IMAGE COMPRESSION IN SIGNAL-DEPENDENT NOISE

by

RUBEENA SHAHNAZ, B.S.E.E.

A THESIS

IN

ELECTRICAL ENGINEERING

Submitted to the Graduate Faculty
of Texas Tech University in
Partial Fulfillment of
the Requirements for
the Degree of

MASTER OF SCIENCE

IN

ELECTRICAL ENGINEERING

Approved

August, 1995

ACKNOWLEDGMENTS

I would like to express my gratitude to Dr. John F. Walkup for his guidance and support throughout the course of this work. I would also like to thank Dr. Thomas F. Krile for his advice and guidance. I am grateful to Dr. Sunanda Mitra for serving on my committee and for her advice and suggestions.

My deepest appreciation goes to my family, especially to my parents for their love, support and encouragement. Finally I would like to thank all my friends.

TABLE OF CONTENTS

ACKNOWLEDGMENTS ii

ABSTRACT vi

LIST OF TABLES vii

LIST OF FIGURES viii

LIST OF ABBREVIATIONS x

CHAPTER

I. INTRODUCTION 1

1.1 Problem Statement 1

1.2 Computer Simulation of the Images 2

1.3 Image Quality Measures 2

1.3.1 Performance Criteria for the Image Restoration Schemes 2

1.3.2 Performance Criteria for the Image Compression Scheme 3

1.4 Thesis Outline 4

II. IMAGE AND NOISE MODELS 5

2.1 Image Model 5

2.2 Noise Models 7

2.2.1 Signal-Independent Additive Noise Model 7

2.2.2 Film-Grain Noise 7

2.2.3 Speckle Noise 8

III. EFFECTS OF NOISE ON IMAGE COMPRESSION 10

3.1 The JPEG Algorithm 11

3.1.1 The Discrete Cosine Transform (DCT) 12

3.1.2 Quantization 14

3.1.3 Quality Factor 15

3.1.4 DC Coding and Zig-zag Sequence 16

3.1.5 Entropy Encoding 16
3.2 Effect of Noise on Compression Using JPEG 17
3.3 Compression Results 23

IV. NOISE SUPPRESSION AND COMPRESSION IN FILM-GRAIN NOISE 29
4.1 Estimator for Signal-Dependent Noise 30
4.2 Adaptive Noise Smoothing Filter 31
4.3 Noise Suppression Results for Images Degraded by SIN 33
4.4 Adaptive Noise Smoothing Filter for Film-Grain Noise 36
4.5 Noise Suppression Results for Film-Grain Noise Using the Adaptive Noise Smoothing Filter 37
4.6 Combined Homomorphic and Local-Statistics Processing for Film-Grain Noise 40
4.7 Noise Suppression Results Using the Combined HT and Local-Statistics Processing 44
4.8 Comparison of the Two Approaches to Film-Grain Noise Suppression 46
4.9 Compression Results 49
4.9.1 Compression in Signal-Independent Noise 49
4.9.2 Compression in Film-Grain Noise 50

V. NOISE SUPPRESSION AND COMPRESSION IN SPECKLE 58
5.1 Single Frame and Multiple Frames Speckle Images 59
5.2 Adaptive Noise Smoothing Filter for Speckle 62
5.3 Speckle Reduction Results Using the Adaptive Noise Smoothing Filter 63
5.4 Combined Homomorphic and Local-Statistics Processing for the Speckle 66
5.5 Speckle Reduction Results Using Combined Homomorphic and Local-Statistics Processing 67
5.6 Compression Results in Speckle 71

VI. CONCLUSIONS AND SUGGESTIONS FOR FUTURE WORK 80

REFERENCES 83

APPENDIX

A. SAMPLE PROGRAM FOR IMAGE RESTORATION 86

B. SOFTWARE USED FOR COMPRESSION 90

ABSTRACT

The performance of an image compression scheme is affected by the presence of noise in an image. This work mainly investigates the effects of signal-dependent noise on image compression using the JPEG image compression algorithm. Simulation results show that the achievable compression is significantly reduced in the presence of noise. The types of noise considered are, signal-independent additive noise, signal-dependent film-grain noise and speckle noise. For improvement of compression ratios noisy images are pre-processed for noise suppression before applying compression. Two approaches are used for reduction of signal-dependent noise prior to compression. In one approach estimator designed specifically for a particular signal-dependent noise model is used on the noise degraded image for noise suppression. In the second approach the signal-dependent noise is transformed into signal-independent noise using a homomorphic transformation. An estimator designed for signal-independent noise is then used on the transformed image for noise suppression followed by an inverse homomorphic transformation. The performances of these two pre-compression noise suppression schemes are compared using different performance criteria. Simulation results show that pre-compression noise suppression significantly increases the amount of compression obtained subsequently. The compression results for the noiseless, noisy and restored images are compared.

LIST OF TABLES

3.1 Compression results for the noiseless Lena image and noisy Lena image with SIN of variance 1000 24

4.1. Noise suppression results for Lena image with SIN (variance 1000) using the adaptive noise smoothing filter for SIN 36

4.2. Noise suppression results for the Lena image with film-grain noise (C=4) using the adaptive noise smoothing filter 40

4.3. Noise suppression results for Lena with film-grain noise (C=4) 46

4.4. Noise suppression results for Lena with film-grain noise (C=2) 49

4.5. Noise suppression results for Airplane with film-grain noise (C=4) 49

4.6. Compression results for different quality factors for noisy (FGN, C=2) and the restored Lena image 54

5.1 Noise suppression results for the Lena with single and four frames-averaged speckle noise using the (5x5) adaptive noise smoothing filter 66

5.2. Noise suppression results for the Airplane with single and four frames-averaged speckle noise using adaptive noise smoothing filter (5x5) 66

5.3. Noise suppression results for Lena with single and four frames-averaged speckle noise using HT and J-S estimator 71

5.4. Noise suppression results for Airplane with single and four frames-averaged speckle noise using HT and J-S estimator 71

5.5. Compression results for Lena with speckle noise 77

5.6. Compression results for Lena restored from four frames-averaged speckle image ... 78

LIST OF FIGURES

2.1 Lena image and its histogram 6

3.1 Image compression/decompression system 12

3.2 DCT-based encoder processing steps 13

3.3 DCT-based decoder processing steps 13

3.4 Quantization matrix for the 8x8 sub-blocks of a monochrome image 15

3.5 Zig-zag sequence of the quantized DCT coefficients 17

3.6 Noiseless Lena image samples and DCT coefficients 18

3.7 Noisy Lena image samples and DCT coefficients 18

3.8 Noisy images and the corresponding histograms 20

3.9 Reconstructed Lena images after compression at different quality factors 25

3.10 Plots of compression results for noiseless Lena and Lena+SIN 27

4.1 Pre-compression noise suppression scheme 29

4.2 Two alternative noise suppression schemes for SDN 30

4.3 Noisy Lena and restored images obtained using the adaptive noise smoothing filter for SIN 34

4.4 Lena image with FGN (C=4) and the restored images using the adaptive noise smoothing filter 38

4.5 Noisy Lena with FGN (C=2) and the restored image using the adaptive noise smoothing filter (5x5) 41

4.6 Airplane image and its histogram 42

4.7 Airplane image with FGN (C=4) and the restored image using the adaptive noise smoothing filter (5x5) 43

4.8 Restored images from the noisy Lena image with FGN (C=4) 45

4.9 Image restored from the noisy Lena image with FGN (C=2) using HT and J-S estimator, MSE=92 47

4.10 Restored images from noisy Airplane image with FGN (C=4) 48

4.11 CR versus Q plot for different Lena images 51

4.12 Plot of bpp versus MSE for different Lena images 53

4.13 DCT coefficients of 32x32 blocks of different Lena images 55

4.14 Lena with FGN (C=2) after compression (Q=65) 56

4.15 Compressed Lena image (Q=65) after restoration from FGN (C=2) 56

5.1 Lena and Airplane images with speckle noise (single frame) 60

5.2 Lena and Airplane images with speckle noise (four frames-averaged) 61

5.3 Images restored from single frame speckle images using adaptive noise smoothing filter (5x5) 64

5.4 Images restored from four frames-averaged speckle images using adaptive noise smoothing filter (5x5) 65

5.5 Images restored from single frame speckle images using HT and J-S (5x5) estimator 69

5.6 Images restored from four frames-averaged speckle images using HT and J-S estimator (5x5) 70

5.7 Images restored from four frames averaged speckle images using HT and J-S (3x3) estimator and two iterations 72

5.8 CR versus Q for the noisy Lena images with speckle 74

5.9 CR versus Q for the restored Lena images with speckle noise 74

5.10 Compressed Lena images at quality factor 65 75

5.11 Plot of bpp versus MSE for the speckle image and the restored image using adaptive noise smoothing filter (5x5) 76

5.12 DCT (32x32 blocks) plots of different Lena images restored from noise 79

LIST OF ABBREVIATIONS

BPP	Bits per pixel
CR	Compression ratio
CCITT	International telegraph and telephone consultative committee
DCT	Discrete cosine transform
FDCT	Forward discrete cosine transform
FGN	Film-grain noise
HT	Homomorphic transformation
IDCT	Inverse discrete cosine transform
IJG	Independent JPEG group
ISO	International organization for standardization
JPEG	Joint photographic experts group
J-S	James-Stein
LLMMSE	Local linear minimum mean square error
LOGMSE	Log mean square error
MLE	Maximum likelihood estimate
NMSE	Normalized mean square error
PSNR	Peak signal-to-noise ratio
RLE	Run-length-encoding
RMSSNR	Root mean square signal-to-noise ratio
SAR	Synthetic aperture radar
SDN	Signal-dependent noise
SIN	Signal-independent noise
SNR	Signal-to-noise ratio

CHAPTER I
INTRODUCTION

At the present time, there is widespread use of continuous tone images in diverse applications. The large file sizes of images can make the storage and transmission of large numbers of images a serious problem [1]. The uses of images in the data processing environment can be difficult as handling large amounts of data requires time, is costly and requires large bandwidths. These problems can be reduced by compressing the image files. Therefore, image compression is a topic of great interest today.

