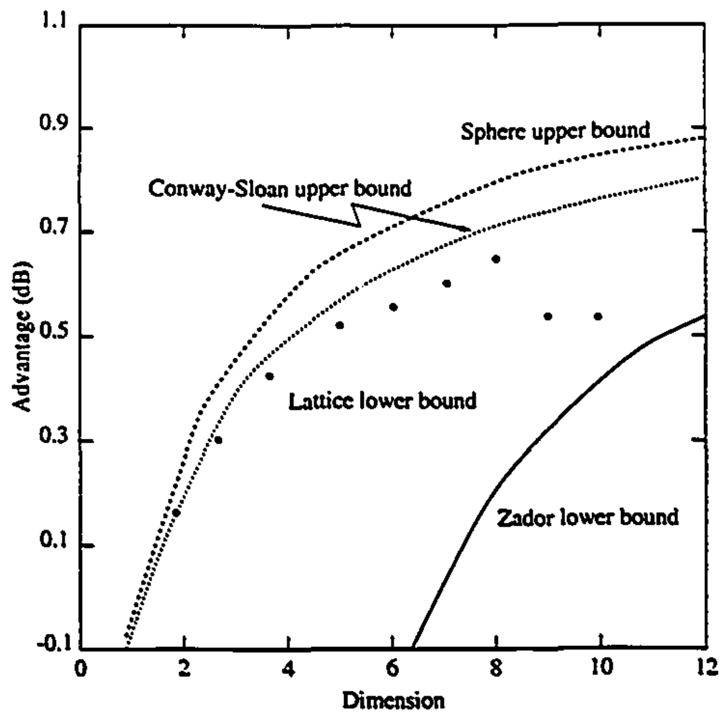


Fig. 2.2 Bounds on high rate VQ space filling advantage.

2.2. EC history from EC-SQ to EC-CELP

In this section, we present the history and context of EC-CELP and its elements. It was mentioned in chapter 1 that to combine the benefits of lossless coding with the lossy coder, one direct approach is to use a configuration in which the lossy coder (quantizer whose output symbols are not usually uniformly distributed) is followed by the lossless (entropy) coding to obtain a more compact signal representation to be transmitted over the channel or storage medium. The formal definitions for the two classical quantizer design alternatives can be given as follows.

Definition 2.1. *In a quantizer design, if the criterion of minimization of average distortion is for a fixed size of quantizer output alphabet, the design and quantizer are referred to as alphabet size or (rate) resolution-constrained (RC).*

Definition 2.2. *In a quantizer design, if the criterion of minimization of average distortion is subject to the constraint that the entropy of the quantizer output alphabet be below* *a certain level, the design and quantizer is referred to as entropy-constrained (EC).*

The EC class has certain well-known advantages [44], [110], [26]. Various numerical schemes for the EC zero memory SQ (EC-SQ) design have been reported. In particular the EC-SQ design algorithms by Wood, Berger, and Farvardin and Modestino [26] are available. The classical results of Gish and Pierce [44], under mild conditions, show the asymptotic optimality of the uniform EC-SQ at high rates. The performance gap with $R(D)$ was shown to be within 0.25 bps. Since then it has been proved that at low bitrates, neither uniform quantizer optimality nor the 0.25 bps gap hold [26].

The necessary condition for the optimality of EC-SQ algorithm for the general distortion measure case is given in [26]. Using the properties of EC quantizer and its relationship with RC (Lloyd-Max) quantizer [41], the tedious EC-SQ design problem can be reduced to a search among quantizers satisfying the necessary condition with an output entropy equal to a fixed level. The EC design algorithm which is a descent (Lagrangian) algorithm uses the known fact that the average distortion of EC quantizer is a non-increasing function of output entropy.

Other than zero-memory SQ, another important class of source coders, directly related to the fundamental theories in information theory, is the class of delayed decision coders (Tree, Trellis, and VQ). The Tree and Trellis quantization schemes can be considered as special class of general VQ [41]. Their merit is in providing coding gains (efficiency of representation) for sources with memory as well as for memoryless sources (e.g. *i.i.d.* Gaussian source). It was Shannon who showed that performance approaching the rate-distortion theoretical limits is possible by quantization of blocks (VQ) of source symbols instead of the individual source symbols (SQ). To use a broader, more suitable terminology, we use the N-dimensional *general VQ* which includes many VQ classes. Gersho and Gray [41] have classified the general VQ to classes of *constrained VQ* (e.g. product VQ, multistage or residual VQ, transform coding VQ), *recursive VQ* [63] (e.g. predictive VQ, finite-state VQ), and *adaptive VQ* (e.g. gain adaptive VQ).

Chou *et al.* [14] made a Lagrangian extension of the EC-SQ algorithms to entropy-constrained VQ (EC-VQ). The iterative descent minimization algorithm for EC-VQ has similarities to the Linde-Buzo-Gray (LBG) algorithm [78] and in the case of the memoryless source is guaranteed to converge to a local minimum.

In the EC class of coders, if there were no constraint on the VQ dimension N and codebook size I, the EC-VQ may be considered as the ultimate coding scheme. This is due to Shannon source coding theories, that in effect says that for sufficiently large N, there exists a VQ coding scheme with a codebook size I, which yields a rate that is arbitrarily close to RDF. The high rate VQ memory and filling gain for EC-VQ were given in Eqn. 2.16.

For sources with memory, the required delay (block size N) and complexity for VQ can increase considerably and may not be practical. Means to address search complexity have been provided at the cost of some degradation through the class of *constrained VQ* [5] and *structured VQ* [41]. The delay requirements for satisfactory performance could still be stringent. This is why recursive procedures of differential or predictive coding (PC) such as DPCM are used in removing memory redundancies.

The numerical design algorithm of EC-SQ [26] or EC-VQ [14] set the path for the design algorithms devised for EC-DPCM, EC block transform quantization (EC-BTQ) [24], EC Trellis quantization (EC-TCQ) [28], [82], EC predictive Trellis quantization (EC-PTCQ) [28], EC residual VQ (EC-RVQ) [68], EC predictive VQ (EC-PVQ) [64], and in this thesis EC-CELP.[8] The results of EC-CELP formulated in this thesis were first published in [32]. As well, the 2-dimensional (2-D) variations of these EC coders have already been formulated for image coding. Examples of recent papers for 2-D EC coders include 2-D EC-PVQ [87], 2-D EC subband coding (EC-SBC) [65], EC mean-gain-shape VQ (EC-MGSVQ) [77], and EC frame-work for wavelet-based image coding [96]. Other than video coding application in chapter 5, we have also studied speech coding application of EC-CELP in [35] (extension to this thesis). Parts of the results of chapters 3 and 4, devoted to EC-CELP design and analysis, are published in [32], [33], [34].

Finally, we would like to motivate the EC-CELP work through looking at the EC-CELP elements. As will be seen in detail in chapter 3, CELP uses PC and VQ. PC combined

[8]Many of these EC coders, appearing quite recently and sometimes in parallel, use the Lagrangian approach of EC-VQ.

with VQ belongs to the larger class of recursive VQ. Effective prediction of the signal in the case of stationary or slowly time-varying signals (locally stationary) is possible. This is done by the good estimation of predictor model parameters. In such cases, PC provides higher memory gain than other alternatives, particularly the non-recursive schemes such as TC [81], [58]. Other coding configurations which also use PC but can be considered as special case of CELP are DPCM and PVQ. Note that TC and other recursive schemes such as RVQ belong to the constrained VQ class which are devised to deal with complexity issues of VQ.

CELP also belongs to the class of adaptive VQ due to its analysis-by-synthesis feature and other adaptive components. The closed-loop analysis-by-synthesis structure of CELP easily lends itself to adaptation of memory removal procedure for the slowly time-varying signal. In chapter 3, we describe the above elements in more detail. Among other things we show that even for stationary signals, non-adaptive CELP has a PC memory advantage over PVQ. For the nonstationary input signals, the advantages are more eminent. We also briefly discuss the low bitrate quantization noise effects. More rigorous analysis of such effects are presented in chapter 4. In the absence of shaping gain due to entropy coding, at low bitrates where the coding gains are scarce and interactive with quantization noise, a joint coding configuration using VQ and EC design becomes preferable. This becomes the motivation for the EC design in EC-CELP. In EC-CELP, the capabilities of VQ can be concentrated on the space-filling gain and the remaining source and quantization memory redundancies. The reasons for the more efficient near rate-distortion performance of EC-CELP than the previous methods principally lie in the better joint memory/space-filling gains in the presence of quantization noise feedback. High coding efficiency obtained is both in the sense of rate-distortion and the imposed low delay (VQ dimension) and low complexity constraints.

2.3. High compression video coding

For time-varying image sequence compression, classical and modern source coding theory and the knowledge of the source characteristics and modeling are used. Various techniques have been suggested in removing the video signal temporal and spatial redundancies with impressive results. Nevertheless, higher quality compression schemes operating at lower bitrates are still in demand [54]. As a result, there has been an intensified activity in video communication research particularly in this decade. Current and emerging technologies and trends such as video telephony, multi-media, and wireless are the driving force behind such demands. Video quality in different applications varies and ranges from large size high quality HDTV to smaller size lower quality video telephony.

High compression video coding usually, but not necessarily, means very low bit rate applications. Very low bitrate time-varying image sequence standards and initiatives such as H.261 and MPEG-4 have been underway [88], [53], [56], [54], [69], [74]. The H.261 standard (also referred to as P×64 where P is from 1 to 30) coders already produce good low bitrate coders for ISDN telephony application. But for applications such as the basic video telephony application[9], considerable research is still in progress. In particular, the initiative under the name of MPEG-4, with target video bitrate of 4.8–64 Kbpsec. and a time schedule between 1993 and 1998 has been very active. Table 2.1 summarizes the format and parameters of some possible very low bitrate configurations [2], [54] [10]. For this thesis too, the target coder would be similar to the ones targeted by the MPEG-4, with rates in the order of 10 Kbpsec. Such high-compression coders must produce high quality coded video at low bitrates of less than one bit per source symbol (bps). Note that for the case of MPEG-4 targets with rates above 4.8 Kbpsec., the bps is given in Table 2.1. To make a meaningful interpretation of the bps, alternative configurations in various coding schemes such as allocation of higher bps for reference frames (for which no temporal redundancy exploitation is done) have to be taken into account.

The three industries of telecommunications, computer, and TV/film traditionally have not had much interaction. Now an emerging intersection between these three industries is the focus of MPEG-4 activity [55]. Such intersection entails mixed elements that have

[9]The channel is the basic twisted wire telephone line with 3.3 KHZ bandwidth. The corresponding current modem rates over such channel and current speech coding technology allow for a video rate of around 10 Kbpsec.

[10]Note that for our purposes, rates should always be compared for a given format. Alternatively, one may consider down-sampling in format as part of rate reduction.

format	resolution	raw rate Mbpsec.	MPEG-4 rate Kbpsec.	compression	bps
QCIF	$176 \times 144 \times 30 \times 8+$ $82 \times 72 \times 30 \times 8 \times 2$	9.1	64–16	142–569	0.056–0.014
AT&T 2500	$128 \times 112 \times 10 \times 8+$ $32 \times 28 \times 10 \times 8 \times 2$	1.3	4.8	269	0.013

Table 2.1 Examples of format, raw, and target MPEG-4 rates, and other parameters for very low bitrate time-varying image sequence coders. Resolution parameters are $B = 8$bps and spatio-temporal dimensions for luminance and color components. MPEG-4 target rate is 4.8–64 Kbpsec.

historically belong to individual industries. This poses new challenges which include trends such as interactive multi-media TV/film, multi-media in interactive computer, and finally audio-visual wireless. A preferred multi-resolution video quality, has prompted research to match this preference. Finally, if video communication is to become reality for the wireless channels, the characteristics of wireless channels and required robustness have to be taken into account.

In the next three subsections we look at the spatial and temporal video signal characteristics from the perspective of video compression. More complex multi-frame modeling of image sequence intensities along motion trajectories (MTs) is devised and the advantages of such models over conventional methods are shown. The hybrid DPCM, the common video coding configuration mentioned in chapter 1, is described. An overview of some of the more advanced coding schemes provides the chronological context for the proposed method of this work.

2.3.1. Image sequence characteristics and modeling

Let us assume that the video input to the source encoder is fully digital. Also without loss of generality, let us focus on the predominant video information of luminance component[11] and assume a progressive scanning and sampling structure on a regular lattice grid [17], [18]. In particular scanning format of QCIF in Table 2.1 can be used.

Let us denote $\boldsymbol{x} = (x, y)$ as the spatial coordinate of the image point. When explicit

[11] from a luminance-chrominance color space video signal or a black and white video signal.

dependence on time t is needed, we will use $(\boldsymbol{x}(t), t) = (x(t), y(t), t)$. If the sampling structure is represented by $\Lambda_{\boldsymbol{x}t}$, we can use the discrete image sequence signal notation

$$g(x, y, t), \quad (x, y, t) \in \Lambda_{\boldsymbol{x}t}. \tag{2.17}$$

For the QCIF luminance format $\Lambda_{\boldsymbol{x}t} = \{(x, y, t) : 0 \geq x \leq 175, 0 \geq y \leq 143, t = 0, \cdots\}$. When only temporal notation is required, the spatial coordinates are eliminated and $g(t)$ notation is used. We also use

$$\boldsymbol{g}(t) = \{g(\boldsymbol{x}, t); \boldsymbol{x} \in \Lambda_{\boldsymbol{x}}\} \quad t \in \Lambda_t, \tag{2.18}$$

referring to a single image frame, with bold face indicating a vector.

The digital video signal $g(\boldsymbol{x}, t)$ temporal and spatial domains have different characteristics. The coding configuration should take the difference and interaction between these two domains into consideration. The important feature in spatial domain video signal is the scene edges. In many traditional methods, square blocks of image are processed. As a result at lower bitrates due to higher quantization noise, subjectively intolerable distortions commonly referred to as "blockiness" will appear. Traditional methods, for the sake of of simplicity, have not benefited from the more complex edge-based (region-based) spatial modeling and coding methods. Region-based video compression schemes have received most of the attention and focus of the recent research activities and obviously can be extremely beneficial. However, since the main focus of the video coding aspect of this thesis is the temporal domain, we will suffice to a brief review of the spatial methods during the next sections. We do this to focus and provide a more detailed treatment of the temporal domain video signal characteristics and modeling.

In the case of temporal domain, the characteristics of the motion plays the important role. Apparent motion in the image sequence is the result of either camera motion, motions in the scene, or both. In video-phone applications, the camera is fixed and the apparent motion (ordinarily low) is due to the person movements. In surveillance applications, most of the apparent motion is due to camera movement (except during an "incident"), while in sports scenes both camera and scene movements may be highly active [19]. The camera motion includes zooming, panning, and other motions. The objects in the scene may have motion, described by translation, rotation and more complex deformation operators. To make this more realistic, one should include light variations and the noise introduced by the camera, sampling, etc.

MTs trace out the projection of scene points in the image plane during the time they are visible in the image. The initial and final frames (t_i and t_f) define the visible period and correspond to the appearance and disappearance of the pixel point in the image sequence. The initial and final points not only happen at the picture frame edges, but also take place when a point is *newly exposed* or *occluded* [18]. Two examples of MTs are shown in Fig. 2.3. Similar to the notation for the spatial coordinates of the image point $\boldsymbol{x} = (x(t), y(t))$, the trajectory coordinate in the 3-D xyt space is denoted by $(x(t), y(t), t)$. The function $\boldsymbol{c}(\tau; \boldsymbol{x}, t)$ describes the 2-D trajectory or the spatial coordinate at time τ of an image point which was at location $\boldsymbol{x}$ at time t. The 3-D trajectory $(\boldsymbol{c}(\tau; \boldsymbol{x}, t), \tau) = (x(\tau), y(\tau), \tau)$ has a unique mapping to the above 2-D function and is defined during the time of visibility in the image sequence. It provides the spatial location of an image point at time τ.

The intensities along MTs which are highly correlated can be modeled as a stationary GM(1) process with $0.95 \leq a < 1.0$.

Motion compensated coding (along the MTs) is an efficient technique in removing temporal redundancies in image sequence coding. Many qualitative (e.g. [53]) and some quantitative arguments [89], [43] have been made to support motion-compensated coding advantage over the alternative. Preferably MT estimation has to be incorporated into the coding scheme or use a measure of rate-distortion (objective or subjective) performance criteria. The idea behind the advantage of motion-compensated coding is that in an ideal situation, given MT parameters, only the MT information and the initial frame need to be coded. The idealization includes simple motion transformations, robust and accurate motion parameter estimation, and low information content of MT parameters. In reality above assumptions are violated and hence there would be residual motion-compensated *intensity* to be quantized and transmitted. As well, the required bitrate for transmission of

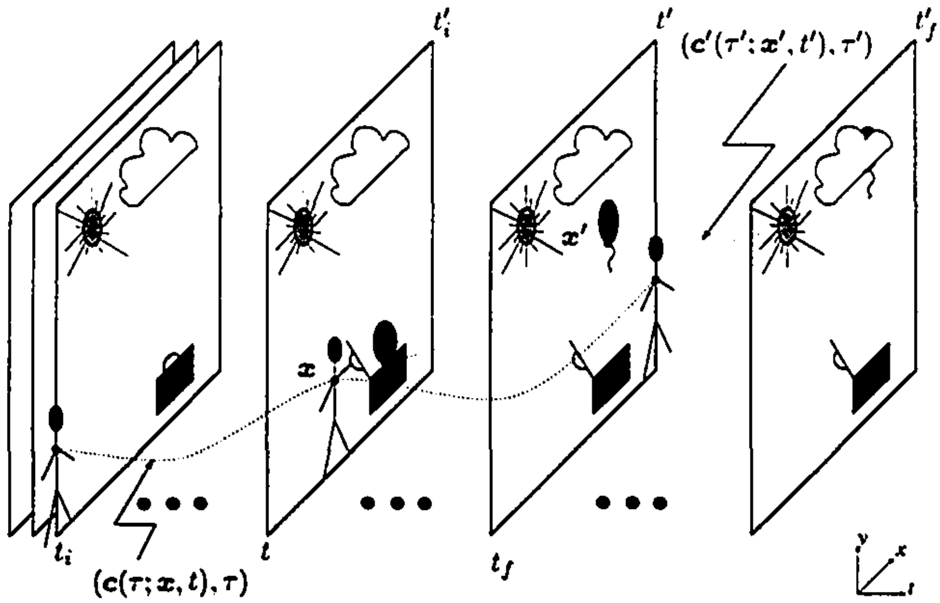


Fig. 2.3 Two motion trajectories are shown. One trajectory lasts from the initial time t_i to the final time t_f, the period during which the point is visible in the picture frame sequence. The second trajectory from t'_i to t'_f traces out a newly exposed point up to the time when it is occluded.

motion parameters can be substantial. It is obvious that the total average bitrate not only includes the bitrate allocated for the coding of intensities but also the side information allocated for the motion parameters

$$R = R_{\text{intensity}} + R_{\text{motion parameters}}. \tag{2.19}$$

As seen from the above, there could be situations where the motion-compensated coding is not advantageous. Such situations arrive when the $R_{\text{motion parameters}}$ exceeds the saving in the bitrate due to motion-compensated coding.

As was pointed out in reviewing source coding principles in section 2.1.1, a highly random source is more difficult to code than a source with memory. The poor MT estimation may induce random noise which would waste parts of the precious available bitrate. The tradeoff between overhead MT information rate and saving in bitrate as a result of motion-compensation (Eqn. 2.19) is an important consideration which needs future investigation. For simplicity, in this work the MT estimation and coding are separate and not jointly done as suggested above. Further description of MTs and MT estimation is presented in chapter 5.

2.3.2. Conventional techniques

The motion-compensated hybrid DPCM configuration video coding has remained popular in recent standardization (H.261 and MPEG-1/2) and research activities. Even among many emerging MPEG-4 coders, underlying assumptions beyond this configuration are kept and some elements of hybrid DPCM configuration are used without major modification. The basic block diagram of this configuration is shown in Fig. 2.4. To show the advantages and limitations of this configuration, we now briefly describe coding elements and basic assumptions leading to such techniques.

Without considerations with respect to the important spatial feature, namely edges, this configuration uses a simple block-based TC to remove spatial correlations. The choice of TC method to remove spatial redundancies is often DCT. At least at higher rates (low quantization noise) good quality coding at low computational costs are produced. DCT belongs to a more general class of unitary transforms. Unitary transformations represent the signal vector in terms of a discrete set of basis functions. The signal vector is only rotated and hence energy is conserved. This is without any loss of information (conservation of entropy). The two most important characteristics of the transformed signal is energy compaction and decorrelation. The tendency of packing a relatively large fraction of the average signal energy into relatively few components is what is referred to as energy compaction. Fine quantization of high energy components and coarser quantization of low energy components results in an efficient use of the available bitrate (assuming proper bit allocation) and a better distribution of distortion. The decorrelation property means that the off-diagonal elements of the covariance matrix of the transformed signal are relatively small and hence the signal is almost decorrelated. In the Karhunen-Loeve optimum transform (KLT) (optimum in a mean-squared sense), the maximum average energy is packed in a given number of coefficients and the signal is completely decorrelated. The choice of TC not only depends on the above considerations, but also on the availability

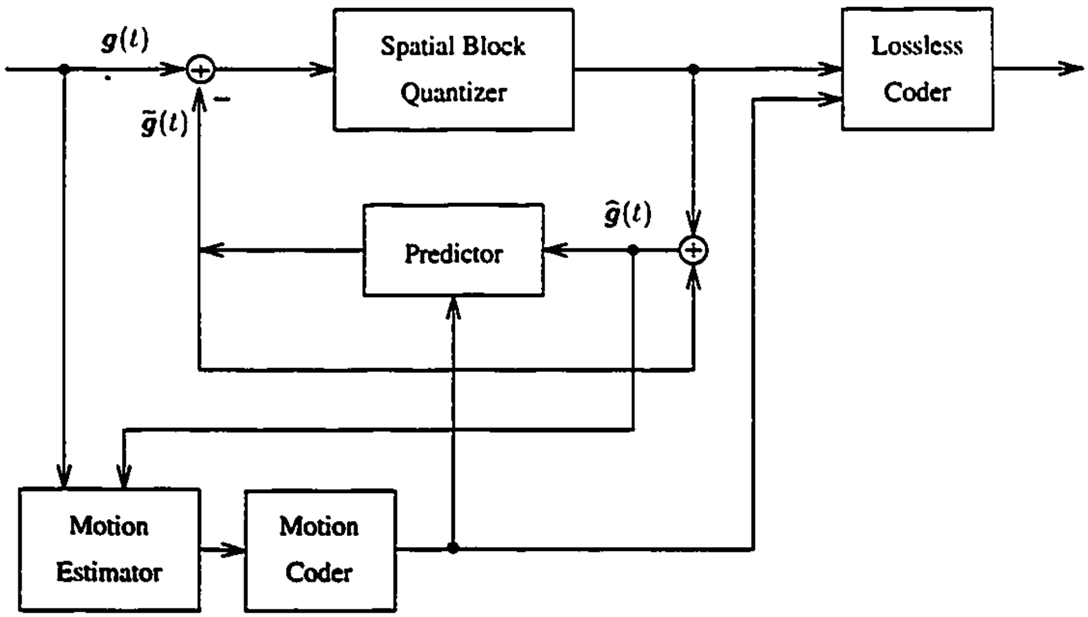


Fig. 2.4 Motion compensated DPCM configuration coder.

of fast algorithms [57], [58]. In hybrid DPCM-TC coding of Fig. 2.4, a VLC lossless coding follows the spatial DCT transformation and quantization. In the temporal domain, the common hybrid configuration incorporates the motion redundancies by using a motion compensated DPCM scheme. However, it assumes a relatively simple motion characteristic and modeling. Spatial blocks of image along the direction of motion for two neighboring frames are matched. The motion estimation of choice hence is the well known *block matching*, possibly using one of the existing fast algorithms [98], [79]. The fact that both spatial coder and motion estimation are block-based, makes the signal processing less complex.

The motion estimation block in Fig. 2.4, produces the description of *motion vector displacement fields* from frame to frame (special simple case of MTs described earlier). The displacement field description is used to obtain the motion-compensated neighboring frame in the predictor of the DPCM coder. The estimation could be either based on the reconstructed signal or, as often is the case, on the original signal. To make the motion information available at the decoder, losslessly encoded motion parameters are also transmitted.

In MPEG-2, in addition to PC (DPCM), inter-frame interpolation is used to further increase the temporal domain compression. The lossless coding techniques of Run-Length and Huffman coding are normally applied.

Using the QCIF format and the hybrid DPCM configuration, a H.261-like coder can only produce acceptable quality at minimum bitrate of around 90 Kbpsec. As one example of lower bitrate video coding using the hybrid coding configuration of Fig. 2.4, the AT&T's 2500 video-phone obtained reasonable quality but with substantial spatial-temporal resolution reduction [21] (Table 2.1). To obtain higher quality with acceptable resolution and without fundamental changes in the coding configuration, attempts in reducing the bitrate, by say one order of magnitude or more, have produced mixed results. Hence, as MPEG-4 study suggests, substantial innovations and higher quality advanced techniques are required to produce high quality video coding at very low bitrate.

2.3.3. Advanced techniques

Most of the conventional methods assume medium to high rate, and hence the effectiveness of these schemes at very low bitrates has to be reexamined. Until recently, image sequence models have been kept simple to comply with the practical constraints imposed by the available technology. Alternative schemes would emerge from a closer study of the temporal and spatial video source characteristics and more complex models. As well, better low bitrate source coding schemes and configurations have to be designed to match such models. The results of many new image coding techniques have appeared in publications during the course of this work. In this introductory section we include references to some of the more relevant works.[12] Such intense research activity also demonstrates the importance of much needed advanced techniques.

Most of the more recent attempts in providing higher compression have focused on spatial domain modeling and compression techniques. As part of the current MPEG-4 and other low rate coders, new directions of morphology, fractals, model based (regional) spatial coding, and alternative spatial source coders such as VQ, subband, and wavelet

[12] Most of the research work referenced appeared before the completion of the thesis work and first draft in December 1994. In this final draft some later references were also added.

coding are considered [76] [85]. Multi-resolution modeling and description have been particularly popular in many of these recent techniques.

Among efforts in reexamining the temporal domain aspects, there has been some success in improving usage of motion effects [52], [76]. Better motion estimation techniques are devised to increase motion accuracy and to match other components of the new configurations [102].

Although many of the recent new video coding configurations still use the temporal DPCM coding component of the conventional configuration, its efficiency for higher compression coding has to be questioned. The departure from the scalar nature of temporal DPCM coding leads to *multi-frame* video signal modeling and coding. Although multi-frame video coding has been around for some time, its practical use involving considerations with regard to the choice of models and suitable motion compensation and coding techniques are fairly recent research topics. When using multi-frame configurations, it should be noted that we are also constrained by the maximum delay requirements (100-150 msec.). This kind of delay is typical of the ones imposed in most low bitrate situations [56] [74]. Also note that in the conventional schemes, to exploit redundancies beyond two frames, of course one alternative already in use is the use of interpolation [53].[13]

Non-recursive multi-frame coding. Although 3-D spatio-temporal video coding was suggested as early as 1977, its success and popularity has been limited until recently. This is due to the less than acceptable tradeoff offered between complexity and quality and also the available computing power. New focus of the more recent 3-D video coders has been on the non-recursive techniques such as 3-D subband coding [11], [97], [94]. Some of these techniques also include motion compensation. Among other suggested variations to the conventional configurations is the use of VQ [11], [72].

Recursive multi-frame coding. Non-recursive (e.g. TC, VQ, subband) multi-frame video coding has certain disadvantages. These disadvantages are due to the fact that the temporal domain signal is highly correlated. For such signals, high-compression using non-

[13] As an example of the new image sequence modeling using the multi-frames see [9].

recursive coding requires long delays (number of frames). From the high rate memory gain results in Fig. 2.1, one may conclude that to obtain good memory gain, the required size for the non-recursive temporal block will be more than ten frames. The proposed scheme in this thesis which is based on EC-CELP falls into the alternative approach of *multi-frame recursive* temporal coding. To our knowledge, there has not been other video coding schemes which suggest the use of such approach. The motivation for such an approach can be best expressed by the above mentioned impractical delay requirement of multi-frame non-recursive schemes and by the study of expected saturation of the DPCM (single frame recursive) alternative at low bitrates. At very low bitrates, the rate-distortion performance of DPCM operating on the highly correlated GM model degenerates substantially. Later in this thesis, we will examine the exact limitations of DPCM coding for such correlated sources. To deal with this problem, we first propose improving the performance of the scalar DPCM configuration by adopting an EC-DPCM design strategy (scalar EC-CELP). Next, we may further improve the coding performance by increasing the number of frames (multi-frame EC-CELP coding).

To obtain higher video compression, in addition to the proposed new temporal domain schemes in this thesis, as suggested by the MPEG-4 studies and the spatio-temporal system in chapter 5, the spatial coding efficiency of the conventional configuration has to be reexamined [54]. As well, the utilization of higher performance lossless coders such as Arithmetic coding may be beneficial. In our coding configuration, we will use the higher quality dense motion compensation. However as mentioned earlier, the tradeoffs of motion and intensity information, reflected in Eqn. 2.19 needs future examination. As well, more efficient representation of motion parameters will be beneficial (e.g. [90]).

Chapter 3

Entropy-Constrained Code-Excited Linear Predictive (EC-CELP) Quantization

3.1. Introduction

As a classical source coding task, it is desirable to obtain high quality low bitrate quantization of (nonstationary) sources with memory while meeting practical constraints of low delay and complexity. The criteria in the above task are in a sense contradictory. For example, Shannon source coding theories suggest that, given sufficiently long delay (dimension) and complexity, block quantization or VQ (EC and RC) can provide performance near the rate-distortion. Practical source coders hence search for a tradeoff point and in effect attempt to deliver the best performance for some practical delay and complexity. Two important considerations which determine the outcome of this quest are the choice of coding configuration and the method of quantizer design. The new method of EC-CELP quantization, introduced in this chapter, offers excellent results with regard to the aforementioned tradeoff. In fact, as will be seen in chapter 4, among a certain group of quantizers and for the GM source, it has the best potential. As mentioned above, the choice of coding configuration, namely entropy-coded CELP and the method of EC design are the key considerations which make this possible.

EC-SQ is the simplest and the earliest EC quantization scheme and its advantages at high rates and for memoryless sources are well known. There are also a variety of algorithms for the design of this EC zero-memory quantizer [26] (section 2.2). For the more practical cases of sources with memory and low bitrates, both EC-DPCM [27] and EC-VQ [14] have been suggested. For the stationary GM(1) source, the advantage of EC-DPCM is its single symbol delay, a consequence of SQ. The recursive nature or use of PC in EC-DPCM results in efficient memory removal performance. However EC-VQ can deliver an overall higher performance provided sufficient delay, thanks to the filling and memory advantages of VQ. To combine the advantages of VQ and recursive schemes, EC-CELP is an excellent general method of choice and as will be seen, it has advantages over its special cases EC-PVQ [87], [64] and EC-DPCM [27].

Similar to other EC coders, EC-CELP suffers from problems associated with variable-length code (VLC) transmission over fixed channels. Buffer overflow, underflow, and delay as well as performance degradation in the presence of channel errors are among such problems. However, in dealing with such draw-backs, it is expected that, through similar techniques used for other EC schemes (e.g. [87]), the resulting degradations can be dealt with and minimized. Better analysis of these issues for the EC-CELP will be the subject of future research.

Temporal video coding is the focus application of this dissertation, where a highly correlated GM(1) source models the intensities along motion trajectories (chapter 5). Due to its formulation, EC-CELP can also be particularly beneficial for nonstationary signals such as speech [35].

In this chapter, unless otherwise indicated, a lossy+lossless coding configuration, inherent to EC quantizers, is assumed. In section 3.2, we introduce the lossy CELP coder as part of the more general class of *adaptive* and *recursive VQ*. At the same time, CELP is presented in a fashion that will include two other important practical recursive coders, namely PVQ and DPCM, as special cases. Advantages of CELP due to the choice and configuration of its main components, PC, VQ, and analysis-by-synthesis are explained.

In section 3.3, the EC design algorithm of EC-CELP along with its special cases is presented. The EC-CELP design algorithm can also furnish new, good, and compact design algorithms for its special cases of adaptive EC-DPCM and EC-PVQ with advantages over previously published algorithms. Section 3.4 contains the simulation results and the concluding remarks. Previous EC work and advantages of EC-CELP over these schemes are given during the chapter presentation. Section 2.2 which already surveyed previous EC quantizers and the history of EC quantization should be consulted for additional references.

3.2. Entropy-coded CELP, adaptability and memory gain

CELP has undoubtedly been the most successful speech coding configuration in the last decade [101], [13]. Recently 2-D CELP was also applied to image coding [91], [50], [1]. In [90], a video coding application of CELP for coding the motion information is investigated. As was mentioned in section 3.1, we may emphasize the nature of CELP by calling it an (adaptive) analysis-by-synthesis PVQ or by referring to its class as recursive and adaptive VQ.

The purpose of this section is to compare the minimum delay (measured in terms of signal dimension N) memory gain and signal adaptation characteristics of the CELP configuration with other alternatives which use PC (recursive) and TC (constrained) VQ for memory redundancy removal. First we present the CELP configuration as a general scheme in a fashion that easily shows that DPCM and PVQ are special cases of CELP. This representation alone can give an intuitive and general characterization for the minimum delay memory gain advantage of CELP over the special cases as well as the constrained VQ alternatives. The robust adaptation characteristics of CELP are also discussed. We will use the GM(1) source in Eqn. 2.3 as the example source. To support the intuitive conclusions, we present the simulation results for the rate-distortion performance of the entropy coded CELP coder without EC design (first published in [30]). EC consideration and advantage will be considered in the following sections. A more detailed analytical analysis is postponed to chapter 4.

The CELP coder elements and features can be summarized as follows:

- o implicit use of analysis-by-synthesis PC (inter-vector and intra-vector PC),
- o implicit use of VQ,
- o closed-loop analysis-by-synthesis configuration,
- o (optional) far-sample prediction filter (used for pitch in speech),
- o (optional) adaptation of prediction filter (backward or forward),
- o (optional) gain scaling and adaptation, and
- o (optional) perceptual weighting (noise-shaping).

In this work we include the features which relate to the coding theory perspective for sources with memory. In the above list of features, one consequence of closed-loop configuration is in the sense of PC being based on reconstructed signal rather than unquantized signal (as in classical DPCM versus D*PCM [58]). Sometimes, the terms closed-loop search and analysis-by-synthesis are used interchangeably. Analysis-by-synthesis or the so called trial-and-error approach means that the (usually) exhaustive search through the excitation codebook uses a copy of the decoder at the encoder. It is also the analysis-by-synthesis which allows for intra-vector PC.

In [63], [41], the closed-loop or feedback PC is presented in a larger class of recursive coders. Such a general class of recursive VQ can be depicted as in Fig. 3.1.[1] We will be presenting CELP as a member of this class. Later we will see that in order to represent and interpret the standard quantizer used by CELP in this general configuration, we would have to use an *effective quantizer*. This is not the case for the DPCM or PVQ coders, where the standard quantizer is a direct SQ or VQ. By a standard quantizer, we simply mean a quantizer which maps from a well-defined or explicit input signal to an output signal. From the source input process $\{s(k)\}_{k=1}^{\infty}$, the encoder produces the output index sequence (channel symbols or inputs to the entropy coder in case of EC coders) $\{i(k)\}_{k=1}^{\infty}$

[1]Throughout this work, an error-free channel is assumed.