The presence of noise in an image affects any kind of image processing scheme, as the noise effectively changes the image properties. An image compression technique would perform differently for a noisy image compared to the noise free case. As most of the noise sources are signal-dependent in nature [2, 3], image compression performance in the presence of signal-dependent noise is a topic worth investigating.

1.1 Problem Statement

The purpose of this thesis is to investigate the performance of image compression techniques in the presence of signal-dependent noise. Film-grain noise and speckle are two types of signal-dependent noise commonly encountered in real world images. Film-grain noise is a source of degradation when images are recorded on photographic films and are digitized [4]. Speckle occurs in all types of coherent imagery, such as synthetic aperture radar (SAR) imagery, laser illuminated imagery, astronomical imagery and also in ultrasonic medical imagery [5, 6]. The presence of signal-dependent noise (SDN) affects the compression of these images for the purpose of archiving and transmission. We have mainly studied the effects of speckle and film-grain noise on image compression.

We have used the JPEG (Joint Photographic Experts Group) standard for image compression in this work. JPEG is the current standard for image compression set under the auspices of the International Organization for Standardization (ISO), the International Telegraph and Telephone Consultative Committee (CCITT) and the International

Electrotechnical Commission (IEC) [7]. We have investigated how the presence of noise reduces the amount of compression achievable. We have also investigated the pre-processing of the noisy images to subsequently achieve higher compression ratios.

1.2 Computer Simulation of the Images

The images used in our work were 512x512 monochrome images having pixel values between 0 and 255. In the original uncompressed images, each pixel is represented by 8 bits. We have simulated the noisy images from the original noiseless images using various noise models. During simulation of the noisy images, all the pixel values generated above 255 were thresholded to 255 and all the pixel values below zero were thresholded to zero.

1.3 Image Quality Measures

The performance evaluation of any image processing scheme is done based on some suitable criterion for image quality. We have studied two types of image processing schemes in our work. They are (1) image restoration or noise suppression and (2) image compression. The image quality measures used for those two approaches are described below.

1.3.1 Performance Criteria for the Image Restoration Schemes

The most commonly used measure of image quality is the mean square error (MSE). The MSE of an NxN image is given by

$$MSE = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left[f(i,j)-\hat{f}(i,j)\right]^2, \tag{1.1}$$

where $f(i,j)$ and $\hat{f}(i,j)$ are the original noiseless image and the restored image gray level values, respectively. We have also used three other image quality measures for the restoration schemes, those are normalized mean square error (NMSE), RMS signal-to-

noise ratio (RMSSNR) and log mean square error (LOGMSE). These are defined as follows:

$$NMSE = \frac{\sum_{i=1}^{N}\sum_{j=1}^{N}\left[f(i,j)-\hat{f}(i,j)\right]^2}{\sum_{i=1}^{N}\sum_{j=1}^{N}\left[f(i,j)\right]^2}, \tag{1.2}$$

$$RMSSNR = 10\log_{10}\left[\frac{\sum_{i=1}^{N}\sum_{j=1}^{N}\left[f(i,j)\right]^2}{\sum_{i=1}^{N}\sum_{j=1}^{N}\left[f(i,j)-\hat{f}(i,j)\right]^2}\right] \text{dB}, \tag{1.3}$$

$$LOGMSE = \frac{\sum_{i=1}^{N}\sum_{j=1}^{N}\left[\log_{10}\{1+f(i,j)\}-\log_{10}\left\{1+\hat{f}(i,j)\right\}\right]^2}{\sum_{i=1}^{N}\sum_{j=1}^{N}\left[\log_{10}\{1+f(i,j)\}\right]^2}, \tag{1.4}$$

where $f(i,j)$ and $\hat{f}(i,j)$ are the noiseless and the restored image gray level values, respectively. The LOGMSE gives an error measurement based on the human visual system. The human visual system response to the input light intensity is nonlinear, which is modeled as a logarithmic function [8]. A mean square error can be calculated after passing the images through the filter with a logarithmic intensity response. Thus the LOGMSE is given by Equation (1.4).

1.3.2 Performance Criteria for the Image Compression Scheme

For the compression scheme we have again used the MSE as an image quality measure, where the MSE is now defined as

$$MSE = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left[x(i,j)-x_c(i,j)\right]^2, \tag{1.5}$$

where $x(i,j)$ is the uncompressed input image to the compression/decompression system and $x_c(i,j)$ is the image after compression and decompression. Another quality measure

used for the image compression scheme is the peak signal-to-noise ratio (PSNR) defined by

$$\mathrm{PSNR} = 10\log_{10}\left[\frac{255^2}{\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\{x(i,j)-x_c(i,j)\}^2}\right] \text{ dB.} \qquad (1.6)$$

1.4 Thesis Outline

Chapter II describes the image and noise models assumed in our work. Chapter III presents a general discussion of the effects of noise on image compression. Chapter IV describes the noise suppression techniques for film-grain noise and presents the restoration and compression results for the noisy and the restored images. Chapter V presents the techniques used for speckle reduction, the results of speckle reduction, the compression results obtained for images degraded by speckle and the restored images. Chapter VI summarizes the work and discusses possible future extensions.

In this thesis, the effects of image noise, specifically signal-dependent noise, on image compression have been studied. We have shown that there is a significant reduction of achievable compression in the presence of noise, such as signal-independent Gaussian noise, signal-dependent film-grain noise and speckle. We have then investigated the effectiveness of processing the image for noise suppression prior to compression in order to obtain higher compression ratios. We have studied two approaches for signal-dependent noise suppression prior to compression. In one approach, we have used local-statistics estimators designed specifically for signal-dependent noise. In the second approach, estimators designed for signal-independent noise (SIN) were applied to the noise degraded image after first transforming the SDN into SIN. The performances of these two schemes for noise suppression were compared. We also studied the improvement in the achievable compression realizable due to pre-compression noise suppression. The improvement in the compression ratios for the two types of noise suppression schemes were also compared.

CHAPTER II

IMAGE AND NOISE MODELS

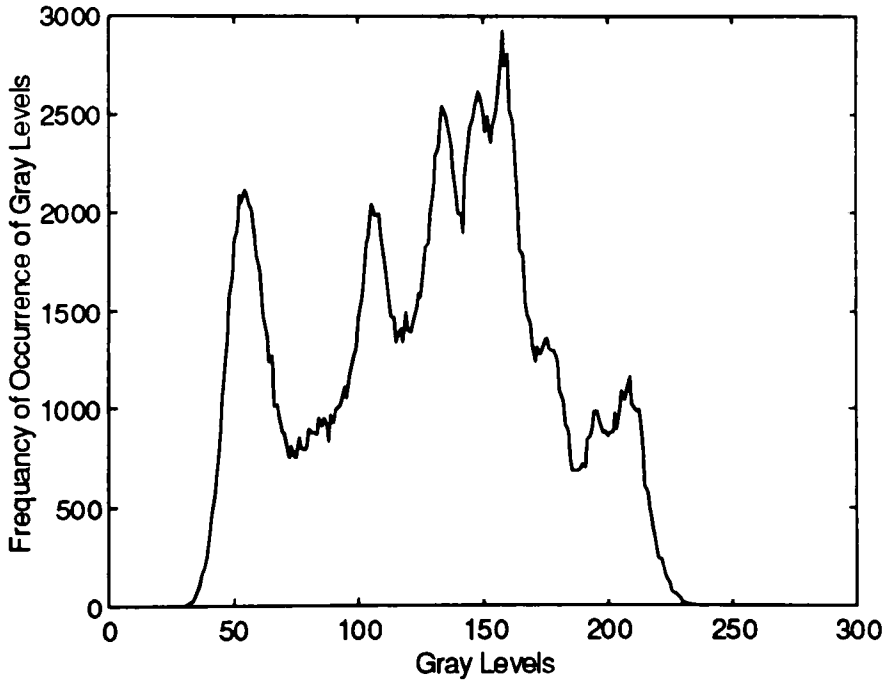
In dealing with the performance of image compression techniques on noisy images, we need to consider both the image model and the noise model. In order to find out how the presence of noise affects the compression scheme, the noise statistics need to be defined. For example, signal-dependent noise will introduce effects significantly different from signal-independent noise. To deal properly with the noise degradation, the noise model should also describe the noise process accurately. The restoration of the images degraded by noise also depends on the image and the noise models. The validity and the effectiveness of the restoration techniques are largely dictated by the models assumed. In this chapter, the image and the noise models used for our work are presented.

2.1 Image Model

Conventionally, an image is described by a stationary model with a constant mean and a stationary correlation function. Any restoration scheme based on this model is, however, insensitive to abrupt changes in the image. So an image model should technically be nonstationary. The stationary model describes an image to a be a wide-sense stationary random field with a constant mean vector and a block Toeplitz covariance matrix [9]. The joint probability density is assumed to be multivariate Gaussian. For a real world image this is not a very accurate description. As seen from Figure 2.1, the histogram of the Lena image is not Gaussian in nature and has multiple peaks. Thus a nonstationary image model is required to describe the image.

The restoration filters used for signal-dependent noise in Chapter IV and Chapter V assume a nonstationary mean and nonstationary variance model. Hunt et al. [10] proposed that an image f can be decomposed into a spatially nonstationary mean and a stationary residual component $f_0 = f - \bar{f}$. The nonstationary mean $\bar{f}$ describes the gross structure of an image and the residual component f_0 describes the detail variation of the

(a) Lena Image



b) Histogram of Lena Image

Figure 2.1 Lena image and its histogram

image. The stationary assumption about f_0 is an approximation for computational simplicity. Without introducing too much additional complication in computation, f_0 can itself be taken as a nonstationary white process. This suggests that f_0 is independent and characterized by its nonstationary variance [9]. This approach yields the nonstationary mean and nonstationary variance (NMNV) image model.

2.2 Noise Models

The effectiveness of the noise suppression and image restoration schemes depends on the statistical models for the noise processes. A lot of work on image restoration has assumed signal-independent noise. Many physical noise sources are, however, inherently signal-dependent. We used the noise models presented below to investigate the compression technique's performance on noisy images. The restoration filters are based on the same noise models. After image restoration using those filters, the compression technique's performance on the restored images was also studied.