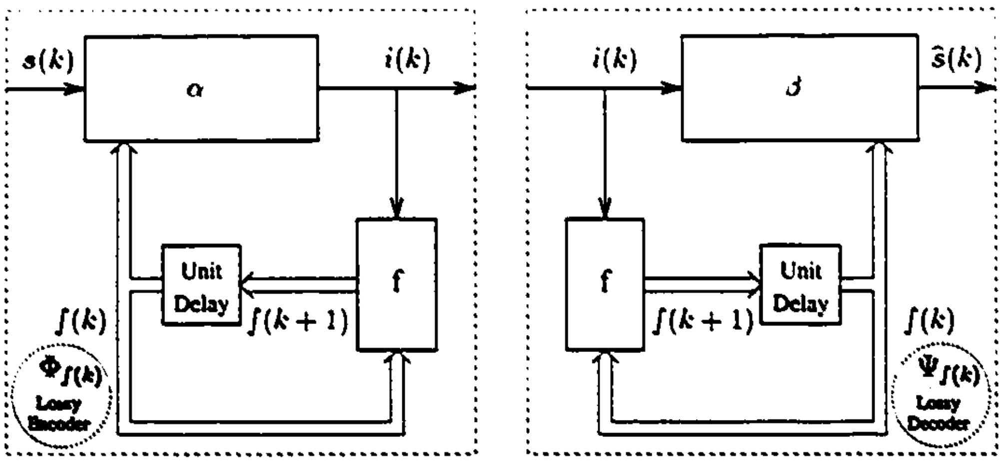


Fig. 3.1 Recursive VQ lossy encoder and decoder.

by means of a state sequence $\{f(k)\}_{k=1}^{\infty}$. States describe the lossy encoder behavior. The decoder produces the reproduction output sequence $\{\hat{s}(k)\}_{k=1}^{\infty}$ moving through the same state sequence as the encoder $\{f(k)\}_{k=1}^{\infty}$, provided common initial state and *state transition* function as the encoder. The state of encoder and decoder at each time is determined from the previous state and previous index output.

Let us assume that the input process values are in the set $\mathcal{S}$ and the states are in a certain metric space Ω. As well, the *state transition* function f, the lossy encoder mapping α, and lossy decoder output mappings β are defined as[2]

$$s(k) \in \mathcal{S}, \quad f(k) \in \Omega, \quad i(k) \in \mathcal{I}, \tag{3.1}$$

$$\alpha : \mathcal{S} \times \Omega \to \mathcal{I}, \quad \beta : \mathcal{I} \times \Omega \to \hat{\mathcal{S}}, \tag{3.2}$$

$$f : \mathcal{I} \times \Omega \to \Omega. \tag{3.3}$$

Additionally we describe the equivalent recursive encoder and decoder mappings

$$\Phi_\Omega \equiv (\alpha, f), \tag{3.4}$$

$$\Psi_\Omega \equiv (\beta, f). \tag{3.5}$$

[2]See chapter 2 for elementary source coding notations used here.

In other words, operations by the state transition, encoder, and decoder for each time instant can be described by

$$f(k+1) = f\left(i(k), f(k)\right), \tag{3.6}$$

$$i(k) = \alpha(s(k), f(k)) = \Phi_{f(k)}\left(s(k)\right), \tag{3.7}$$

$$\hat{s}(k) = \beta(i(k), f(k)) = \Psi_{f(k)}\left(i(k)\right) \quad k = 1, 2, \ldots \tag{3.8}$$

Assuming the encoder and decoder are both in state f, the so called *state codebook* alphabet of state f, a collection of state-dependent outputs, is defined by

$$\mathcal{A}_f = \{\Psi_f(i);\ \text{all } i \in \mathcal{I}\}. \tag{3.9}$$

This allows the definition of the required *recursive VQ nearest neighbor* or minimum distortion property for a given state f[3]

$$\Phi_f(s) = \arg\min_{i \in \mathcal{I}} \rho(s, \Psi_f(i)), \quad \text{or equivalently,} \tag{3.10}$$

$$\rho\left(s, \Psi_f(\Phi_f(s))\right) = \min_{\hat{s} \in \mathcal{A}_f} \rho\left(s, \hat{s}\right), \tag{3.11}$$

where ρ is the distortion measure (in this case squared-error). As mentioned earlier, the analysis-by-synthesis configuration does not allow as to specify a quantizer (VQ) in the conventional sense. However, the quantizer may be seen in the following conceptual (effective) analysis-by-synthesis fashion. It is possible to show that an input to this VQ exists in the analysis-by-synthesis sense. As an analogy to DPCM and PVQ cases, we will use $\{d(k)\}_{k=1}^{\infty}$ for the "difference" input signal to the quantizer. The N dimensional VQ is defined as the mapping from $\mathcal{D}$ (subset of the N-dimensional Euclidean space for which we have $d(k) \in \mathcal{D} \subset \mathbb{R}^N$) to a finite set of reconstruction codevectors in codebook $\mathcal{V}_f$.

[3]In general we will use the subscript of metric space Ω to indicate dependence on state or the recursive nature of the coder. Other times for the encoder/decoder operations where being in a particular state is meant, we will use subscript f or $f(k)$.

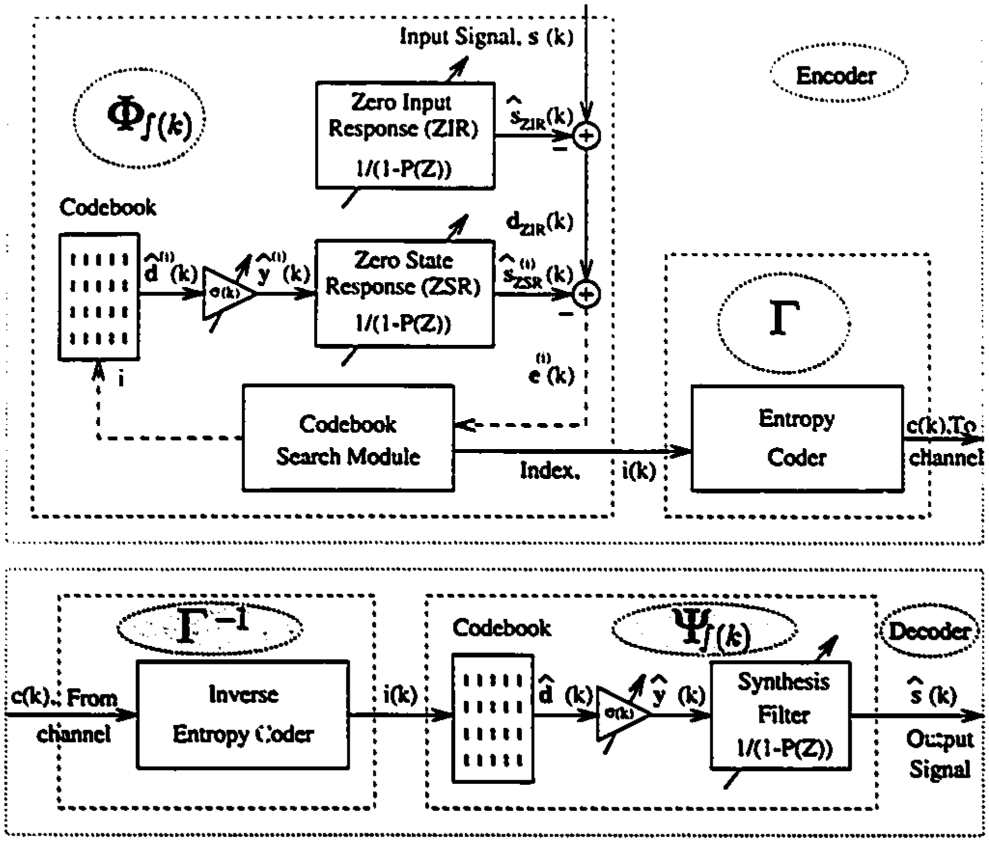


Fig. 3.2 Entropy-coded CELP and EC-CELP encoder and decoder block diagram

The quantization can be decomposed to encoder $\overset{\text{enc.}}{Q}$ and decoder $\overset{\text{dec.}}{Q}$

$$\overset{\text{enc.}}{Q}: \mathcal{D} \to \mathcal{I}, \quad \text{and} \quad \overset{\text{dec.}}{Q}: \mathcal{I} \to \mathcal{V}_I, \quad \text{where} \tag{3.12}$$

$$i(k) = \overset{\text{enc.}}{Q} (d(k)) \quad \text{and} \quad \hat{d}(k) = \overset{\text{dec.}}{Q} (i(k)) \quad k = 1, 2, \ldots, \tag{3.13}$$

where the codebook is defined as $\mathcal{V}_I = \{v^{(i)}; i \in \mathcal{I}\}$, $\mathcal{I} = \{1, 2, \ldots, I\}$ and we also have $\{v^{(i)} \in \mathbb{R}^N : i \in \mathcal{I}\}$. The *partition cell* (Voronoi cell) associated with every $i \in \mathcal{I}$ is

$$\mathcal{R}^{(i)} = \{d \in \mathcal{D} : \overset{\text{enc.}}{Q} (d) = i\} \tag{3.14}$$

so that the cells partition $\mathcal{D}$ or $\mathbb{R}^N$.

Prior to more detailed description of the CELP and its special cases let us provide some related definitions for the prediction and synthesis filters used. The Z transform of the M-th order predictor and corresponding synthesis filter (the inverse of the prediction error filter, $\frac{1}{1-P(Z)}$) are defined as

$$P(Z)=\sum_{m=1}^{M} a_m Z^{-m}, \tag{3.15}$$

$$H(Z)=\frac{1}{1-P(Z)} = \frac{1}{1-\sum_{m=1}^{M} a_m Z^{-m}} = \sum_{m=0}^{\infty} h_m Z^{-m}. \tag{3.16}$$

For the k-th time index (dependence on k indicates the adaptiveness of PC), we may also define the following useful $N \times N$ lower triangular matrix $H(k)$ with its special simple case for the stationary GM(1) case ($h = a$)

$$H(k)=\begin{bmatrix} h_0 & 0 & \dots & 0 \\ h_1 & h_0 & \dots & 0 \\ \vdots & \vdots & \ddots & 0 \\ h_{N-1} & h_{N-2} & \cdots & h_0 \end{bmatrix} \overset{\mathrm{GM}(1)}{=} \begin{bmatrix} 1 & 0 & \dots & 0 \\ h & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & 0 \\ h^{N-1} & h^{N-2} & \dots & 1 \end{bmatrix}. \tag{3.17}$$

As seen in Eqn. 3.16, in the above $\{h_0, h_1, \dots h_{N-1}\}$ are samples of the impulse response of the synthesis filter $\frac{1}{1-P(Z)}$. The choice of $h = a$ for the stationary GM(1) source, as seen in chapter 4, is close to optimum for most cases of interest. Note that if the synthesis filter $\frac{1}{1-P(Z)}$ is driven by the signal $\{\hat{y}(n)\}_{n=1}^{\infty}$, the synthesis filter output $\{\hat{s}(n)\}_{n=0}^{\infty}$ can be written as

$$\hat{s}(n) = \sum_{m=0}^{\infty} h_m \hat{y}(n-m) = \hat{y}(n) + \sum_{m=1}^{M} a_m \hat{s}(n-m). \tag{3.18}$$

Next we look at the detailed operation of the CELP coder as shown in Fig. 3.2.[4] In this figure CELP is the lossy coder in a lossy+lossless configuration. When the EC strategy is not used, we refer to the coder as entropy-coded CELP. The use of the EC strategy results in the EC-CELP quantization. The block diagram also represents CELP as the general recursive VQ with its states as the M synthesis filter states or memories. For the M-th

[4] Also see [35] for the more general EC-CELP which includes other components required for a speech coding application.

order synthesis filter as in Eqn. 3.18, the recursive VQ coder will have M states which at time instant n are $\{\hat{s}(n-m) : m = 1, 2, \ldots, M\}$, M previous reconstructed samples. For $M = 1$, matched to the GM(1) source, there is a single state or expressed in vector notation

$$f(k) = \hat{s}_N(k-1).$$

The CELP encoder uses an exhaustive search through the excitation signal codebook. The search proceeds by passing each of the I excitation vectors from the codebook $\mathcal{V}_I$ through the synthesis filter to obtain the candidates for the current input vector. The synthesis filters ZIR and ZSR are separated to reduce the complexity[5] and to help analyze the coders. In chapter 2 we provided the definition for ZIR and ZSR in the context of GM(1) vector representation. One representation of ZIR and ZSR in the context of CELP will be given later in Eqn. 3.24. The CELP encoder first generates the ZIR difference signal $\boldsymbol{d}_{\text{ZSR}}(k)$ (which can also be called ZIR residual or ZIR error). Then each of the excitation vector candidates $\widehat{\boldsymbol{d}}^{(i)}(k)$ from the codebook (where $\widehat{\boldsymbol{d}}^{(i)}(k)$ is the same as codevector $\boldsymbol{v}^{(i(k))}$, $i \in \mathcal{I}$) is passed through the gain scaling unit and the zero-state synthesis filter. The results are the ZSR candidates $\widehat{\boldsymbol{s}}^{(i)}_{\text{ZSR}}(k)$, $\;i \in \mathcal{I}$ for the current input signal vector $\boldsymbol{s}(k)$. We identify the winning excitation codevector as $\widehat{\boldsymbol{d}}^{(i(k))}(k) = \boldsymbol{v}^{(i(k))}(k) \in \mathcal{V}_I$. For the block (vector) processing, suitable for CELP, the block ZIR $\widehat{\boldsymbol{s}}_{\text{ZIR}}(k)$ can be viewed as the output of the synthesis filter driven by the signal

$$\{\widehat{\boldsymbol{y}}(1), \widehat{\boldsymbol{y}}(2), \ldots, \widehat{\boldsymbol{y}}(k-1), \boldsymbol{0}\},$$

where $\boldsymbol{0}$ zero vector has dimension N. Also the ZSR signal $\widehat{\boldsymbol{s}}_{\text{ZIR}}(k)$ can be represented as the synthesis filter output driven by the input signal

$$\{\underbrace{\boldsymbol{0}, \boldsymbol{0}, \ldots, \boldsymbol{0}}_{k-1}, \widehat{\boldsymbol{y}}(k)\}.$$

[5] See the discussion and formulation details later in this section, comparing equations 3.34 and 3.20.

Then it is clear that the reconstructed signal can be written as

$$\hat{s}(k) = \hat{s}_{\text{ZIR}}(k) + \hat{s}_{\text{ZSR}}(k).$$

where only ZSR depends on $\hat{y}(k)$ or $v^{i(k)}$.

The output of the codebook search module is the winning index $i(k)$ which yields minimum cost. As will be seen later, this cost for the case of entropy-coded CELP is the squared error and in the case of EC-CELP is the EC squared error. This index is sent[6] to the lossless entropy coder.[7] In the configuration of Fig. 3.2, with or without EC, we are interested in entropy-coded performance and hence the coders always includes entropy coding. For a given entropy rate, to measure the coding performance, the normalized average distortion D/σ_s^2, or the signal-to-noise ratio defined as $\text{SNR} = 10\log_{10}\frac{\sigma_s^2}{D}$ and measured in dB is used. For index k, the resulting error and squared-error between the input vector $s(k)$ and the reconstructed vector $\hat{s}(k)$ are given by

$$e(k)=s(k) - \hat{s}(k), \tag{3.19}$$

$$D(k)=\|e(k)\|^2. \tag{3.20}$$

3.2.1. Comparisons between CELP and its special cases

To better demonstrate the comparisons of CELP with DPCM and PVQ coders, we rewrite $\hat{s}(k)$ as the summation of the separated ZIR and ZSR. As mentioned, the ZIR vector $s_{\text{ZIR}}(k)$ is the "ringing" from the previously coded vectors and is fixed throughout the search. The ZSR is the result of passing the gain scaled codebook entry through the zero-state synthesis filter, $s_{\text{ZSR}}(k) = s_{\text{ZSR}}^{i(k)}(k)$. It is obvious that throughout the search, the ZSR will be different for each codevector entry index $i \in \mathcal{I}$. Let us denote $\hat{d}(k) = \left[\hat{d}_1(k)\hat{d}_2(k)\ldots\hat{d}_N(k)\right]^T = v^{(i(k))}(k) \in \mathcal{V}_I$ to be the selected codebook entry at time vector

[6] In the case of fixed-rate coding, this index is directly sent over the channel.

[7] CELP coding may have alternative parameter configurations. Note that the forward adaptation alternative requires transmission of predictor parameters. We confine ourselves to the backward adaptation case which is more appropriate for low delay coding [36], [45]. Extensions of the formulations for other configurations should be straightforward.

instant k. Using the definition of $H(k)$ in Eqn. 3.17, we may rewrite $\hat{\mathbf{s}}(k)$ as

$$\hat{\mathbf{s}}(k)=\hat{\mathbf{s}}_{\mathrm{ZIR}}(k)+\hat{\mathbf{s}}_{\mathrm{ZSR}}(k)=\hat{\mathbf{s}}_{\mathrm{ZIR}}(k)+\sigma(k)H(k)\hat{\mathbf{d}}(k). \tag{3.21}$$

We may also represent the error vector for the CELP by

$$\begin{aligned}(\mathrm{CELP})\quad \mathbf{e}(k)&=\mathbf{s}(k)-\hat{\mathbf{s}}_{\mathrm{ZIR}}(k)-\hat{\mathbf{s}}_{\mathrm{ZSR}}(k)\\ &=\mathbf{d}_{\mathrm{ZIR}}(k)-\sigma(k)H(k)\hat{\mathbf{d}}(k).\end{aligned} \tag{3.22}$$

For the stationary GM(1) source we have $\sigma(k)=1$ and we may use $h=a$. We may write the reconstructed vector as

$$\hat{\mathbf{s}}(k)=\begin{bmatrix}\hat{s}_1(k)\\ \hat{s}_2(k)\\ \cdots\\ \hat{s}_N(k)\end{bmatrix}=h\begin{bmatrix}\hat{s}_N(k-1)\\ \hat{s}_1(k)\\ \cdots\\ \hat{s}_{N-1}(k)\end{bmatrix}+\begin{bmatrix}\hat{d}_1(k)\\ \hat{d}_2(k)\\ \cdots\\ \hat{d}_N(k)\end{bmatrix}, \tag{3.23}$$

The separated of ZIR and ZSR in Eqn. 3.21 can be rewritten in detail as

$$\hat{\mathbf{s}}(k)=\underbrace{\overbrace{\begin{bmatrix}h\\ h^2\\ \cdots\\ h^N\end{bmatrix}\hat{s}_N(k-1)}^{\text{inter-block memory}}}_{\hat{\mathbf{s}}_{\mathrm{ZIR}}}+\underbrace{\overbrace{\begin{bmatrix}0&\cdots&0&0\\ h&\cdots&0&0\\ \cdots&\cdots&\cdots&\cdots\\ h^{N-1}&\cdots&h&0\end{bmatrix}\begin{bmatrix}\hat{d}_1(k)\\ \hat{d}_2(k)\\ \cdots\\ \hat{d}_N(k)\end{bmatrix}}^{\text{intra-Block memory}}+\overbrace{\begin{bmatrix}\hat{d}_1(k)\\ \hat{d}_2(k)\\ \cdots\\ \hat{d}_N(k)\end{bmatrix}}^{\text{“innovation”}}}_{\hat{\mathbf{s}}_{\mathrm{ZSR}}}. \tag{3.24}$$

Using the above representation for $\hat{\mathbf{s}}(k)$ and the representation of GM(1) input signal $\mathbf{s}(k)$ in Eqn. 2.3, it is interesting to see that the ZIR difference vector $\mathbf{d}_{\mathrm{ZIR}}(k)$ includes the filtered excitation signal quantization error. For the case of stationary GM(1) source, using $h = a$, we get

$$\boldsymbol{d}_{\text{ZIR}}(k) = \begin{bmatrix} a \\ a^2 \\ \cdots \\ a^N \end{bmatrix} e_N(k-1) + \begin{bmatrix} w_1(k) \\ a w_1(k) + w_2(k) \\ \cdots \\ \sum_{i=1}^{N} a^{N-i} w_i(k) \end{bmatrix}. \tag{3.25}$$

where the first term of the LHS is the filtered excitation signal quantization error, $e_N(k-1)$.

At this point we use Fig. 3.2, and Eqn. 3.24 to compare CELP and its special cases. Table 3.1 summarizes the comparison of basic features of CELP and other coders. Among the recursive coders, DPCM and PVQ can be considered as special cases of CELP. Consequently, in the above equations for CELP, the scalar case of $N = 1$ furnishes the DPCM signal formulations. To obtain the PVQ formulas, we need to eliminate the intra-block memory term from the above CELP formulas. As an example, by removing such term from Eqn. 3.22 we get the following description for the error signal in PVQ

$$\begin{aligned} \text{(PVQ)} \quad \boldsymbol{e}(k) &= \boldsymbol{s}(k) - \hat{\boldsymbol{s}}_{\text{ZIR}}(k) - \sigma(k)\hat{\boldsymbol{d}}(k) \\ &= \boldsymbol{d}_{\text{ZIR}}(k) - \sigma(k)\hat{\boldsymbol{d}}(k). \end{aligned} \tag{3.26}$$

From the above we can also easily verify that codebook search in PVQ does not involve analysis-by-synthesis. It is obvious that by combining more features, CELP is providing better performance. This for example is reflected in the synthesis structure of signals in CELP (Equations 3.23 and 3.24) which resembles the GM(1) input process in Eqn. 2.3. This cannot be said about the PVQ case. As another example, since the ZIR difference signal represented in Eqn. 3.25 is actually the input to the quantizer (VQ) in the PVQ case, this input still contains intra-block innovation correlations, whereas the intra-block PC feature of CELP attempts to remove such correlations. This allows for a more efficient use and design of VQ in CELP. The consequence of such efficient PC memory removal results in overall improved efficiency of CELP over PVQ, particularly in the case of nonstationary signals. The result can be a smaller dimension (delay) or codebook size (complexity) for the VQ. In chapter 4, we will provide a more detailed analysis of available memory gain for various coders and we will consider the effects of low bitrates on the available memory gain. It is obvious that the advantages of VQ, available to PVQ and CELP, are absent from DPCM and SQ. At low bit-rates, as seen from the simulation results, these advantages play an important role.

We may use the above presentation of PVQ as a special case of CELP to draw the following analogy for the effective VQ for CELP. A comparison of equations 3.22 and 3.26 demonstrates the nature of analysis-by-synthesis VQ mapping in CELP. We then can easily derive the CELP analysis-by-synthesis conceptual VQ mapping and make the following comparison with the PVQ case. Assuming no gain scaling, we have

$$\text{(CELP)} \quad \overset{\text{dec.}}{Q} \circ \overset{\text{enc.}}{Q} : d(k) = d_{\text{ZIR}}(k) - (H - \underline{I})\hat{d}(k) \longrightarrow \hat{d}(k) = v^{i(k)} \tag{3.27}$$

$$\text{(PVQ)} \quad \overset{\text{dec.}}{Q} \circ \overset{\text{enc.}}{Q} : d(k) = d_{\text{ZIR}}(k) \longrightarrow \hat{d}(k) = v^{i(k)}, \tag{3.28}$$

where $\underline{I}$ is the identity matrix. We will reuse the above comparison between CELP and its special cases in other places later on.

Quantization	(inter-vector) PC	VQ	intra-vector PC
SQ			
DPCM	√		
VQ		√	
PVQ	√	√	
CELP	√	√	√

Table 3.1 A comparison of basic features of CELP versus other coders is shown. DPCM and PVQ are also recursive and can be considered as special cases of CELP.

3.2.2. Search and codebook design

In this subsection we describe the search module in Fig. 3.2 and present the codebook design procedure in more detail. The search and design procedures in this subsection are the well known classical techniques for CELP. While presenting these techniques for the non-EC case and as the background material for the EC case, we provide some analysis which will also be relevant for the EC case (section 3.3).

At time instant k and for an input signal $s(k)$ and given codebook $\mathcal{V}_I = \{v^{(i)}; i \in \mathcal{I}\}$, $\mathcal{I} = \{1, 2, \ldots, I\}$, the search through the codebook for the codevector with the minimum cost (squared-error distortion D) results in the winning output index $i(k)$,

$$i(k)=\arg\min_{i\in\mathcal{I}} D^{(i)}(k) \tag{3.29}$$

$$=\arg\min_{i\in\mathcal{I}} \|s(k) - \hat{s}^{(i)}(k)\|^2 \tag{3.30}$$

$$=\arg\min_{i\in\mathcal{I}} \|d_{\text{ZIR}}(k) - \sigma(k)H(k)v^{(i)}\|^2, \tag{3.31}$$

where we know that $\hat{d}^{(i)}(k) = v^{(i(k))}$. Note that $d_{\text{ZIR}}(k)$ depends on the result of the quantization of the previous input vector due to the recursive nature.

The above search is usually exhaustive and hence has to be computationally simple. Let us use the description of the error signal in Eqn. 3.22 and derive a simpler formulation for $D^{(i)}(k)$, the squared-error for each codevector with index $i \in \mathcal{I}$. We denote the gain normalized difference ZIR signal by $\tilde{d}_{\text{ZIR}}(k) = d_{\text{ZIR}}(k)/\sigma(k)$ and obtain the following expanded expression

$$D^{(i)}(k)=\|d_{\text{ZIR}}(k) - \sigma(k)H(k)v^{(i)}\|^2 = \sigma^2(k)\|\tilde{d}_{\text{ZIR}}(k) - H(k)v^{(i)}\|^2 \tag{3.32}$$

$$=\sigma^2(k)\left[\|\tilde{d}_{\text{ZIR}}(k)\|^2 + \|H(k)v^{(i)}\|^2 - 2\tilde{d}_{\text{ZIR}}(k)^T H(k)v^{(i)}\right]. \tag{3.33}$$

Since during the search, the first term is constant, the minimization argument can be chosen to be

$$\tilde{D}^{(i)}(k) = -2p^T(k)v^{(i)} + E^{(i)}(k), \quad \text{where}$$
$$p(k) = H(k)^T\tilde{d}_{\text{ZIR}}(k), \quad \text{and} \quad E^{(i)}(k) = \|H(k)v^{(i)}\|^2; \quad i \in \mathcal{I}. \tag{3.34}$$

Notice that $p(k)$ is calculated once during the search and is constant for all i. As well $E^{(i)}(k)$ for each $i \in \mathcal{I}$ is pre-calculated and can remain constant as long as $H(k)$ is not updated (filter adaptation period).

Next we describe the codebook design procedure in CELP. We will be using the training sequence $\{s(k)\}_{k=1}^{K}$. The codebook can be designed using *closed-loop*, *open-loop* [37], [29], or *successive clustering* [105] LBG-like algorithms for the CELP coder. We experimented with all three schemes and it was confirmed that for low bitrates, where quantization noise is high, the *closed-loop* design is more appropriate. The reason as seen shortly is that it incorporates the effect of quantization error in the formulation. In all of these schemes, iterations on *nearest neighbor* search strategy and *clustering and centroid* calculations have to done. The iteration is between new *partitioning* and *new codebook* calculation. Let m be the iteration index and let $\overset{m}{\mathcal{V}}_I$ be the codebook at iteration m. In the *nearest neighbor* search, for notational convenience we define the set of training indices included in the i-th partition $\overset{m}{\mathcal{R}}^{(i)}$ as

$$\overset{m}{\mathcal{K}}^{(i)} = \{k : \arg\min_{\hat{d}\in\mathcal{V}_I} \|d_{\text{ZIR}}(k) - \sigma(k)H(k)\hat{d}\|^2 = \overset{m}{v}^{(i)}\}. \tag{3.35}$$

During the iteration, the N-dimensional space is divided into I non-standard[8] Voronoi cells (Eqn. 3.14). During the clustering iteration, the i-th cell $\overset{m}{\mathcal{R}}^{(i)}$ with $i \in \mathcal{I}$ will be populated with $\overset{m}{K}^{(i)}$ elements.

Using the equations 3.22 and 3.27, we can obtain the squared-error distortion for the i-th Voronoi cell (with codevector $\overset{m}{v}^{(i)}$)

$$\overset{m}{D}^{(i)} = \frac{1}{\overset{m}{K}^{(i)}} \sum_{k\in\overset{m}{\mathcal{K}}^{(i)}} \|s(k) - \hat{s}^{(i)}(k)\|^2 \tag{3.36}$$

$$= \frac{1}{\overset{m}{K}^{(i)}} \sum_{k\in\overset{m}{\mathcal{K}}^{(i)}} \left[d_{\text{ZIR}}(k) - \sigma(k)H(k)v^{(i)}\right]^T \left[d_{\text{ZIR}}(k) - \sigma(k)H(k)v^{(i)}\right]. \tag{3.37}$$

When using the *closed-loop* design, we will find the unknown i-th cluster new centroid codevector $\overset{m+1}{v}^{(i)}$, by minimizing

$$\overset{m}{D}^{(i)}(\hat{d}^{(i)}) = \frac{1}{\overset{m}{K}^{(i)}} \sum_{k\in\overset{m}{\mathcal{K}}^{(i)}} \left[d_{\text{ZIR}}(k) - \sigma(k)H(k)\hat{d}^{(i)}\right]^T \left[d_{\text{ZIR}}(k) - \sigma(k)H(k)\hat{d}^{(i)}\right], \tag{3.38}$$

[8] Voronoi cells in the case of CELP are non-standard as VQ is an *effective quantizer*.

the total distortion for the cluster with respect to $\hat{d}^{(i)}$. The centroid condition is the necessary and sufficient condition for the squared-error distortion minimization

$$\overset{m+1}{v}{}^{(i)} = \mathrm{centroid}(\overset{m}{\mathcal{R}}{}^{(i)}) = \arg\min_{\hat{d}^{(i)}} \overset{m}{D}{}^{(i)}\,(\hat{d}^{(i)}). \tag{3.39}$$

To minimize $\overset{m}{D}{}^{(i)}\,(\hat{d}^{(i)})$ with respect to $\hat{d}^{(i)}$, we take the derivative of $\overset{m}{D}{}^{(i)}\,(\hat{d}^{(i)})$ with respect to $\hat{d}^{(i)}$ and we set the result to zero. As a result we get the following linear system

$$\sum_{k\in\overset{m}{\mathcal{K}}{}^{(i)}} \sigma^2(k)\left[H(k)^T H(k)\right]\hat{d}^{(i)} = \sum_{k\in\overset{m}{\mathcal{K}}{}^{(i)}} \sigma(k)H(k)^T d_{\mathrm{ZIR}}(k). \tag{3.40}$$

The matrix $\sigma^2(k)H^T(k)H(k)$ and the vector $\sigma(k)d_{\mathrm{ZIR}}(k)$ are accumulated separately for each cluster and the I linear systems are solved to obtain the I new codevectors $\{\overset{m+1}{v}{}^{(i)}\}$. The matrix $H(k)$ implicitly incorporates the time variations of the predictor filter in the formulation. The signal dynamic range variations are reflected in $\sigma(k)$. We refer to the above as the closed-loop *centroid rule* for CELP quantization design.

Next we examine and compare the solutions of open-loop and closed-loop centroid updates. To simplify this analysis, let us assume a stationary input signal. In that case there is no need for solving a linear system and only the accumulation of the difference vector $d_{\mathrm{ZIR}}(k)$ for $k\in\overset{m}{\mathcal{K}}{}^{(i)}$ is sufficient and the closed-loop *centroid rule* simplifies as follows

$$\overset{m+1}{v}{}^{(i)} = \frac{1}{\overset{m}{K}{}^{(i)}}\left[H^T H\right]^{-1} H^T \sum_{k\in\overset{m}{\mathcal{K}}{}^{(i)}} d_{\mathrm{ZIR}}\,(k). \tag{3.41}$$

The above simplification certainly makes the stationary design procedure simple. To further examine the above expression, let us assume that the input source is a GM(1) source with innovation signal $\{w(k)\}_{k=0}^{\infty}$ (Eqn. 2.3). After some algebra, the centroid amounts to the following expression

$$\overset{m+1}{\boldsymbol{v}^{(i)}} = \frac{1}{\overset{m}{K^{(i)}}} \sum_{k \in \overset{m}{\mathcal{K}^{(i)}}} \begin{bmatrix} ae_N(k-1) + w_1(k) \\ w_2(k) \\ \cdots \\ w_N(k) \end{bmatrix}. \tag{3.42}$$

It is easy to derive the expression for the open-loop design which is identical to the above (accumulated innovation samples w) except it does not include the accumulated error effect reflected in the first dimension. i.e.

$$\overset{m+1}{\boldsymbol{v}^{(i)}} = \frac{1}{\overset{m}{K^{(i)}}} \sum_{k \in \overset{m}{\mathcal{K}^{(i)}}} \boldsymbol{w}(k) = \frac{1}{\overset{m}{K^{(i)}}} \sum_{k \in \overset{m}{\mathcal{K}^{(i)}}} \begin{bmatrix} w_1(k) \\ w_2(k) \\ \cdots \\ w_N(k) \end{bmatrix}. \tag{3.43}$$

This further qualifies what earlier was said in favor of closed-loop design for low rate coding. The above analysis can also be extended to the case of higher order GM input source.

For the PVQ design, the special case of CELP, we assumed a first order prediction with coefficient h which is matched to the GM(1) source regression coefficient $h = a$. Appendix B provides details for the choice of predictor coefficients in PVQ. Similar to the case of CELP, we may arrive at the following centroid rule for the PVQ

$$\overset{m+1}{\boldsymbol{v}^{(i)}} = \frac{1}{\overset{m}{K^{(i)}}} \sum_{k \in \overset{m}{\mathcal{K}^{(i)}}} \boldsymbol{e}_{\mathrm{ZIR}}(k) = \frac{1}{\overset{m}{K^{(i)}}} \sum_{k \in \overset{m}{\mathcal{K}^{(i)}}} \begin{bmatrix} ae_N(k-1) & + & w_1(k) \\ ae_1(k) & + & w_2(k) \\ & \cdots & \\ ae_{N-1}(k) & + & w_N(k) \end{bmatrix}. \tag{3.44}$$

Comparison of closed-loop case for CELP and PVQ shows how the focus of codevector representation is deviated from representing innovation signal within the block. This is due to the lack of memory in the PC ZSR. This amounts to "more work for VQ" and hence less coding efficiency "where it would be needed" (eminent in the nonstationary input signal case).

Finally, one reason for the success of CELP for the slowly time-varying signals such as speech is the ease and nature of adaptability in CELP. As seen, the *formulation of closed-loop design* for CELP quantization, the synthesis filter ($H(k)$) and gain $\sigma(k)$ variation are easily and implicitly reflected in the *nearest neighbor* and *centroid* design formulation.

3.2.3. Rate-distortion performance of entropy-coded CELP

In this subsection, we present the rate-distortion results for the entropy coded CELP (without EC). We used this configuration early in our work to evaluate the performance of CELP without entropy-constrained (EC) consideration. In the simulation results for this case we allow the fiction that codewords can have non-integer length. We used the first-order entropy of the lossy coder output (CELP) to represent the average rate (entropy of the index set $\{i(k)\}_{k=1}^{\infty}$). To provide a basic theoretical justification for this, we first extend a similar required theory, expressed for EC-DPCM [27], to the more general PC case (CELP). This theory and the one in [27] extend the zero-memory case to CELP and DPCM respectively. The theorem qualifies the utilized fact that the amount of information delivered by the output process about the input process equals the entropy of the output process. It applies to the general entropy-coded CELP (with EC or without EC).

Theorem 3.1. *Let us group K vectors, each with length N, to form the KN sample vector*

$$
\begin{aligned}
{}^{KN}s &= \left({}^{N}s(1), {}^{N}s(2), \ldots, {}^{N}s(K)\right), \quad \textit{where} \\
{}^{N}s(k) &= \left(s\left((k-1)N+1\right), s\left((k-1)N+2\right), \ldots, s\left((k-1)N+N\right)\right).
\end{aligned}
$$

For the CELP coder with reconstruction formulation described by 3.23, we then have

$$
\lim_{K\to\infty} \frac{1}{KN} I\left({}^{KN}s; {}^{KN}\hat{d}\right) = H_{\infty}(\hat{d}),
$$

with $I(.;.)$ denoting mutual information and $H_\infty(\hat{d})$ being the entropy rate of $\{\hat{d}\}$.