2.2.1 Signal-Independent Additive Noise Model

This model is very common in digital image restoration and is given by

$$r(k,l) = f(k,l) + n(k,l), \tag{2.1}$$

where r(k,l) is the degraded image, f(k,l) is the original image and n(k,l) is the signal-independent additive Gaussian noise characterized by

$$n(k,l) \sim N\left(0,\sigma_n^2\right), \tag{2.2}$$

where N(m,b) denotes a Gaussian distribution with mean m and variance b. Here no blurring function is assumed.

2.2.2 Film-Grain Noise

Film-grain noise is produced when images are recorded on photographic film and are digitized for processing. This type of noise is signal-dependent [3]. The film-grain noise model [11] is generally described as

$$r(k,l) = f(k,l) + Cf^{p}(k,l)n(k,l),$$

where r(k,l) is the noisy image, f(k,l) is the original image, n(k,l) is

$$n(k,l) \sim N(0,1), \tag{2.3}$$

C is a scalar constant and p varies between 0.3 to 0.7. Here p=0.5 is assumed, so the model becomes

$$r(k,l) = f(k,l) + C\sqrt{f(k,l)}n(k,l). \tag{2.4}$$

2.2.3 Speckle Noise

Speckle noise occurs in all types of coherent imaging systems such as synthetic aperture radar (SAR) images for remote sensing, acoustic images, laser illuminated images and astronomical images. In coherent illumination, objects with roughness of the order of a wavelength cause speckle to appear in their images as formed by imaging systems which cannot resolve the microscale of the object roughness [12]. A general noise model for speckle is the multiplicative model [13]

$$r(k,l) = f(k,l)n(k,l), \tag{2.5}$$

where r(k,l) is the speckled image, f(k,l) is the original image and n(k,l) is the random noise, which is signal-independent with a negative exponential distribution. This multiplicative noise is a correlated noise. If we are only interested in signals which have smaller bandwidths than the noise spectrum we can undersample the speckle intensity image r(k,l) such that the sampling frequency is comparable to the bandwidth of f(k,l). In that case n(k,l) becomes uncorrelated as the correlation length of the noise is smaller than the sampling interval [14]. Then a single frame speckle image can be described by an uncorrelated multiplicative model where n(k,l) has a normalized negative exponential distribution, i.e., its probability distribution function (PDF) is described by

$$p(n(k,l)) = \begin{cases} \exp(-n(k,l)) & \text{for } n(k,l) \geq 0 \\ 0 & \text{otherwise} \end{cases}. \tag{2.6}$$

Many investigators have used the multiplicative model for speckle without any limitations on it. But the multiplicative model is subject to the limitations mentioned

above and is restricted to the situation where the degraded image has been sampled coarsely enough so that the degradation at any point can be assumed to be independent of all the other points [13].

In many applications, several independent speckle images of the same object are available. Frame averaging techniques can be applied to these images to increase the signal-to-noise ratio (SNR). The average of M independent speckled images is

$$r_a(k,l) = \frac{1}{M}\sum_{i=1}^{M} r_i(k,l), \tag{2.7}$$

where $r_i(k,l)$ is the i^{th} image frame. This average of M speckle frames can be shown to be the maximum likelihood estimate (MLE) of the undegraded image [13]. The average of M independent negative exponential random variables, n_a has a Gamma probability density function given by

$$p(n_a(k,l)) = \begin{cases} \dfrac{M^M}{\Gamma(M)} n_a^{M-1}(k,l)\exp(-Mn_a(k,l)) & \text{for } n_a(k,l) \geq 0 \\ 0 & \text{otherwise} \end{cases}, \tag{2.8}$$

where $\Gamma(M)$ is the Gamma function of order M. The signal-to-noise ratio (SNR) measured pointwise in a single frame speckle image is unity. The SNR of the image after averaging M frames is improved by the factor $\sqrt{M}$ [5].

CHAPTER III

EFFECTS OF NOISE ON IMAGE COMPRESSION

Image compression is desirable both to minimize storage in case of data archiving and to improve transmission speed. The aim of all compression schemes is to obtain a high compression ratio yet to reproduce the image with high fidelity. The compression ratio (CR) is given by

$$CR = \frac{n_o}{n_c}, \tag{3.1}$$

where n_o is the number of bits/pixel of the original image and n_c is the bits/pixel in the compressed image. In other words we can express the CR as the ratio of the sizes of the original image file and the compressed image file. The quality of the reconstructed (decompressed) image can be measured in terms of different error measurements, some of which were defined in Chapter I. The amount of achievable compression is affected by the presence of noise in an image [15, 16] and so is the quality of the reconstructed image. Before exploring how noise affects image compression, we will consider the image compression technique used in our work.

Image compression has two major categories: lossless and lossy compression. In lossless compression the original data can be recovered completely. This type of compression can give compression ratios ranging from 1.5 to 3 [16]. Lossy compression allows a slight loss of accuracy in order to achieve higher compression ratios. Due to the loss of information, the original signal cannot be recovered completely, resulting in some distortion in the image. In the case of images, these distortions can be made very small, so that visually they are undetectable. Clearly, a lossless method is desired when no information loss is tolerable. For graphics files, lossy methods may be used [17], which easily yield a compression ratio of 20:1 or higher [7].

At present, different image compression schemes are available. Adaptive Discrete Cosine Transform [18], vector quantization [19], fractals [20] and subband coding [21] are some of the techniques used in image compression. A more recent trend is the use of

the wavelet transform [22]. Thus a significant amount of research is underway which aims at the improvement and optimization of image compression techniques.

A lot of image compression is performed using special-purpose systems for individual requirements. With an increasing need for compressed images, users must archive and exchange compressed images between applications. Consequently a standard is required for the compression and the decompression of the images. A joint ISO/CCITT standardization group JPEG (Joint Photographic Experts Group) has developed standards for compressing continuous tone images. The compression technique specified by this group is called JPEG and this standard is widely used in different applications. JPEG defines both lossy and lossless processes. The lossy JPEG is based on the Discrete Cosine Transform (DCT) and the lossless JPEG process is based on Differential Pulse Code Modulation (DPCM) [7]. In our work, we have used lossy JPEG to study the effects of image noise on image compression. DCT-based JPEG is very effective in minimizing visible distortion and well suited for images. Our study used the fifth public release of the Independent JPEG Group's (IJG) software. This software implements baseline and extended-sequential compression processes. The "baseline" system is the minimum capability defined by JPEG that must be present in all modes of JPEG using DCT. In the sequential process, the encoder encodes the image in a single scan or pass through the data and the decoder also reconstructs the image at full quality in a single scan. Sequential DCT-based systems that have capabilities beyond baseline requirements are called "extended sequential systems" [7].

3.1 The JPEG Algorithm

Like all other image compression schemes JPEG has two basic components as shown in Figure 3.1. The output of the encoder is the compressed image which may either be stored or transmitted. Later, at some point, the compressed image has to be reconstructed by a decoder for further use. The steps in the JPEG compression algorithm are as follows:

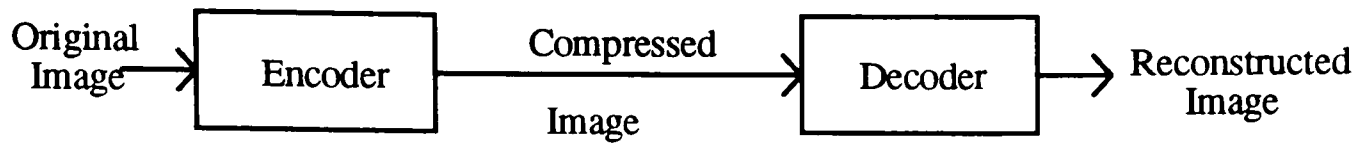


Figure 3.1 Image compression/decompression system

1. The image is subdivided into 8x8 pixel sub-blocks. If the image dimensions are not a multiple of 8, zero padding is used.

2. The 8 bit image data from the range 0 to 255 is converted to the range -128 to 127 by subtracting 128 from the image data. This reduces the precision required in DCT calculations [23].

3. Each 8x8 block is processed using the DCT and the transformed values are stored using 12 bits per pixel giving it a precision of 11 bits plus a sign bit.

4. The 64 DCT coefficients obtained are quantized to a lower precision (8 bit) using a user defined quantization table and any value smaller than some threshold is set to zero. This is the lossy step in the DCT-based JPEG.

5. The 1st of the 64 coefficients is the average or "DC" term. It is represented as a difference from the DC term of the previous block using a DPCM scheme.

6. The remaining 63 coefficients are scanned in a zig-zag order that starts with the lowest frequency and progresses to the highest. The terms are collected together and are coded as the number of consecutive zero terms.

7. The entire data is then further compressed using Huffman coding. This part of JPEG provides the lossless compression.

Figures 3.2 and 3.3 show the DCT-based encoder and decoder processing steps, respectively.

3.1.1 The Discrete Cosine Transform (DCT)

The key to the compression of images by JPEG is the Discrete Cosine Transform [17], which decomposes the image into its underlying spatial frequencies. Most of the graphical images are composed of low frequency information and the DCT has the ability

to pack most of the information in the first few low frequency coefficients. When DCT is applied to an 8x8 sub-block of the image, we have an 8x8 matrix of DCT coefficients. The component in row and column zero (DC component) carries more information than the higher frequency components. As we move further away from the DC component, the coefficients are smaller, as is the information content. Thus the DCT identifies the pieces of information which can be discarded without seriously compromising the quality of the image [17]. Before applying the DCT, a level shift is performed on the 8x8 sub-blocks by subtracting 128 from the 8 bit data image. This reduces the internal precision required for the DCT calculation.

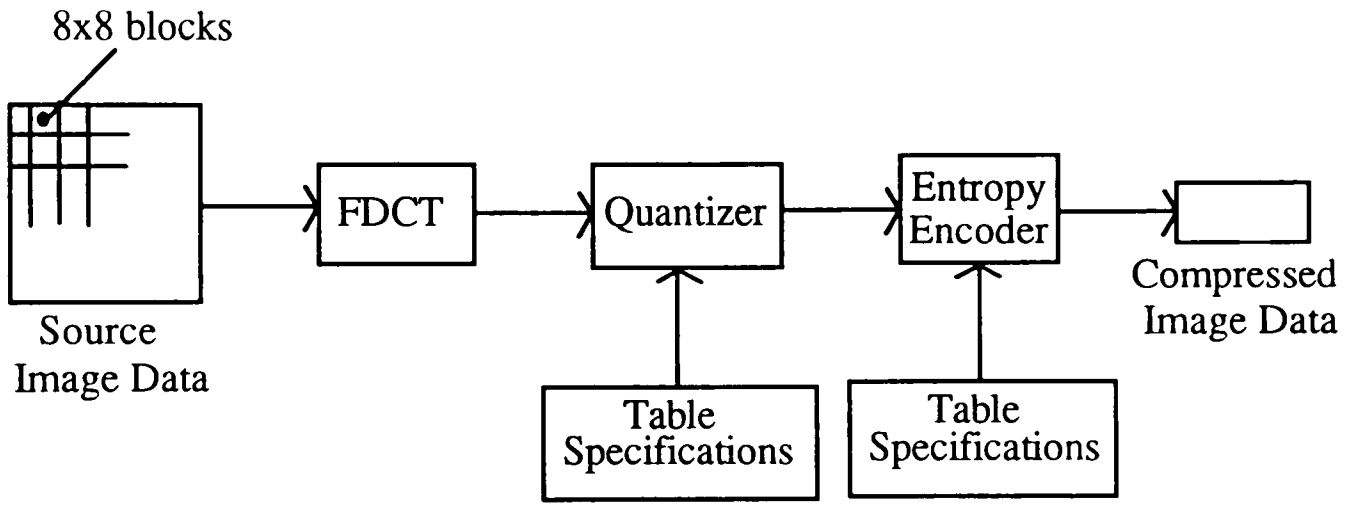


Figure 3.2 DCT-based encoder processing steps

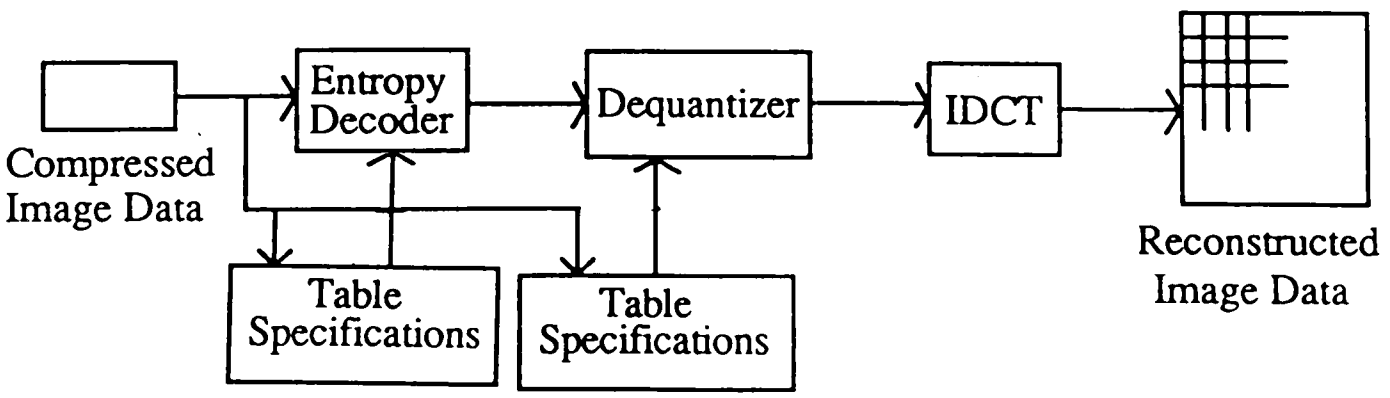


Figure 3.3 DCT-based decoder processing steps

3.1.2 Quantization

The DCT matrix takes up more space than the original pixels. Thus quantization is performed on the DCT matrix, which is the process of reducing the number of bits required to store an integer by reducing the precision of the integer. JPEG thus quantizes each of the 64 DCT coefficients employing a 64 element quantization matrix which is specified by the user. The elements in the matrix may be an integer value from 1 to 255 which define the quantization step size for the corresponding DCT coefficients. Quantization is a many-to-one mapping and is the main source of lossiness in JPEG. Quantization is defined by the following equation [23]

$$F^{Q}(u,v) = \text{Integer Round}\left(\frac{F(u,v)}{Q(u,v)}\right), \tag{3.2}$$

where F(u,v) is the forward DCT coefficient and Q(u,v) is the corresponding quantizer step size from a user-defined 8x8 quantization matrix. The high frequency coefficients of an 8x8 sub-block of the image generally have smaller amplitudes compared to the low frequency components. The quantizer step sizes are larger for the high frequency coefficients than for the low frequency coefficients. Consequently a large number of high frequency components will round off to zero. On the other hand significant DCT coefficients with non zero values are represented with no greater precision than necessary for the desired image quality. The dequantization which occurs while performing decompression is [23]

$$F^{Q'}(u,v) = F^{Q}(u,v)Q(u,v). \tag{3.3}$$

A quantization matrix for a monochrome image, as given in JPEG specification, is shown in Figure 3.4. This quantization matrix is given as an example in the JPEG specifications and is said to give good results on 8 bits per pixel monochrome images [7]. The user, however, may define any quantization matrix suitable for his/her own application. The aim of the compression scheme is to compress the image as much as possible without visible artifacts. Thus in the quantization matrix each step size should be chosen as the perceptual threshold for the visual contribution of its corresponding cosine

basis function. These thresholds are functions of image characteristics, viewing distance etc. [23]. With these parameters defined, the thresholds can be determined by psychovisual experiments. Based on such experiments the quantization matrix shown in Figure 3.4 was derived by the JPEG committee [23].

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Figure 3.4 Quantization matrix for the 8x8 sub-blocks of a monochrome image

3.1.3 Quality Factor

The quantization step sizes of the quantization matrix determine how much compression can be achieved for a particular image. If we make the quantizer step sizes coarse, we would get high compression at the expense of losing some information due to quantization errors. Finer quantization step sizes would give less compression; however, reconstructed image quality would be better because of less quantization loss. We can trade-off the compression ratio against the quality of the reconstructed image. In general, the higher the compression ratio the lower will be the quality of the image. The IJG software for JPEG lets the user define a quality factor for the reconstructed image on a scale of 1 to 100. A quality factor (Q) of 100 makes all the entries in the quantization matrix unity and this will result in no quantization loss. According to the JPEG specification the quantization matrix given in Figure 3.4 yields a 'good' visual quality of the reconstructed image. For a quality factor (Q) of 50 the IJG JPEG software uses the quantization matrix of Figure 3.4 as it is. For any quality factor other than 100, each

element of the quantization table of Figure 3.4 is multiplied by a scaling factor SF which is defined by [24]

$$SF = \begin{cases} \dfrac{50}{Q} & \text{for } 1 \le Q \le 50 \\ \dfrac{200 - 2Q}{100} & \text{for } 50 \langle Q \langle 100 \end{cases} . \qquad (3.4)$$

For a desired quality factor of 75, SF=0.5. Thus each element of the quantization matrix shown in Figure 3.4 is divided by 2 and the resulting quantization matrix is used for compressing the images with a quality factor of 75.

3.1.4 DC Coding and Zig-zag Sequence

After quantization, the DC coefficient is treated differently than the 63 AC coefficients. As the adjacent blocks of an image exhibit a high degree of correlation, the DC coefficient is coded as a difference from the DC element of the previous block. The rest of the 63 AC coefficients are arranged in a zig-zag sequence as shown in Figure 3.5, and are then encoded. This facilitates entropy coding by placing low frequency components (more likely to be non-zero) before high frequency coefficients [23].

3.1.5 Entropy Encoding

After zig-zag sequencing, the coefficients are coded using Run-Length-Encoding (RLE) [25]. It gives a count of consecutive zero values in the images. Since a lot of the coefficients are rounded off to zero in most of the images, RLE yields outstanding compression [17]. Then JPEG uses entropy coding which can be either Huffman coding or Arithmetic coding. In our work Huffman coding has been used, which is a variable length coding technique. This combination of RLE and Huffman coding yields lossless compression.

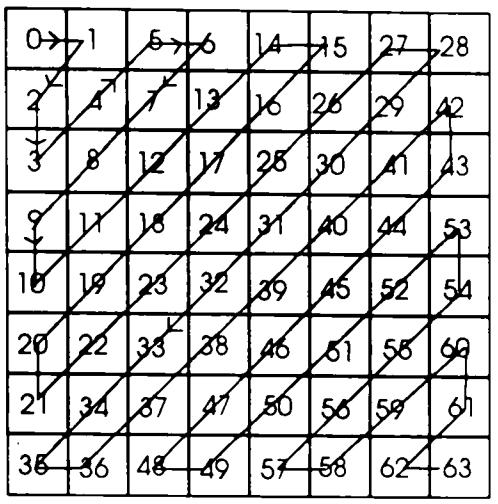


Figure 3.5 Zig-zag sequence of the quantized DCT coefficients

3.2 Effect of Noise on Compression Using JPEG

The compression scheme of JPEG is based on the statistical properties of the image and also on the frequency components present in an image. Both signal-dependent noise (SDN) and signal-independent noise (SIN) affect these properties and significantly change the resulting compression ratio.

First, we will consider the DCT and the quantization step size of JPEG. As mentioned earlier, noiseless images carry most of the significant information in the low frequency regions. Noisy images, on the other hand, have a wide range of spatial frequency components. The forward DCT has been applied on the 8x8 blocks of noiseless and noisy images to illustrate this point. Figure 3.6(a) shows an 8x8 sub-block of the noiseless Lena image and Figure 3.6(b) shows the corresponding forward DCT coefficients. After adding signal-independent, zero mean Gaussian noise of a variance of 1000 to the Lena image according to the model presented in section 2.2.1, the DCT is again applied on the same 8x8 sub-block. The DCT result for the noisy image is shown in Figure 3.7(b). In both of the cases, the DCT is applied directly to the image pixels without performing any level shifting.