Proof: Using the property of mutual information we have

$$I\left({}^{KN}s;\, {}^{KN}\hat{d}\right) = H({}^{KN}\hat{d}) - H\left({}^{KN}\hat{d} \mid {}^{KN}s\right).$$

Assuming common initial state and state transition rule for recursive encoder and decoder, we now use the following one-to-one relationship for vectors of size N,

$${}^{N}s \leftrightarrow {}^{N}d \leftrightarrow {}^{N}\hat{d},$$

to arrive at the following one-to-one conclusion for blocks of size KN,

$${}^{KN}s \leftrightarrow {}^{KN}d \leftrightarrow {}^{KN}\hat{d}.$$

Hence there is no uncertainty for $\hat{d}$ or

$$H\left({}^{KN}\hat{d} \mid {}^{KN}s\right) = 0 \quad \text{and hence}$$
$$I\left({}^{KN}s;\, {}^{KN}\hat{d}\right) = H({}^{KN}\hat{d}).$$

Now we can easily arrive at the desired result by dividing both sides by KN and letting $K \to \infty$. □

Simulation results: The rate-distortion performance results of CELP (without EC design) in Fig. 3.3 were obtained using 10^6 samples from the input GM(1) source. The goal of these experiments was to obtain enough rate-distortion performance data to characterize the coder performance. To do this we designed CELP coders with a variety of codebook sizes (I) and vector lengths (N). The maximum vector size $N = 8$ and maximum codebook size $I = 512$ were used. The simulations revealed that most points on the rate-distortion curves were obtained using lower dimensions than the maximum values. For comparison we used an ideal ("infinite" level I) entropy-coded uniform DPCM (UDPCM). As seen in the results of Fig. 3.3, the entropy-coded CELP coder provides close to

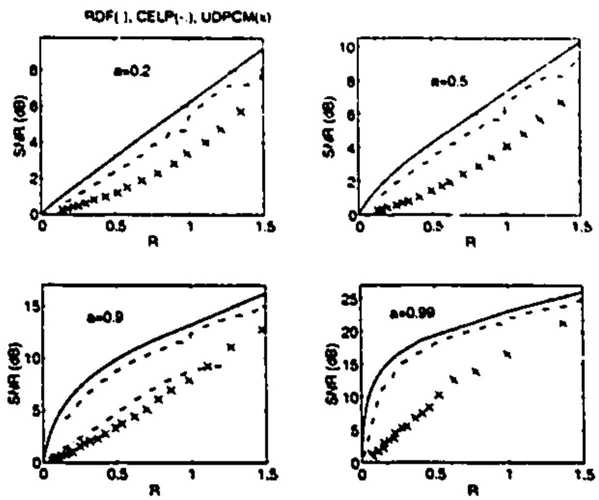


Fig. 3.3 Quantizers rate-distortion performance versus RDF (solid) for various regression coefficients a. Performance of entropy coded CELP with maximum $N = 8$ and various codebook sizes I is shown in dash-dot (-.). Entropy-coded ideal DPCM (UDPCM) is shown in cross (x). For $a = 0.9$, a second curve (the lower curve further from RDF) in dash-dot (-.) shows the D4 Lattice performance as another reference.

RDF performance. From these results, there are two reasons to motivate the EC design, the main subject of this work. First EC strategy matches the entropy-coded configuration better than the above RC design approach. The second reason which follows from the first is that lower delay (N) and complexity (efficiency of quantization through lower codebook size I) may be possible. This was indeed verified as later EC-CELP results were compared with the above case. These results showed that EC design strategy can provide substantial reduction in coding delay and complexity for similar performance.

As seen in the results of Fig. 3.3, the performance gap between entropy-coded UDPCM and CELP increases as the source correlation (regression coefficient) a is increased from 0.2 to 0.99. It is easy to verify that for the GM(M) source and the DPCM quantizer, the input to the quantizer is also an M-th order Markovian process. In [3] it is shown that although this process is not Gaussian it is close to Gaussian. There, based on numerical results, it is shown that higher data rate saving over DPCM for higher correlation should be possible. As explained in [3] and in chapter 4, the higher the correlation of the source, the higher the correlation of the input to the quantizer. Later detailed analyses show at lower bitrates the quantization noise feedback limits the performance of DPCM. We will also see that the EC-DPCM (not uniform) has similar trends. It is the VQ feature of the CELP coupled with its closed-loop configuration that reduces the quantization noise effects particularly at very low bitrates.[9]

3.3. EC-CELP Design algorithm

Results of section 3.2 showed that even at low bitrates and for highly correlated sources, performance near the rate-distortion bound is possible by using the entropy-coded CELP. This motivates the goal of designing the CELP quantizer codebook in a rate-distortion theoretic sense. More precisely, we wish to minimize the overall average distortion, while the average transmission rate or entropy rate is kept below certain level. This is the goal of the EC quantization design. Using Theorem 3.1, we may restate the above problem as designing a codebook for the EC-CELP quantization scheme such that the overall average squared error distortion is minimized, while the entropy rate at the CELP output (codevector indices) is held below a prescribed value say H_0. In this section we present an iterative design algorithm for EC-CELP using a suitable empirical approach. The simulation results in the next section show that the EC design strategy can significantly improve the efficiency of RC entropy-coded CELP of previous section by reducing the delay and coder complexity.[10] The notations in this section correspond to the block diagram in Fig. 3.2.

First let us summarize the recursive (predictive) and adaptive VQ (CELP) variable rate encoder and decoder pair $(\bar{\Phi}_\Omega, \bar{\Psi}_\Omega)$ by the following mappings:

$$\bar{\Phi}_\Omega = \Gamma \circ \Phi_\Omega : \qquad \mathcal{S} \mapsto \mathcal{C}, \tag{3.45}$$

$$\bar{\Psi}_\Omega = \Psi_\Omega \circ \Gamma^{-1} : \qquad \mathcal{C} \mapsto \hat{\mathcal{S}}, \tag{3.46}$$

The implicit state-transition properties are shown by the subscript Ω and the recursive

[9] The aforementioned advantages are not limited to CELP and extend to other delayed decision predictive coders such as EC-PTCQ [28] [4] or predictive Tree coders.

[10] We first reported these EC-CELP quantization performance results for stationary GM source in [32], [33].

VQ lossy and lossless decomposition components are defined as before by

$$\Phi_\Omega : \mathcal{S} \mapsto \mathcal{I} \text{ and } \Gamma : \mathcal{I} \mapsto \mathcal{C}, \tag{3.47}$$

$$\Gamma^{-1} : \mathcal{C} \mapsto \mathcal{I} \text{ and } \Psi_\Omega : \mathcal{I} \mapsto \hat{\mathcal{S}}. \tag{3.48}$$

For EC-CELP quantization design, to obtain locally optimum variable-length recursive and adaptive VQ (CELP) coding with respect to a fidelity criterion, an extension of previous EC algorithms (particularly [14]) is formulated. Similar to [14], the proposed EC-CELP algorithm uses a Lagrangian formulation. However it differs from such EC formulations by being recursive in nature (indicated by the state space Ω subscripts or f subscript to indicate a particular state) and being empirical as opposed to analytical. As in other empirical EC cases, the goal is to find the convex hull of the *N-th order operational distortion rate trajectory* of the EC-CELP quantizer,[11]

$$^N\tilde{D}(R) = \inf_{N(\Gamma\circ\Phi_\Omega, \Psi_\Omega\circ\Gamma^{-1})} \left\{ \frac{1}{N}\hat{E}\left[\rho(s,\hat{s})\right] \mid \frac{1}{N}\hat{E}\left[\mathrm{len}(s)\right] < {}^N R \right\}, \tag{3.49}$$

where $\rho(s,\hat{s})$ is the distortion (here squared-error) between s and $\hat{s}$. In the above equation $D = (1/N)\hat{E}(\rho(s,\hat{s}))$ is the average distortion and $R = (1/N)\hat{E}[\mathrm{len}(s)]$ is the average rate.[12] $\mathrm{len}(s) = |\Gamma(i)| = |\Gamma(\Phi_f(s))|$ is the length of the codevector[13] representing s (an approximation of *self-information* or an approximation to the optimal codeword length) in bits.[14]

[11] A given initial state for the recursive encoder/decoder system is assumed.

[12] $\hat{E}(\cdot)$ is the sample average or sample estimation of expectation operator.

[13] $\mathrm{len}(s)$ depends on the initial state or state which is assumed to be given.

[14] For the case of EC-VQ, the lower bound to the N-th order operational (VQ) RDF is the *N-th order RDF*,

$$^N D(R) = \inf_{P_{\hat{s}|s}} \left\{ \frac{1}{N}\hat{E}\left[\rho(s,\hat{s})\right] \mid \frac{1}{N} I(s;\hat{s}) < {}^N R \right\},$$

which as N goes to infinity (for squared-error) becomes RDF. In the case of EC-CELP, due to the recursive nature of PC, such a lower bound can not be stated. Assuming signal stationarity and by using the effective block length N corresponding to PC, the value of N can not be clearly defined. In the next chapter we will see that effective quantizer error and overall error for the CELP coder are equal. From this, we may speculate that the optimum operational performance by the EC-CELP sought in the EC-CELP algorithm is related to the *N-th order RDF* of the residual signal $d(k)$. This decouples the PC effective block size from the residual signal $d(k)$ block size N.

The convex hull of the ${}^N\tilde{D}(R)$ is found by minimization of the functional

$$J_\lambda(\bar{\Phi}_\Omega, \bar{\Psi}_\Omega) = \hat{E}\left[\rho(s,\hat{s})\right] + \lambda \hat{E}\left[\text{len}(s)\right], \quad \text{or}$$
$$J_\lambda(\Phi_\Omega, \Gamma, \Psi_\Omega) = \hat{E}\left[\rho\left(s, \Psi_\Omega\left(\Phi_\Omega(s)\right)\right) + \lambda \mid \Gamma\left(\Phi_\Omega(s)\right) \mid\right], \tag{3.50}$$

where $-\lambda$ graphically represents the slope of the line passing through the point $(R(\Phi_\Omega, \Gamma, \Psi_\Omega), D(\Phi_\Omega, \Gamma, \Psi_\Omega))$ in the EC-CELP rate-distortion trajectory plane and supporting the convex-hull.

3.3.1. Summary of algorithm

For each λ, giving a point on the operational rate-distortion convex hull, starting from an initial coder, the iterative descent algorithm repeatedly updates the mappings $(\overset{m}{\Phi}_\Omega, \overset{m}{\Gamma}, \overset{m}{\Psi}_\Omega)$ for increasing index m until some stopping criterion for the convergence of the above functional is met. The stacked index m annotates the design algorithm iteration index. The resulting EC coder triplet, winning with the lowest EC cost, would be denoted by $(\overset{*}{\Phi}_\Omega, \overset{*}{\Gamma}, \overset{*}{\Psi}_\Omega)$.

Each of the three main steps of algorithm fixes two of the triplet coding components in order to obtain the third one,

$$(\overset{m}{\Phi}_\Omega, \overset{m}{\Gamma}, \overset{m}{\Psi}_\Omega) \overset{\text{step 1}}{\Longrightarrow} (\overset{m+1}{\Phi}_\Omega, \overset{m}{\Gamma}, \overset{m}{\Psi}_\Omega) \overset{\text{step 2}}{\Longrightarrow} (\overset{m+1}{\Phi}_\Omega, \overset{m+1}{\Gamma}, \overset{m}{\Psi}_\Omega) \overset{\text{step 3}}{\Longrightarrow} (\overset{m+1}{\Phi}_\Omega, \overset{m+1}{\Gamma}, \overset{m+1}{\Psi}_\Omega).$$

First, for the given coding components $(\overset{m}{\Gamma}, \overset{m}{\Psi}_\Omega)$, the mapping $(\overset{m+1}{\Phi}_\Omega)$ is obtained by using the codebook search with the EC cost defined as $\rho(s, \overset{m}{\Psi}_f(\overset{m+1}{\Phi}_f(s))) + \lambda |\overset{m}{\Gamma}(\overset{m+1}{\Phi}_f(s))| = \rho(s, \overset{m}{\Psi}_f(i)) + \lambda |\overset{m}{\Gamma}(i)|$. This is the ***EC nearest neighbor rule.*** In the second step, given $(\overset{m}{\Phi}_\Omega, \overset{m}{\Psi}_\Omega)$, the codevector lengths $(\overset{m+1}{\Gamma})$ are updated. Finally, given $(\overset{m}{\Phi}_\Omega, \overset{m}{\Gamma})$, the mapping $(\overset{m+1}{\Psi}_\Omega)$ is obtained. This last step amounts to the ***EC centroid rule*** in the closed-loop CELP design. As a result, as in the non-EC case of the previous section, for each nonempty Voronoi cell, a set of linear equations has to be solved (Eqn. 3.40). The solution constitutes the new codevector. As in [14], the empty Voronoi cell codevectors are discarded. Again this formulation is necessary only for the general case of nonstationary Gauss-Markov source model, where the predictor $P(Z)$ in the CELP coder is adaptive. For the stationary Gauss-Markov case, there is no need for solving such linear systems. For each Voronoi cell, it suffices to accumulate an N-dimensional vector and divide this vector by the Voronoi cell population number. The accumulated vector is a simple function (scaled by fixed matrix multiplication term) of the ZIR error signals (d_{ZIR}) for that cell (Eqn. 3.41).

For the EC-CELP *nearest neighbor rule*, the search procedure of previous section for entropy-coded CELP has to be modified to include the EC component. We will use the non-EC cost notation in Eqn. 3.33 to obtain the EC-CELP EC-squared-error cost function. The resulting EC cost function, for the codebook entry i, will have the form

$$D_{\mathrm{EC}}^{(i)}(k)=\rho(s(k),\Psi_{f(k)}(\Phi_{f(k)}(s(k)))) + \lambda|(\Gamma(\Phi_{f(k)}(s(k))))| \tag{3.51}$$

$$=D^{(i)}(k) + \lambda|i|. \tag{3.52}$$

Similarly, the reduced complexity EC-squared-error cost function using Eqn. 3.34 has the form

$$\tilde{D}_{\mathrm{EC}}^{(i)}(k) = \tilde{D}^{(i)}(k) + \lambda|(i)|. \tag{3.53}$$

Before presenting the complete iterative algorithm, let us define the following notations. The relative frequency of an event or the empirical probability mass function (PMF) will be denoted by $\widehat{P}$. The sample average or estimated expectation operator notation $\widehat{E}[\cdot]$ and the Euclidean distance notation $\|\cdot\|$ are also used.

For a fixed N and a given λ, providing a point in the N-th order rate-distortion convex-hull, the algorithm will start with the following initial conditions:

(*i*) $m = 0$, index of iteration,

(*ii*) $\{s(k)\}_{k=1}^{K}$, training vector sequence,

(*iii*) λ, Lagrangian multiplier,

(*iv*) $\overset{m}{\Omega} = \mathbf{0}$, the zero initial state,

(*v*) $\overset{m}{\mathcal{V}}_{\overset{m}{\mathcal{I}}} = \mathcal{V}_{\mathcal{I}}$, a properly chosen random codebook, index set $\overset{m}{\mathcal{I}} = \mathcal{I}$, and associated lossy encoder/decoder pair $(\overset{m}{\Phi}_\Omega, \overset{m}{\Psi}_\Omega)$,

(*vi*) $\{|i|; i \in \mathcal{I}\} = \left\{|i| = -\log_2 \hat{P}(i); i \in \mathcal{I}\right\} = \{|i| = \log_2 I; i \in \mathcal{I}\}$, the equi-probable self-information set associated with the lossless coder $\overset{m}{\Gamma}$,

(*vii*) $\overset{*}{J} = \infty$, optimum EC cost.

We iterate on m according to the following algorithm steps and rules,

(1) Update *EC Nearest neighbor rule* by sequentially encoding $s(k), k = 1, \ldots, K$, simplified using equations 3.31 and 3.34 in the following manner

$$
\begin{aligned}
\overset{m+1}{\Phi}_{f(k)}(s(k)) = i(k) &= \arg\min_{i \in \overset{m}{\mathcal{I}}} \left[\rho(s(k), \overset{m}{\Psi}_{f(k)}(i)) + \lambda |\overset{m}{\Gamma}(i)|\right] \\
&= \arg\min_{i \in \overset{m}{\mathcal{I}}} \left[\|d_{\mathrm{ZIR}}(k) - \sigma(k) H(k) \overset{m}{v}^{(i)}\|^2 + \lambda |\overset{m}{\Gamma}(i)|\right] \\
&= \arg\min_{i \in \overset{m}{\mathcal{I}}} \left[\tilde{D}^{(i)}(k) + \lambda |\overset{m}{\Gamma}(i)|\right]
\end{aligned}
$$

(2) Update the functional $\overset{m}{J} = \hat{E}\left[\rho(s, \overset{m}{\Psi}_f(\overset{m+1}{\Phi}_f(s))) + \lambda |\overset{m}{\Gamma}(\overset{m+1}{\Phi}_f(s))|\right]$.

(3) If $\overset{m}{J} < \overset{*}{J}$, set $\begin{cases} & \text{(possibly reduced index set } \overset{m}{\mathcal{I}} \text{ and lossless coder,)} \\ \overset{*}{J} = & \overset{m}{J}, \\ \overset{*}{\mathcal{V}}_{\mathcal{I}} = & \overset{m}{\mathcal{V}}_{\mathcal{I}}, \\ \overset{*}{\Gamma} = & \overset{m}{\Gamma}. \end{cases}$

If $m > m_{max}$ quit, otherwise set $m = m + 1$ and continue,

(4) Update lossy coding rule for each non-empty cell $i \in \overset{m+1}{\mathcal{I}}$ (possibly reduced size) according to

$$|\overset{m+1}{\Gamma}(i)| = -\log_2 \hat{P}\left\{\overset{m+1}{\Phi}_f(s) = i\right\}$$

(5) Update *EC centroid rule* (see equations 3.38 and 3.40), for each non-empty cell $i \in \overset{m+1}{\mathcal{I}}$ with population $\overset{m+1}{K^{(i)}}$, find $\overset{m+1}{v^{(i)}} \in \overset{m+1}{\mathcal{V}}$ according to

$$
\begin{aligned}
\overset{m+1}{\Psi}_{f}(i) &= \arg\min_{\hat{s}\in\hat{\mathcal{S}}} \hat{E}\left[\rho(s,\hat{s}) \mid \overset{m+1}{\Phi}_{f}(s)=i\right], \\
&= \arg\min_{\hat{s}^{(i)}(k)} \frac{1}{\overset{m}{K^{(i)}}} \sum_{k\in\overset{m}{\mathcal{K}^{(i)}}} \|s(k)-\hat{s}^{(i)}(k)\| = \arg\min_{\hat{d}^{(i)}} \overset{m}{D^{(i)}}(\hat{d}^{(i)}), \\
\text{satisfying} &\quad \left(\sum_{k\in\overset{m}{\mathcal{K}^{(i)}}} \sigma^2(k)\left[H(k)^T H(k)\right]\hat{d}^{(i)} = \sum_{k\in\overset{m+1}{\mathcal{K}^{(i)}}} \sigma(k) H(k)^T d_{\text{zin}}(k)\right).
\end{aligned}
$$

As seen in the above algorithm, unlike EC-VQ, for each new λ we initialized with a full codebook size. The reason for this is the consequence of sensitivity to selection scheme of λ which is explained in next subsection. In our notation the minimum cost coder parameters are indicated by stacked notation ($*$). In section 3.3.3 we will see that in some cases, due to instability of the convergence, a "memory" of minimum cost coder parameters may be required.

In our simulations, other than the above EC closed-loop design alternative, we experimented with the EC open-loop and successive clustering alternatives. The previous section provided some analysis for the CELP close-loop and open-loop alternatives without EC strategy. Consistent with those understandings in our experiments, the EC closed-loop strategy provided better performance. Following the above procedure outlined for the EC closed-loop centroid rule, the detailed *open-loop* and *successive clustering* EC centroid rules can also be easily derived.

3.3.2. Initial codebook, choice of λ, and lossless coders

One of the important issues in general VQ design is the choice of initial codebook. For the recursive VQ schemes such as EC-CELP, the choice of initial codebook can even have greater consequences. In particular by using a "richer" initial codebook, the final performance of the coding scheme can be substantially improved. In the case of stationary GM source however, one easy choice would be a randomly selected codebook. Intuitively, samples from a pseudo-random sequence close to the innovation process would be reasonable. However, if the initial codebook size I is small, this may not result in very good performance. Another alternative is to use the open-loop design strategy on random samples from an innovation-like process sequence. The *split method* similar to simple VQ LBG-type design [78] can also generate a richer initial codebook. For nonstationary signals such as speech, the process of selection of initial codebook is more problematic. For these cases, good results were obtained using the extension of the above ideas coupled with suitable choices of predictor and gain initial state and adaptation. Finally another alternative which in cases of unknown input signal produced satisfactory results was structured VQ codebook (designed based on variance of the innovation signal).

For the selection of λ, the same strategy used for the EC-VQ design in [14] produced satisfactory results in most stationary cases. In this strategy, the two extreme cases in the range $\infty \geq \lambda \geq 0$ ($\lambda = 0$ corresponding to non EC case and large λ or ∞ corresponding to rate zero) are designed first. The ordered sequence of λ in between this two values were selected by a certain intuitive rule. In our experiments, we encountered great sensitivity to the choice of λ. Hence we used *a priori* selected values in between the two extreme λ values.

The lossless entropy coding alternatives have been the subject of many EC papers (e.g. [51], [65], [60]). Both infinite buffer option and finite buffer cases include variations of classical Huffman, Arithmetic, and Lempel-Ziv coding [6], [107]. For cases where long delay can be tolerable, Jones [59] has suggested an efficient Arithmetic coding scheme. Such entropy encoder maps long strings of lossy encoder output into long strings. Such strategy requires either availability of length of string (at the encoder) to the decoder or use of end of block special character to be used by encoder and decoder. For applications such as transmission over fixed-rate channels which necessitate buffering, there is the associated overflow/underflow problems.

The common conclusion from previous investigations is that there is a tradeoff between performance and delay. It is also concluded that the advantages of EC strategy can be maintained with small degradation in coder quality. In the EC formulation presented earlier, it is easy to see that the infinite delay ideal entropy coder mapping can be easily replaced by a specific entropy coder of choice. EC design procedure in that case will incorporate the tradeoffs resulting from the lossless coding of choice. Note that in this thesis minimum delay property refers to the VQ dimension delay. Considerations with regard to the buffer delay are not taken into consideration. Harrison and Modestino [51] have studied a *modified Huffman* buffer-instrumented strategy which resulted in entropy coding with robust buffer management. Kim and Modestino [65] have suggested an adaptive buffer control modification to Arithmetic coding of [59]. The long lossy encoder output is divided into sub-blocks marked by end-of-block character. We have assumed that the conclusions from the above studies will extend to the EC-CELP case. However further simulations and studies will be required to obtain quantitative evaluation of delay and performance tradeoffs. Such evaluations will be particularly important for the case of low delay (end-to-end delay) applications.

3.3.3. Algorithm convergence

In [14], it is suggested that the EC-VQ design algorithm for the case of stationary signal and under mild conditions has a stable convergence property. For the case of (closed-loop) general EC-CELP design algorithm unlike EC-VQ (being empirical rather than analytical as well as being recursive rather than non-recursive), derivation of such theoretical convergence properties will be difficult. On one hand, for the open-loop design algorithm, extension of similar conclusions as made in the case of EC-DPCM [27] may be possible. The closed-loop CELP design case (and a nonstationary input signal) on the other hand is known [12] not to have a monotonic convergence characteristics and the general EC-CELP design case will obviously have similar unstable characteristics. Experimentally however, certain conclusions regarding convergence properties of the EC-CELP algorithm can be made which will be presented next.

Since closed-loop design convergence characteristics are not always monotonic, in our design algorithm we used a similar strategy as for the CELP coder design [12]. This strategy amounts to the following "memory" in the design. During the iterations, the minimum cost coder parameters, indicated by stacked notation $(*)$, are remembered. This allows for possible (numerical instability) violation of non-increasing behavior in cost function. Note that for the stationary input case, there is no need for such strategy and changes smaller than a given epsilon in the cost value can be used as a stopping criterion.[15]

[15]Exception were numerically unstable cases where for example the GM(1) source regression coefficient was very close to 1 (e.g. 0.999)

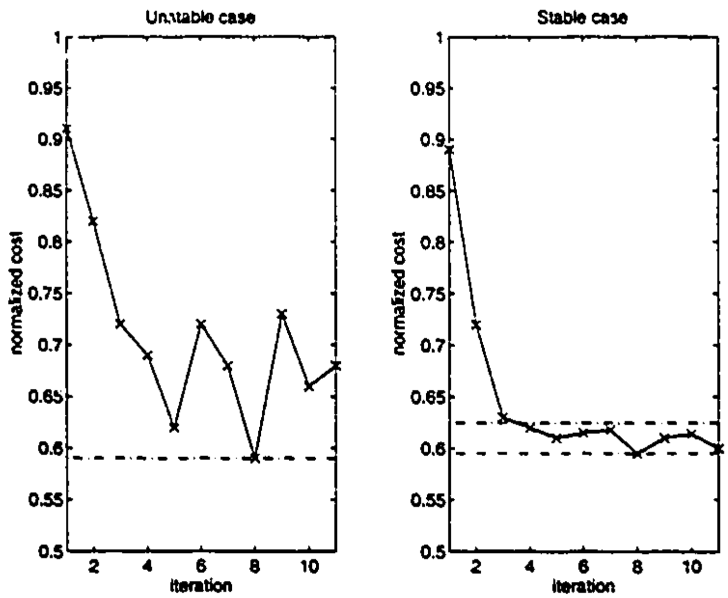


Fig. 3.4 Trend of convergence of EC-CELP design algorithm for a given λ: cost value as a function of iteration number in stable cases (most stationary cases) and unstable cases (nonstationary cases or stationary case with GM(1) regression coefficient a very close to 1.0). For the stable case changes less than an ϵ (epsilon value indicated by the gap between the two dashed lines) in the cost value may be used as a stopping criteria.

In our simulations, other than the sensitivity of the algorithm convergence to λ value (EC-VQ algorithm is reported to have similar tendencies), the source characteristics played the most important role in convergence. In Fig. 3.4, typically observed convergence trends are depicted. Stationary GM(1) source generally showed a stable convergence property while nonstationary source such as speech signal [35] showed mostly unstable property (especially for non-zero λ values). The GM(1) source with regression coefficient a very close to 1.0 (e.g. $a > 0.99$), as may be expected, also showed unstable characteristics.

3.3.4. EC-DPCM and EC-PVQ special cases

As mentioned previously, both EC-DPCM and EC-PVQ can be considered as special cases of EC-CELP. However, in our early survey of literature before and up to the time of publication of EC-CELP results [32], we had only come across the EC-DPCM work [27]. Later we learned about two independent works on EC-PVQ [64], [87]. The authors of these two independent works did not seem to know about the other work. The design approach in the above schemes are direct extensions of EC-SQ and EC-VQ to differential configurations. More specifically, direct applications of EC-SQ and EC-PVQ to the scalar (case of EC-DPCM) and vector (case of EC-PVQ) quantizer design are made. For the recursive analysis-by-synthesis configuration of EC-CELP, such approach would have not been applicable. Moreover, as previous discussions suggested, the EC-DPCM and EC-PVQ algorithm using the special cases of EC-CELP is formulated in the more suitable recursive (predictive) frame work. The coupling and joint use of coding components in design formulations of EC-CELP allows for an implicit and joint design that merges the two steps of the previous algorithms. This coupling is especially beneficial in the case of adaptive source signals.[16] As later simulation results and analysis in chapter 4 show, the EC-DPCM algorithm resulting from EC-CELP ($N = 1$) is both easier and shows better numerical performance over the previously reported scheme [27]. Additionally this approach can easily be used for an adaptive EC-DPCM (an obvious easy extension of the algorithm of [27] is not possible). Similarly the additional step related to PC utilized in EC-PVQ design of [64], [87] is absent from the EC-PVQ as special case EC-CELP (with no ZSR). Hence in the case of EC-CELP algorithm, adaptiveness can be easily and implicitly incorporated.

3.4. Simulation Results and Conclusions

In the presentation of selected simulation results in this section, the following objectives are pursued. First, the benefits of the EC design strategy (EC-CELP) over the entropy-coded (with no EC) CELP configuration is shown. The second goal is to show the advantages of EC-CELP special case algorithms, configured as EC-DPCM and EC-PVQ, over the previously published algorithms for EC-DPCM and EC-PVQ. Finally simulation results

[16] For the non-adaptive case, from Eqn. 3.31 it is easy to see that by modifying the codebook of EC-PVQ, $\mathcal{V}_I = \{v'^{(i)} = Hv^{(i)}; i \in \mathcal{I}\}$, $\mathcal{I} = \{1, 2, \ldots, I\}$, we may obtain an equivalence between EC-PVQ and EC-CELP. Nevertheless, due to the difficulties associated with the choice of initial codebook and as a consequence of other advantages of EC-CELP over EC-PVQ during the design (particularly in the presence of quantization noise effects at low bitrates), EC-CELP should provide better overall results. For the adaptive case or nonstationary input signal, the advantages of EC-CELP over EC-PVQ are obviously more eminent.

are used to show the EC-CELP advantages in comparison with its special recursive cases and other non-recursive EC alternatives such as EC-VQ and EC-BTQ.

The results shown in this section are all based on a training sequence of 10^5 samples from the simulation of stationary GM(1) input source with various regression coefficients a. Therefore, gain and predictor adaptation are not used. We did not optimize the predictor coefficient and used the suboptimal $h = a$. As analysis of chapter 4 and our experiments showed, in the case of GM(1) input source, the effect of deviation of h from a does not have a sizable effect on the performance (adaptation is important in the case of nonstationary signals such as speech [35].). In our experiments, for block length values of $N \in \{1, 2, 3\}$ and $N = 8$, the initial codebook sizes used are $I = 512$ and 1024 respectively.

As was shown in section 3.2, performance of an (RC) entropy-coded CELP configuration for GM(1) input source is close to the rate-distortion-bound. However, in order to show the EC design advantage, we compared the delay (block size N) and complexity (codebook size $\dot{I}$) at a given rate. We observed that for the GM(1) source, in almost all cases of rate less than one bps, through EC design strategy, the delay N and complexity-related codebook size $\dot{I}$ was substantially reduced. Table 3.2 provides a comparison for some selected rates. Although we may expect this trend, the required delay and complexity for the high-quality EC-CELP are surprisingly low. This confirms the efficiency of the combination of PC, VQ, analysis-by-synthesis , and EC in the algorithm.

We now compare the performance of EC-DPCM and EC-PVQ using special case EC-CELP design algorithm with previously published results for EC-DPCM [27] and EC-PVQ [64]. The expected advantages of EC-CELP-special case algorithms are due to joint steps in the closed-loop design which implicitly combine the features. The resulting advantage manifests itself in two cases. First, at low rates when the quantization noise is high, the joint steps in the closed-loop design is more effective. Hence for the GM(1) input source with higher correlations, there is substantial improvement over the EC-DPCM results from [27]. This is seen in the results of simulations shown in Figure 3.5. The second case where the joint steps in the closed-loop design should provide a better alternative is in the case of a nonstationary input source. The previously published EC-PVQ adaptation strategy

	N (delay)	$\dot{I}$ (complexity)	R	SNR	a
CELP	7	128	1.0	18.5	0.98
EC-CELP	3	26	1.0	18.6	0.98
EC-CELP	1	4	1.0	16.7	0.98
CELP	8	32	0.6	16.0	0.98
EC-CELP	3	7	0.6	16.0	0.98
CELP	8	512	1.1	22.1	0.99
EC-CELP	3	31	1.1	22.4	0.99
EC-CELP	1	4	1.0	20.4	0.99
CELP	8	32	0.6	18.9	0.99
CELP	5	8	0.6	18.3	0.99
EC-CELP	3	7	0.6	18.9	0.99

Table 3.2 Performance comparison between CELP (entropy-coded without EC design) and EC-CELP, showing the EC lower delay and complexity advantages (a is the regression coefficient of GM (1) source).

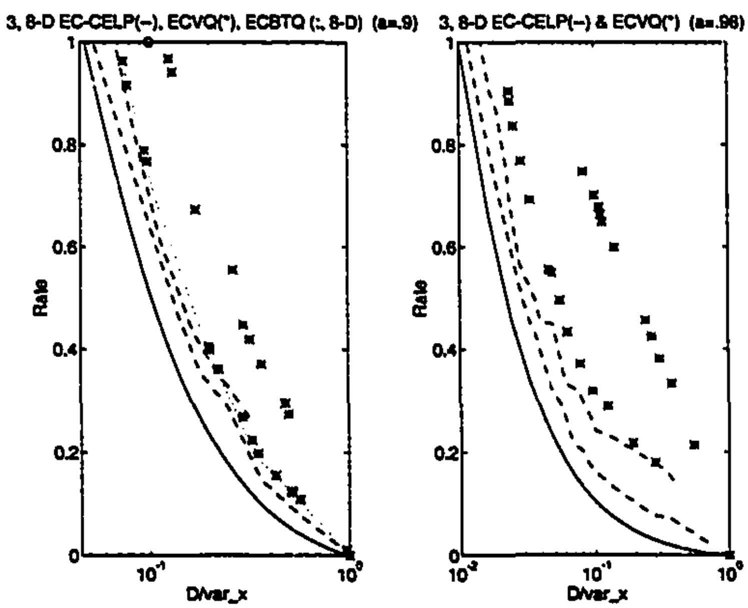


Fig. 3.5 Rate-distortion performance of EC-CELP (N = 3, 8, dashed), EC-VQ (N = 3, 8, *), EC-BTQ ($N = 8$, dotted, expected to be similar to EC-VQ for $a = .99$), and RDF (solid) for a=0.9 and a=0.98.

is a *classified adaptation*. One advantage provided by the EC-CELP-special case EC-PVQ over such algorithms would be the implicit and *continuous adaptation*, which in the case of nonstationary signals can be highly beneficial. Unfortunately, for better comparisons of EC-CELP with previous EC-PVQ algorithms, reference simulation results from [64], [87] with considerations of size of N and I were not available. However, the analytical analysis provided in chapter 4, predicts the possible gains (reduced size of I and N) of EC-CELP over special case EC-PVQ, especially in the nonstationary input signal case.

Figure 3.5 compares the EC-CELP and its special cases EC-DPCM performance with other EC alternatives. The comparison is with alternative coding configurations of EC-DPCM in [27] and EC-VQ in [14], and entropy constrained block transform quantization (EC-BTQ) [24]. The performance of EC-CELP was the closest to the RDF for a given low delay (small block length N) with relatively low computational cost (small codebook size I). EC-VQ results are our simulations (consistent with [14]) and EC-BTQ results are taken from the available results in [24]. As expected, the EC-CELP performance gap over EC-VQ increases as the correlation coefficient is increased (e.g. $a = 0.98$). Since the codebook size for rates below one bps becomes relatively small ($\bar{I} < I$), the CELP coder complexity is also low. More importantly, for highly correlated signals, the EC-CELP is the only coder which can achieve close to RDF with practical block length $N = 3$ and hence low delay. Although for $a = 0.98$, results for EC-BTQ were not available, it is safe to predict that like EC-VQ for block size $N = 8$, only a fraction of possible memory gain can be obtained using EC-BTQ (see chapter 4 for reasoning).

Results from the application of EC-CELP to speech signals in [35] and the video application results in chapter 5 extend the GM(1) model results of this chapter to more practical situations. The speech application also shows that the adaptation and EC strategy can be effective. Future work should include lossless coding considerations. Also more extensive investigations of full coders aimed at particular speech, image, and video applications is necessary.

Chapter 4

Low Rate Entropy-Coded Quantization Theory and Comparative Analysis of Entropy-Constrained Predictive Quantizers

4.1. Introduction

In chapter 3, the EC-CELP quantizer with its special cases of EC-DPCM and EC-PVQ was introduced. The performance advantage of the recursive quantization technique of EC-CELP over its special cases and the class of non-recursive EC quantizers such as EC-BTQ and EC-VQ were shown. This chapter is devoted to some new results on low rate entropy-coded quantization theory and a more rigorous analysis of various entropy-constrained predictive quantizers.

The contributions of this chapter are divided into two parts. (1) Results are provided on quantization theory for EC quantizers at low bitrate and operating on correlated sources. Coding gains and classifications under such conditions are suggested and calculated. These gain measures provide the reference for analysis of various quantizers in the second part. (2) For various quantization schemes, analytical analysis with numerical results are provided to give better insight into the EC-CELP performance advantages. Factors such as quantization noise effects at low bitrates, coding block size N (delay), and coding complexity are taken into consideration.

In our analysis, we confine ourselves to the Gauss-Markov processes. The choice of Gaussian is primarily due to the fact that we will be dealing with entropy-coded quantization for which effects of source symbol probability distribution function (PDF) or *shaping* is absent. Simplicity of analysis and consequences of the Central Limit Theorem for real world signals are other reasons for the choice of signal PDF.

The rigorous analytic expressions for *memory* and *filling* gains in the case of EC coders at high rates [80] which were presented in section 2.1.1, although informative, are not suited for low rates. Since the above categorization is a powerful tool and extremely useful for coding performance reference, in section 4.2, we make appropriate modifications to adapt and formulate analogous gains for low bitrates. The modification incorporates dependence on rate R as well as dimension N. The resulting categorization provides better understanding of the advantages of EC quantization at low rates.

In section 4.3, we use the analytical formulations of [3], [92], [93] on predictive coding (DPCM) and make an extension to the vector generalized cases of PVQ and CELP. These analyses provide insight into performance limits of various quantization schemes.

4.2. Rate-distortion theory analysis of maximum EC gains

In section 2.1.1 we reviewed the categorization of maximum EC coding gains over basic EC-SQ by Lookabaugh and Gray [80] to *memory* and *dimensionality filling* gains. However at low rates, analytic expressions for these maximum gains are not available and the high-rate formulations in equations 2.13 and 2.15 are not appropriate. In this section, we propose two *computational* formulations for the memory and dimensionality filling maximum EC coding gains. These formulations have an explicit dependence on average rate R as well as dimension N. As seen shortly, to obtain these gains we need the computed parametric RDF and N-th order RDF for GM(1). As well, the results are based on EC quantization theory and EC-VQ empirical performance. These gains will be used as an appropriate reference in analyzing the available low rate coding gains and in evaluating the performance of the EC coders at low rates.

Let us use the parametric formulas for $R(D)$ and ${}^{N}R(D)$ of GM(1) from section 2.1 and compute these functions for various regression coefficients a. Instead of normalized distortion in RDF (D/σ_s^2 versus R), we use SNR values dB $\left(\frac{\sigma_s^2}{D}\right)$ and denote these values by $\mathrm{SNR_{RDF}}$. For the GM(1) source, the resulting low rate $\mathrm{SNR}^{\mathrm{GM(1)}}_{\mathrm{RDF}}(R, N)$ and $\mathrm{SNR}^{\mathrm{GM(1)}}_{\mathrm{RDF}}(R) = \mathrm{SNR}^{\mathrm{GM(1)}}_{\mathrm{RDF}}(R, \infty)$ curves are shown in Fig. 4.1. The parametric formulation of RDF for correlated Gaussian sources [84], [7] has the well known graphical interpretation of "water-filling". The total average distortion D of the error spectrum "fills up" part of the area under the signal spectrum much as a liquid fills an irregular container. Frequency ranges which are completely "filled up" make no contribution to the information rate. The above parametric formulation for GM(1) RDF at low rates provided the intuition for the proposed computational approach in this section. The analytical formulation of such gains, as done at high rates, does not seem feasible.

Eqn. 2.16 expressed the total EC coding gain at high rates as the sum of memory and filling gains. To extend this formulation for low bitrates, we must incorporate the "water-filling" phenomenon as discussed above. This is done based on a parametric formulation and will use a numerical procedure detailed in the next two subsections. The resulting estimated maximum coding gains available to the EC coding is divided into *RDF memory gain* and *empirical space filling gain*. For the source with memory S, we may write

$$\Delta^{S}(N, R)\mid_{\mathrm{RDF}} \approx \Delta^{S}_{\mathrm{memory}}(N, R)\mid_{\mathrm{RDF}} + \Delta^{*}_{\mathrm{filling}}(N, R)\mid_{\mathrm{RDF}} . \tag{4.1}$$

Note that dependence on R is emphasized by introducing the functional dependence on R in the notation (compare this with Eqn. 2.12). For the GM(1), case we use the following

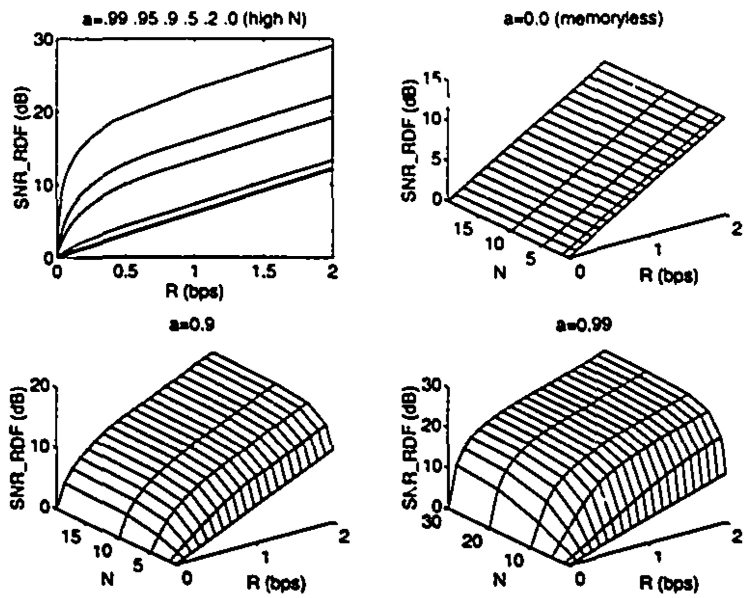


Fig. 4.1 RDF and N-th order RDF plots (SNR) for Gaussian and GM(1) sources with coefficient a (in the LHS top graph, the higher the graph the higher the a value). Note the significance of N and R in SNR figures and the relationship with values of coefficient a.

notation

$$\Delta^{\mathrm{GM(1)}}(N,R)\mid_{\mathrm{RDF}} \approx \Delta^{\mathrm{GM(1)}}_{\mathrm{memory}}(N,R)\mid_{\mathrm{RDF}} + \Delta^{\mathrm{i.i.d.\ Gaussian}}_{\mathrm{filling}}(N,R)\mid_{\mathrm{RDF}}. \tag{4.2}$$

4.2.1. RDF memory gain

For a given dimension N and rate R, and for the source with memory S, $\{s(n)\}_{n=0}^{\infty}$ and its corresponding memoryless source $\{s^{*}(n)\}_{n=0}^{\infty}$, we can describe the *RDF memory gain* as

$$\Delta^{S}_{\mathrm{memory}}(N,R)\mid_{\mathrm{RDF}} = \mathrm{SNR}^{S}_{\mathrm{RDF}}(N,R) - \mathrm{SNR}^{*}_{\mathrm{RDF}}(R), \tag{4.3}$$

where as in section 2.1, the memoryless source (*) can be unambiguously defined [108]. For the GM(1) source, the memoryless source is the memoryless (*i.i.d.*) Gaussian source, with the same variance,

$$\Delta^{\mathrm{GM(1)}}_{\mathrm{memory}}(N,R)\mid_{\mathrm{RDF}} = \mathrm{SNR}^{\mathrm{GM(1)}}_{\mathrm{RDF}}(N,R) - \mathrm{SNR}^{\mathrm{i.i.d.\ Gaussian}}_{\mathrm{RDF}}(R) \tag{4.4}$$

For the GM(1) source we use the explicit parametric formulations of RDF in section 2.1. To extend this to a general *source with memory* and its corresponding memoryless source (*), the memory gain definition relies on the availability of source RDF. When generation of typical memory and memoryless sources is possible, we may use the numerical algorithm of Blahut [8] for computation of the RDF to obtain the above maximum memory gain.

If the quantization method only uses block quantization of block size N (e.g. EC-VQ) the above memory gain can be interpreted as (similar to [80]) maximum memory gain over basic EC-SQ. For other quantization schemes such as DPCM or CELP which use recursive schemes (e.g. PC), the block size N has to be interpreted as the effective infinite block length. When prediction order in PC is appropriately selected, we may assume an infinite effective block length.

4.2.2. Space filling or dimensionality gain

Assuming relatively high correlation, at high rates and for the Gaussian memoryless source, as shown in Fig. 2.2 and indicated by the small gap between the EC-SQ rate-distortion performance and the RDF results in [14] [80], the filling gain advantage available through dimensionality advantage of EC-VQ is consequently small. Available filling gain at low rates decreases proportionally much like the memory gain does. Once again, for the filling gain we need to impose the dependence on rate R as well as the dimension N. We propose a new algorithmic definition for the filling gain whose computation particularly seems reliable at low rates. We make the assumption that the algorithmic performance of a well-designed N-dimensional EC-VQ is available. We can use such performance results to quantify the maximum achievable filling gain for a memoryless source. [1]

For low rates and general EC coders and for memoryless source (*), we propose the algorithmically computable maximum ***empirical space filling gain*** over basic EC-SQ as

$$\Delta^{*}_{\text{filling}}(N,R)\mid_{\text{RDF}} \approx \text{SNR}^{*}_{\text{EC-VQ}}(N,R) - \text{SNR}^{*}_{\text{EC-SQ}}(R). \tag{4.5}$$

[1] Such assumption is realistic since design of EC-VQ for low rates and low dimensions for memoryless sources has been successful.

For the GM(1) source the memoryless source (*), is the memoryless (*i.i.d.*) Gaussian source with the same variance.

4.2.3. Simulation results

To obtain numerical results for the maximum memory and filling gains of the GM(1) source we proceeded as follows. We first computed parametric RDF results shown in Fig. 4.1 and then simulated the EC-VQ rate-distortion performance results for various block sizes N. The EC-VQ simulations were based on 10^5 samples and were obtained using the EC algorithms in [14] [32]. Figures 4.2 and 4.3 show the simulation results for RDF memory and space filling gains for GM(1) input source. The RDF memory gains for GM(1) source with $a = 0.9, 0.99$, $\Delta^{\mathrm{GM(1)}}_{\mathrm{memory}}(N, R)\,|_{\mathrm{RDF}}$, are shown in two bottom graphs of Fig. 4.2. High N gain, $\Delta^{\mathrm{GM(1)}}_{\mathrm{memory}}(\infty, R)\,|_{\mathrm{RDF}}$, is shown in the top graph. Judging by the near RDF performance of EC-CELP at low rates, one may postulate that EC-CELP nearly provides the sum of empirical filling gain and the high N RDF memory gain (top graph) (due to PC's effective high N). Note that an EC coder SNR is approximately the $\mathrm{SNR}^*_{\mathrm{EC\text{-}SQ}}$+ coder memory gain+filling gain. As may be concluded from the results in Fig. 3.5, the combined memory and filling gains over EC-SQ of EC-CELP for a given N and R is the highest. Hence it yields the highest SNR. Next in section 4.3, we will provide a rigorous analysis of the memory gain for various EC quantizers which includes the effects of high quantization noise at low bit rates.

4.3. Memory gain in EC-CELP and other EC-quantizers

In section 4.2, it was seen that for relatively highly correlated sources, the memory gain is the dominant available coding gain, even at low rates. Nevertheless, due to reduced amount of gains, "every bit" of gain may count. Among known techniques to exploit memory redundancy in the source, (linear) PC, TC, and VQ have been popular. The objective, as mentioned before, is achieving the highest SNR at low rates with the additional constraints of minimum delay (dimension N) and low complexity and considerations with regard to nonstationary input signal. This section investigates the available memory gain to coders

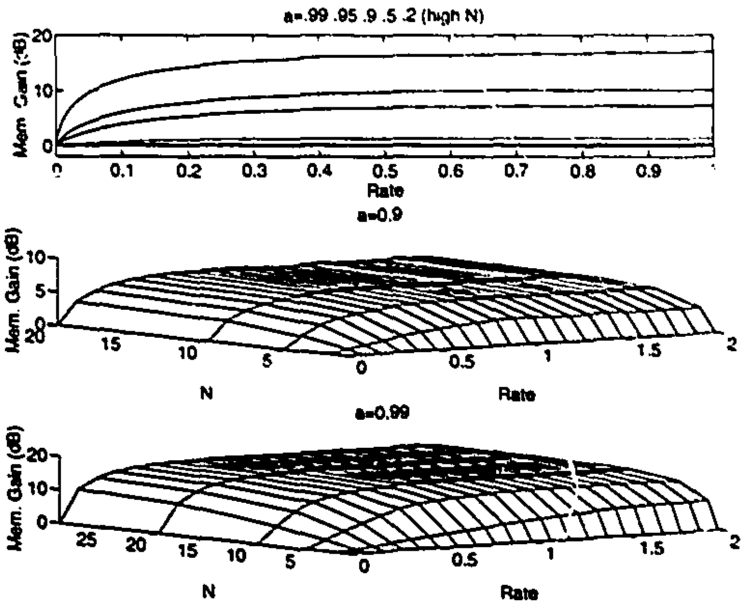


Fig. 4.2 RDF memory gain for GM(1) source. Graphs from top down: High and low N RDF memory gains for Gauss-Markov source with coefficient a.

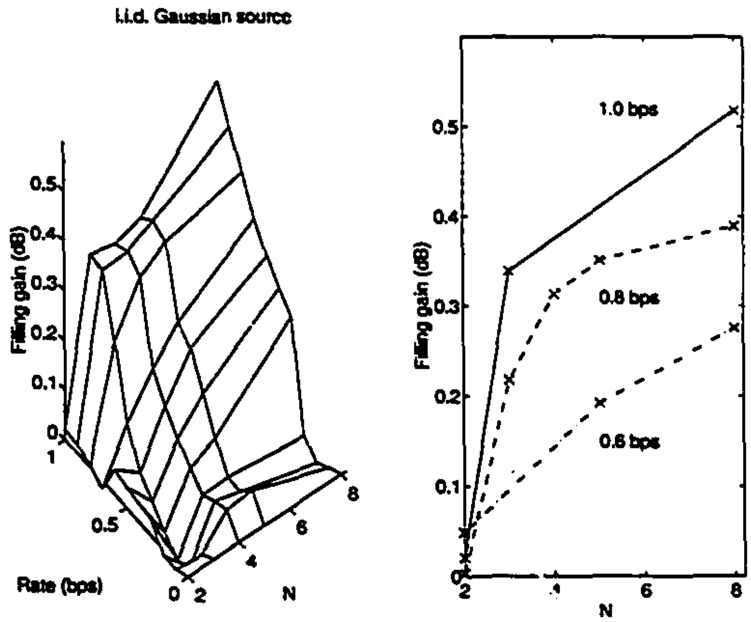


Fig. 4.3 Low rate empirical filling gain for (memoryless) Gaussian source.

which use these methods and suggests the full utilization of PC advantage as an important factor among available choices. We analyze various recursive PC schemes namely CELP, PVQ, and DPCM. We will see how at low bitrates a closer to optimal gain may become possible. Since at low bitrates, the quantization error feedback effect is not negligible, we have to take such effects into consideration.

Kolomogoroff provided the minimum prediction error variance μ_s^2 (described in terms of SFM in Eqn. 2.10) which also directly relates to the ideal maximum prediction gain

$$\mu_s^2 = \gamma_s^2 \sigma_s^2 \tag{4.6}$$

$$\Delta_{\text{memory, PC}} \big|_{\text{ideal}} = \mathrm{dB}\left(\frac{1}{\gamma_s^2}\right). \tag{4.7}$$

This gain is defined for the asymptotic case of high bitrates.

The simplest practical coder is the memoryless PCM quantizer or SQ. The error variance for SQ is sometimes expressed in terms of the useful factor of *quantizer performance factor (qpf)* [58],

$$\sigma_e^2 \triangleq \epsilon_{q,\text{SQ}}^2 \sigma_s^2. \tag{4.8}$$

The analysis in this section incorporates the low bitrate effect through the qpf. This means that higher qpf corresponds to coarser quantization. From the rate-distortion curves of a quantizer, we may estimate the correspondence between the qpf and the rate R using the SNR value of the quantizer for a given rate and source, $\text{SNR}_{\text{VQ}} = -\mathrm{dB}(\epsilon_q^2)$. For example assuming a memoryless Gaussian source and using the EC-SQ or EC-VQ rate-distortion curves we get the correspondence between rate and qpf in table 4.1. Note that for the case of recursive quantizers, the qpf is different for the effective quantizer whose input was previously denoted by the process $\{d(k)\}_{k=1}^{\infty}$. It is also important to note that the analysis of this section based on qpf has to viewed in the light of the low-rate rate-distortion analysis of section 4.2. The two approaches of these two sections provide complementary analysis with dependency on rate and error feedback. Finally, in these analyses, we assume that the PC order used for a given source is properly chosen. As a result, in the analysis of memory gain for the PC, we do not need to consider any dependence on the block size N.

	$R = 1$ bps		$R = 0.4$ bps	
	$N = 1$	$N = 4$	$N = 1$	$N = 4$
SNR	4.64	5.02	1.69	1.69
ϵ_q^2	0.34	0.31	0.68	0.68

Table 4.1 Correspondence between qpf ϵ_q^2 and rate R for Gaussian memoryless source as function of rate R and dimension N.

In other words, an infinite effective N can be assumed.

4.3.1. Review of PC memory gain for DPCM case

In this subsection we review PC memory gain in the DPCM quantizer. DPCM is the simplest quantization scheme which uses PC and may be characterized as a recursive SQ. The analysis of DPCM has already been the subject of a large number of previous publications. New analysis of PC for other quantization schemes in the preceding discussions will be essentially based on the conclusions of the previous section and extensions of the DPCM case.

To demonstrate the role of memory, the SQ error variance in Eqn. 4.8 should be contrasted against the DPCM case[2] given by

$$\min\{\sigma_e^2\}\,|_{\text{DPCM}} = \epsilon_{q,\text{DPCM}}^2 \gamma_s^2 \sigma_s^2, \quad \text{without feedback effect in PC.} \tag{4.9}$$

We assume that the qpf of the simple SQ case is approximately the same as the qpf of the SQ within the DPCM ($\epsilon_{q,\text{DPCM}}^2$),

$$\epsilon_{q,\text{DPCM}}^2 \approx \epsilon_{q,\text{SQ}}^2\,|_{\text{i.i.d. Gaussian}}\,.$$

Obviously similar assumption cannot be made about the qpf of the VQ and SQ. However, once the input source characteristics to the VQ is known, we may obtain approximations similar to the ones obtained for the memoryless Gaussian source in Table 4.1.

[2]Eqn. 4.9 is the special case of Eqn. 4.13, where the formulation includes the effect of feedback in PC.

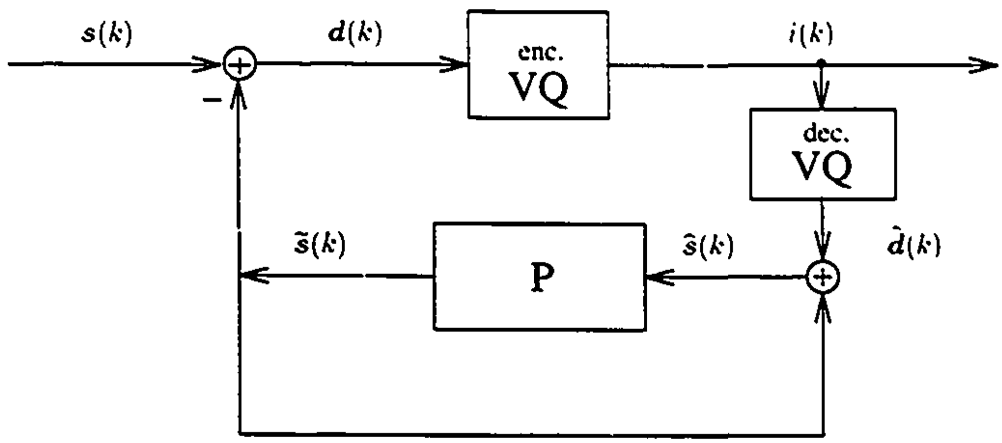


Fig. 4.4 PVQ encoder block diagram

The block diagram of the PVQ encoder is shown in Fig. 4.4. This commonly used representation, resembling the common DPCM structure, is equivalent to the special case CELP representation provided in chapter 3. There, it was shown that both DPCM and PVQ can be considered as special cases of the CELP quantizer.

In the current discussion, let us assume an additive (but not always white) noise model for the quantization error. Without loss of generality of our discussion in most cases, we will be using the GM(1) source with the regression coefficient a as the input source. The corresponding PC coefficient is denoted by h. Notations for DPCM correspond to the ones in Fig. 4.4 with SQ replacing VQ. As seen in the figure, PC is based on the reconstructed signal $\hat{s}(k)$, and an additive quantizer model is used, $\hat{d}(k) = d(k) - q(k)$. Using simple algebra describing the signals, we obtain

$$\begin{aligned} e(k) &= q(k), \\ d(k) &= s(k) - hs(k-1) + hq(k-1), \end{aligned} \tag{4.10}$$

Assuming vanishing correlation between the input signal and the quantization error, we get

$$\sigma_d^2 \mid_{\mathrm{DPCM}} = \frac{1 - 2ah + h^2}{1 - \epsilon_{q,\mathrm{DPCM}}^2 h^2} \sigma_s^2. \tag{4.11}$$

The effect of feedback in DPCM is known to result in the following interesting conclusions. For the additive quantizer model, although the output $\hat{d}(k)$ signal is white, the quantization noise $q(k)$ and the input $d(k)$ are both non-white, no matter how fine the quantization. However it is suggested that that $q(k)$ is close to Gaussian [3], [58]. Based on these conclusions, by using the near optimum predictor $h_{\text{opt}} \approx a$ in equations 4.11 and 4.10, we get the following performance limits for the PC in the DPCM context

$$\min\{\sigma_d^2\}\ |_{\text{DPCM}} = \frac{1-a^2}{1-\epsilon_{q,\text{DPCM}}^2 a^2}\sigma_s^2, \tag{4.12}$$

$$\min\{\sigma_e^2\}\ |_{\text{DPCM}} = \epsilon_{q,\text{DPCM}}^2 \sigma_d^2 = \epsilon_{q,\text{DPCM}}^2 \gamma_s^2 (1-\epsilon_{q,\text{DPCM}}^2 a^2)^{-1}\sigma_s^2, \quad \text{feedback effect.} \tag{4.13}$$

Comparing this with Eqn. 4.9, reveals the effect of feedback on PC that appears as the term in parentheses. The resulting memory prediction gain then may be defined as

$$\Delta_{\text{memory PC}}\ |_{\text{DPCM}} = \text{dB}\left(\frac{\sigma_s^2}{\sigma_d^2}\right) = \text{dB}\left(\frac{1-\epsilon_{q,\text{DPCM}}^2 a^2}{1-a^2}\right) = \text{dB}\left((\gamma_s^2)^{-1}(1-\epsilon_{q,\text{DPCM}}^2 a^2)\right). \tag{4.14}$$

In Fig. 4.5, it is shown that even for very high qpf at low rates, the deviation of h_{opt} from the value of a is very small and has little effect on DPCM performance gain (SNR or PC gain). This justifies the choice of $h_{\text{opt}} \approx a$. However from Eqn. 4.13 and Fig. 4.5, it is easy to see that at low rates, where the qpf has a larger value, the memory gain due to PC has a sizable decrease. It is due to such effect that VQ memory gain can supplement the PC effective memory gain and improve the qpf. Hence through the use of PC, close to maximum memory gain at lower rates becomes possible.

4.3.2. Prediction gain in CELP

Using the extension of the analysis used for DPCM, in this and the next subsection, we derive expressions for prediction memory gain for CELP and PVQ. For the case of CELP, since there is no direct quantizer in the usual sense (mapping from an input to an output), we use its *effective quantizer*. The details of such an effective quantizer were explained in chapter 3. This allows us to extend the DPCM formulation to the case of CELP. This *effective quantizer*, just like the standard quantizer, has an input and an output and

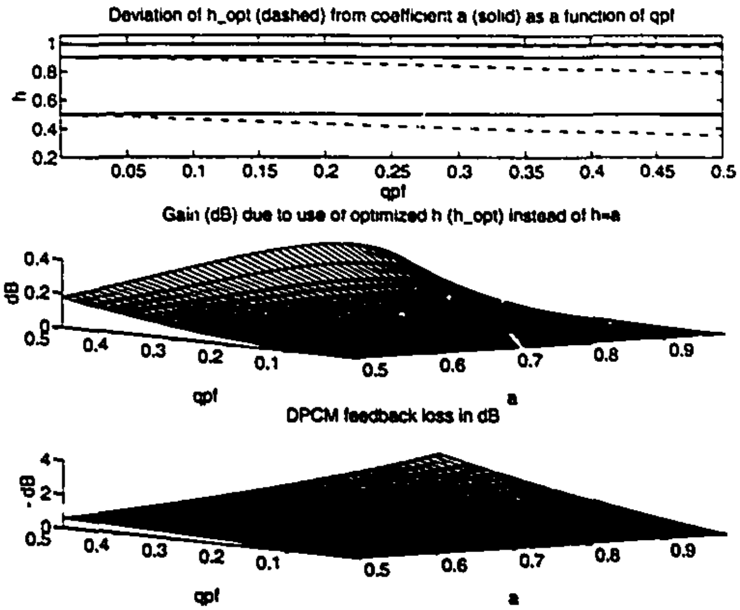


Fig. 4.5 Top two graphs show deviation of h_{opt} from matched $h = a$ in DPCM and corresponding gain in PC memory gain (or DPCM SNR). Bottom graph shows loss in PC memory gain due to feedback in DPCM. These curves will also be valid for CELP with a lower qpf value.

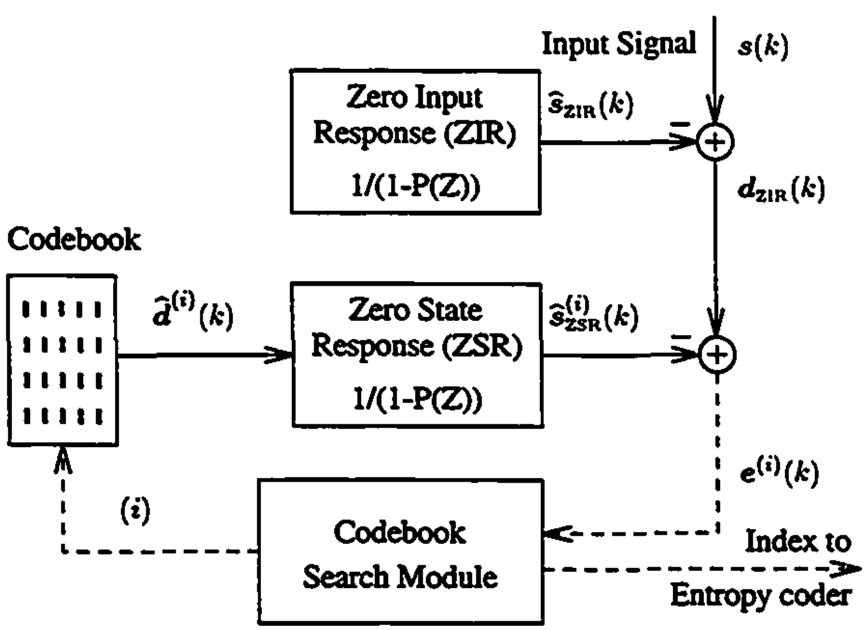


Fig. 4.6 Lossy encoder in EC-CELP: encoder (CELP) block diagram.

is based on an additive noise model $q(k)$. Again a GM(1) input source is assumed for simplicity. To avoid repetition we first refer the reader to the notations and description of signals used in chapter 3. The block diagram of the CELP coder is shown in Fig. 4.6. The codebook search module in CELP minimizes the squared error described by Eqn. 3.22. The effective quantizer was shown in Eqn. 3.27 to be

$$\text{(CELP)} \quad \boldsymbol{d}(k) = \boldsymbol{d}_{\text{ZIR}}(k) - (H(k) - \underline{I})\,\hat{\boldsymbol{d}}(k). \tag{4.15}$$

which allows us to rewrite the error in the normal fashion of (vector) quantizer

$$\boldsymbol{e}(k) = \boldsymbol{d}(k) - \hat{\boldsymbol{d}}(k). \tag{4.16}$$

Note that the second term of the RHS in Eqn. 4.15, which represents the ZSR, results from the analysis-by-synthesis operation and is absent in the case of PVQ. Using the above *effective VQ*, we write the vector extension of DPCM derivations for PC memory gain for the CELP

$$\hat{\boldsymbol{d}}(k) = \boldsymbol{d}(k) - \boldsymbol{q}(k), \tag{4.17}$$

$$\boldsymbol{d}(k) = \boldsymbol{s}(k) - \tilde{\boldsymbol{s}}(k), \tag{4.18}$$

$$\boldsymbol{e}(k) = \boldsymbol{s}(k) - \hat{\boldsymbol{s}}(k), \tag{4.19}$$

$$\hat{\boldsymbol{s}}(k) = \hat{\boldsymbol{d}}(k) + \tilde{\boldsymbol{s}}(k), \tag{4.20}$$

$$\tilde{\boldsymbol{s}}(k) = \hat{\boldsymbol{s}}_{\text{ZIR}}(k) + (H(k) - \underline{I})\,\hat{\boldsymbol{d}}(k) \tag{4.21}$$

$$= \underbrace{\overbrace{\begin{bmatrix} h \\ h^2 \\ \cdots \\ h^N \end{bmatrix} \hat{s}_N(k-1)}^{\text{inter-block memory}}}_{\hat{\boldsymbol{s}}_{\text{ZIR}}} + \underbrace{\overbrace{\begin{bmatrix} 0 & \cdots & 0 \\ h & \cdots & 0 \\ \cdots & \cdots & \cdots \\ h^{N-1} & \cdots & h \end{bmatrix} \begin{bmatrix} \hat{d}_1(k) \\ \hat{d}_2(k) \\ \cdots \\ \hat{d}_N(k) \end{bmatrix}}^{\text{intra-Block memory}}}_{\text{due to ZSR filtering}}. \tag{4.22}$$

Again for the PVQ case, the intra-block PC memory or the RHS in the above equation will be absent. Following similar algebra as for the DPCM case, we may now derive similar results for the CELP. For the CELP coder we get,

$$e(k) = q(k) \tag{4.23}$$

$$d(k) = \begin{bmatrix} s_1(k) \\ s_2(k) \\ \cdots \\ s_N(k) \end{bmatrix} - h \begin{bmatrix} s_N(k-1) \\ s_1(k) \\ \cdots \\ s_{N-1}(k) \end{bmatrix} + h \begin{bmatrix} q_N(k-1) \\ q_1(k) \\ \cdots \\ q_{N-1}(k) \end{bmatrix}. \tag{4.24}$$

Let us assume equal per sample quantization noise characteristics for all samples within the VQ block. As before let us also assume a vanishing correlation between input and quantization error. Using similar steps as for DPCM for each sample in the vector d we get identical formulation as for DPCM for PC memory gain and hence

$$\sigma_d^2 \mid_{\text{CELP}} = \frac{1 - 2ah + h^2}{1 - \epsilon^2_{q,\text{CELP}} h^2} \sigma_s^2, \tag{4.25}$$

$$\Delta_{\text{memory PC}} \mid_{\text{CELP}} = \text{dB}\left(\frac{1 - \epsilon^2_{q,\text{CELP}} a^2}{1 - a^2}\right) = \text{dB}\left(\gamma_s^{-2}(1 - \epsilon^2_{q,\text{CELP}} a^2)\right). \tag{4.26}$$

As the dimension N of the VQ increases, we suggest that the following approximation for the ϵ^2_q, CELP value,

$$\epsilon^2_q, \text{CELP} \mid_{\text{GM(1)}} \approx \epsilon^2_q, \text{VQ} \mid_{\text{i.i.d. Gaussian}},$$

decreases (Table 4.1). In other words, due to use of VQ in CELP ($N \geq 1$), for a given rate, the qpf value will be the same or better (lower) than the one in DPCM case ($N = 1$). Hence for the same rate, the PC memory gain of CELP will be equally good or better than the one of DPCM. i.e. There will be less quantization error feedback in CELP,

$$\epsilon^2_{q,\text{CELP}} \leq \epsilon^2_{q,\text{DPCM}} \Rightarrow \Delta_{\text{memory PC}} \mid_{\text{CELP}} \geq \Delta_{\text{memory PC}} \mid_{\text{DPCM}}. \tag{4.27}$$

Evidently in the CELP case, VQ may also provide additional (not removed by PC) available memory gain. The total memory gain is the sum of PC and VQ memory gains.

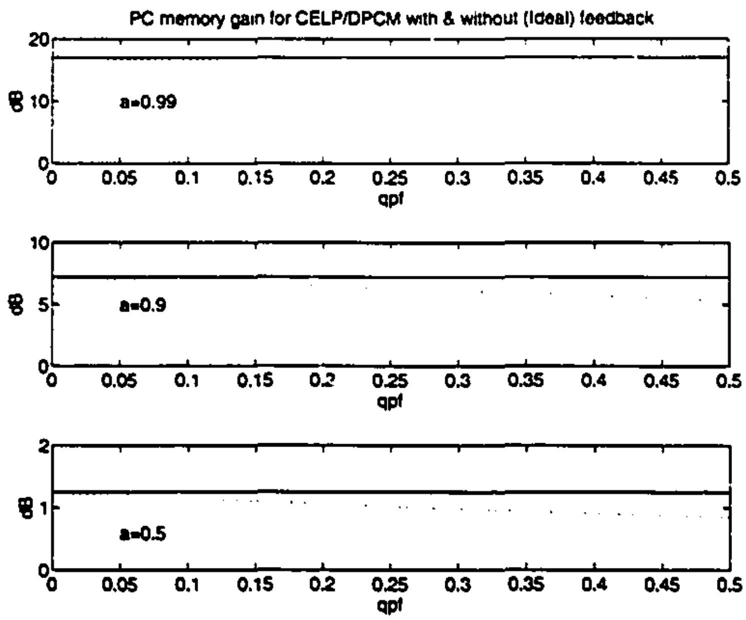


Fig. 4.7 PC memory gain for CELP and DPCM with (dot) and without (solid) feedback effect. The input source is a GM(1) with $a = 0.9, 0.99$.

Due to the similarity of the CELP PC memory gain expressions to the ones for the case of DPCM, the results for the DPCM case in figures 4.5 ($h_{opt} \approx a$), 4.7, and 4.5 will also valid for the CELP case. Again, for the case of CELP we need to make the adjustment that for the same input source to the effective quantizer, the qpf will slightly decrease as values of N higher than one are used.