196	175	173	172	147	114	106	97
189	174	172	154	141	118	107	100
186	184	161	143	125	116	112	100
200	172	159	138	124	122	110	104
201	161	152	130	132	109	115	96
194	153	149	136	125	118	112	106
178	151	145	124	124	116	107	106
183	163	128	122	121	120	120	107

(a) Image samples

3.5E4	5008	760	640	303	343	98	272
902	553	–487	–334	31	18	–45	–23
49	–119	–228	–77	72	–61	–98	–87
98	77	–202	–17	262	164	114	151
122	83	72	138	70	59	–93	12
–14	7	–29	–88	–5	–18	94	–89
81	10	62	4	–19	49	–112	–123
–157	37	–16	–20	–37	–35	–121	90

(b) Forward DCT coefficients

Figure 3.6 Noiseless Lena image samples and DCT coefficients

209	183	190	164	94	83	107	94
159	226	208	160	129	144	107	96
185	236	145	155	129	127	124	60
206	166	137	108	89	141	113	145
163	183	204	145	102	177	112	119
187	172	157	131	114	147	124	112
207	155	165	137	86	97	110	115
179	106	125	135	175	99	125	121

(a) Image samples

3.6E4	5034	1128	-11	-806	394	454	-429
713	1105	-220	-432	-807	-230	-359	-477
-453	281	-357	-324	846	400	-425	698
229	268	159	-553	-368	334	628	-164
-303	-574	241	-39	558	196	483	264
-804	359	1075	61	493	501	-643	128
-170	10	-189	567	240	63	-643	128
146	537	-391	-222	-330	181	-200	-133

(b) Forward DCT coefficients

Figure 3.7 Noisy Lena image samples and DCT coefficients

From the DCT results obtained, we see that the high frequency DCT coefficients of the noisy image are significantly greater in amplitude when compared to the noiseless case. Thus when quantization is applied to the DCT coefficients of the noisy image, most of the coefficients will have a non-zero value, whereas in the case of the noiseless image a lot of the coefficients will round off to zero. Even if the noiseless image coefficients have a non-zero value after quantization, they are generally quantized to a very low precision, which is not true for the noisy case.

The concept of the zig-zag sequence also fails to yield compression, as the noise spectrum occupies almost all the frequency range from low to high values. The run-length-coding will yield less compression in the case of noisy images compared to noiseless images, as most of the DCT coefficients after quantization are non-zero.

The presence of noise significantly destroys the correlation between image pixels [15]. The DC coefficient encoding is based on inter-pixel correlation. With the absence of strong correlation between adjacent 8x8 sub-blocks of pixels, the DC coding suffers in the presence of noise.

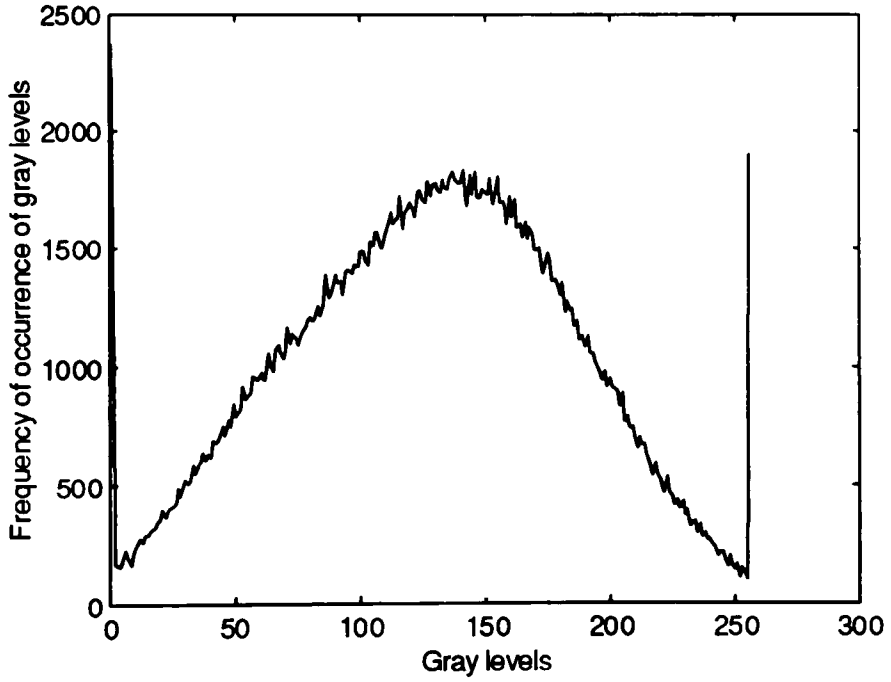
We have considered three types of noisy images, which were simulated according to the models presented in section 2.2. The noisy image with additive signal-independent noise (SIN) was simulated using Gaussian noise n(k,l) where $n(k,l) \sim N(0,1000)$. The image with film-grain noise (FGN) was simulated using a factor C=4 (as defined in equation 2.4) and the speckle image was simulated by averaging four frames of independent speckle images that were generated using the multiplicative model. Figure 3.8 shows the noisy images and the corresponding histograms.

The histogram and the entropy of an image give a fair idea of how Huffman coding will be affected by the presence of noise. Huffman coding is a variable length coding scheme with shorter codes for highly probable symbols and longer codes for symbols with low probability of occurrence. For Huffman coding to be efficient, the histogram of an image needs to deviate from a uniform probability density, which has the maximum entropy when compared to other density functions. Entropy is defined as

$$H = -\sum_{i=1}^{m} P_i \log_2 P_i \quad \text{bits}, \tag{3.5}$$

where P_i is the probability of occurrence of the i^{th} gray level and m is the number of gray levels. Using Equation (3.5), the entropies for the noiseless image and noisy images are: H (noiseless)=7.4037 bits, H (with additive SIN)=7.7703 bits, H (with FGN) =7.7966 bits and H (with speckle noise)=7.6291 bits.

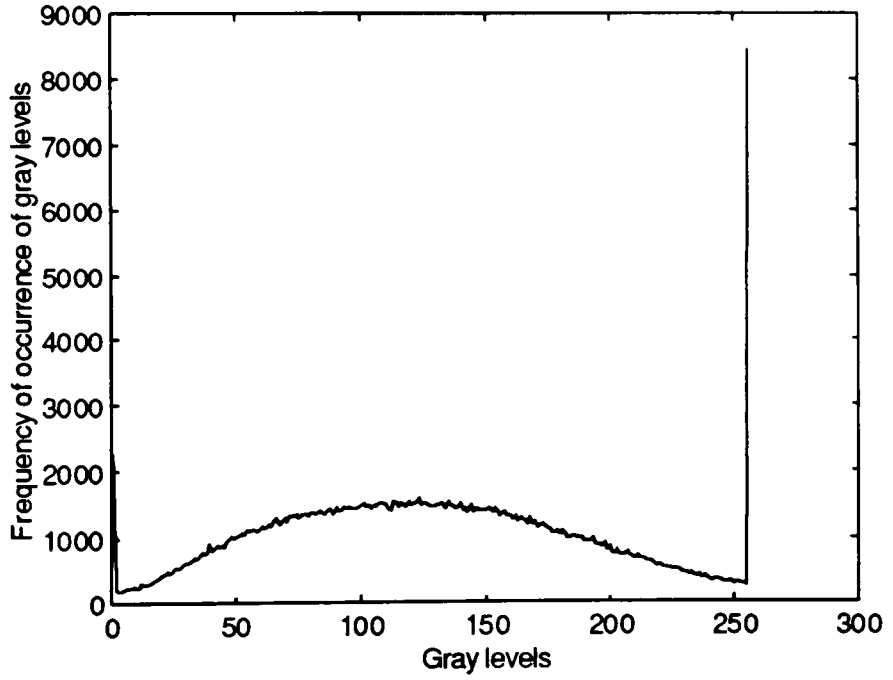
(a) Lena image with additive SIN



(b) Histogram of (a)

Figure 3.8 Noisy images and the corresponding histograms

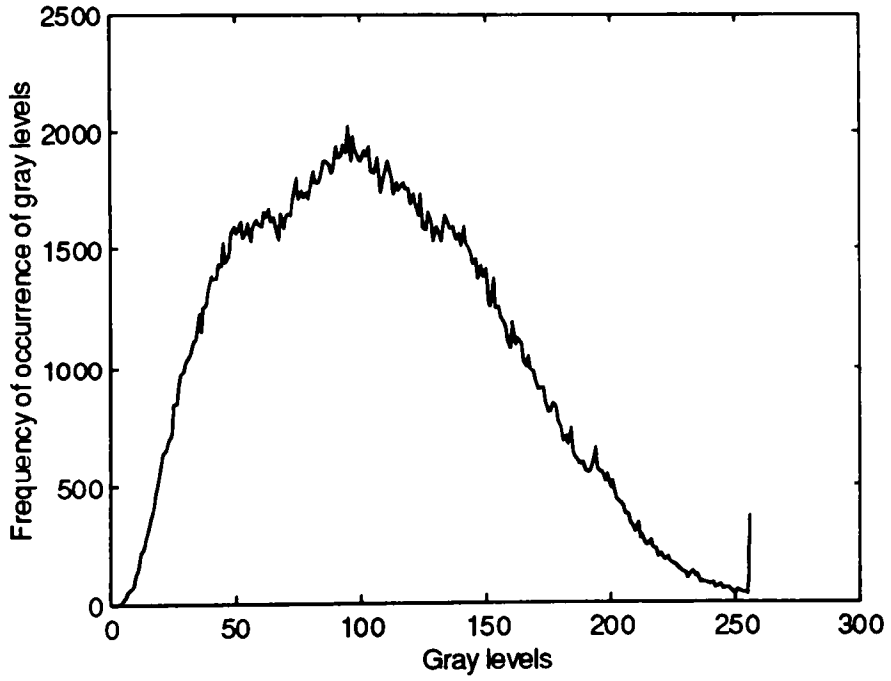
(c) Lena image with film-grain noise (C=4)



(d) Histogram of (c)

Figure 3.8 Continued

(e) Lena image with speckle noise (4 frames-averaged)



(f) Histogram of (e)

Figure 3.8 Continued

The noiseless Lena image has a multipeak histogram that deviates greatly from the uniform distribution and as a result its entropy is low. When SIN is added to the image, the histogram of the noisy image looks almost Gaussian and the histogram of the speckle image has the shape of the Gamma distribution density. Thus these histograms are flatter than the noiseless image histograms. The image with film-grain noise also has a flat histogram. As a result the noisy images have higher entropies, so that Huffman coding should be more efficient for the noiseless images than for the noisy images.