4.3.3. Prediction gain in PVQ

VQ is the common feature between PVQ and CELP. However, in these coders PC is not used in exactly the same fashion. The difference is that there is some loss in the PC memory gain for the PVQ. This will become clear as PC memory gain for PVQ is formulated next. This is done by eliminating the intra-block term in the expressions derived for CELP. As a result, we will get an "unbalanced" (unequal for samples within a vector) quantization noise effect as reflected in the following expression for the $d(k)$

$$e(k) = q(k), \tag{4.28}$$

$$d(k) = \begin{bmatrix} s_1(k) \\ s_2(k) \\ \cdots \\ s_N(k) \end{bmatrix} - \begin{bmatrix} h \\ h^2 \\ \cdots \\ h^N \end{bmatrix} s_N(k-1) + \begin{bmatrix} h \\ h^2 \\ \cdots \\ h^N \end{bmatrix} q_N(k-1). \tag{4.29}$$

In appendix B, we calculated the coefficients of the predictor for the case of GM(1) source, when the quantization noise effect is not taken into consideration. These coefficients are used in the above expression. Using similar assumptions as for CELP now we can derive the "unbalanced" expressions for quantizer input sample variances with the vector (size N)

$$\sigma^2_{d_i} = \sigma^2_s \left(1 + h^{2i} - 2\,(ah)^{2i} + h^{2i}\epsilon^2_{q_N}\sigma^2_{dN}\right), \quad i = 1, \ldots, N-1, \tag{4.30}$$

$$\sigma^2_{dN} = \sigma^2_s \frac{(1 + h^{2N} - 2(ah)^{2N})}{1 - \epsilon^2_{q_N} h^2}, \tag{4.31}$$

$$\sigma^2_d = \frac{1}{N}(\sigma^2_{dN} + \sum_{i=1}^{N-1} \sigma^2_{di}). \tag{4.32}$$

Using the above expression, in top graph of Fig. 4.8, we demonstrate that despite different structure in PVQ difference signal d, the optimum h value is also close to a. But the final objective of this discussion was to compare the PC memory gains. Using the above relationships we obtain a closed form expression for the PVQ PC memory gain which again includes the effects of feedback

$$\Delta_{\text{memory PC}}\,|_{\text{PVQ}} = \text{dB}\left(\frac{N\left(1 - \epsilon^2_{q,\text{PVQ}}a^{2N}\right)(1 - a^2)}{N(1 - a^2)\left(1 - \epsilon^2_{q,\text{PVQ}}a^{2N}\right) - \left(a^2 - a^{2(N+1)}\right)\left(1 - \epsilon^2_{q,\text{PVQ}}\right)}\right). \tag{4.33}$$

Unlike the CELP case, the qpf for PVQ cannot be assumed to be close to qpf of VQ for *i.i.d.* Gaussian source. In this case since VQ input still carries intra-block correlations, the VQ in PVQ has a higher (worse) qpf than the CELP case. Nevertheless we make this assumption in view of the fact that the loss in PC memory gain due to lack of intra-block memory is much more significant than the small degradation for small N and R.[3] The PC

[3]Note that theoretically speaking, in the case of the stationary input signal, if the design of the VQ

memory gain for the case of ideal PVQ (when no feedback effect) in Eqn. 4.33 simplifies to

$$\Delta_{\text{memory PC}}\mid_{\text{PVQ}} = \text{dB}\left(\frac{(1-a^2)}{N(1-a^2)-\left(a^2-a^{2(N+1)}\right)}\right).$$
$$= \text{dB}\left(\frac{1}{1-\frac{1}{N}\sum_{k=1}^{N} a^{2i}}\right) \quad \text{without feedback effect.} \qquad (4.34)$$

As seen in the bottom two graphs in Fig. 4.8, as a result of lack of intra-block PC, as the size of N is increased, there is loss in the overall PC memory gain. The loss is more significant (up to 6 dB) in the case of highly correlated GM(1) process with $a = 0.99$ for the vector length $N = 10$. In EC-CELP the analysis-by-synthesis feature provides the intra-block PC gain and avoids such loss. The effect of quantization noise feedback, or the difference between the equations 4.33 and 4.34 is depicted in Fig. 4.9.

4.3.4. Memory gain for VQ and TC

Eqn. 2.13 gave the general maximum memory gain formulation as a function of dimension size N. For the jointly Gaussian process, the following explicit formulation can be computed [80] (Eqn. 2.14)

$$\text{High rate:} \quad \Delta_{\text{memory}}^{\text{Gaussian}}(N)\mid_{\text{VQ}} = \text{dB}\left(\frac{\sigma_s^2}{\mid {}^N R_{ss}\mid^{1/N}}\right). \qquad (4.35)$$

For the GM(1) process, further simplification leads to the familiar formulation below

$$\text{High rate:} \quad \Delta_{\text{memory}}^{\text{GM(1)}}(N)\mid_{\text{VQ}} = \text{dB}\left((1-a^2)^{-\frac{N-1}{N}}\right). \qquad (4.36)$$

TC, which can be considered as a constrained VQ [41], will provide the maximum memory gain when the Karhunen-Loeve optimum transform (KLT) is used. Hence the

codebook in PVQ is properly done, the VQ memory gain should almost compensate for the intra-block PC memory loss (see previous chapter). Nevertheless, such task may not prove to be fully practical due to the difficulties associated with the choice of initial codebook in the design and quantization noise effects at low bitrates. Moreover, for the nonstationary input signal case which is often of interest, the PC memory gain and other advantages of CELP over PVQ will obviously result in better overall PC+VQ memory gain and coder performance.

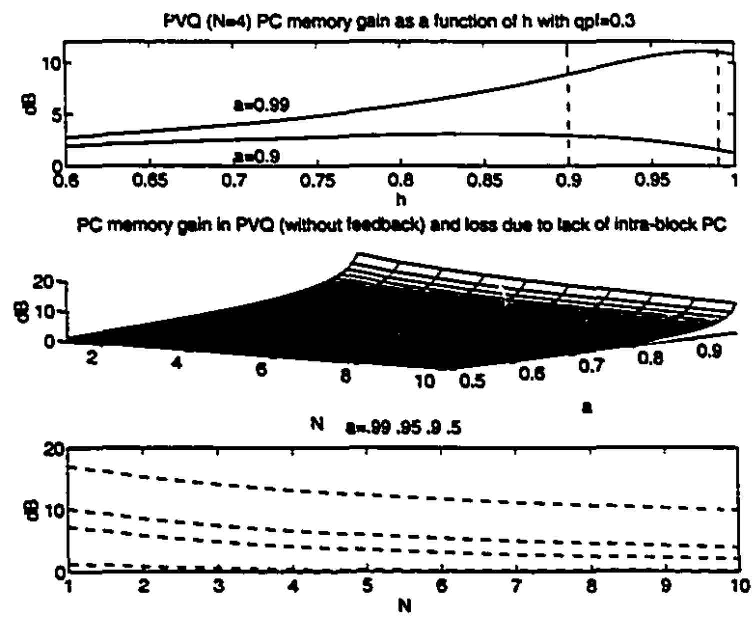


Fig. 4.8 Top graph: PC memory gain variation for an $N = 4$ dimensional PVQ as a function of h value. The input source is the GM(1) source with $a = 0.9, 0.99$. The qpf is fixed at $\epsilon^2_{q,\mathrm{PVQ}} = 0.3$. The memory gain for optimum h is only nominally better than the matched case of $h = a$. Bottom graphs: PC memory gain for PVQ (with no feedback effect), showing the loss due to lack of intra-block PC for various a values.

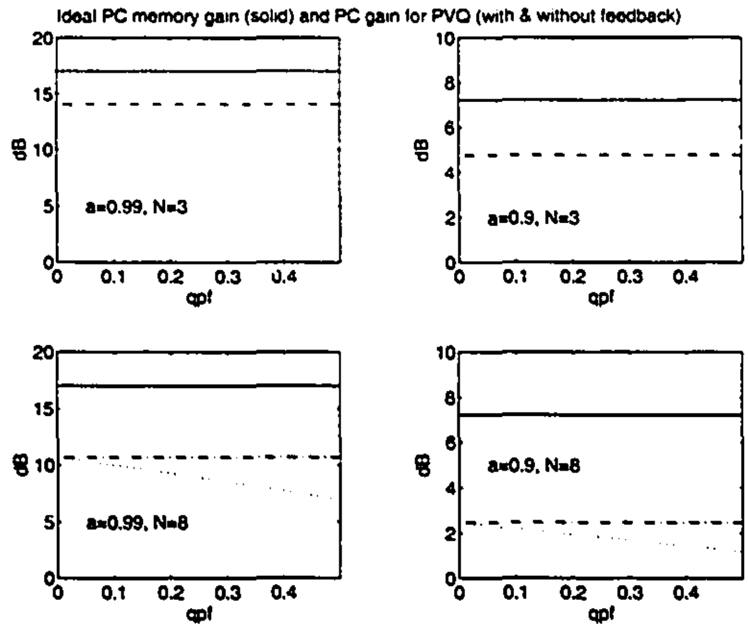


Fig. 4.9 PC memory gain for PVQ with VQ of dimension $N = 3, 8$ with (dot) and without (dash-dot) feedback effect. The Ideal PC gain is shown by solid lines. The input source is a GM(1) with $a = 0.9, 0.99$.

TC maximum memory gain for the GM(1) source will also have the form [58]

$$\text{High rate:} \quad \Delta^{\mathrm{GM(1)}}_{\mathrm{memory}}(N)\mid_{\mathrm{TC}} = \Delta^{\mathrm{GM(1)}}_{\mathrm{memory}}(N)\mid_{\mathrm{KLT}} = \mathrm{dB}(1-a^2)^{-\frac{N-1}{N}} \tag{4.37}$$

Due to non-recursive nature of VQ and TC schemes, it is clear that there is no need for feedback considerations at low bitrates. Rather than using qpf, we can directly use the rate dependent results from section 4.2. Hence, at lower rates we have the maximum memory gains

$$\Delta^{\mathrm{GM(1)}}_{\mathrm{memory}}(R,N)\mid_{\mathrm{VQ}} = \Delta^{\mathrm{GM(1)}}_{\mathrm{memory}}(R,N)\mid_{\mathrm{TC}} = \Delta^{\mathrm{GM(1)}}_{\mathrm{memory}}(R,N)\mid_{\mathrm{RDF}}, \tag{4.38}$$

where $\Delta^{\mathrm{GM(1)}}_{\mathrm{memory}}(R,N)\mid_{\mathrm{RDF}}$ was given in Eqn. 4.4.

Throughout this thesis, the objective has been to obtain close to RDF performance with low delay and complexity for sources with memory in general and highly correlated source in particular. For such sources, a significant part of memory extends beyond the block size

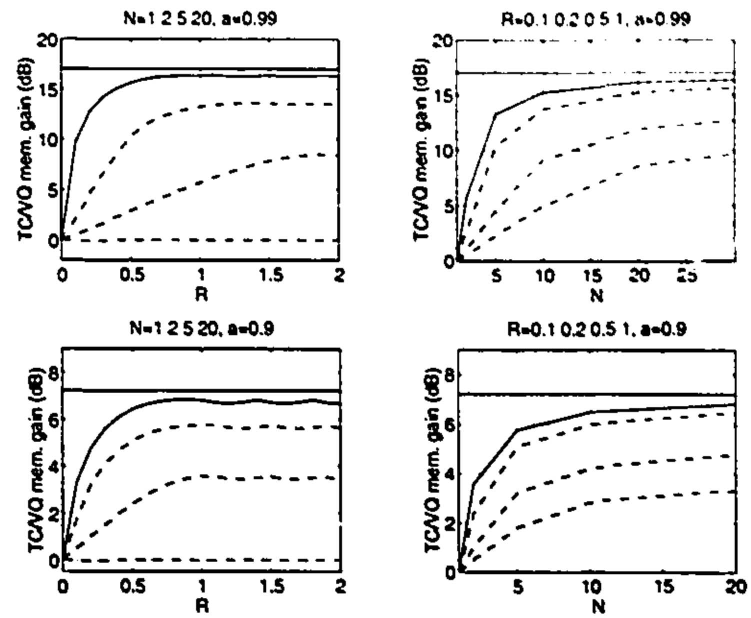


Fig. 4.10 Analysis to show a comparison of theoretical memory gains between VQ/TC. Straight solid lines show the ideal PC memory gain. The PC memory gain will be similar to the ones for TC/VQ for large N (effective N is PC)

N. Therefore low delay non-recursive quantizers such as VQ and TC cannot provide full memory gain advantage. On the other hand, such techniques may have advantages over PC in the case of highly nonstationary signals, where PC adaptation may be difficult. However, for the locally stationary or slowly time-varying signals such as speech and temporal video signals, (adaptive) PC with its recursive advantage is more effective. Common examples for the above two cases are the use of DCT in spatial video coding and the use of CELP in speech coding. For the temporal video coding application considered in this thesis, the PC (commonly used DPCM or the proposed CELP configuration in chapter 5) would clearly be preferred. As shown in Fig. 4.10, for the GM(1) source with regression coefficient 0.99, modeling the intensities along MTs in video signals, the required delay by VQ or TC which could provide sufficiently high memory gain is not practical. From the analysis in this section, it is clear that the evaluation of memory gain in recursive and non-recursive alternatives highly depends on the particular input source and application.

4.4. Delay and complexity

In section 4.2, we saw that at lower rates, due to the "water-filling" phenomenon, the size of RDF maximum available coding gains are reduced. In section 4.3, we separately analyzed the memory gain for PC and VQ. Such analysis has to be considered in the light of RDF maximum available gains of section 4.2. The analysis of those two sections facilitated the assessment of tradeoffs in using various schemes (DPCM, PVQ, CELP) and provided means to assess the overall performance of a coding configuration. We have been using the quantizer rate-distortion performance, delay, and complexity to evaluate the effectiveness and efficiency of the technique. In this section, once again we review and reconsider some delay and complexity issues.

From the analysis of section 4.3 for PVQ, one may suggest that VQ can partly exploit the memory which PC fails to remove. However, this can be both inefficient and ineffective. It is clear that the advantage of CELP over PVQ is in removing "maximum" possible memory available to PC. This allows the VQ in CELP to focus on the residual memory in the (nonstationary) signal and on the available filling gain. Therefore, the result of better efficiency not only may appear in rate-distortion gain for a given delay N, but also in the size I of the final codebook (complexity). In our experiments these trends were clearly observed. i.e. the codebook of a less efficient system had to be richer. This often meant a larger N and I.

In chapter 3, we first showed that entropy coded CELP without EC feature provides near rate-distortion performance. Design under the EC criterion resulted in improving the quantizer efficiency. The EC design not only reduces the required block size N or delay, but also the codebook size I which directly dictates coding complexity. Table 4.2 shows the delay/complexity results for GM(1) source with regression coefficient 0.9. More results were provided in Table 3.2.

Previously, we saw that the special cases of design algorithm of EC-CELP for EC-DPCM and EC-PVQ showed performance advantages over the previously reported similar algorithms (see chapter 3 for a comparison with EC-DPCM results of [27] with special case EC-CELP). Close to rate-distortion performance for EC-PVQ has been reported for

	N (delay)	$\bar{I}$ (complexity)	R	SNR
EC-CELP	3	28	1.0	11.8
EC-CELP	1	4	1.0	10.7
EC-CELP	3	4	0.6	9.0

Table 4.2 Examples of order of coding complexity (codebook size $\bar{I}$) and delay (N) or various coders for GM(1) input source with $a = 0.9$.

GM(1) with coefficient $a = 0.9$ [64], [87]. Since the results in [64], [87] are limited and do not include codebook size, proper comparisons with EC-CELP to support the theoretical conclusions in this chapter would not be possible. Nevertheless for higher correlations, from the analysis here we may expect that the performance may not be as close to the rate-distortion of EC-CELP, particularly in the case of nonstationary input signal. In Table 3.2, the improved coder efficiency, resulting from addition of EC feature (EC-CELP over CELP), was shown to be reflected in the reduced codebook size I or complexity and in the size of delay N. Similarly here, the improved coder efficiency of EC-CELP over EC-PVQ, resulting from addition of intra-block PC feature, will also be reflected in reduced codebook size I or complexity and in the size of delay N.

4.5. Conclusion

In section 4.2, a new formulation for low rate entropy-coded quantization gains was devised. Based on low rate rate-distortion theory, we formulated RDF memory gain and empirical space filling gains for EC coding. Using GM(1) source, we evaluated and verified these formulations. In section 4.3, we presented new analytical results for available PC memory gain in EC-CELP and its special cases. These analyses in conjunction with RDF gains in section 4.2 provided better insight for the low rate quantization using alternative schemes. The benefit of these tools is not limited to the EC quantization but can also extend to the RC case. We paid special attention to the analysis of PVQ in comparison with CELP. This provided a clear picture of different combinations of PC and VQ and differences of CELP and PVQ. These differences can be a source of confusion, as we have experienced this in discussions with some of the source coding researchers (particularly the ones who are not very familiar with CELP).

The analyses of this chapter have shown how EC-CELP configuration allows for the joint full benefits of PC and VQ (particularly in the nonstationary input signal case). The analysis presented addressed the contradicting issues of high quality and low delay and complexity in source coding of (nonstationary) sources with memory. As a consequence of efficient PC memory removal in EC-CELP, VQ can concentrate on the remaining memory redundancies[4] and the filling gain. This makes the low delay and complexity high quality quantization possible.

In this thesis, the name EC-CELP is chosen to implicitly imply adaptiveness. Adaptiveness is both in the sense of PC adaptation and the VQ capability to remove the linear and non-linear memory. In this chapter, we did not consider an extensive analysis of the adaptive features of EC-CELP. Nevertheless, many of the issues discussed, would still have pertinence. This is reflected in the results of the application of EC-CELP to speech signals [35]. More rigorous analysis has been reported for the Adaptive DPCM [39], [67]. Future work may extend such analysis to the case of CELP.

[4] mostly due to coarse quantization

Chapter 5

Multi-Frame Recursive Motion-Compensated Image Sequence Coding Using EC-CELP

5.1. Introduction

In this chapter we introduce and evaluate a new motion-compensated image sequence coding technique. The proposed new video coder has a recursive (predictive) multi-frame VQ configuration. The quantization technique of EC-CELP in chapter 3 is used to provide high performance low delay quantization in the temporal domain.

Chapter 2 introduced the motivation and context of high compression video coders. We mentioned that in motion-compensated video coding, the input source to the quantizer is the set of intensity signals along the MTs. This source is modeled as a highly correlated GM(1) source with a regression coefficient close to one. It was then suggested that for such highly correlated source and at low bitrates, the commonly used temporal DPCM quantization is in its performance saturation region. Using GM(1) models, chapters 3 and 4 quantified the above suggestion and offered the alternative of EC-CELP. Simulation results and theoretical analyses demonstrated close to optimal performance of the EC-CELP quantization technique under the assumed coding conditions and source model.

Advantages of EC-CELP were shown over non-recursive and other alternatives.

The goal of this chapter is to provide a framework, a feasibility study, and a preliminary evaluation of a new recursive multi-frame video coding configuration based on EC-CELP quantization. Prior to this work only multi-frame non-recursive video coding configurations have been suggested [94]. As seen later in the chapter, there are a number of issues and problems pertinent to the proposed new coding system. For instance, a new motion estimation scheme suitable for the multi-frame recursive configuration is needed. Also since the outputs of the temporal EC-CELP quantizer are codebook indices, spatial lossy coding in the usual fashion is not possible. We will suggest a number of solutions to the above problems and we will use simulations to examine the performance of the proposed solutions. We will show that significant bitrate reduction over the conventional schemes using the proposed temporal compression technique is possible. Full treatment of the above and other issues for this new system will be beyond the scope of this thesis and should be the subject of future research.

The chapter is organized in the following manner. In section 5.2, we introduce the non-recursive and recursive multi-frame video coding techniques. In section 5.3, we present several motion estimation schemes and a suitable configuration for the proposed recursive multi-frame coding system. The temporal redundancy reduction scheme using EC-CELP is presented in section 5.4. The full multi-frame hybrid EC-CELP image sequence coding system is described in section 5.5. Simulation results along with discussions on alternative schemes are given in section 5.6. Appendix C provides some details for the basic multi-frame MT estimation algorithm.

5.2. Multi-Frame Motion-Compensated Video Coding

Before introducing the multi-frame motion-compensated coding configurations, the presentation of the following remarks and underlying assumptions is in order. The efficiency of the motion-compensated hybrid video coding (Fig. 2.4) or any alternative video coding technique is dependent on the quality and accuracy of its components, namely spatio-temporal redundancy removal and quantization. In particular, the accuracy of motion estimation and compensation plays an important role in the temporal redundancy removal and the overall performance of the coder. In fact it has been suggested [43] that if an accurate motion compensation is available, the additional gain by the spatial encoding of the prediction error in the hybrid DPCM configuration will be small. To incorporate this conclusion and to maximize the motion compensation gain, in this chapter we will use the *dense* motion-compensation of [20], [66] as opposed to the commonly used block matching techniques [94]. The general formulation of dense motion-compensated techniques with fractional pixel accuracy that is used encompasses cases of pixel or subpixel accuracy block matching techniques. As well, when desired, such a formulation can be adapted for the region-based video coding systems. It is obvious that the gain due to motion compensation highly depends on the particular input sequence. It has been also suggested [43] that such gain for moving areas is limited. On the other hand, as was pointed out in section 2.3, the coder total bitrate includes the allocated bitrate for the motion parameters (Eqn. 2.19),

$$R = R_{\text{intensity}} + R_{\text{motion parameters}}.$$

Any high performance motion-compensated coding scheme should take the tradeoffs reflected in the above formula into consideration. Some detailed discussion concerning such tradeoffs for hybrid video coding systems is found in [42], [43]. In summary, one may conclude that in using motion-compensated coding, particularly at low bitrates, the benefits and tradeoffs of motion compensation have to be treated with some caution.

The primary goal of this chapter being the evaluation of the temporal quantization component in the video coder, we need to minimize the effects of the above tradeoffs in our evaluation. To do this, we assume that all coding systems evaluated benefit equally from the dense motion compensation and consequently the cost $R_{\text{motion parameters}}$ can be assumed to be equal for all systems. Therefore the presented bitrates will only be for the $R_{\text{intensity}}$. Additional considerations for the above tradeoff should be investigated as a topic of future research.

Finally, some treatment of the spatial redundancy removal aspect of the proposed recursive multi-frame coder will be investigated but will be postponed until section 5.5. However for the rest of this chapter, to simplify the simulations, we have assumed that due to accuracy of motion parameters and efficiency of temporal compression, the additional gain by the intra-frame spatial encoding is small ([43]). As a result we will not be considering the effects of such gains in our simulations. This assumption or condition is again similar for all coding configurations compared. It is obvious that a target maximum compression video coder can derive a benefit (however small) from the exploitation of the spatial coding gains. Nevertheless the above assumption permits us to conduct the required simulations within the scope of this thesis and to draw important conclusions.

We now proceed with the introduction of the multi-frame coding configuration. As mentioned, the motivation for using a multi-frame video configuration is to overcome the limitations of single-frame or temporal differential scalar quantization (DPCM) at low bitrates. The aim is to apply the theoretical conclusions of previous chapters on EC-CELP quantizer (and its special cases) operating on the GM(1) source model to the video signals. Other than the multi-frame (VQ) quantization advantage, due to additional data available in the case of multi-frame motion estimation, more effective motion compensation should be possible (assuming the use of a suitable multi-frame motion estimation model). This can also lead to better temporal redundancy reduction through better motion compensation.

Fig. 5.1 shows a proposed general *motion-compensated multi-frame image sequence coding* configuration. The encoding process involves multi-frame motion estimation (module $\underline{2}$) and compensation (module $\underline{3}$) and multi-frame quantization (module $\underline{4}$). The decoder performs the inverse operations (modules $\underline{5}$ and $\underline{6}$). The *buffer* module $\underline{1}$, groups a sequence of image frames into multi-frame coding blocks or *coding image decks* of temporal size N,

$$D(k) = \{g(\underbrace{kN - N + l}_{\tau\in\Lambda_t}) : l = 1, 2, \ldots, N\} \qquad k = 1, 2, \ldots \tag{5.1}$$

The grouping of the frames into image decks, for the non-recursive and recursive cases, is demonstrated in figures 5.2 and 5.3 respectively.[1] Note that temporal block index k is different from the time index t. Assuming that the quantizer (e.g. EC-CELP, EC-PVQ) uses

[1] Only spatial y dimension is shown.

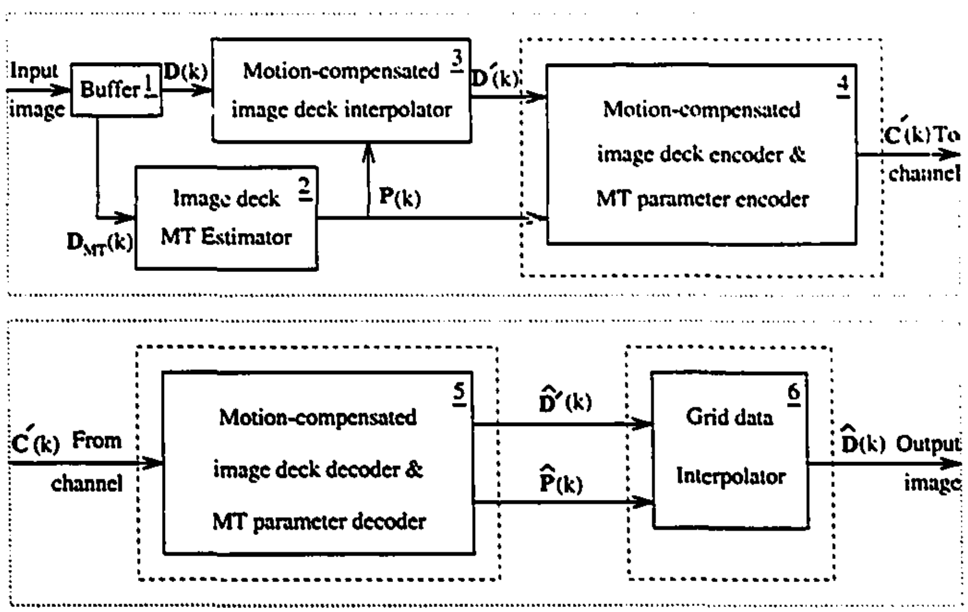


Fig. 5.1 Motion-compensated multi-frame image sequence coding configuration block diagram.

temporal coding blocks of N samples, the encoder will have a coding delay of N frames. The motion compensated samples along a single MT form a signal vector into the encoder which belong to the motion-compensated coding deck $\boldsymbol{D}'(k)$. For each spatial position $\boldsymbol{x} \in \Lambda_{\boldsymbol{x}}$ in $\boldsymbol{D}'(k)$, the above temporal vector is denoted by $\boldsymbol{s}(k) = [s_1(k)s_2(k)\dots s_N(k)]^T$ as depicted by figures 5.3 and 5.5. Also note that in $\boldsymbol{s}(k)$, the implicit dependence on $\boldsymbol{x} \in \Lambda_{\boldsymbol{x}}$ is not represented. Although the coding image decks in both recursive and non-recursive cases are identical, in the case of recursive coding, the recursive predictor incorporates the effect of "infinite" past memory. This effect requires changes in deck configuration shown in Fig. 5.3. Further details regarding this effect will be discussed later.

Recent implementations of non-recursive multi-frame video coding configurations have shown the benefits of the use of multiple frames. The three-dimensional motion-compensated subband coding of [94], [95] is an example of such a class. In previous chapters, using the GM(1) model of the temporal signal, we made the following important conclusion, comparing the non-recursive and recursive class. To obtain similar predictive coding gains

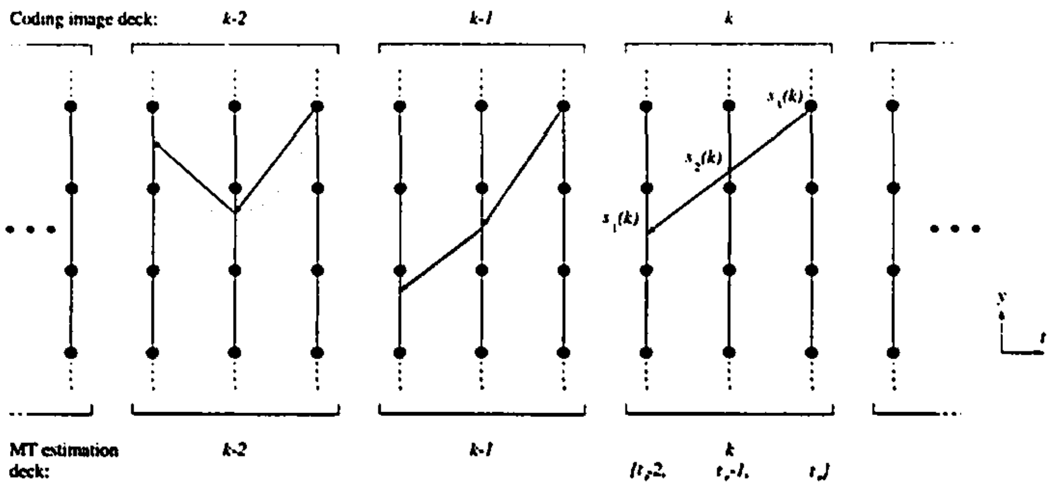


Fig. 5.2 Demonstration of non-recursive image deck $\boldsymbol{D}(k)$ with temporal block size $N = 3$, the corresponding motion-compensated coding deck $\boldsymbol{D}'(k)$ (a single vector $\boldsymbol{s}(k) = [s_1(k)s_1(k)\ldots s_N(k)]^T$ is shown), and the N frame MT estimation image deck $\boldsymbol{D}_{MT}(k)$, with $\mathcal{I}_{t_r} = \{t_r - N - 1, \ldots, t_r - 1, t_r\}$ and $t_r = kN$.

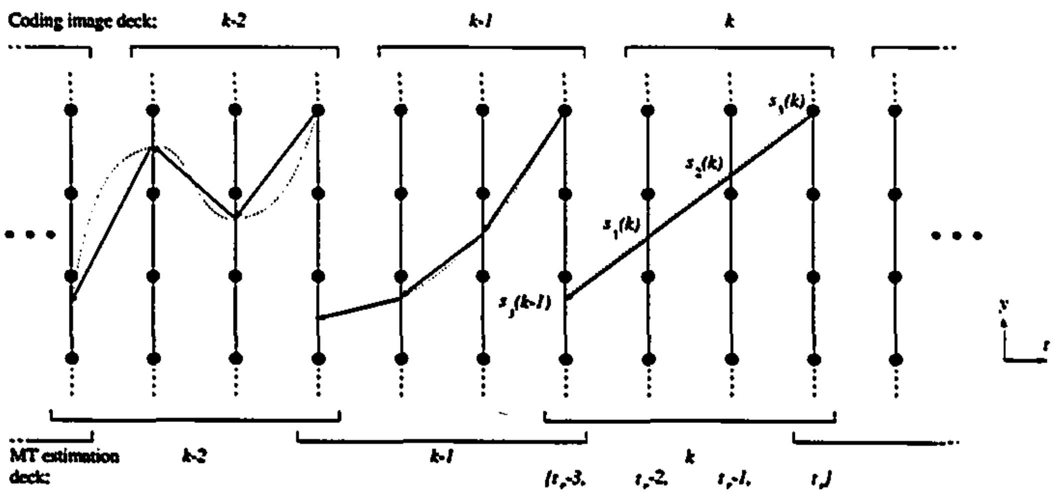


Fig. 5.3 Demonstration of recursive coding image deck $\boldsymbol{D}(k)$ with temporal block size $N = 3$, the corresponding motion-compensated coding deck $\boldsymbol{D}'(k)$ (a single vector $\boldsymbol{s}(k) = [s_1(k)s_1(k)\ldots s_N(k)]^T$ is shown), and the $N+1$ frame MT estimation image deck $\boldsymbol{D}_{MT}(k)$, with $\mathcal{I}_{t_r} = \{t_r - N, \ldots, t_r - 1, t_r\}$ and $t_r = kN$.

as provided by the recursive approach, a non-recursive multi-frame scheme would require much larger number of frames or delay (an order of 10 frames was estimated).

Other than the impractical delay and complexity of non-recursive schemes, as will be explained next, as the number of frames increase, the adverse effect of higher interpolation error due to "sparse" MTs worsens. Consequently, for the non-recursive configurations, the simultaneous maintenance of the benefits of motion compensation and non-recursive decorrelation becomes difficult.

The difficulty of "sparse" and *non-homogeneous* MTs, inherent to all motion-compensated multi-frame configurations, arises in reconstruction of pixel intensity values at the decoder for positions beyond the immediate neighboring frame of the reference frame. This phenomenon is due to the fact that as motion compensation extends to a higher number of frames, the effects of *occlusion* and *newly exposed* areas, defining the motion boundaries, becomes more complex. In fact there may be positions where MTs densely overlap or certain positions in the image where no MT hits within a large distance. It is the latter case with the "sparse" MT region which poses difficulty. This difficulty is particularly acute for the conditions of highly moving regions, dilation, occluded/newly exposed regions, and large number of utilized frames N. This is demonstrated in Fig. 5.4 (also see Fig. 2.3). One may also see how various kinds of object motions (i.e. translational, rotation, and dilation) will result in phenomena such as occlusion and new exposition with not necessarily homogeneous MTs (e.g. "sparse" MTs in k-th deck in Fig. 5.4).

The solution to the above difficulty lies in the more complex modeling and signal processing for motion compensation. The previous research on multi-frame video coders has been limited and as mentioned has been done in the context of a non-recursive configuration.[2] In [94], for the case of non-recursive multi-frame, relatively simple solutions to the above problem are based on classification to "covered" and "uncovered" areas with respect to the reference frame. Focusing on the original goals of this chapter and in view of emerging region-based video coders, we will not consider any complex MT

[2]The emerging direction of region-based motion-compensated video coding will have an important impact on this topic and future research probably should be in the context of region-based configurations. Region-based video coding can alleviate the "sparse" MT problem but other issues as discussed in [94] may rise.

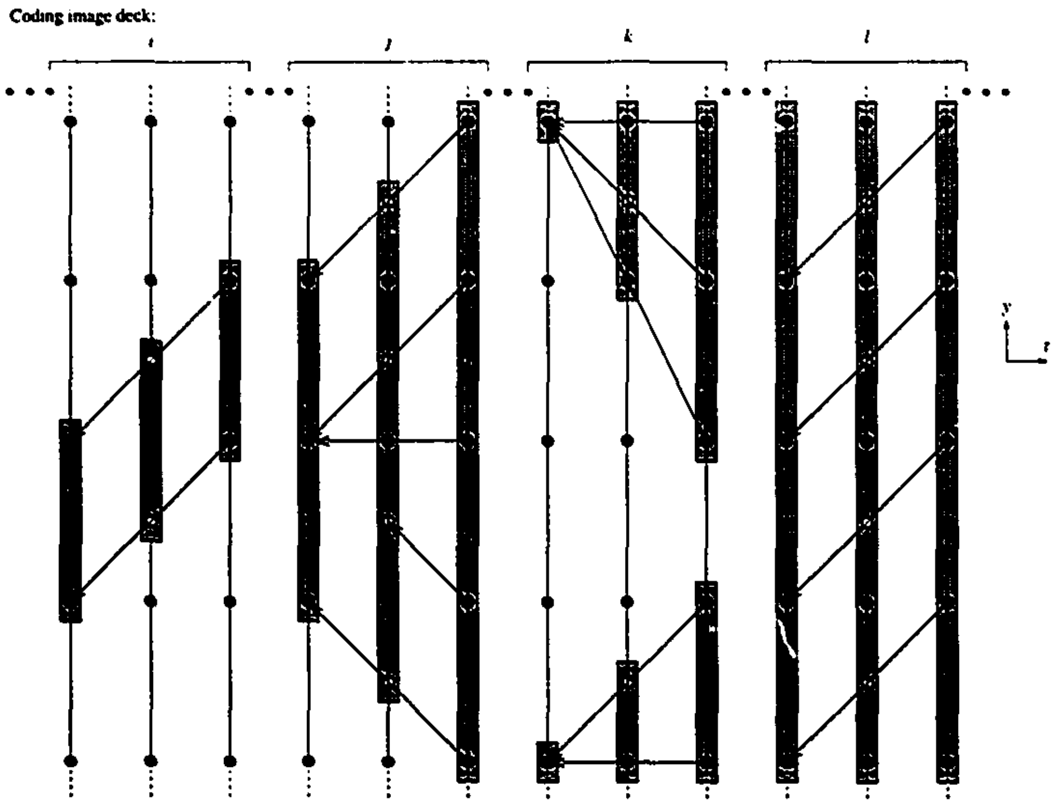


Fig. 5.4 Demonstration of non-recursive coding image deck types with locally moving object (deck i), change of scale (deck j), "sparse" *non-homogeneous* motion trajectory (deck k), and *homogeneous* motion trajectory (deck l). In this example MTs have a half-pixel accuracy and the block size is $N = 3$.

treatments and assume low motion activity[3] and MT "connectivity" (terminology in [94]).[4] Better understanding and thorough research on more complex motion treatment especially in the context of region-based coders would play an important role in full realization of practical multi-frame video coders.