3.3 Compression Results

JPEG compression was applied to the noiseless Lena image and the noisy Lena image with additive SIN having a noise variance of 1000. Results obtained with different quality factors are presented in Table 3.1. The results for the film-grain noise and the speckle noise are presented in Chapters IV and V. The errors are calculated between the input image X to the compression scheme and the reconstructed image X_c after compression, thus in the case of noisy images, X is the noisy image before any compression is applied and X_c is the reconstructed image after compression is applied to the noisy image.

The JPEG specified quantization matrix shown in Figure 3.4 was used for a quality factor of 50 in our computer simulations and according to the JPEG specification it yields a good visual quality of the reconstruction image [7]. A peak signal-to-noise ratio (PSNR) of 35 dB and above is considered satisfactory for a compression scheme. From Table 3.1, we see that for the noiseless Lena image, a quality factor (Q) of 45 and above gives good PSNR. Within this range of Q, the highest compression ratio obtained is 13:1. Considering the visual quality of the reconstructed image, a quality factor as low as 25 may be tolerated for some applications and it yielded a CR of approximately 19:1. Below that range the reconstructed image shows blocking effects and the visual quality of the reconstructed images deteriorates. Figure 3.9 shows the reconstructed Lena images for different quality factors.

Plots of bpp (bits per pixel) and MSE (log based 10 scale) and CR versus Q for the noiseless and the noisy case are shown in Figure 3.10. We see that as the compression ratios are less for the noisy images compared to the noiseless images, the bpp required are greater for the noisy images. For noisy images, the MSEs are greater than the noiseless images. From Table 3.1, it is seen that the Lena image with SIN after decompression has a very low PSNR even for a quality factor of 75 or 65. But this result can be somewhat misleading. The lossy compression part of JPEG discards the high frequency information and in the case of Lena with SIN, the high frequency information comes from the noise. So if the significant information of the original noiseless image is contained in the low frequency part (as is true for most real world images) then JPEG discards some of the high frequency noise component and acts like a low pass filter. Thus the error between the reconstructed noisy image and the uncompressed noisy image is larger as compared to the noiseless case. The information loss, however, may not be great even if the PSNR is below 30 dB. We have investigated this aspect in detail for film-grain noise in Chapter IV.

Table 3.1 Compression results for the noiseless Lena image and noisy Lena image with SIN of variance 1000

Quality	Noiseless image			Noisy image		
factor	CR	MSE	PSNR(dB)	CR	MSE	PSNR(dB)
95	2.9157	3	43.90	1.3057	4	42.32
75	8.159	10	37.97	2.4437	94	28.42
65	10.005	13	36.99	2.8663	176	25.67
45	13.496	18	35.63	3.8194	366	22.50
35	15.7103	21	34.88	4.65	482	21.30
25	19.3289	27	33.79	6.3933	617	20.23
15	26.033	41	32.02	10.325	757	19.34

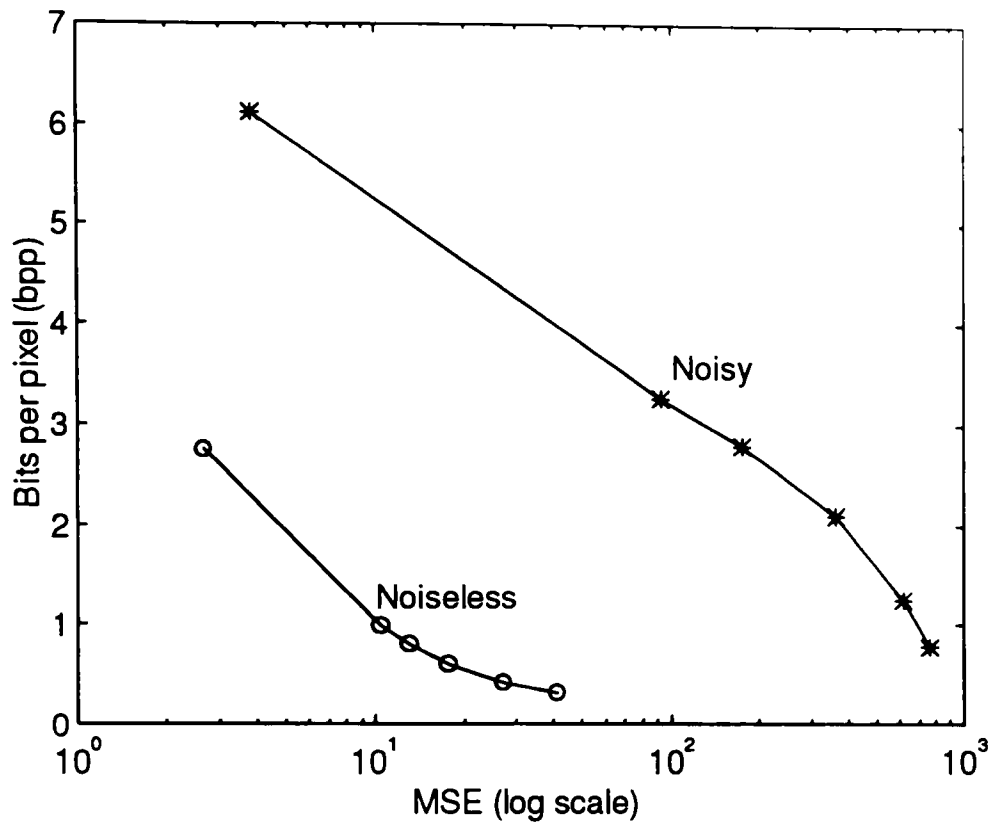
(a) Quality factor 65

(b) Quality factor 25

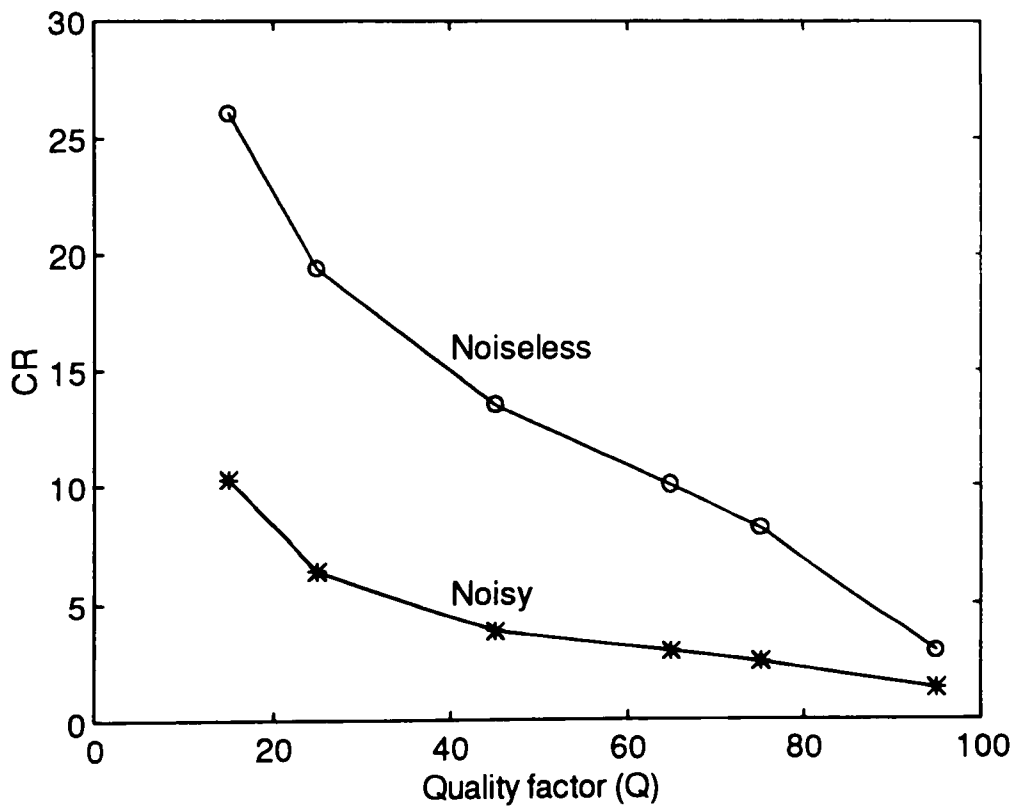
Figure 3.9 Reconstructed Lena images after compression at different quality factors

(c) Quality factor 15

Figure 3.9 Continued



(a) Plots of bpp versus MSE



(b) Plots of compression ratio (CR) versus quality factor

Figure 3.10 Plots of compression results for noiseless Lena and Lena+SIN

The main disadvantage due to the presence of noise is the reduction of the achievable compression. This fact is illustrated in the plots of the compression ratios versus the quality factors (Figure 3.10(b)) for both the noiseless and noisy cases. For a quality factor of 65, we obtained a CR of 10 for the noiseless Lena image. For the same quality factor, the CR for the noisy image with SIN is reduced by a factor of 3.5. For a quality factor of 25, the CR is reduced by a factor of 3. So if we want the same amount of compression for the noisy image and the noiseless image, we need to use a lower quality factor for the noisy image. A compression ratio of 10 is considered a reasonable compression factor in many applications. A quality factor of 65 yields that for Lena without noise and this also provides a good picture quality. For Lena with signal-independent additive noise of variance 1000, however, a quality factor of 15 gives a CR of 10. As mentioned earlier a quality factor of 15 gives a quantization too coarse for baseline JPEG and results in blocking effects in the reconstructed image.