5.3. Multi-Frame Trajectory Estimation

Since motion is not directly observable and measurable, it has to be estimated. It is known that the estimation process is uncertain, complicated, and ill-posed. There are a variety of existing practical motion estimation schemes, some particularly oriented for specific applications. Here, the goal would be to find a suitable framework, to configure, and to formulate the estimation process for the multi-frame quantizer in general and the recursive multi-frame quantizer in particular. As the starting scheme, among many existing approaches we chose the *maximum a posteriori probability* (MAP) criterion and cost function formulation of [20] due to ease of adaptation for multi-frame MT estimation. Unlike the commonly used *block matching* schemes, an elegant statistical approach is used which simplifies to a non-statistical formulation of the cost minimization. Markov random field (MRF) models are used both for observation process and MT description. It is the *Hammersley-Clifford theorem* that makes the simplification of characterization of MRF for the MAP estimation to a minimization of a cost function possible. This cost function is the weighted sum of certain "energies" (see Appendix C).

MT estimation can be done in backward, forward, or combined fashions [20]. To obtain the motion-compensated coding deck $D'(k)$, we need to estimate the required MT parameters. Hence a corresponding MT estimation image deck $D_{MT}(k)$ is defined. For the case of the non-recursive configuration, this deck is the same as the coding deck $D(k)$ (Fig. 5.2) with size N. For the recursive configuration (Fig. 5.3), we propose that the backward MT estimation to be carried out over $N+1$ frames of the MT estimation deck.

[3] Any higher motion activity while using this strategy would result in reconstruction with interpolation of values from a larger neighborhood in regions with "no-hit" MTs.

[4] For real video sequence, the simple strategy of *motion adaptive* as discussed in [94] may also be adopted.

From here on we will concentrate on the recursive case which is of particular interest. By proper modifications, the simpler case of non-recursive configuration can be easily obtained.

In each of th : MT estimation decks $k-2, k-1$, and k in Fig. 5.3, one MT is illustrated within each estimation deck. In deck k, the actual MT (dotted) is shown to be linear over the estimation deck. The MT of deck $(k-1)$ is quadratic. Finally a higher order MT is shown for the deck $(k-2)$. Notice that the MT estimation deck in Fig. 5.3 includes a frame from the previous coding deck. The reason for this becomes clear as the required "continuity" for the MTs is explained later. This "continuity" requirement is due to the recursive nature and need for incorporation of "infinite" past memory in the multi-frame configuration used.

For the reference time instant $t_r = kN$ in the MT estimation image deck $D_{MT}(k)$, we define the set $\mathcal{I}_{t_r}$ to denote the finite set of time instants of image frames in temporal block k, used to estimate trajectory c_{t_r} (see section 2.3.2 or the MT model later in this section for definition of c_{t_r}),

$$\mathcal{I}_{t_r} = \{\tau : g_\tau \text{ is used in estimation of } c_{t_r}\}, \tag{5.2}$$

$$\text{(backward MTs)} = \{\tau = t_r - l : l = N, \ldots, 1, 0\} = \{t_r - N, \ldots, t_r - 1, t_r\}. \tag{5.3}$$

As seen in Fig. 5.3, each MT passes through one spatial sampling point of the reference frame (t_r) and extends to the last frame in the previous block $(t_r - N)$. The k-th MT estimation deck, with MTs passing through the grid points at time instant t_r, may therefore have the following representation

$$D_{MT}(k) = \{g(\tau) : \tau \in \mathcal{I}_{kN}\}, \tag{5.4}$$

$$= \{g(kN - l) : l = N, \ldots, 1, 0\}. \tag{5.5}$$

The output of *image deck MT estimator* (module 2) is the MT parameter field $P(k) = \{p_x(k) : x \in \Lambda_x\}$, where each MT is described by a set of parameters[5] $p_x(k)$. $P(k)$

[5] The motion parameters are similar to the motion vector data in the conventional video coders.

parameter field is used to obtain the N frame motion-compensated coding image deck by the *motion-compensated image deck interpolator* module 3.

$$D'(k) = \{\tilde{g}(\underbrace{kN - N + l}_{\tau \in \Lambda_t}) : l = N, \ldots, 2, 1\}. \tag{5.6}$$

In Eqn. 5.6, $\tilde{g}(\tau)$ is the motion-compensated image at discrete time index τ which is obtained using the MT parameters estimated from MT deck $D_{MT}(k)$. Such image at non-grid image points along the MTs are obtained using one of the known two dimensional interpolation schemes. Obviously this operation (indicated by tilde) introduces some noise.

The estimation criterion. The estimation of MTs can be formulated as minimization of an objective function decomposed into a *structural model* and a *motion field model energy* [18]. The structural model corresponds to the assumptions about the properties of objects undergoing motion. More sophisticated models represent true motion with increased degrees of quality at the cost of increased complexity. Frequently, the simplified assumption of constant brightness along MTs is used. This assumption obviously is less valid for larger frame size N. As seen in more detail in Appendix C, the structural term of the objective function amounts to the sample variance along MTs. The MT field model energy captures the *a priori* knowledge about motion fields or the spatial smoothness of MTs over certain neighborhoods. The discontinuities of the motion process over borders of objects may be captured using region-based motion estimation or the use of occlusion and line fields [20]. In [20], a formulation based on MRF description and Gibbs distribution is formulated. A more popular and simpler approach simply uses rectangular blocks to force the coherence of motion fields. A weighting factor for the *motion field model energy* enforces the desired relative importance of the spatial smoothness. The above objective function can be derived using a Bayesian formulation and MRF statistical modeling of the observation process (MAP criterion) and motion fields. Alternatively, a deterministic formulation such as regularization can be used.

MT models. We will adopt and adapt the approach and notation of [20] [18], with no occlusion model, for our multi-frame recursive MT estimator as follows. We first let the "visible" period be $\{\tau : \tau \in \mathcal{I}_{t_r}\}$. Then, the MT function $c(\tau; \boldsymbol{x}, t_r)$ describes the spatial position at time τ of an image point which at time t_r was located at spatial location $\boldsymbol{x} \in \Lambda_{\boldsymbol{x}}$. $\mathbf{c}(\tau; \boldsymbol{x}, t_r)$ describes a 2-dimensional trajectory in the image plane while $(\mathbf{c}(\tau; \boldsymbol{x}, t_r), \tau) = c(x(\tau), y(\tau), \tau)$ describes a 3-dimensional trajectory in the xyt space with a one-to-one correspondence. The instantaneous velocity $\mathbf{v}$ and acceleration[6] $\mathbf{a}$ are defined as the first and second derivative of $\mathbf{c}(\tau; \boldsymbol{x}, t_r), \tau)$ with respect to τ and evaluated at $\tau = t$. Each motion trajectory $\mathbf{c}$ is superscripted by vector of motion parameters $\boldsymbol{p}$ as $\mathbf{c}^{\boldsymbol{p}}$. Hence, for the linear and quadratic trajectory model, we have $\boldsymbol{p} = \mathbf{v}$ and $\boldsymbol{p} = [\mathbf{v}\ \mathbf{a}]$ respectively, which may also be explicitly given for the position $\boldsymbol{x}$ and time reference t_r as

$$\boldsymbol{p}(\boldsymbol{x}, t_r) = [\mathbf{v}(\boldsymbol{x}, t_r)\ \ \mathbf{a}(\boldsymbol{x}, t_r)]\,. \tag{5.7}$$

The MT estimation is primarily based on a motion model. The multi-frame linear or quadratic motion models (constant $\mathbf{v}$ or $\mathbf{a}$) for the configuration in Fig. 5.3 can be respectively described by

$$\begin{aligned}
&\mathbf{c}^{\boldsymbol{p}}(\tau; \boldsymbol{x}, t_r) = \boldsymbol{x} + \mathbf{v}(\boldsymbol{x}, t_r)(\tau - t_r), && (5.8)\\
&\mathbf{c}^{\boldsymbol{p}}(\tau; \boldsymbol{x}, t_r) = \boldsymbol{x} + \mathbf{v}(\boldsymbol{x}, t_r)(\tau - t_r) + \frac{1}{2}\mathbf{a}(\boldsymbol{x}, t_r)(\tau - t_r)^2, \quad \tau \in \{\mathcal{I}_{t_r} - \{t_r\}\} \quad \boldsymbol{x} \in \Lambda_{\boldsymbol{x}},
\end{aligned}$$

where for the linear backward MTs, also a corresponding displacement field $\boldsymbol{d}(\tau; \boldsymbol{x}, t_r) = \boldsymbol{x} - \mathbf{c}(\tau; \boldsymbol{x}, t_r) = \mathbf{v}(\boldsymbol{x}, t_r)(t_r - \tau)$ is defined. Consequently, for each pixel $(\boldsymbol{x}, t_r)$ in reference frame t_r where a MT passes, the task is to find two or four parameters or components of $\mathbf{v}$ or $[\mathbf{v}\ \mathbf{a}]$, depending on the use of linear or quadratic motion parameters.

Minimization of the cost function. Among the spatio-temporal domain approaches to minimization of the cost function, the simplest is the block-oriented direct search, commonly referred to as block matching (use of above structural model term is easily possible). Gradient-based descent algorithms can provide better results which utilize both terms of the cost function. Since the resulting cost function has sum of square terms, the

[6] Row vectors $\mathbf{v}$ and $\mathbf{a}$ are two dimensional vectors with x and y components.

Gauss-Newton algorithm or an equivalent form algorithm can be used [20].[7] As shown in [20], multi-resolution estimation provides better efficiency and improves the performance of the algorit!.m.

MT estimation alternativ?s for multi-frame coding. We propose the following three alternatives for computing the multi-frame MT parameters. The *first alternative* is to use the *linear* model over the $N+1$ frame MT estimation deck. Since the MT estimation is meant to be carried over several frames, this simple model ($\boldsymbol{p} = \mathbf{v}$) may not correspond to realistic image motion situations. A *second alternative* of *quadratic* motion model, requiring twice as many parameters, is more suitable yet more complex [20] [10]. Some details of the above alternatives are presented in Appendix C. [8]

The *third alternative* is a MT estimation with a *piece-wise linear* model. The basic linear model estimation is carried out over each of the N overlapped pair of frames $(\tau, \tau+1)$ of the estimation deck. The resulting MT parameter field can be described by $\boldsymbol{P} = \{\mathbf{v}_\tau : \tau \in \{\mathcal{I}_{t_r} - \{t_r\}\}$. When $N > 2$ this alternative has larger number of parameters over the above quadratic model. The advantage of this alternative, as shown by the solid displacement vectors in Fig. 5.3, is that all three MT types shown, including higher than quadratic motion model, can be approximated closely. The basic linear estimation has to be carried out for pairs of consecutive neighboring frames starting from the spatial sampling points of the last frame in the deck.

Each MT in the third method formed by the N-piece trajectory between the N pairs of frame $(\tau, \tau+1)$, traces a non-grid image point $(\boldsymbol{x}(t_r - l), t_r - l)$ to the corresponding non-grid point in the neighboring frame $(\boldsymbol{x}(t_r - l - 1),\ t_r - l - 1)$. The MT can be described

[7] At the cost of increased complexity, one may pursue a rather globally optimum approach, using stochastic relaxation algorithms such as simulated annealing as suggested by [20].

[8] To complete the formulation for the MT estimation process, the basic quadratic MT estimation algorithm of Appendix C additionally uses the structural model and spatial smoothing term. A Gibbs-Markovian model and MAP criterion are used to formulate an objective function to be minimized. Multi-resolution deterministic relaxation [10] is the method of choice for this minimization.

by the recursive MT piecewise-linear estimate[9]

$$c^P(t_r - l - 1; x(t_r - l), t_r - l) = x(t_r - l) + \mathrm{v}\,(x(t_r - l), t_r - l)\,(-1),$$
$$\text{where} \quad x(t_r - l) = c^P\,(t_r - l; x(t_r - l - 1), t_r - l - 1) \quad \text{for} \quad l = N - 1, \ldots, 0.$$

The basic algorithm outlined in Appendix C again would be used for each of the linear MT estimation operating on each pair of frames. Simulation results shown in section 5.6 confirms the tradeoffs of complexity and accuracy among the above three image deck MT parameter estimation alternatives.

Using the above notations and assuming $\mathcal{I}_{t_r}$ and N as in Fig. 5.3, intensities along a single MT (output of module $\underline{2}$), can be represented in the following fashion

$$s(k) = \begin{bmatrix} s_1(k) \\ s_2(k) \\ \cdots \\ s_N(k) \end{bmatrix} = \begin{bmatrix} \tilde{g}\,(c^P\,(t_r - N - 1; x, t_r)\,, t_r - N - 1) \\ \tilde{g}\,(c^P\,(t_r - N - 2; x, t_r)\,, t_r - N - 2) \\ \cdots \\ \tilde{g}\,(c^P(t_r; x, t_r), t_r) \end{bmatrix}, \quad t_r = kN. \tag{5.9}$$

Note that in the above $\tilde{g}\,(c^P(t_r; x, t_r), t_r) = g(x, t_r)$, the grid or pixel values at $x \in \Lambda_x$. For the required MT "continuity", the last frame of the previous deck is obtained similarly but interpolated at MT position, based on the reconstructed samples (video decoder output) of that previous deck, i.e.

$$\hat{s}_N(k-1) = \tilde{\hat{g}}\,(c^P\,(t_r - N; x, t_r)\,, t_r - N)\,, \tag{5.10}$$

where $\hat{}$ indicates the reconstructed nature of the signal and $\tilde{}$ indicates the interpolation operation. This means that using the reconstructed signals at grid positions x, the interpolated values at non-grid MT positions are obtained.

[9]Note that for the special case of $l = 0$ and $l = -1$, we have

$$c^P(t_r - 1; x, t_r) = x + \mathrm{v}(x, t_r)(-1) \qquad l = 0,$$
$$c^P(t_r; x, t_r) = x \qquad l = -1 \quad x \in \Lambda_x.$$

The special case of $l = -1$ shows the exception of the initial coordinate of the piece-wise linear MT pivoted at the spatial grid points of the last frame.

5.4. Temporal EC-CELP Quantization

Let us now describe the configuration and formulation of the EC-CELP quantizer as applied to temporal redundancy removal in motion-compensated image sequence coding. In this section we will not be concerned with the spatial redundancies, the treatment of which is postponed until section 5.5. The raster-scanned temporal signal vectors from the motion-compensated coding image deck $\boldsymbol{D}'(k)$ described in the previous section will be the input to the temporal EC-CELP. In an inverse operation at the decoder, the outputs will form the reconstructed motion-compensated coding image deck $\widehat{\boldsymbol{D}}'(k)$. The prediction filter of the EC-CELP operates on the highly correlated samples along each of the MTs or temporal vectors (Fig. 5.3), modeled by the stationary GM(1) process. Although MT continuity is violated at the image deck boundary, their "continuity" is emulated as described by Eqn. 5.10.

Fig. 5.5 shows the block diagram of the temporal EC-CELP encoder and decoder. A buffer in the front end of the encoder will hold the temporal vectors, grouped into the motion-compensated coding image deck $\boldsymbol{D}'(k)$. Each vector is associated with the corresponding MT "continuity" sample, interpolated based on the reconstructed previous image deck (Eqn. 5.10). First using $\boldsymbol{P}(k)$ and the last frame of the previous reconstructed deck, $\hat{g}\,(\boldsymbol{x}, t_r - N) \in \boldsymbol{D}'(k)$, the MT "continuity" or ZIR memory is obtained by interpolation (Eqn. 5.10). Then the raster-scanned vectors in $\boldsymbol{D}'(k)$ along with the corresponding ZIR memory are fed into the encoder. Note that as before the dependence of the signal $s(k)$ on $\boldsymbol{x}$ (i.e. $s(\boldsymbol{x}, k)$) is not shown.

Apart from this special treatment of the ZIR memory and input/output signal buffering, the temporal EC-CELP encoder and decoder in Fig. 5.5 are similar to the ones for the EC-CELP described in Fig. 3.2. As shown in the figure, lossy+lossless coding configuration is inherent in the EC-CELP configuration. However, the simplified lossy coder in temporal EC-CELP in Fig. 5.5, uses a non-adaptive prediction and does not include gain scaling and adaptation. Nevertheless, as previously discussed, the EC-CELP allows for a combined advantages from entropy coding, analysis-by-synthesis PC, VQ, and closed-loop

EC design.[10]

As shown in the block diagram of EC-CELP quantization in Fig. 5.5, the CELP encoder with an analysis-by-synthesis structure uses an exhaustive search through the excitation signal codebook. The filter ZSR and ZIR are separated to reduce the complexity. However, as explained earlier using the formulation in Eqn. 5.10 and as seen in Fig. 5.3, due to discontinuity of the signal at the image deck boundaries, $s(k)$ at position x is not the "continuation" of $s(k-1)$ at the same position x but rather as an emulated (interpolated reconstructed) version of it.[11] As seen from the later simulation results, the introduced interpolation and MT estimation error have a tolerable adverse effect, particularly in the closed-loop CELP.

The N sample temporal input vector, is raster-scanned by the front-end encoder buffer from the coding image deck k. From each vector $s(k) = (s_1(k), s_2(k), s_N(k))$ the CELP encoder ($\Phi_{f(k)}$) generates the output index $i(k) \in \mathcal{I}$. This index is the input to the entropy encoder (Γ) which results in the output $c(k) \in C$ ($C = \{C_i\}_{i \in \mathcal{I}}$). The inverse of the entropy coder at the decoder (Γ^{-1}) generates the output $i(k)$. The CELP decoder ($\Psi_{f(k)}$) reconstructs the signal $\hat{s}(k)$ from this index. Similar to encoder front-end buffer, buffering at the tail-end of the decoder will form the reconstructed motion-compensated image deck $\widehat{\boldsymbol{D}}'(k)$ along with the required associated ZIR memory or emulated ZIR memory samples $\tilde{\hat{s}}_N(k-1)$.

As shown in Fig. 5.5, the CELP encoder first generates the ZIR difference signal $\boldsymbol{d}_{\mathrm{ZIR}}(k)$ with the above MT "continuity" consideration. Then each code vector $\boldsymbol{v}^{(i)}(k)$ (with index $i \in \mathcal{I}$) from the codebook is passed through the zero-state synthesis filter to obtain ZSR candidates $\hat{\boldsymbol{s}}^{(i)}_{\mathrm{ZSR}}$ for the current input signal vector $s(k)$. The index of the

[10]Note that a multi-frame recursive video coder based on an adaptive (general) EC-CELP has obvious advantages. The investigation of such coding system and the corresponding performance and complexity tradeoffs should be the subject of future research. In the non-adaptive case, chapters 3 and 4 suggested that an equivalence between the EC-CELP and its special case EC-PVQ can be derived. Consequently, a satisfactory tradeoff between performance and complexity may be possible through the use of appropriately modified EC-PVQ-based coder in place of the more general EC-CELP. Note that both EC-CELP and its special case EC-PVQ can be the temporal quantizer of choice in the proposed recursive video coding system. Obviously we are treating the more general case of EC-CELP in this chapter.

[11]This value must also be available to the decoder which is the reason why the reconstructed signal is used.

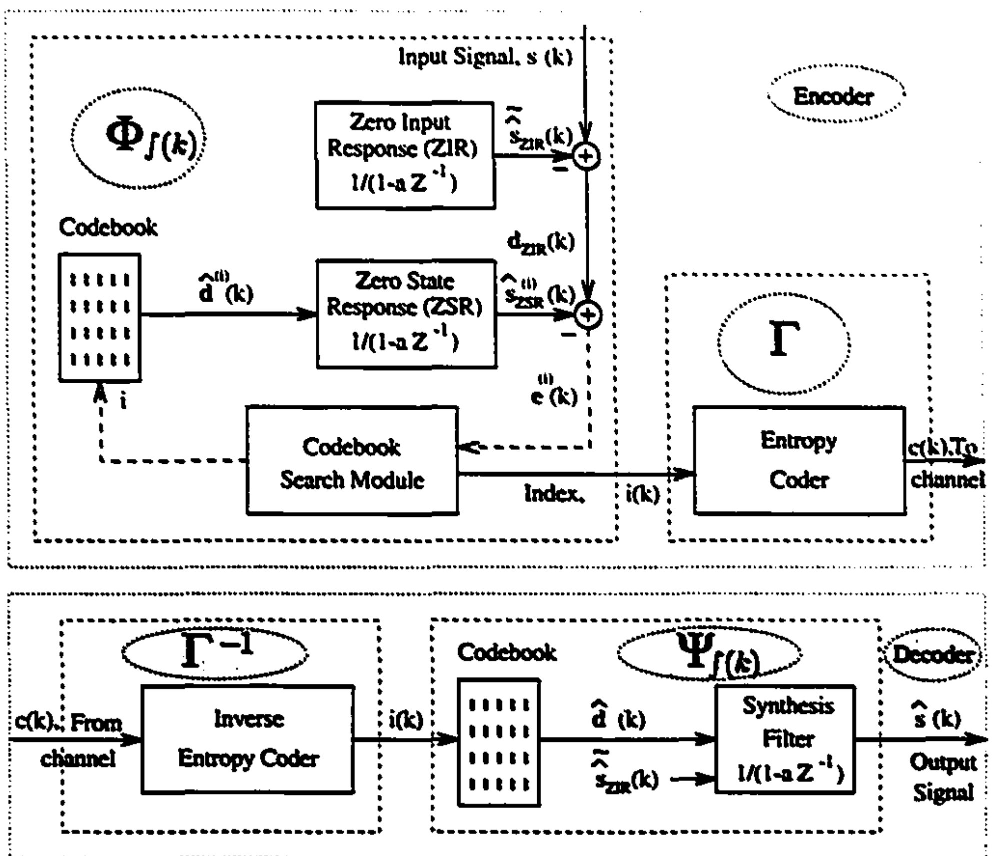


Fig. 5.5 Non adaptive EC-CELP encoder and decoder block diagram (motion-compensated image deck temporal coding with first order predictor $P(Z) = aZ^{-1}$)

codevector which yields minimum entropy-constrained cost, $i(k)$, is sent to the lossless entropy coder. We have assumed a stationary GM(1) source and hence the predictor coefficient value is the same as the GM(1) coefficient a ($P(Z) = aZ^{-1}$). The reconstructed vector is the summation of ZIR of ZSR of the synthesis filter using the excitation signal $\hat{d}(k) = \left[\hat{d}_1(k)\hat{d}_2(k)\ldots\hat{d}_N(k)\right]^T = v^{(i(k))}(k) \in \mathcal{V}$,

$$\hat{s}(k) = \underbrace{\overbrace{\begin{bmatrix} a \\ a^2 \\ \cdots \\ a^N \end{bmatrix}}^{\text{inter-block memory}} \tilde{s}_N(k-1)}_{\tilde{\hat{s}}_{ZIR}} + \underbrace{\overbrace{\begin{bmatrix} 0 \\ a\hat{d}_1(k) \\ \cdots \\ a^{N-1}\hat{d}_1(k) + \cdots a\hat{d}_{N-1}(k) \end{bmatrix}}^{\text{intra-Block memory}} + \overbrace{\begin{bmatrix} \hat{d}_1(k) \\ \hat{d}_2(k) \\ \cdots \\ \hat{d}_N(k) \end{bmatrix}}^{\text{codevector}}}_{\hat{s}_{ZSR}}. \tag{5.11}$$

Note that as in chapter 3, we may also use the more compact notation $\hat{s}_{ZSR}(k) = H(k)\hat{d}(k)$ which uses the filter response lower triangular matrix H. Using the stationary assumption $H(k) = H$ has the representation

$$H = \begin{bmatrix} 1 & 0 & \ldots & 0 \\ a & 1 & \ldots & 0 \\ a^2 & a & \ldots & 0 \\ \vdots & \vdots & \ddots & 0 \\ a^{N-1} & a^{N-2} & \ldots & 1 \end{bmatrix}. \tag{5.12}$$

In the above $\{1, a, a^2, \ldots a^{N-1}\}$ are the N samples of the impulse response of the synthesis filter $\frac{1}{1-aZ^{-1}}$.

It is obvious that due to the absence of gain scaling and adaptation and the assumption of stationarity for the source model, the complexity cost for the exhaustive search reduces considerably (see equations 3.34 and 3.53). Pre-calculation and lookup tables, as described in section 3.3, must be used to help the reduction of computation cost.

As far as the codebook design for the temporal EC-CELP is concerned, it is the special case of the general design procedure in section 3.3. As it was seen there, the algorithm simplifies considerably due to stationarity and absence of the gain scaling and adaptation. Other factors in low complexity here (as seen in the results of section 5.6) are the relatively small codebook size and low vector dimension N.

5.5. Hybrid temporal EC-CELP Image Sequence Coder

In this section we will review the complete coding procedure of the proposed hybrid motion-compensated multi-frame temporal EC-CELP image sequence coder referring to the coder's modules as was shown in Fig. 5.1. We then propose some suitable spatial decorrelation configurations.

For the k-th deck, $N+1$ and N input image frames are buffered in decks $\boldsymbol{D}(k)$ and $\boldsymbol{D}_{MT}(k)$ as inputs into *image deck MT estimator* and *motion-compensated image deck interpolator* modules (modules $\underline{2}$ and $\underline{3}$ in Fig. 5.1) respectively. The estimation module $\underline{2}$ calculates the MT parameter field $\boldsymbol{P}(k)$, using one of the methods of section 5.3. Using $\boldsymbol{P}(k)$, the interpolator module $\underline{3}$ obtains the image values along the MTs. To do this, one of the known two dimensional interpolation schemes such as Keys' bicubic interpolation scheme [61] is used. The output of this module is the (motion-compensated) coding image deck $\boldsymbol{D}'(k)$. The *motion-compensated image deck encoder* $\underline{4}$ encodes $\boldsymbol{D}'(k)$. The EC-CELP encoder described in section 5.4 is the main element of this module. As well, the MT parameter field $\boldsymbol{P}(k)$ is to be coded by this module (*MT parameter encoder*). The decoder performs the reverse operation of the encoder. Module $\underline{5}$ generates the reconstructed motion-compensated image deck, $\widehat{\boldsymbol{D}'}(k)$. The EC-CELP decoder described in section 5.4 and the decoder of choice for the MT parameter field $\boldsymbol{P}(k)$ would form this module.

To obtain the image deck at image grid points forming $\widehat{\boldsymbol{D}}(k)$, module $\underline{6}$ needs to perform *scattered data* to *grid data* an interpolation. Note that such interpolation operation is more complex and introduces more error than the interpolation operation performed by module $\underline{3}$. The need for this operation is one disadvantage and difficulty associated with the multi-frame coding. There are a number of grid data interpolation algorithms in the literature. We used the method of [100] for this grid data interpolation. Since most of the gain of the EC-CELP is obtained with low temporal block size (N =2 or 3), the relatively regular scattered data (motion-compensated deck) does not result in large errors (see experimental results of the next section). Although we managed to provide alternative simpler interpolation methods for the special scattered data interpolation problem at hand, further work on this module may be needed. We refer the reader for a discussion on this topic in [100].

To obtain the EC coder codebook, a training image sequence is used. Using the modules 1-3 we generate a sequence of training motion-compensated image decks $\boldsymbol{D}'(k)$, ($k = 1, 2, \ldots$). The corresponding training signal from this deck sequence is used to design an EC codebook and the corresponding codevector code length book as described in chapter 3. Obviously any test sequence is not included in the training image sequence.

When using the EC-CELP coding scheme, the output of the EC-CELP is a spatial field of integer-valued indices of the codevectors $\boldsymbol{i}(k)$. From the indices $\boldsymbol{i}(k)$ of all spatial coordinates $\boldsymbol{x} \in \Lambda_{\boldsymbol{x}}$, we construct the index field of the image deck k denoted by $\boldsymbol{I}(k) = \{\boldsymbol{i}_{\boldsymbol{x}}(k) \in \mathcal{I} : \boldsymbol{x} \in \Lambda_{\boldsymbol{x}}\}$. The entropy of these indices constitutes the rate for the coding scheme. In real situation, lossless coding has to be applied to these index fields. For a simple comparison of the above EC-CELP configuration with the common motion-compensated coding, the EC-CELP may be replaced with DPCM coder, and the first order entropy of $\boldsymbol{I}(k)$ for the configurations can be compared. As mentioned in our experiments, we also did not consider the coding of the trajectory parameter fields, $\boldsymbol{P}(k)$. Section 5.2 assumed that the cost of such coding will be similar for the compared coders. Similarly, we obtained results for an EC-DPCM as a special case of EC-CELP with temporal block size $N = 1$ (see results in the next section). This design approach showed improvement over the alternative design method of EC-DPCM in [27] for GM(1) source for rates less than one bps. Obviously it is expected that the performance of EC-CELP improves as larger N values are used. Due to the EC design advantage, even an EC-CELP with $N = 1$ (our EC-DPCM) outperforms the simple DPCM significantly.

The encoder and decoder blocks should include both temporal and spatial lossy and lossless compression techniques. As mentioned before, in this study we emphasized the examination of advantages of better exploitation of temporal redundancy. To simplify the simulations, we assumed that the EC-CELP is to code intensities along MTs without being concerned with the spatial redundancies of the EC-CELP codevector index fields.

Nevertheless in the next two subsections we will consider the particularity of the spatial redundancy removal in the hybrid temporal EC-CELP video coding configuration. A number of proposals[12] with regard to spatial redundancy removal using lossless and lossy coding will be presented.

5.5.1. Spatial redundancy reduction using lossless coding

Up to this point we have focused on the temporal redundancy removal coding gains. Subsequent to the use of temporal EC-CELP, direct use of conventional spatial redundancy removal schemes (i.e. hybrid DCT-VLC) will not be applicable. This can be easily observed from the fact that the EC-CELP output index field $I(k)$ can only be losslessly encoded. Similar to the case of one dimensional temporal signal, discussed in previous chapters, the spatial coding gains may be classified into the three categories of *memory*, *shaping*, and *space filling*. The first order entropy coding can only provide the shaping gain.

High order lossless coding. In the conventional video coding of hybrid DCT-entropy, DCT furnishes the memory coding gain and first order entropy coder provides the shaping gain. One alternative is not to use any spatial lossy coder. If we were to use only the first order entropy coding of EC-CELP output index field $I(k)$, any possible spatial memory gain, provided ordinarily by the DCT in conventional hybrid video coder, would not be available. To maintain lossless coding while exploiting memory, one may resort to one of higher order entropy coding schemes such as the one in [75]. The spatial memory coding gain will be additional to the excellent temporal coding efficiency obtained by the EC-CELP. Such overall spatio-temporal coding efficiency then should be superior to the one of the conventional schemes.

The advantage of higher-order entropy coding over the normally used first order entropy coding is obtained by using the joint or conditional entropy. In our case of an image

[12]Providing simulations for the proposed schemes would have been out of the scope of this thesis.

sequence coder, the higher order lossless coding would be applied on the output index field $I(k)$ with joint/conditional entropy set selected on the immediate spatial neighborhoods. The fundamental information theory [38] reveals that the conditional entropy approach potentially has a lower rate than the joint entropy approach. Obviously such gain is at the cost of increased complexity. In [75], it shown how to reduce the normally impractical higher-order entropy complexity.

5.5.2. Spatial redundancy reduction using lossy coding

Due to the particularity of the temporal EC-CELP configuration, combining such a configuration with both lossy and lossless spatial coding would require a coupled spatio-temporal quantization structure. Such a coupled configuration is in fact advantageous due to its joint spatio-temporal quantization. In this configuration a spatial intra-frame coding scheme, namely a 2D-lossy coder (a simple and suitable EC or non-EC lossy coder), would be coupled with the temporal EC-CELP coder. The encoding will be done in a joint spatio-temporal fashion using small spatial blocks of image decks. This means that at each coding instant the encoder will encode a small 3-dimensional volume of the motion-compensated image deck with an index $\boldsymbol{k} = [k^h k^v k^t]$ (horizontal-vertical-temporal) by generating a spatio-temporal codevector index $i(\boldsymbol{k})$. The minimization of the coding cost for volume $\boldsymbol{k}$ in such configuration could be formulated as the joint spatio-temporal lossy+lossless coder. The codevector spatial (horizontal-vertical $[N^h N^v]$) dimension can be simply chosen so as to have the image spatial dimension ($[M^h M^v]$) as its multiple. The temporal vector dimension will be denoted by N^t.

Joint spatial EC-VQ temporal EC-CELP. The following proposed configuration combines spatial EC-VQ quantization and temporal EC-CELP quantization. At the cost of increased complexity, it allows for simultaneous and joint memory, space-filling, and shaping gains both in spatial and temporal domains.[13] Such a coder, which we may refer to as spatio-temporal EC-VQ-CELP (spatial VQ and temporal CELP with EC), is depicted

[13] An even more general configuration would combine temporal EC-CELP with spatial EC-CELP. This alternative which is in fact a spatio-temporal domain EC-CELP may particularly be attractive in the case of region-based video coders where spatial signal is locally stationary and spatial predictive coding can be most effective.

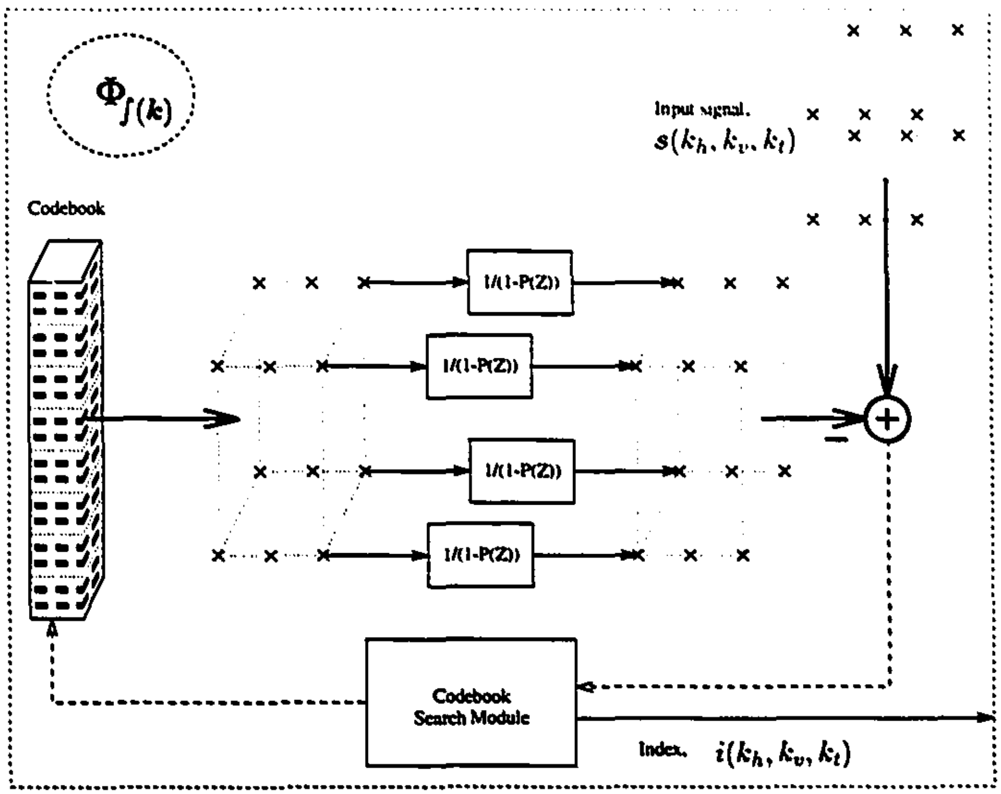


Fig. 5.6 VQ-CELP spatio-temporal lossy encoder (lossy coder in EC-VQ-CELP).

in Fig. 5.6. As seen in this figure the codebook is a 3-*domain*[14] spatio-temporal VQ (each vector has 3 domains). Obviously, $I^{(\text{spatio-temporal})}$ the size of the 3-domain codebook $\mathcal{V}_I^{(\text{spatio-temporal})}$ has to be larger than the temporal codebook $\mathcal{V}_I^{(\text{temporal})} = \mathcal{V}_I$. In fact, if we were to use spatial and temporal codebooks of sizes $I^{(\text{spatial})}$ and $I^{(\text{temporal})}$ for separate EC-VQ and EC-CELP respectively, one would expect that the 3-domain spatio-temporal combined size would be $I^{(\text{temporal})} \times I^{(\text{spatial})}$. In Fig. 5.6, the 3-domain VQ is shown to have domain dimensions $N = [N^h, N^v, N^t] = [2, 2, 3]$, resulting in the total VQ dimension 12. In the figure, the depicted codebook size is 8.

Direct extension of formulations of signal representation in chapter 3, to the spatio-

[14]We use 3-domain rather than 3-dimension to refer to image sequence spatio-temporal dimensions in order to reserve the term dimension for VQ.

temporal domain signals is possible. To illustrate an example, let us represent the formulation of the k-th reconstructed sample (extension of Eqn. 3.23). We may group the temporal vectors to provide a more compact temporal signal representation (with index $k^t = 1, 2, \ldots$) to get

$$\hat{s}_{l^h,l^v}(k) = \begin{bmatrix} \hat{s}_{l^h,l^v,1}(k) \\ \hat{s}_{l^h,l^v,2}(k) \\ \cdots \\ \hat{s}_{l^h,l^v,N^t}(k) \end{bmatrix} = h \begin{bmatrix} \hat{s}_{l^h,l^v,N^t}(k^h, k^v, k^t - 1) \\ \hat{s}_{l^h,l^v,1}(k^h, k^v, k^t) \\ \cdots \\ \hat{s}_{l^h,l^v,N^t-1}(k^h, k^v, k^t) \end{bmatrix} + \begin{bmatrix} \hat{d}_{l^h,l^v,1}(k) \\ \hat{d}_{l^h,l^v,2}(k) \\ \cdots \\ \hat{d}_{l^h,l^v,N^t}(k) \end{bmatrix}. \quad (5.13)$$

The temporal domain CELP using the first order prediction with coefficient h is combined with the spatial VQ. The three domain excitation signal is $\hat{d}(k) = \left[\hat{d}_1(k)\hat{d}_2(k)\ldots\hat{d}_N(k)\right]^T = v^{(i(k))}(k) \in \mathcal{V}_I^{(\text{spatio-temporal})}$. We have the following corresponding spatial vector indices (k^h, k^v) as well as the associated sample (within each vector) indices (l^h, l^v),

$$k^h \in \{1, 2, \ldots, M^h/N^h\} \quad \text{and} \quad k^v \in \{1, 2, \ldots, M^v/N^v\}$$
$$l^h \in \{1, 2, \ldots, N^h\} \quad \text{and} \quad l^v \in \{1, 2, \ldots, N^v\}.$$

The evaluation of the above spatial lossless or lossy redundancy removal techniques is beyond the scope of this thesis. As mentioned, simulation results in the next section (5.6) does not include spatial coding. It solely focuses on an evaluation of the EC-CELP configuration temporal coding efficiency. The choice of spatial coding for any future EC-CELP full video coder would depend primarily on tradeoffs between complexity and available gains, which needs further research.

5.6. Simulation Results and Conclusions

In this section we present the simulation results and draw some concluding remarks. We first examine the proposed multi-frame MT estimation techniques. Then we present the results related to the coding gains of the proposed temporal EC-CELP video coding configuration.

Multi-frame MT estimation results. As the first set of experiments, the three alternatives of MT estimation for multi-frame coding in section 5.3, were simulated and compared. They all successfully, but with varying quality (consistent with the conclusions of section 5.3), resulted in generating highly correlated signal with "continuity" of MTs. Fig. 5.7 pictorially demonstrates the resulting motion compensated image deck $D'(k)$ using the second (*quadratic*) MT estimation alternative. It is clearly seen that the signal after motion compensation has higher correlation along the temporal axis (for three frames from the sequence *Miss America*).

Next, to have a more quantitative comparison among the three alternatives, we required a precise knowledge of the real motion in the image sequence. To guarantee such knowledge while maintaining near "real"-video quality we used the ray tracing software Persistence of Vision Ray-tracer (POV) [109]. Experiments with a number of test synthetic sequences with varied image and motion description were conducted. The synthetic motion had a fractional pixel accuracy based on different motion types with different motion orders (constant v or constant a, etc). We then compared the MSE of the known true motion field and the estimated motion field $P(k)$, as defined in section 5.3. For accelerated motion, as expected the simulations showed that the second model using a quadratic motion model over the MT estimation deck is more suitable than the first model (linear motion model over deck). The *piecewise-linear* (third) alternative is superior to the other two alternatives if higher than quadratic synthetic motion test sequence were used.

As previously discussed, another factor playing an important role, is the number of frames used the MT estimation. As shown in Table 5.1, lower MSE of the MT parameters is possible when higher estimation window $N+1$ (the size of MT estimation deck) is used. This of course is a function of the known MT parameters and in practice may not be as important. Using the above results and subjective evaluation, we may conclude that for coding configurations discussed here, the *quadratic* (second) alternative probably produces sufficiently satisfying results in most cases with acceptable complexity.

Figures 5.9 and 5.10 depict some simulation results using real image sequences of *Salesman* and *Miss America*. These results were consistent with the above conclusions, comparing various MT estimation alternatives. In these figures using third *piecewise-linear* alternative, estimated MTs with $\mathcal{I}_{t_r} = \{t_r - 3, \ldots, t_r - 1, t_r\}$ are depicted. A visual inspection verifies the advantage of higher order MT estimation, particularly for the rapidly moving areas (hand in *Salesman*).

$\mathcal{I}_{t_r}$	v_x	v_y	a_x	a_y
$\{t_r - 1, t_r\}$	0.04	0.04	0.3	0.8
$\{t_r - 2, t_r - 1, t_r\}$	0.06	0.07	0.02	0.03
$\{t_r - 3, t_r - 2, t_r - 1, t_r\}$	0.04	0.04	0.02	0.02

Table 5.1 MSE between the known MT parameter and the estimated MT parameters for a selected spatial moving region and frame of a synthesized test sequence with known $\boldsymbol{p}$: $[1.5\ 1.5\ 0.5\ 1.0]^T$ for different sets $\mathcal{I}_{t_r}$.

Temporal coding gains after motion compensation. To focus the simulations on temporal coding gains, we chose to minimize the effects of motion estimation and the grid data interpolation error on the second set of simulations. We generated synthetic image sequences (using POV) with known simple (translation, rotation, etc) motion and with homogeneous MT fields. For such test sequences, the motion estimation methods of section 5.2 resulted in near perfect trajectory description. One example sequence used is the sequence *marble*[15] which has a simple fractional pixel translational motion. As seen in Fig. 5.8, the sequence does not include occlusion or newly exposed areas except at the image borders. As a result, the MTs sufficiently away from the image borders are homogeneous.

Both subjective and objective evaluation of of temporal coding performance for alternative coders was used. For the subjective measure, informal viewing of the sequence and the individual image frames was used. For the objective measure the commonly used measure of Peak-SNR (PSNR)[16] were used. Using the test sequence *marble*, we proceeded by fixing the coded image (subjective and objective) quality while comparing the required average bitrate in terms of estimated first order entropy for each quantization scheme.

Fig. 5.8 and Table 5.2 compare the average bit rates and quantization performance

[15] A camera moves in front of the textured *marble* wall.

[16] Using the usual definition of PSNR= dB $\left(\frac{\text{Peak value}(s)^2}{D}\right)$.

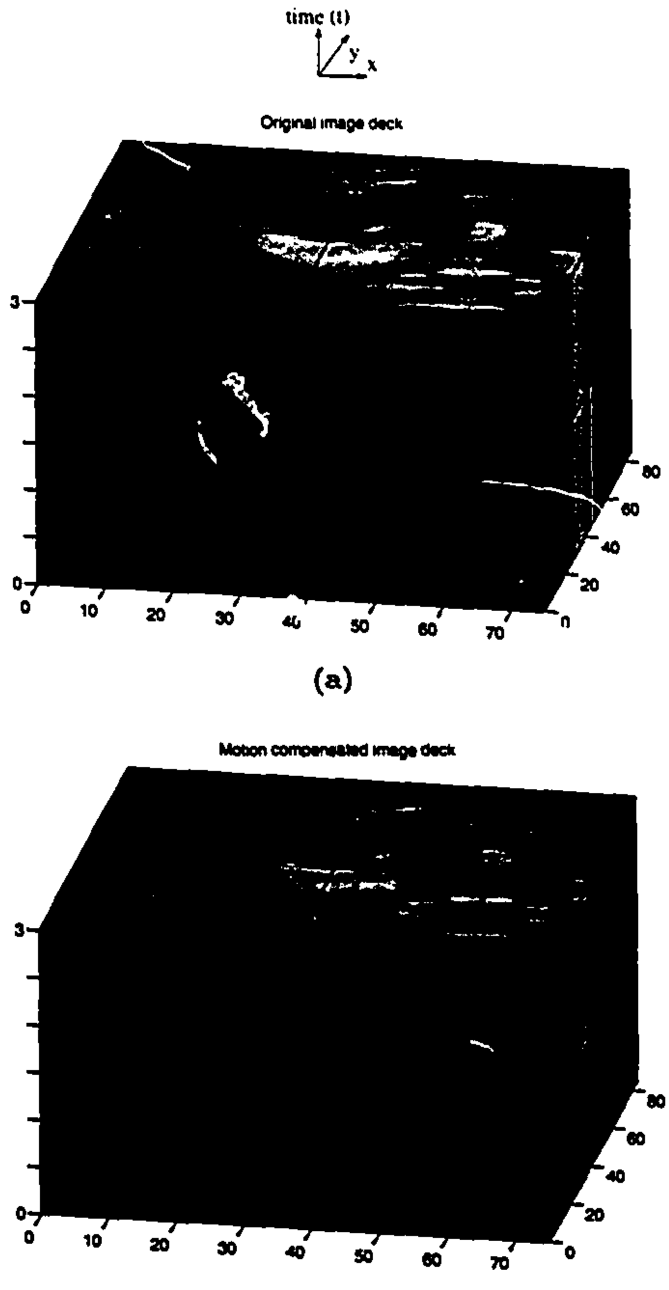
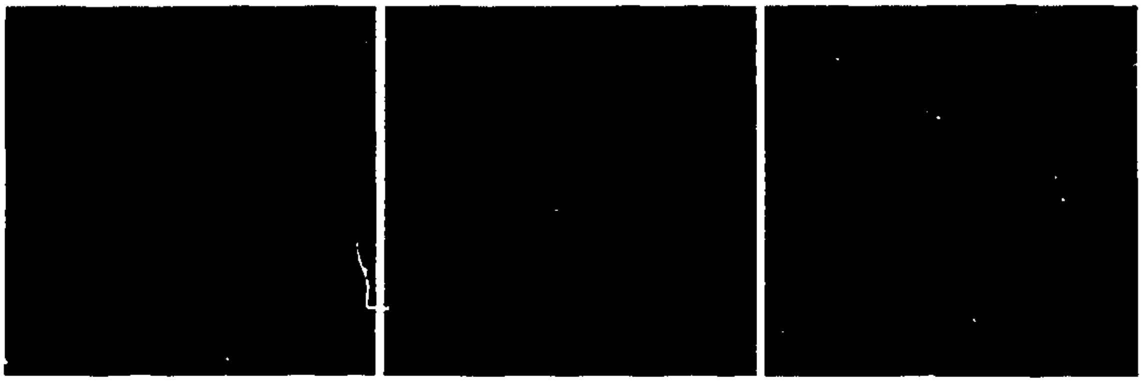


Fig. 5.7 Three-dimensional depiction of 3 frames from *Miss America* sequence as (a) original image deck and (b) motion-compensated image deck.



(a) (b) (c)

Fig. 5.8 Image sequence *marble*, (a) frame 5 of original (b) frame 6 of reconstructed sequence (overall in Table 5.2) using EC-CELP $N = 2$ (similar quality for EC-CELP $N = 1$ and DPCM, rates of Table 5.2 (c) frame 5 of reconstructed sequence (overall) using EC-CELP $N = 2$ showing adverse effect of grid data interpolation. As also shown in Table 5.2 DPCM rate is 1.2 bps, EC-CELP $N = 1$ rate is 0.6 and EC-CELP $N = 2$ rate is 0.4 bps.

coder	PSNR, coder	PSNR, overall	Rate
DPCM	40.4	40.4	1.2
EC-CELP $N = 1$	40.1	40.1	0.6
EC-CELP $N = 2$	40.3	39.4	0.4

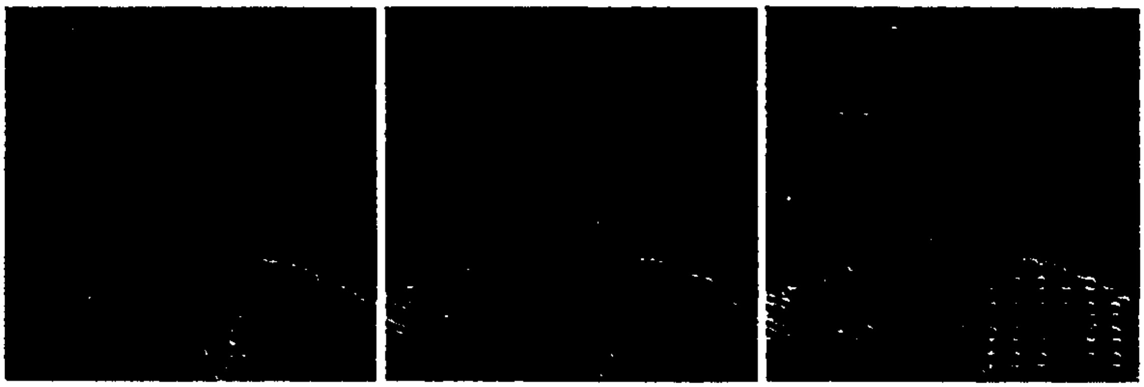
Table 5.2 Performance comparison among DPCM and EC-CELP ($N = 1, 2$) for image sequence *marble*, excluding few image border samples. Overall PSNR means that both reconstruction and when applicable grid data interpolation effect are included.

quality for EC-CELP with block sizes $N = 1$ and 2 ($N = 1$ is EC-DPCM) with DPCM. The significant performance advantage of EC-CELP demonstrates that EC and EC-CELP can overcome the DPCM performance saturation at lower bitrates. The simulations using the synthetic sequences also showed that this improved performance is possible at nominal cost. Such cost is evaluated in terms of marginal increase in complexity and delay. In particular as seen from the results, the vector dimension or delay of 2 can already provide most of available memory gain. The complexity is directly related to the codebook size which in these simulations were quite low. In these experiments however, the effect of complex and non-homogeneous motion as well as the effect of spatial redundancy were deliberately not considered. Obviously such cases would arise in more realistic image sequences. The non-homogeneous motion in particular can result in some adverse effects on the proposed system results. On the other hand, particularly in the context of region-based coders and by utilization of sufficiently comprehensive motion classification, such adverse effects can be minimized. Other improvements may come through the adaptation of PC in the quantizer.

The above comparison mainly demonstrates the coding gain due to better temporal redundancy removal. The EC gain is shown by comparing the DPCM results and the results of EC-CELP with $N = 1$ (our EC-DPCM). EC-CELP with $N = 2$ gives another substantial improvement over $N = 1$.

Previously we mentioned that one disadvantage of the motion-compensated schemes is in possible grid data interpolation error accumulation, particularly for larger N. In the image border area in Fig. 5.7-c this effect is clearly visible. Computation of PSNR for the affected areas showed a loss of about 1 dB for border areas for N value as small as 2. Again this problem would be less significant in the emerging region-based systems were the moving areas are better defined and have a more homogeneous characteristic. Obviously the difficulties resulting from the grid data error accumulation, particularly for larger N, needs further investigation.

To summarize, the simulation results in this section demonstrated that EC and EC-CELP (multi-frame recursive system) can overcome the DPCM performance limitation at low bitrates. The gain was shown to be substantial for the case of synthetic moving image sequences and was shown to come at relatively low cost in terms of delay and increased coding complexity. Equivalent coding performance using non-recursive multi-frame configuration which in theory can only be provided using a large number of frames would probably suffer from motion compensation adverse effects for large temporal blocks. In these experiments however, the effect of MT estimation for non-homogeneous motion and consequent adverse effects were deliberately isolated and eliminated. In region-based coding, and by using sufficiently complex classified MT estimation, it should be possible to lower such adverse effects. Nevertheless the future investigation of the above issues plays an important role in maintaining the demonstrated temporal coding gain advantages. This study would be in the context of full coders, operating on real image sequences with varied motion characteristics. Finally as mentioned, the feasibility of the proposed spatial redundancy removal in spatio-temporal coding configuration needs further investigation. Although, as shown in [43], for accurate motion-compensated coding, the spatial coding gain could be small, at low bit rates such coding gain could be needed (however small). The necessity for spatial lossy coding however depends on the required quality and tolerable complexity as well as the size of the coding gain due to better temporal coding. Future work should also include investigation of interpolation effects for real image sequences with various motion effects (occlusion, etc), performance and complexity tradeoffs resulting from PC adaptation, and other aspects of the complete coder described in this chapter.



(a) (b) (c)

Fig. 5.9 Image sequence *Salesman (face)* with $\mathcal{I}_{t_r} = \{t_r - 3, \ldots, t_r - 1, t_r\}$ MTs, **(a)** reference frame $t_r = 12$ **(b)** reference frame $t_r = 18$ **(c)** reference frame $t_r = 18$, also showing reference and MT points.

(a) (b) (c)

Fig. 5.10 Image sequence *Miss America (face)* with $\mathcal{I}_{t_r} = \{t_r - 3, \ldots, t_r - 1, t_r\}$ MTs, (a) reference frame $t_r = 6$ (b) reference frame $t_r = 9$ (c) reference frame $t_r = 21$.

Chapter 6

Summary and Conclusions

The first focus of this dissertation was source coding theory for input sources with memory. In Chapter 3, we addressed the problems associated with practical high quality, low delay, low complexity, low bitrate source coding of nonstationary sources with memory. Although VQ with a sufficiently high dimension, can yield performance close to the rate-distortion bound, for (nonstationary) sources with memory, the required VQ dimension or delay and complexity is usually not practical. Other than the class of constrained VQ techniques, classes of recursive (e.g. predictive) and adaptive VQ techniques have been already used to deal with this problem. To effectively combine the benefits of the latter two class with entropy coding, we introduced the EC-CELP quantization design scheme. EC-CELP implicitly and jointly combines the advantages of adaptive VQ, PC, and analysis-by-synthesis with merits of EC codebook design. We also showed that the EC-CELP design algorithm in its special cases can be used to design the EC adaptive predictive VQ (EC-APVQ) and EC-ADPCM. For these coding configurations, good general algorithms did not exist. Simulation results showed that, at low bitrates and for a given signal dimension N (delay), compared to all known EC alternatives, EC-CELP provides the closest performance to the rate-distortion bound for the Gauss-Markov source. Due to the efficiently designed small size codebook, the complexity of the coding scheme is also relatively low. As the success of CELP in the past decade and the follow up studies of the application of EC-CELP to speech coding have suggested, for nonstationary signals such as speech and still images, the adaptive VQ feature of EC-CELP can be most beneficial. The obvious disadvantage comes from the complexity and adverse effects associated with entropy coding. Studies in the case of other EC coders have shown that such adverse effects can be minimized. Future work must verify that one may draw similar conclusions in the case of EC-CELP quantization.

In Chapter 4, we presented some new results on low rate entropy-coded quantization theory for sources with memory and analyzed the advantages of EC-CELP in comparison with other EC quantizers, especially the EC predictive quantizers. The analysis and comparisons were at low bitrates and took the quantization noise effects at such bitrates into consideration. Based on N-th order RDF, EC quantization theory, and empirical methods, ***RDF memory gain*** and ***empirical space filling gain*** (dimensionality N) at low bitrates were formulated and calculated. These gains categorized and helped us analyze and compare the available coding gains for various EC coders for a given rate and delay (N). Based on extensions of previous prediction coding analyses to predictive VQ and CELP, closed-form formulations for predictive memory gain for various quantization schemes were obtained.

The second focus of this thesis was problems associated with high-compression video coding. The current intense demand for high-compression image sequence coding is expected to continue into the next century. Better temporal and spatial modeling of the video signal and suitable high quality quantizers play vital roles in the required coding systems. Motion is the key factor in the temporal domain signal. Conventional entropy-coded motion-compensated differential pulse code modulation (DPCM) quantization of temporal video signals and its performance limitations at low bitrates is one of the bottlenecks in achieving higher compression video coding. For the first order Gauss-Markov source, modeling the intensities along motion trajectories (MTs), we observed that due to the high quantization noise feedback at low bitrates, DPCM performance is very poor. Simultaneous quantization of temporal multi-frame blocks is one alternative to overcome this performance bottleneck. Although the idea of multi-frame video coding has been around for some time, the feasibility of high performance practical multi-frame systems was not shown until quite recently. All such new techniques, however, belong to the class of non-recursive quantizers. For this class and for the highly correlated input source, performance near the rate-distortion is only possible with large delay (number of image frames) and complexity. In the video coding portion of this thesis in chapter 5, we proposed a recursive and multi-frame video coding system using EC-CELP quantization in the temporal domain. The conclusions of chapter 3 and 4 were utilized and provided solid theoretical basis for better temporal video coding. The proposed new approach can provide low delay (few frames) high temporal video signal compression at low bitrates. A suitable motion estimation and coding configuration is also suggested. Within the scope of this thesis, some of the problems and issues pertinent to the proposed coding system are addressed. Significant bitrate reduction can be obtained by using the proposed multi-frame temporal quantizer over the conventional scheme. The proposed scheme can play a role in high compression video coding systems at low bitrates.

6.1. Future work

In chapter 3 we used the fictional assumption that entropy coding codewords can have non-integer length. We did not concern ourselves with the type of lossless coding used. We also did not took into consideration the drawbacks and difficulties associated with VLC over fixed channels such as buffer delay, overflow, and underflow or adverse effects in the presence of channel errors. Such problems have been studied previously both for variations of Huffman and Arithmetic lossless coders [51], [65]. Effects of the utilization of such practical lossless coding techniques in EC-CELP and proper tradeoff between delay and performance have to be investigated.

More work is warranted on the multi-frame recursive video coding system proposed in chapter 5. A full coding system based on EC-CELP or its special case EC-PVQ for the temporal domain coding needs to be simulated and compared against the alternative full coder of conventional hybrid DPCM and recursive multi-frame configurations [94]. In chapter 5, it was suggested that to incorporate spatial redundancy removal in the EC-CELP-based multi-frame coder, high order lossless coding of spatial index fields is one alternative. Implementation of this alternative and other suggested alternatives should be the subject of future research.

For still image coding applications and as was suggested in chapter 5, as an alternative for spatial redundancy removal in video coding, spatial multi-domain EC-CELP quantization or EC-VQ schemes can be formulated. Such formulations will use the extension of the formulation of single domain (e.g. temporal dimension) EC-CELP in Chapter 3 and details of existing results on application of spatial CELP to image coding [91], [50] [1], [87]. Finally, the simulation of the proposed spatio-temporal EC-VQ-CELP in chapter 5 and evaluation of its performance and complexity tradeoffs should be the subject of future research.

Appendix A

Basics of Information

A.1. Introduction

In this appendix some of the basics of information and some of the theories of source coding are reviewed for reference [16], [7], [58], [8], [38].

Fig. A.1 shows a discrete transition mapping model, where the mapping may be due to the channel or the source encoder. The finite set $A_J = \{a_1, a_2, \ldots, a_J\}$ of size J is referred to as the *alphabet* and is used to represent the specific discrete symbol source output (random variable (r.v.) X, the input to the mapping). A *probability mass function* (PMF) P is defined and the pair (A_J, P) is referred to as the *finite ensemble*. Therefore X assumes letter j with probability $p(j)$. For the pair of alphabets A_J and B_K, we may define their *product space*, with joint PMF $P(j,k)$ abbreviated as P_{jk}. The output of the mapping (channel or source encoder output), the r.v. Y takes values in the set B_K. The conditional probability distribution, called transition matrix, is denoted by $\boldsymbol{Q}$ with elements $Q_{k|j}$ ($p_{jk} = p_j Q_{k|j}$). Often one may speak of N-tuple source output, the random vector $\boldsymbol{X}$, with each of N random variables in the vector assuming values from the alphabet. If the probability of a N-tuple block is equal to the product of the probabilities of the individual letters, the source is independent, identically distributed (*i.i.d.*). A source with *i.i.d.* probability distribution is also called a *memoryless* source.

self-information $I(j)$ is the basic measure of information upon receiving the r.v. j (unit

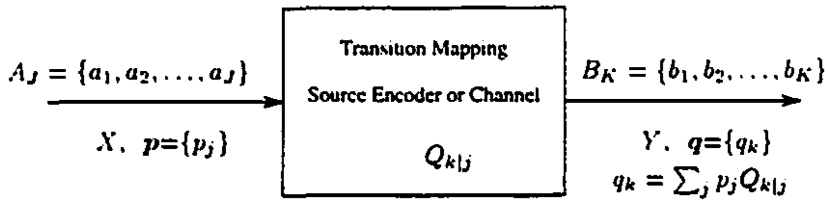


Fig. A.1 A model for source mapping.

is bit for base 2 log and nat for natural log)

$$I(j) = -\log_2 p_j. \tag{A.1}$$

The *conditional self-information* is a measure of information one acquires upon being told that event $X = j$ has occurred given event $Y = k$

$$I(j|k) = -log_2 p_{j|k}. \tag{A.2}$$

The *mutual information*, the difference between self-information of r.v. j and the conditional self-information $I(j|k)$, $(I(j;k) = I(j) - I(j|k))$ may also be expressed as

$$I(j;k) = I(k;j) = \log_2 \frac{p_{j|k}}{p_j} = \log_2 \frac{Q_{k|j}}{q_k} = \log_2 \frac{p_{jk}}{p_j q_k}. \tag{A.3}$$

The expected values of the above information measures are of essential interest. The expected value of the self-information is the *entropy*

$$H(X) = E\{I(x)\} = -\sum_j p_j \log_2 p_j, \tag{A.4}$$

which is the average amount of information upon acquiring the knowledge of the value of r.v. X. Also entropy is interpreted as the measure of average a priori uncertainty regarding which value r.v. X will assume. The *conditional entropy* is the average conditional self-information or the measure of average uncertainty remaining regarding which value X has assumed after the knowledge of value of r.v. Y. The above entropy definition is the single symbol entropy (which requires the assumption of independence among consecutive symbols X). From the case of N-th order or length-N symbol entropy (unit is bit per length-N symbol or, $1/N$ of that has unit bit per symbol (bps) where symbols may have dependence), the more appropriate entropy definition (per symbol) for source, specially for the one with memory, is defined as the limit:

$$\begin{aligned} {}^{\infty}H(X) &= \lim_{N\to\infty} \frac{1}{N} H_N(X) \\ &= \lim_{N\to\infty} [-\frac{1}{N} \sum\sum \cdots \sum_{\text{all } X} p(X) \log_2 p(X)]. \end{aligned}$$

The limit exists for the asymptotically mean stationary process with discrete alphabet. For the stationary case, ${}^{N}H(X)/N$ is non-increasing in N, hence

$$ {}^{\infty}H(X) = \inf_N \frac{1}{N} {}^{N}H(X) \tag{A.5}$$

and $N = 1$ for single symbol entropy ($H(X) = {}^{1}H(X)$). For the *i.i.d.* source ${}^{\infty}H(X) = H(X)$. For the sources with memory and without memory (with identical alphabet and probabilities) we have

$$ {}^{\infty}H(X)\Big|_{\text{with memory}} < H(X)\Big|_{\text{memoryless}} \le \log_2 J = H_U(X)\Big|_{\text{memoryless}} \tag{A.6}$$

where subscript U stands for Uniform distribution. This inequality clearly demonstrates that the *nonuniform* distribution of the PMF and the presence of *memory* constitute the source redundancies which is the difference between $\log_2 J$ and entropy $H(X)$. The redundancy is only zero for the equi-probable (uniformly distributed PMF, $p_j = 1/J$) memoryless source ($H_{\text{U}}(X)|_{\text{memoryless}} = \log_2 J$).

Since the entropy of a continuous-amplitude r.v. is infinite, ***differential entropy*** is defined analogously as

$$ h(X) = -\int_x f_X(x) \log_2 f_X(x) dx, \tag{A.7}$$

where $f_X(x)$ describes the probability density function (PDF). But the differential entropy may also have negative or infinite value. It measures randomness relative to the coordinate system [7]. It is not the limiting case of entropy and the entropy of r.v. with a continuous distribution is infinite. Consequently, when such sources are represented with a finite number of bits, distortion must always occur. Lossless coding or data compaction do not have interesting generalizations but mutual information (defined shortly) notion and lossy or compression coding, formulated based on such notions do have interesting generalizations (all well-behaved in the continuous case). For the above reasons, the following relationship (Appendix C in [58]) between the differential entropy of various continuous sources (with a given variance σ_x^2) prove to be informative. We have

$$ {}^{\infty}h(X)\Big|_{\text{with memory}} < h(X)\Big|_{\text{memoryless}} \leq \frac{1}{2}\log_2(2\pi e\sigma_X^2) \tag{A.8} $$

where the RHS limit is for the discrete memoryless *i.i.d.* Gaussian source

$$ h_{\,\mathrm{G}}(X)\Big|_{\text{memoryless}} = \frac{1}{2}\log_2(2\pi e\sigma_X^2), \tag{A.9} $$

and

$$ \begin{aligned} {}^{\infty}h(X) &= \lim_{N\to\infty}\frac{1}{N}h_N(X) \\ &= \lim_{N\to\infty}[-\frac{1}{N}\int\int\cdots\int_{\text{all } X} p(X)\log_2 p(X)]. \end{aligned} $$

Eqn. A.8 states that the redundancies are due to non-Gaussian PDF and presence of memory which amounts to non-flat PSD or SFM or $\gamma_X^2 < 1$ (for a source with variance σ_X^2. RHS is the bound for $h(X)$).[1] The notion of entropy power is defined as

$$ Q_e(X) = (2\pi e)^{-1}2^{2h(X)}, \tag{A.10} $$

which has the maximum value for the Gaussian *i.i.d.* memoryless source equal to σ_X^2.[2]

[1]The above relationships can be easier interpreted in the context of lossy or compression RDF formulations [7], [8] later summarized in section A.1.2.

[2]$Q_{e,\,\mathrm{G}}(X)|_{\text{with memory}} = \gamma_X^2\sigma_X^2$.

The *average mutual information* another important measure, is defined as

$$I(Q,p) = I(X;Y) = \sum_{j,k} p_{jk} \log_2 \frac{p_{jk}}{p_j q_k}. \tag{A.11}$$

which is the measure of average uncertainty remaining regarding which value r.v. X has assumed given the value of Y. Alternatively, $I(X;Y)$, is the average information given by Y value about the value of X, or vice versa ($I(Y;X) = I(X;Y)$). It can be shown that $I(X;Y) \leq H(X)$ in general and we also have $I(X;Y) = H(X) - H(X|Y)$ for the discrete case and $I(X;Y) = h(X) - h(X|Y)$ for the continuous case. Also the above definition of average mutual information is valid for the random vector if r.v. are replaced by random vectors (assuming they are finite valued). For the cases of infinite valued and continuous, integral replaces sum and PDF replace PMF. Also random vector may be used with appropriate modifications to the definitions. For example for the random vector continuous valued case we have

$$I(Q,f_{\boldsymbol{X}}(\cdot)) = \int_{\boldsymbol{x},\boldsymbol{y}} f_{\boldsymbol{X},\boldsymbol{Y}}(\boldsymbol{x},\boldsymbol{y}) \log_2 \frac{f_{\boldsymbol{X},\boldsymbol{Y}}(\boldsymbol{x},\boldsymbol{y})}{f_{\boldsymbol{X}}(\boldsymbol{x}) q_{\boldsymbol{Y}}(\boldsymbol{y})} d\boldsymbol{x} d\boldsymbol{y}. \tag{A.12}$$

For coders with fidelity criterion (lossy or compression), we measure the performance of the coding using the tradeoff between code rate and *distortion measure* ρ. If we have a mapping from random process $\{X\}$ with r.v. values in alphabet A to $\{Y\}$ (alphabet A') (e.g. Fig. 1.2), the distortion is a mapping $\rho : A \times A' \to [0,\infty)$. The choice of distortion measure is varied. The most common one is the MS error defined for the N-dimensional vectors $\boldsymbol{x}$ (e.g $A = \mathcal{R}^N$and A' a subset of $\mathcal{R}^N$) is

$$\rho(\boldsymbol{x},\boldsymbol{y}) = \frac{1}{N}\sum_{i=1}^{N}(x_i - y_i)^2. \tag{A.13}$$

Any *norm*, *semi-norm* or *distance* measure (with conventional definitions of norm, semi-norm and distance) provides a distortion measure. The frequently used L_p distance is defined as

$$\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|^p = [\sum_{l=1}^{N}(x_l - \hat{x}_l)^p]^{1/p}. \tag{A.14}$$

Also a *difference distortion measure* using only dependence on $x - \hat{x}$ is commonly used. Other distortion measures such as *weighted* squared-error may also be used

The family of distortion measures for N-dimensional sequences ${}^N x$, $\{{}^N\rho : N = 1, 2, \ldots\}$ called *fidelity criterion*. The special case of *single letter* or *additive* fidelity criterion has the additive property

$$ {}^N\rho({}^N\boldsymbol{x}, {}^N\boldsymbol{y}) = \sum_{l=1}^{N} {}^1\rho(x_l, y_l), \tag{A.15} $$

where x_l is the l-th element of vector $\boldsymbol{x}$ (time average distortion for sequence). The long term time average distortion

$$ \lim_{N\to\infty} \frac{1}{N} \sum_{l=1}^{N} {}^1\rho(x_l, y_l), \tag{A.16} $$

measures sample performance with such distortion. This limit exists for asymptotically mean stationary process pair. With the additional characteristic of ergodicity, the time average is the expected distortion D. The single-letter distortion may be represented by the $J \times K$ distortion matrix compactly with entries $\rho_{jk} = {}^1\rho(x_j, y_k)$ (the discrete alphabet case).

A.1.1. Data compaction theories

For data compaction (lossless coding), there are fixed and variable categories of codes. The *fixed-length block* or *vector* codes, encodes N-dimensional input vector $\boldsymbol{X}$ into r bits where N and r are fixed. This is done for each input vector independent of previous vectors. Another case is the *tree* code case where encoding is done with some knowledge of earlier ones. The special case of $N = 1$ is the *scalar* coder. If the source is discrete and memoryless and each source symbol (r.v. X) takes values from alphabet A_J with probability $\boldsymbol{p}$, there are J^N possibilities for source output ($\log_2 J^N$ bits for each vector). This results in a $\log_2 J$ bit per symbol. This is one way of measuring rate. The alternative is to measure rate by the source entropy (about $H(X)$ bit per symbol). But for the block codes this efficient coding is only possible if N is large and the $2^{NH(X)}$ codeword are assigned to high probability source vectors and as a result there is arbitrary small probability of an error (undecodable source vector). It is of related interest to mention at this point that that the concave entropy function of the discrete memoryless source has the property of $H(X) \leq \log_2 K$.

Theorem A.1 . ***Shannon first coding Theorem*** *For a discrete memoryless source, a fixed code with block length N can be produced mapping source blocks of length N into codewords of length r and with alphabet A_K so the probability of error p_e can be made smaller than ϵ ($\epsilon > 0$). This is provided that N is large enough and*

$$\frac{r}{N}\log_2 K > H(X). \tag{A.17}$$

The rate $\frac{N}{r}$ (bit per source symbol) here is related to information content as entropy (for $K = 2$, $\log_2 K = 1$). The *converse theorem* states that if the rate is less than the source entropy, for large block length the decoding error approaches 1. There are important bounds [8] which show that the probability of error decreases exponentially with the increase in the block length N.