CHAPTER IV
NOISE SUPPRESSION AND COMPRESSION IN FILM-GRAIN NOISE

The compression results from Chapter III show that there is a significant reduction in the compression ratio when noise is present. A noisy image requiring compression can either be pre-processed or post-processed for noise suppression. Any application that can tolerate low compression ratios can use the compression technique as it is and then post-process the image for noise suppression after compression/decompression. An alternative is to devise a compression algorithm that can give high compression ratios even in the presence of noise. A third approach, useful when high compression ratios are required, is to pre-process the image for noise suppression and then use compression on the restored image in order to obtain high compression ratios. We have mainly explored the third approach in our work. We have used different image restoration techniques on the noise-degraded images and then used JPEG to compress the restored images. The block diagram in Figure 4.1 shows the pre-compression noise suppression scheme for the noisy images.

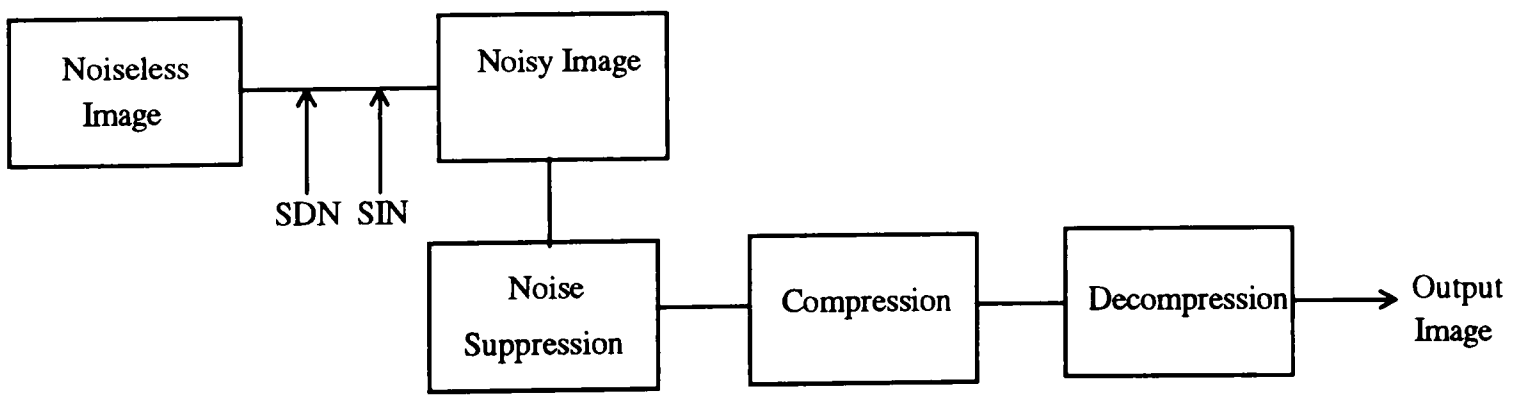


Figure 4.1 Pre-compression noise suppression scheme

We can deal with the signal-dependent noise (SDN) in two ways. In one approach (scheme 1), estimators specifically designed for a particular SDN model are applied to the

image degraded by SDN for noise suppression purposes [3, 5, 9]. In the second approach (scheme 2) the SDN is transformed into signal-independent noise(SIN) [26, 27] and then an estimator designed for SIN is applied to the transformed image [11]. After that an inverse transformation is applied to transform the image back to the original space. For the images degraded by SDN, we have used both the filters designed for SDN models on the noisy images and the filters designed for SIN on the transformed images for noise suppression. We have compared the performances of the two approaches. For images with SIN, we used the filters designed for SIN. The transformation used to convert the SDN into SIN is called homomorphic transformation (HT) [4, 11]. The noise suppression block in Figure 4.1 can be expanded in the following block diagram shown in Figure 4.2. After noise suppression the restored images were compressed and the results were compared again.

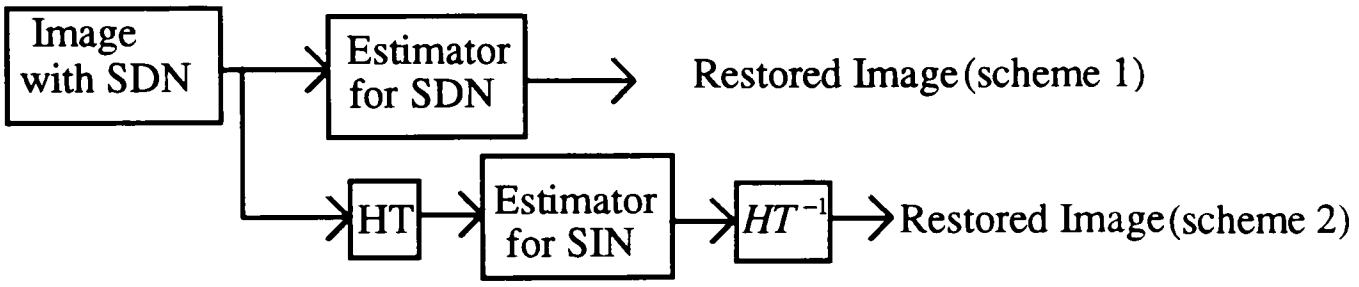


Figure 4.2 Two alternative noise suppression schemes for SDN

In this chapter, we will present the general forms of the filters for SDN and SIN and the homomorphic transformation for SDN. Then the results of noise suppression and image compression are presented. Some results for additive Gaussian signal-independent noise are also presented.

4.1 Estimator for Signal-Dependent Noise

The estimator for signal-dependent noise that we have used was designed by Kuan et al. [9]. This estimator is an adaptive noise smoothing filter applicable for signal-dependent noise. The filter was developed using the local linear minimum mean square

error (LLMMSE) criterion and a nonstationary image model [9]. Kuan used this filter for multiplicative noise and Poisson noise and suggested its possible use for film-grain noise. We have used the adaptive noise smoothing filter for the multiplicative noise model for speckle suppression in Chapter V. In this chapter, the general form of the filter is modified for film-grain noise (FGN) and used for images degraded by FGN.

For the derivation of the LLMMSE filter, the equation used for the degraded image is [9]

$$\underline{r} = H\underline{f} + \underline{u}, \tag{4.1}$$

where $\underline{r}$ is the degraded image, $\underline{f}$ is the original image, $\underline{u}$ is a zero mean noise that can be either signal-dependent or signal-independent and H is the blurring matrix. When H is an identity matrix, i.e., there is no blurring, the LLMMSE filter is given by a set of scalar equations [9]

$$\hat{f}_{LLMMSE}(k,l) = \bar{f}(k,l) + \frac{\sigma_f^2(k,l)}{\sigma_f^2(k,l) + \sigma_u^2(k,l)}\left[r(k,l) - \bar{r}(k,l)\right]. \tag{4.2}$$

Here $\bar{f}(k,l)$ and $\bar{r}(k,l)$ are the means of f(k,l) and r(k,l), respectively, and σ_f and σ_u are the standard deviations of f(k,l) and u(k,l), respectively. As there is no blur, we can write the degraded image model shown in Equation (4.1) as

$$r(k,l) = f(k,l) + u(k,l). \tag{4.3}$$

In the above equation, u(k,l) is an uncorrelated noise. Though u(k,l) is shown as an additive noise, it does not have to be additive in the usual sense. Any noise degradation can be expressed in terms of a signal and a noise term. The models described in section 2.2 for FGN and speckle noise can be expressed in this form [9].

4.2 Adaptive Noise Smoothing Filter

The LLMMSE filter shown in Equation (4.2) uses the ensemble mean and variance of the original image pixel values, f(k,l). These statistics are generally not known a priori, as most of the time we only have the degraded image. Thus statistics may need to be

estimated from the noisy image. When local statistics are used in place of the ensemble statistics in Equation (4.2), it yields the adaptive noise smoothing filter [9]

$$\hat{f}(k,l) = m_f(k,l) + \frac{V_f(k,l)}{V_f(k,l) + V_u(k,l)}\left[r(k,l) - m_r(k,l)\right], \tag{4.4}$$

where $m_f(k,l)$ and $m_r(k,l)$ are the local means of $f(k,l)$ and $r(k,l)$, respectively, and $V_f(k,l)$ and $V_u(k,l)$ are the local variances of $f(k,l)$ and $u(k,l)$, respectively. The ratio of local variances introduces a nonlinearity in the filter as they are estimated by using nonlinear functions of the noisy observations. Thus the adaptive noise smoothing filter is a nonlinear filter even though the LLMMSE filter is a linear filter.

The local statistics are calculated over a uniform moving average window of size (2M+1)x(2N+1). The local mean of r(k,l) is given by

$$m_r(k,l) = \frac{1}{(2M+1)(2N+1)} \sum_{i=k-M}^{k+M} \sum_{j=l-N}^{l+N} r(i,j), \tag{4.5}$$

and the local variance is given by

$$V_r(k,l) = \frac{1}{\left[(2M+1)(2N+1) - 1\right]} \sum_{i=k-M}^{k+M} \sum_{j=l-N}^{l+N} \left[r(i,j) - m_r(k,l)\right]^2 . \tag{4.6}$$

$m_r(k,l)$ and $V_r(k,l)$ are the sample mean and sample variance, respectively.

The local statistics of $f(k,l)$ can be calculated from the local statistics of $r(k,l)$ with the assumption that the relationship between their ensemble statistics also holds for local statistics [9]. These calculations depend on the particular noise structure under consideration. For additive signal-independent noise, we have the model $r(k,l) = f(k,l) + n(k,l)$, where $n(k,l)$ is a zero mean noise. For this type of SIN model, the local mean of $f(k,l)$ is given by the local mean of $r(k,l)$ and the local variance of $f(k,l)$ is [9]

$$V_f(k,l) = V_r(k,l) - \sigma_n^2(k,l), \tag{4.7}$$

where $\sigma_n^2(k,l)$ is the noise variance. Thus the filter for SIN is given by,

$$\hat{f}(k,l) = m_r(k,l) + \frac{V_r(k,l) - \sigma_n^2(k,l)}{V_r(k,l)}\left[r(k,l) - m_r(k,l)\right],$$

or

$$\hat{f}(k,l) = m_r(k,l) + W\left[r(k,l) - m_r(k,l)\right], \tag{4.8}$$

where

$$W = Max\left[\left(1 - \frac{\sigma_n^2}{V_r}\right), 0\right]. \tag{4.9}$$

Equation (4.9) ensures that W will always be positive and that there will be no noise amplification. Equation (4.8) yields the same form as Lee's local statistics estimator [9]. The James-Stein estimator for additive Gaussian noise also has the same form [11, 28].