With *variable length* codes, the N or r, or both are variable and there is no need for the probability of error allowed for the fixed length code, while still allowing coding at a rate close to source entropy. Of practical importance are only the code classes *uniquely decodable* codes (reverse mapping unambiguous) and *prefix* codes. The uniquely decodable codes with code alphabet B_K ($K = 2$ for binary) with M code words, the m-th one having length l_m, satisfy the Kraft inequality

$$\sum_{m=0}^{M-1} K^{-l_m} \leq 1, \tag{A.18}$$

and the average code length $\bar{l}$ per source letter of uniquely decodable codes and prefix codes satisfies

$$\bar{l} \geq \frac{H(X)}{\log K}. \tag{A.19}$$

There is a theorem for a memoryless discrete source stating that there exists a prefix code with average code length close ($\epsilon > 0$, e.g. $\epsilon = 1/N$) to the entropy

$$\frac{H(X)}{\log K} \leq \bar{l} < \frac{H(X)}{\log K} + \epsilon. \tag{A.20}$$

The proof is using the application of law of large numbers. For the more general case of discrete stationary but not memoryless source, the above theorem holds if we use the definition of entropy per source symbol earlier expressed for such sources (${}^{N}H(X)/N$ or ${}^{\infty}H(X)$)

Theorem A.2 . *Theorem (Variable Length codes) there exists a prefix code with average code length close to the entropy per symbol ($\epsilon > 0$)*

$$\frac{{}^{N}H(X)}{N\log K} \leq \bar{l} < \frac{{}^{N}H(X)}{N\log K} + \frac{1}{N}. \tag{A.21}$$

If N is large enough, ${}^{\infty}H(X)$ is used in place of ${}^{N}H(X)/N$ and ϵ replaces $1/N$.

Huffman coding is a practical and optimum (minimum average codeword length) code of this kind [38]. This kind of VLC has limitations (e.g. buffer length selection and *a priori* knowledge of probabilities). Practical solutions to these limitations makes the code sub-optimum. The *vector* Huffman coder with inputs of "extended source" (blocks on N) can achieve in principal rates close to the limit in the above theorem. The minimum average length has to be taken over all N which is essentially achieving entropy rate ${}^{\infty}H(X)$. This scheme which belongs to the class of vector entropy coding is however complex and sub-optimum methods again have to be used.

The other class of codes for data compaction are the tree codes where, unlike block codes, the blocks are not independent (e.g. Elias code). For example for the Elias code, a classification tree is used. This tree has as input real numbers $r \in [0, 1)$ which results in a node 0 or 1. For this code unlike vector entropy codes, the number of input symbols grouped together will vary. Asymptotically, the average word length for this code will converge to the input entropy. The infinite precision resulting from the binary expansion of

r makes the code impractical. Arithmetic codes are like the Elias code but with finite (not arbitrary) precision. This makes the coding more complex and sub-optimal. Nevertheless, it performs better than the Huffman coder at the cost of increased complexity. This code is quite popular for its high performance.

When the source statistics are incompletely or inaccurately specified, the source is stationary but not ergodic, or the source is known to be a member of some class (this third class includes the first two), *Universal codes* perform well. These codes are known to perform asymptotically as well as the custom designed code for the source. Alternatively, adaptive Huffman codes can be used when there is no a priori knowledge of the input probabilities. Also among more popular data compaction codes are: *Run-Length* codes, *Ziv-Lempel*, and *Markov* codes as well as various adaptive schemes [107], [41].

A.1.2. Data compression and Rate-Distortion theories

Data compression codes or coding with a fidelity criterion, unlike data compaction allows distortion between the input and output sequence. The resulting theories are also referred to as rate-distortion theories. Starting with discrete memoryless sources (*d.m.s.*), the theories are stated, with a brief statement of the generalizations to ergodic stationary sources. Using the transition mapping of Fig. A.1, single letter distortion and conditional transition elements (ρ_{jk} and $Q_{k|j}$) and appropriate probability measure (p), the *average distortion* associated with this mapping is defined as (distortion being a r.v.)

$$D(Q) = \sum_{j,k} p_j Q_{k|j} \rho_{jk}. \tag{A.22}$$

Additionally, we define the set of D-admissible transition mappings as

$$Q_D = \{Q_{k|j} : \rho(Q) \leq D\}. \tag{A.23}$$

Also associated with the mapping is the average mutual information ($I(Q,p)$ as defined in Eqn. A.11). The *Rate Distortion Function* (RDF) is defined as the

$$R(D) = \min_{Q \in Q_D} I(Q,p) = \min_{Q \in Q_D} \left[\sum_{jk} p_j Q_{k|j} \log_2 \frac{Q_{k|j}}{q_k} \right]. \quad \text{(A.24)}$$

Note that as was mentioned before, for the case of continuous amplitude stationary sources, we need to use the appropriate formulation for average mutual information, replace PMFs with PDFs, and use integrals in place of sums in the formulations (e.g. compare equations A.11 and A.12).

R(D) is a positive continuous monotonic decreasing convex $\cup$ function in the interval of interest $0 \le D < D_{max}$, where

$$D_{max} = \min_k \sum_j p_j \rho_{jk}. \quad \text{(A.25)}$$

R(D) vanishes for $D \ge D_{max}$. When the reproduction alphabet A_K is an image of source alphabet A_J (for each source alphabet j, there is reproducing letter $k(j) \in A_K$ with $\rho_{jk} = 0$), then $R(0) = H(X)$.

The RDF can be obtained analytically for simple sources or as a minimization problem using the Lagrangian convex programming problem. Given probability p, the problem is to minimize the average mutual information $I(Q,p)$ by choosing a proper transition mapping Q subject to constraints

$$\begin{gathered} Q_{k|j} \ge 0, \\ \textstyle\sum_k Q_{k|j} = 1, \quad \text{and} \\ \textstyle\sum_{j,k} p_j Q_{k|j} \rho_{jk} = D. \end{gathered}$$

The Kuhn-Tucker conditions on the solutions of the minimization problem are

$$c_k = \sum_j p_j e^{s\rho_{jk}} \lambda_j^{-1} \le 1, \quad \text{(A.26)}$$

where $\lambda_j = \sum_k q_k e^{s\rho_{jk}}$ The RDF is sometimes expressed as maximization formulation

$$R(D) = \max_{s<0, \lambda\in\Lambda_s} \Big[sD + \sum_j p_j log_2 \lambda_j \Big]. \tag{A.27}$$

where

$$\Lambda_s = \{\lambda \in \mathcal{R}^J : \lambda_j \geq 0 : \sum_j p_j \lambda_j e^{s\rho_{jk}} \leq 1\}. \tag{A.28}$$

and s is the Lagrangian multiplier. The argument of the maximization is a Lower Bound for RDF often expressed in the alternative form

$$R(D) \geq sD + \sum_j p_j \log_2 \lambda_j = H + sD + \sum_j p_j log_2(\lambda_j p_j). \tag{A.29}$$

Blahut, before providing a detail iterative algorithm for solving the problem (Theorem 6.3.8 and 6.3.10 in [8]), expresses it in terms of the following double minimization problem with parameter s

$$R(D) = sD + \min_q \min_Q \Big[\sum_{j,k} p_j Q_{k|j} \log_2 \frac{Q_{k|j}}{q_k} - s \sum_{j,k} p_j Q_{k|j} \rho_{jk} \Big], \tag{A.30}$$

where

$$D = \sum_{j,k} p_j Q^*_{k|j} \rho_{jk} \tag{A.31}$$

and Q^* achieves minimum. For fixed Q, the RHS is minimized by $q_k = \sum_j p_j Q_{k|j}$ and for fixed q, the RHS is minimized by

$$Q_{k|j} = \frac{q_k e^{s\rho_{jk}}}{\sum_k q_k e^{s\rho_{jk}}}. \tag{A.32}$$

N-th order RDF For discrete time continuous stationary sources with memory (e.g. GM(1)), with a better potential for compression due to inherent statistical dependencies, we can define the useful notion of N-th order rate-distortion function in the following manner. To incorporate the N-tuple symbol memory effects, for the N-tuple source outputs $\boldsymbol{X}$, the joint PDF governing such random vectors and all conditional PDF, $f_{\boldsymbol{Y}|\boldsymbol{X}}(\cdot)$, are considered. Parallel to the formulation of *d.m.s.* (Eqn. A.24), for the N-tuple source with the above modifications to the source mapping model (using random vector PDF in place of PMF in Fig. A.1), using appropriate definition of average mutual information in Eqn. A.12, we can define N-th order RDF

$$^{N}R(D) = \frac{1}{N} \inf_{Q \in Q_D} I(Q, f_{\mathbf{X}}(\cdot)). \tag{A.33}$$

The limit of $^{N}R(D)$ for large N becomes the RDF $R(D)$, and it can be shown that for stationary sources the limit always exists,

$$R(D) = \lim_{n \to \infty} {}^{N}R(D). \tag{A.34}$$

Theorem A.3 . ***Shannon third coding Theorem*** *For the d.m.s. $\{X\}$ with finite alphabet, and using a single-letter fidelity criterion, it is possible to find a block code with rate R and average distortion per letter less than D, if*

$$R \geq R(D), \tag{A.35}$$

and block length N is sufficiently large. As seen before, for the code of size $M = K^r$ (e.g. $K = 2$) and block length N, the rate is $R = \frac{r}{N}$ bit per source symbol.

The converse theorem states that every block code for data-compression of block length n and average per-letter distortion D for a finite alphabet *d.m.s.* has rate R satisfying $R \geq R(D)$.

For the continuous amplitude sources, differential entropy $h(X)$ is used in place of entropy $H(X)$ and integrals replace summations to obtain formulation analogous to the discrete case. It is obvious that PDF will replace the PMFs. However, an important difference is that, unlike the discrete case, the continuity at $D = 0$ does not hold for the continuous case. In most cases $R(D) \to \infty$ as $D \to 0$. due to the fact that absolute entropy of the continuously distributed r.v. is infinity (unlike discrete case, where $H(X)$ is the upper bound, $h(X)$ is not).

N-th order RDF for Gaussian sources with memory As is discussed in detail by Berger in [7], the derivation of N-th order RDF for GM sources is based on properties of symmetric Toeplitz correlation matrix of time-discrete stationary Gaussian sources. Using eigenvalue representation of $N \times N$ correlation matrix and the following Corollary (2.8.3 in [7]) we can arrive at the N-th order RDF in Eqn. 2.8.

Corollary A.1 . *For an N-fold product of statistically independent discrete memoryless sources with RDFs R_k^*, $k = 1, 2, \ldots, N$, and per symbol N-fold distortion measure, the parametric representation of the RDF is*

$$D = \frac{1}{N} \sum_{k=1}^{N} D_k, \tag{A.36}$$

and

$$R(D) = \frac{1}{N} \sum_{k=1}^{N} R_k^*(D_k). \tag{A.37}$$

To arrive at the RDF we use

$$R(D) = \lim_{N \to \infty} {}^N R(D)$$

and the Toeplitz Distribution theorem [49] which provides the limit behavior as $N \to \infty$ for correlation matrix.

Appendix B

PVQ predictor for the GM(1)

In this appendix we provide the derivation of prediction coefficients for the PVQ PC in the case of GM(1) input source. Here the quantization noise effect is not taken into consideration. In chapter 4, the quantization noise effect is also taken into consideration. The block diagram of the PVQ encoder used here is shown in Fig. 4.4. The encoder has a vector-generalized DPCM closed-loop structure. The residual vector which is the difference between the input vector and the predicted signal is the input to the VQ block. An exhaustive search through the residual codebook of size I, finds the closest codevector to represent this residual vector. The reconstructed vector becomes available with one vector delay at the predictor input of the encoder. The first order (one prior block) predictor operation based on the unquantized signal is $\tilde{s}(k) = A_1 s(k)$. To obtain the optimum prediction coefficients for the case of (open-loop) PVQ, we use the formulation of [41] and for $N = 3$ we get

$$A_1 = E\{s(k)s^T(k-1)\}R_{11}^{-1} = R_{01}R_{11}^{-1}. \tag{B.1}$$

For the Gauss-Markov process we have

$$A_1 = \begin{bmatrix} \boldsymbol{a}_1^T \\ \boldsymbol{a}_2^T \\ \boldsymbol{a}_3^T \end{bmatrix} = \begin{bmatrix} a^3 & a^2 & a \\ a^4 & a^3 & a^2 \\ a^5 & a^4 & a^3 \end{bmatrix} \begin{bmatrix} 1 & a & a^2 \\ a & 1 & a \\ a^2 & a & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 0 & 0 & a \\ 0 & 0 & a^2 \\ 0 & 0 & a^3 \end{bmatrix}. \tag{B.2}$$

Appendix C

Multi-frame motion trajectory estimation

The goal of this appendix is to summarize the basic estimation method of MT parameter $\boldsymbol{p}$ for a linear or quadratic motion from an image sequence (first proposed in [20], [10]). The objective function whose minimization will yield an estimate $\boldsymbol{p}_{\text{estimate}}$ of the true motion field, is composed of a structural model term and an *a priori* motion model term that enforces spatial smoothness [20]. The *structural model* term $S(\boldsymbol{p})$ at $(\boldsymbol{x}, t_r)$ is chosen to be defined as the sample variance:

$$S(\boldsymbol{p}) = \sum_{l=1}^{N}[\tilde{g}(\boldsymbol{x}(t_r - l), t_r - l) - \zeta(\boldsymbol{x}, t_r)]^2, \qquad \zeta(\boldsymbol{x}, t_r) = \frac{1}{N}\sum_{l=1}^{N}\tilde{g}(\boldsymbol{x}(t_r - l), t_r - l) \quad \text{(C.1)}$$

where $\tilde{g}(\boldsymbol{x}(t_r - l), t_r - l)$ is the interpolated intensity at time $t_r - l$ and position $\boldsymbol{x}(t_r - l)$ obtained using the linear or quadratic motion model in Eqn. 5.9.

The overall objective function for the quadratic motion model to be minimized over the entire frame (total size in pixels) at time t_r is then expressed as follows:

$$U(\boldsymbol{p}) = \sum_{i=1}^{\text{total no of pixels}} \left\{ \sum_{l=1}^{N} \left[\tilde{g}\left((\boldsymbol{x}_i + \boldsymbol{v}(\boldsymbol{x}, t_r)l + \frac{1}{2}\boldsymbol{a}(\boldsymbol{x}, t_r)l^2), t_r - l \right) - \zeta(\boldsymbol{x}_i, t_r) \right]^2 \right.$$

$$+\lambda' \sum_{\{\boldsymbol{x}_m,\boldsymbol{x}_n\}\in C} \| \boldsymbol{p}_m - \boldsymbol{p}_n \|^2 \Bigg\}$$

where $\boldsymbol{p}_i$ is defined at position $\boldsymbol{x}_i$. The second term in the equation represents the cost associated with the smoothness of the motion field over the ensemble of all 2-element cliques C [10]. λ' is the regularization parameter that plays a vital role in weighting the importance of the *a priori* motion model with respect to the structural model. Using the Taylor expansion of $\tilde{g}(\cdot)$ about some intermediate solution $\dot{\boldsymbol{p}}$, the non-linear objective function in the above equation is approximated by a quadratic function of $\boldsymbol{p}$.

The resulting motion field estimate is given by:

$$\boldsymbol{p}_{\text{estimate}} = \arg\min_{\boldsymbol{p}} U(\boldsymbol{p}) \tag{C.2}$$

Calculating the necessary condition for optimality $\partial U(\boldsymbol{p})/\partial \boldsymbol{p}_i = 0$, and letting $\dot{\boldsymbol{p}}_i = \bar{\boldsymbol{p}}_i$ at every iteration, where $\bar{\boldsymbol{p}}_i$ is the average motion vector at pixel i, the resulting estimate $\boldsymbol{p}_{i,\text{estimate}}$ at every iteration can be determined by solving a linear system of the form $\boldsymbol{A}\boldsymbol{p}_i = \boldsymbol{b}$. The Gauss Seidel relaxation method is used in iterating a complete motion field, and this process is repeated until a suitable convergence is achieved. The computational efficiency of the algorithm, and the likelihood of convergence to the global optimum are improved by considering a multi-resolution approach. For more details of the estimation algorithm see [20] [10].

Bibliography

[1] S. Aissa and E. Dubois, "Block adaptive prediction of still images," in *15th (Canadian) Biennial Symp. Comm., Kingston, Canada*, May 1992.

[2] R. Aravind, G. Cash, D. Hang, H.-M. Haskell, and A. Puri, "Image and video coding standards," *AT&T Tech. J.*, vol. 72, no. 1, pp. 67–89, 1993.

[3] D. S. Arnstein, "Quantization error in predictive coders," *IEEE Trans. Commun.*, vol. 23, pp. 423–429, Apr. 1975.

[4] E. Ayanoglu and R. M. Gray, "The design of predictive trellis waveform coders using the generalized Lloyd algorithm," *IEEE Trans. Commun.*, vol. 34, pp. 1073–1080, Nov. 1986.

[5] C. F. Barnes and R. L. Frost, "Vector quantizers with direct sum codebooks," *IEEE Trans. Inf. Theory*, vol. 39, pp. 565–580, Mar. 1993.

[6] T. Bell, J. Cleary, and I. Whitten, *Text Compression*. Englewood Cliffs, NJ, Prentice-Hall, 1990.

[7] T. Berger, *Rate Distortion Theory: a mathematical basis for data compression*. Prentice-Hall, 1972.

[8] R. E. Blahut, *Principles and Practices of Information Theory*. Reading, Mass.: Addison-Wesley, 1987.

[9] M. Buck and N. Diehl, "Model-based image sequence processing," in *Motion Analysis and Image Sequence Processing* (M. Sezan and R. Lagendijk, eds.), ch. 10, pp. 285–316, Kluwer Academic Publishers, 1993.

[10] M. Chahine and J. Konrad, "Estimation of trajectories for accelerated motion from time-varying imagery," in *Proc. IEEE Int. Conf. Image Processing*, Nov. 1994.

[11] C.-T. Chen and T. R. Hsing, "Review: digital coding techniques for visual communications," *Proc. SPIE (Visual Communications and Image Process.)*, pp. 1–16, Mar. 1991.

[12] J.-H. Chen and A. Gersho, "Gain-adaptive vector quantization for medium-rate speech coding," in *Proc. IEEE Int. Conf. Acoustics Speech Signal Processing*, pp. 45.61–45.6.4, Mar. 1985.

[13] J.-H. Chen and A. Gersho, "A Robust low-delay CELP speech coder at 16 kb/s," in *IEEE workshop on speech coding for telecommunications*, Sept. 1989.

[14] P. A. Chou, T. Lookabaugh, and R. M. Gray, "Entropy-constrained vector quantization," *IEEE Trans. Acoust. Speech Signal Process.*, pp. 31–42, Jan. 1989.

[15] J. H. Conway and N. J. A. Sloan, "Voronoi regions of lattices, second moments of polytops, and quantization," *IEEE Trans. Inf. Theory*, vol. 28, pp. 211–226, Mar. 1982.

[16] T. M. Cover and J. A Thomas, *Elements of information theory*. New York: John Wiley & Sons, 1991.

[17] E. Dubois, "The sampling and reconstruction of time-varying imagery with application in video systems," *Proc. IEEE*, vol. 73, pp. 505–522, Apr. 1985.

[18] E. Dubois, "Motion-compensated filtering of time varying images," *Multidimens. Syst. Signal Process.*, vol. 3, pp. 211–239, 1992.

[19] E. Dubois, *CRC communication handbook*, ch. Source compression–video. CRC, 1994.

[20] E. Dubois and J. Konrad, "Estimation of 2-D motion fields from image sequences with application to motion-compensated processing," in *Motion Analysis and Image Sequence Processing* (M. Sezan and R. Lagendijk, eds.), ch. 3, pp. 53–87, Kluwer Academic Publishers, 1993.

[21] S. H. Early, A. Kuzma, and E. Dorsey, "The videophone 2500–video telephony on the public switched telephone network," *AT&T Tech. J.*, vol. 72, no. 1, pp. 22–32, 1993.

[22] M. V. Eyuboglu and A. Balamesh, "Lattice and trellis quantization with Lattice-and-Trellis-bounded codebooks–Implementation techniques for memoryless sources," *IEEE Trans. Inf. Theory*, 1993.

[23] M. V. Eyuboglu and G. D. F. Jr., "Lattice and trellis quantization with Lattice-and-Trellis-bounded codebooks–High-rate theory for memoryless sources," *IEEE Trans. Inf. Theory*, vol. 39, pp. 46–59, Jan. 1993.

[24] N. Farvardin and F. Y. Lin, "Performance of entropy-constrained block transform quantizers," *IEEE Trans. Inf. Theory*, vol. 37, pp. 1433–1439, Sept. 1991.

[25] N. Farvardin and J. Modestino, "Adaptive buffer-instrumented entropy-coded quantization for memoryless sources," *IEEE Trans. Inf. Theory*, vol. 32, pp. 9–22, June 1986.

[26] N. Farvardin and J. W. Modestino, "Optimum quantizer performance for a class of non-Gaussian memoryless sources," *IEEE Trans. Inf. Theory*, vol. 30, pp. 485–497, May 1984.

[27] N. Farvardin and J. W. Modestino, "Rate-Distortion performance of DPCM schemes for Autoregressive sources," *IEEE Trans. Inf. Theory*, vol. 31, pp. 402–418, May 1985.

[28] T. R. Fischer and M. Wang, "Entropy-constrained Trellis coded quantization," *IEEE Trans. Inf. Theory*, vol. 38, pp. 415–426, Mar. 1992.

[29] M. Foodeei, "Low-delay speech coding at 16 kb/s and below," Master's thesis, McGill University, May 1991.

[30] M. Foodeei and E. Dubois, "Rate-distortion performance of source coders in the low bit-rate region for highly correlated Gauss-Markov source," in *GlobCOM Conf.-Comm.Theory Mini-Conf.*, pp. 123–127, Dec. 1993.

[31] M. Foodeei and E. Dubois, "Coding image sequence intensities along motion trajectories using entropy-constrained code-excited linear predictive (EC-CELP) quantization," in *Proc. IEEE Int. Conf. Image Processing*, Nov. 1994.

[32] M. Foodeei and E. Dubois, "Entropy-constrained code-excited linear predictive quantization (EC-CELP)," in *IEEE Inter. Symp. Inf. Theory*, June 1994.

[33] M. Foodeei and E. Dubois, "Quantization theory and EC-CELP advantages at low bit rates," in *IEEE/IMS Inf. Theory Workshop*, Nov. 1994.

[34] M. Foodeei and E. Dubois, "Comparative analysis of entropy-constrained predictive quantizers," in *ICEE*, (Iran), May 1995.

[35] M. Foodeei, E. Dubois, and P. Mermelstein, "Speech coding using entropy-constrained code-excited linear predictive (EC-CELP) quantizer," in *IEEE workshop on speech coding*, (Annapolis, MD), Sept. 1995.

[36] M. Foodeei and P. Kabal, "Backward Adaptive Prediction: High–Order Predictors and Formant–Pitch Configuration," in *Proc. IEEE Int. Conf. Acoustics Speech Signal Processing*, pp. 2405–2408, May 1991.

[37] M. Foodeei and P. Kabal, "Low-delay CELP and tree coders: comparisons and performance improvements," in *Proc. IEEE Int. Conf. Acoustics Speech Signal Processing*, pp. 25–28, May 1991.

[38] R. G. Gallager, *Information Theory and Reliable Communication*. New York: John Wiley & Sons, 1968.

[39] N. L. Gerr and S. Cambanis, "Analysis of adaptive differential PCM of a stationary Gauss-Markov input," *IEEE Trans. Inf. Theory*, vol. 33, pp. 350–359, May 1987.

[40] A. Gersho, "Asymptotic optimal block quantization," *IEEE Trans. Inf. Theory*, vol. 25, pp. 373–380, July 1979.

[41] A. Gersho and R. M. Gray, *Vector quantization and signal compression*. Boston: Kluwer Academic Press, 1990.

[42] B. Girod, "Rate-constrained motion estimation," in *Proc. SPIE (Visual Communications and Image Process.)*, vol. 2308, pp. 1026–1034, Nov. 19.

[43] B. Girod, "The efficiency of motion-compensating prediction for hybrid coding of video sequences," *IEEE J. Sel. Areas Commun.*, pp. 1140–1154, Aug. 1987.

[44] H. Gish and J. N. Pierce, "Asymptotically efficient quantizing," *IEEE Trans. Inf. Theory*, vol. 14, pp. 676–683, Sept. 1968.

[45] J. Grass, P. Kabal, M. Foodeei, and P. Mermelstein, "High quality low-delay speech coding at 12 kb/s," *IEEE Workshop on Speech Coding for Telecomm., Vancouver, Canada*, Sept. 1992.

[46] R. M. Gray, "Information rates of autoregressive processes," *IEEE Trans. Inf. Theory*, vol. 16, pp. 412–421, July 1970.

[47] R. M. Gray, "Rate distortion functions for finite-state finite-alphabet Markov sources," *IEEE Trans. Inf. Theory*, vol. 17, pp. 127–134, Mar. 1971.

[48] R. M. Gray, *Source Coding Theory*. Boston: Kluwer Academic Press, 1990.

[49] U. Grenander and G. Szego, *Toeplitz forms and their applications*. Berkeley and Los Angeles, Calif.: University of California Press, 1958.

[50] S. Gupta and A. Gersho, "Image vector quantization with block adaptive scalar prediction," *Visual communications and image processing*, vol. 1605, pp. 179–189, 1991.

[51] D. D. Harrsion and J. W. Modestino, "Analysis and further results on adaptive entropy-coded quantization," *IEEE Trans. Inf. Theory*, vol. 36, pp. 1069–1088, Sept. 1990.

[52] ISO/IEC JTC1/SC29/WG11 (MPEG), "Report of the ITU-TS WP 15/1 special rapporteur for very low bitrate visual telephony," tech. rep., ISO N0 MPEG 93/73, Geneva, Sept. 1993.

[53] ISO/IEC JTC1/SC29/WG11 (MPEG), "Generic coding of moving pictures and associated audio information: video," tech. rep., ISO draft 13818-2, Paris, May 1994.

[54] ISO/IEC JTC1/SC29/WG11 (MPEG), "Very-low bit rate coding of moving pictures and associated audio," tech. rep., ISO, Paris, May 1994.

[55] ISO/IEC JTC1/SC29/WG11 (MPEG), "Very-low bit rate coding of moving pictures and associated audio," tech. rep., ISO N0820, Nov. 1994.

[56] ITU-T, CCITT, "Video codec for audiovisual services at $p \times 64$ kbits/sec.," tech. rep., Recommendation H.261, Geneva, 1990.

[57] A. Jain, *Fundamentals of digital image processing*. Information and System Sciences Series, Prentice Hall, 1989.

[58] N. S. Jayant and P. Noll, *Digital coding of waveforms principles and applications to speech and video*. Prentice-Hall, 1984.

[59] C. B. Jones, "An efficient coding system for long source sequences," *IEEE Trans. Inf. Theory*, vol. 27, pp. 280–291, May 1981.

[60] R. Joshi, T. Fischer, M. Marcellin, and J. Kasner, "Arithmetic and trellis coded quantization," in *IEEE Inter. Symp. Inf. Theory*, June 1994.

[61] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. Acoust. Speech Signal Process.*, vol. ASSP-29, pp. 1153–1160, Dec. 1981.

[62] A. K. Khandani, P. Kabal, and E. Dubois, "Efficient decomposition algorithm for the fixed rate, entropy-coded vector quantization," in *Conf. on Inform. Sci. and Sys.*, (Johns Hopkins University), 1993.

[63] J. C. Kieffer, "Stochastic stability for feedback quantization schemes," *IEEE Trans. Inf. Theory*, vol. 28, pp. 248–254, 1982.

[64] R. C. Kim and S. U. Lee, "Entropy constrained predictive vector quantization of speech," *Signal processing 28*, pp. 77–89, July 92.

[65] Y. H. Kim and J. W. Modestino, "Adaptive entropy-coded subband coding of images," *IEEE Trans. Image Process.*, vol. 1, pp. 31–48, Jan. 1992.

[66] J. Konrad, *Bayesian estimation of motion fields from image sequences*. PhD thesis, McGill University, Dept. Electr. Eng., June 1989.

[67] T. Koski and S. Cambanis, "On the statics of error in predictive coding for stationary Ornstein–Uhlenbeck processes," *IEEE Trans. Inf. Theory*, vol. 38, pp. 1029–1040, May 1992.

[68] F. Kossentini and M. J. T. Smith, "Adaptive entropy constrained residual vector quantization," *Signal processing letters*, pp. 121–123, Aug. 94.

[69] M. Kunt, "Recent HDTV systems," in *EUSIPCO-90*, pp. 83–99, 1992.

[70] R. Laroia, *Design and analysis of a fixed-rate structured vector quantization derived from variable-length quantizers*. PhD thesis, University Of Maryland, 1992.

[71] R. Laroia and N. Farvardin, "Trellis-based-scalar-vector quantization for memoryless sources," *IEEE Trans. Inf. Theory*, (to appear) 1995.

[72] R. Laroia, H. Shahri, and T. Alonso, "On TB-SQ based subband coding of video conference sequence," in *IEEE Inter. Symp. Inf. Theory*, June 1994.

[73] C. Lee and N. Farvardin, "Entropy-constrained Trellis coded quantization: Implementation and adaptation," in *The John Hopkins Uni. 27th Annual Conf. Inf. Sci. and Sys.*, 1993.

[74] D. LeGall, "MPEG: A video compression standard for multimedia applications," *Communications ACM*, vol. 34, pp. 46–58, Apr. 1991.

[75] S.-M. Lei, T.-C. Chen, and K.-H. Tzou, "Subband HDTV coding using high-order conditional statistics," *IEEE J. Sel. Areas Commun.*, pp. 65–76, Jan. 1993.

[76] H. Li and A. Lundmark, "Image sequence coding at very low bitrates: A review," in *IEEE Trans. Image Process.*, pp. 589–609, Sept. 1994.

[77] M. Lightstone and S. K. Mira, "Entropy-constrained, mean-gain-shape vector quantization for image compression," in *Proc. SPIE (Visual Communications and Image Process.)*, pp. 389–399, 1994.

[78] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantization design," *IEEE Trans. Inf. Theory*, vol. 28, pp. 84–95, Jan. 1980.

[79] B. Liu and A. Zaccarin, "New fast algorithms for the estimation of block motion vectors," in *IEEE Trans. Circuits Syst. Video Technol.*, pp. 148–157, Apr. 1994.

[80] T. D. Lookabaugh and R. M. Gray, "High-resolution quantization theory and the vector quantization advantage," *IEEE Trans. Inf. Theory*, vol. 35, pp. 1020–1033, Sept. 1989.

[81] D. F. Lyones-III, *Fundamental limits of low-rate transform codes*. PhD thesis, University of Michigan, 1992.

[82] M. W. Marcellin, "On entropy-constrained Trellis coded quantization," *IEEE Trans. Commun.*, vol. 42, pp. 14–16, Jan. 1994.

[83] M. W. Marcellin and T. R. Fischer, "Trellis coded quantization of memoryless and Gauss-Markov sources," *IEEE Trans. Commun.*, vol. 38, pp. 82–93, Jan. 1990.

[84] R. A. McDonald and P. M. Schultheiss, "Information rates of Gaussian signals under constraing the error spectrum," *Proc. IEEE*, vol. 52, pp. 415–416, 1964.

[85] R. Mersereau, M. Smith, C. Kim, and F. Kossentini, "Vector quantization for video data compression," in *Motion Analysis and Image Sequence Processing* (M. Sezan and R. Lagendijk, eds.), ch. 9, pp. 257–284, Kluwer Academic Publishers, 1993.

[86] J. Modestino, D. Harrison, and N. Farvardin, "Robust adaptive buffer-instrumented entropy-coded quantization of stationary sources," *IEEE Trans. Commun.*, vol. 38, pp. 859–867, June 1990.

[87] J. W. Modestino and Y. H. Kim, "Adaptive entropy-coded predictive VQ of images," *IEEE Trans. Signal Process.*, vol. 40, pp. 633–644, Mar. 1992.

[88] MPEG, "International standard, Information Technology – coding of moving pictures and associated audio for digital storage media at upto 1.5 Mbit/s," tech. rep., ISO/IEC 11172, 1993.

[89] A. Netravali and J. Robbins, "Motion-compensated television coding: Part I," *Bell Syst. Tech. J.*, vol. 58, pp. 631–670, Mar. 1979.

[90] H. Q. Nguyen and E. Dubois, "Representation of motion fields for image coding," in *Proc. Picture Coding Symposium*, pp. 8.4.1–8.4.5, 1990.

[91] H. Q. Nguyen and E. Dubois, "Predictive vector quantization of images," *SPIE/SPSE Symp. Elec. Imag.*, pp. 190–198, Feb. 1991.

[92] P. Noll, "On predictive quantization schemes," *Bell System Tech. T.*, pp. 1499–1532, May–June 1978.

[93] P. Noll and R. Zelinski, "Bounds on quantizer performance in the low bit-rate region," *IEEE Trans. Commun.*, vol. 26, pp. 300–304, Feb. 1978.

[94] J.-R. Ohm, "3-D subbband coding with motion-compensation" in *IEEE Trans. Image Process.*, pp. 559–571, Sept. 1994.

[95] J.-R. Ohm, "Motion-compensated 3-D subbband coding with multiresolution representation of motion parameters," in *Proc. IEEE Int. Conf. Image Processing*, pp. III.250–III.254, Nov. 1994.

[96] M. T. Orchard and K. Ramchandran, "An investigation of wavelet-based image coding using entropy-constrained quantization frame-work," in *Proc. of Data Compression Conf.* , pp. 341–350, Mar. 1994.

[97] C. Podilchuk, "Low bit rate subband video coding," in *Proc. IEEE Int. Conf. Image Processing*, pp. III.280–III.284, Nov. 1994.

[98] A. Puri, H.-M. Hang, and L. Schilling, "An efficient block matching algorithms for motion-compensated coding," in *Proc. IEEE Int. Conf. Acoustics Speech Signal Processing*, pp. 25.4.1–25.4.4, 1987.

[99] R. Salami, C. Laflamme, J.-P. Adoul, and D. Massaloux, "A toll qualty 8 Kb/s speech codec for the personal communications systems (PCS)," *IEEE Trans. Vehicular Technol.*, vol. 43, pp. 808–816, Aug. 1994.

[100] D. T. Sandwell, "Biharmonic spline interpolation of GEOS-3 and SEASAT altimeter data," *Geophysical Research Letters*, vol. 14, pp. 139–142, Feb. 1987.

[101] M. R. Schroeder and B. S. Atal, "Code-excited linear prediction (CELP): high-quality speech at very low bit rates," in *Proc. IEEE Int. Conf. Acoustics Speech Signal Processing*, pp. 937–940, Mar. 1985.

[102] M. Sezan and R. Lagendijk, eds., *Motion Analysis and Image Sequence Processing*. Kluwer Academic Publishers, 1993.

[103] C. E. Shannon, "A mathematical theory of communication," *Bell Systems Technical Journal*, vol. 27, pp. 379–423, July 1948.

[104] C. E. Shannon, "Coding theorems for discrete source with fidelity criterion," *IRE National Convention Record*, pp. 142–163, Mar. 1959.

[105] K. Shinohara and T. Minami, "Encoding of still pictures with successive clustering and predictive VQ," in *Electronics and communications in Japan*, pp. 204–213, Feb. 1988.

[106] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Trans. Inf. Theory*, vol. 28, pp. 56–67, Jan. 1982.

[107] R. N. Williams, *Adaptive Data Compression*. Kluwer Academic, 1991.

[108] A. D. Wyner and J. Ziv, "Bounds on the rate-distortion function for stationary sources with memory," *IEEE Trans. Inf. Theory*, vol. 17, pp. 508–513, Sept. 1971.

[109] C. Young, *Persistence of Vision Raytracer, User's Documentation*. team coordinator, version 2.0 ed.

[110] P. L. Zador, "Asymptotic quantization error of continuous signals and the quantization dimension," *IEEE Trans. Inf. Theory*, vol. 28, pp. 373–380, July 1982.