4.3 Noise Suppression Results for Images Degraded by SIN

We have used the filter given by Equation (4.8), i.e., the adaptive noise smoothing filter for SIN, which is the same as the James-Stein estimator, on the Lena image with additive zero mean Gaussian noise of variance 1000. We have calculated different errors as a measure of performance of the noise suppression schemes. All the errors presented in relation to noise suppression or image restoration have been calculated w.r.t. the original noiseless image.

The general window sizes used for the estimators are 3x3, 5x5 and 7x7. A smaller window size, such as 3x3, preserves the edge information in the image better than the larger window sizes, but fails to smooth the noise sufficiently. The 5x5 and 7x7 windows smooth the noise better than the 3x3 window at the expense of smoothing some edge information in the image. We have used 3x3, 5x5 and 7x7 windows for the James-Stein estimator. Figure 4.3 shows the restored images with different window sizes. From the restored images we see that the 5x5 and 7x7 windows give more noise suppression compared to the 3x3 window, however edges are blurred. The blurring effect is more noticeable in the case of the 7x7 window. From Table 4.1, we see that the MSE and the NMSE are smaller for the 7x7 window, as it can smooth out more noise than the other window sizes. The LOGMSE obtained, however, is slightly greater than for the 5x5 window. This is because of the distortion introduced by the edge blurring. The LOGMSE

(a) Lena with SIN (variance 1000), MSE=972

(b) Restored image with 3x3 window, MSE=203

Figure 4.3 Noisy Lena and restored images obtained using the adaptive noise smoothing filter for SIN

(c) Restored image using 5x5 filter, MSE=132

(d) Restored image using 7x7 filter, MSE=128

Figure 4.3 Continued

error measurement is based on the human visual model. Based on this error criterion we can say that the image restored by the 5x5 window gives the best result which is a trade-off between the noise smoothing and the preservation of edge details.

Table 4.1 Noise suppression results for Lena image with SIN (variance 1000) using the adaptive noise smoothing filter for SIN

Image type	MSE	NMSE	LOGMSE	RMSSNR (dB)
Noisy image	972	0.053	0.0122	12.78
SIN filter (3x3)	203	0.011	9.48e-4	19.58
SIN filter (5x5)	132	0.007	6.26e-4	21.46
SIN filter (7x7)	128	0.007	6.38e-4	21.57

4.4 Adaptive Noise Smoothing Filter for Film-Grain Noise

The model employed for film-grain noise is

$$r(k,l) = f(k,l) + C\sqrt{f(k,l)}n(k,l), \tag{4.10}$$

where $n(k,l) \sim N(0,1)$ and C is a scalar constant. Comparing the above equation with Eq. (4.3) we have,

$$u(k,l) = C\sqrt{f(k,l)}n(k,l). \tag{4.11}$$

We can also say that $E[r(k,l)] = E[f(k,l)]$ i.e.,

$$m_r(k,l) = m_f(k,l). \tag{4.12}$$

From Equation (4.11) we obtain the variance of u(k,l),

$$V_u(k,l) = \sigma_u^2(k,l) = C^2 m_f(k,l)\sigma_n^2(k,l), \tag{4.13}$$

where $\sigma_n^2(k,l)$ is the variance of $n(k,l)$ and $m_f(k,l)$ is the mean of $f(k,l)$. The local variance of $f(k,l)$ is $V_f(k,l) = V_r(k,l) - \sigma_u^2(k,l)$ [9]. Using Equation (4.13), the local variance of $f(k,l)$ is

$$V_f(k,l) = V_r(k,l) - C^2 m_f(k,l)\sigma_n^2(k,l). \tag{4.14}$$

Using all these results in Equation (4.4) we obtain the adaptive noise smoothing filter for the film-grain noise

$$\hat{f}(k,l) = m_r(k,l) + \frac{V_r(k,l) - C^2 m_f(k,l)\sigma_n^2(k,l)}{V_r(k,l) - V_u(k,l) + V_u(k,l)}\left[r(k,l) - m_r(k,l)\right].$$

or

$$\hat{f}(k,l) = m_r(k,l) + W_F\left[r(k,l) - m_r(k,l)\right], \tag{4.15}$$

where

$$W_F = Max\left[\left(1 - \frac{C^2 m_r(k,l)\sigma_n^2(k,l)}{V_r(k,l)}\right), 0\right]. \tag{4.16}$$

4.5 Noise Suppression Results for Film-Grain Noise Using the Adaptive Noise Smoothing Filter

We have used C=4 and C=2 and thus generated noisy images of Lena with two different levels of film-grain noise. For noise suppression, the adaptive noise smoothing filter was used on the noisy images. We have used 3x3, 5x5 and 7x7 windows on the noisy Lena image generated with C=4. The MSE obtained for different window sizes are 356, 198 and 176, respectively, whereas the MSE of the noisy image is 1869. The 7x7 window gives the lowest error, however, from Figure 4.4, we see that there is significant blurring of the image. As mentioned earlier, use of larger window sizes smooth out more noise at the expense of blurring. On the other hand, the image restored by a 5x5 window gave less noise suppression, but kept more edge information. The difference between the MSE obtained for the 5x5 and 7x7 windows is not large, thus we considered the 5x5 window to be optimum. Later in the chapter, while using compression on the restored images, we have used the images restored using a 5x5 window. The different restoration results obtained for different noisy images are presented in Table 4.2.

The results of restoration from the noisy Lena with FGN (C=2) are shown in Figure 4.5, where a 5x5 window was used. The MSEs are shown in the figure captions. Comparing the restored images (5x5 window) in Figures 4.4(c) and 4.5, where the noisy images had C=4 and C=2, respectively, it is seen that the restored image is less blurred in the second case. This is because the amount of noise was less.

(a) Lena with FGN (C=4), MSE=1869

(b) Restored image using 3x3 window, MSE=356

Figure 4.4 Lena image with FGN (C=4) and the restored images using the adaptive noise smoothing filter

(c) Restored image using 5x5 window, MSE=198

(d) Restored image using 7x7 window, MSE=176

Figure 4.4 Continued

Besides the Lena image we also considered another image, the "Airplane" image in this section. From Figure 4.6, we see that the Airplane image has a significantly different histogram from the histogram of the Lena image. The adaptive noise smoothing filter (5x5) reduced the MSE of the noisy Airplane (C=4) from 2243 to 221, which is approximately a factor of 10. The MSE of the noisy (C=4) Lena image was reduced by a factor of 9.44 when a 5x5 window was used. The restoration results for the Airplane image are shown in Figure 4.7

Table 4.2 Noise suppression results for the Lena image with film-grain noise (C=4) using the adaptive noise smoothing filter

Image type	MSE	NMSE	LOGMSE	RMSSNR (dB)
Noisy image	1869	0.1015	0.016	9.94
Filter (3x3)	356	0.0193	0.0014	17.14
Filter (5x5)	198	0.0107	8.168E-4	19.69
Filter (7x7)	176	0.0096	7.7052E-4	20.19

4.6 Combined Homomorphic and Local-Statistics Processing for Film-Grain Noise

In the previous sections, we have presented and used a filter that takes into account the signal dependence of the noise. In this section we will present the second approach for dealing with the signal-dependent noise. The technique uses a transformation to map the degraded image into a space where the noise becomes signal-independent, additive and Gaussian in nature [4, 26, 27]. Then the local-statistics algorithm for SIN is applied to the transformed image. After processing, the inverse transform is applied to the image to get the image back into the original space.

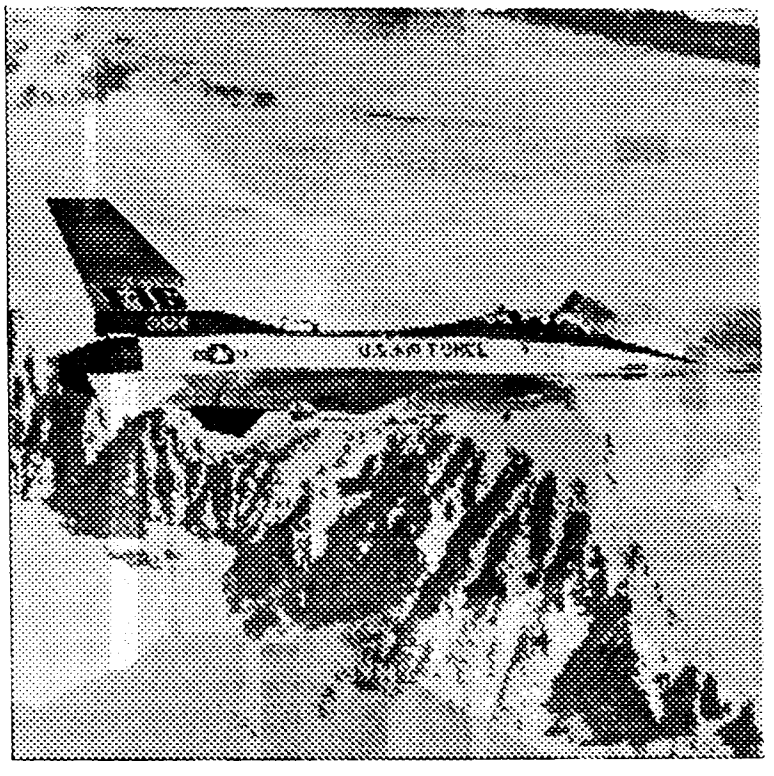
The transformation we have used in this section was suggested by Arsenault et al. [4, 11] and is called a homomorphic transformation (HT). For the film-grain noise model, that we have used, the transformation is given by [4, 11]

$$r'(k,l) = \alpha\sqrt{r(k,l)}\,.$$

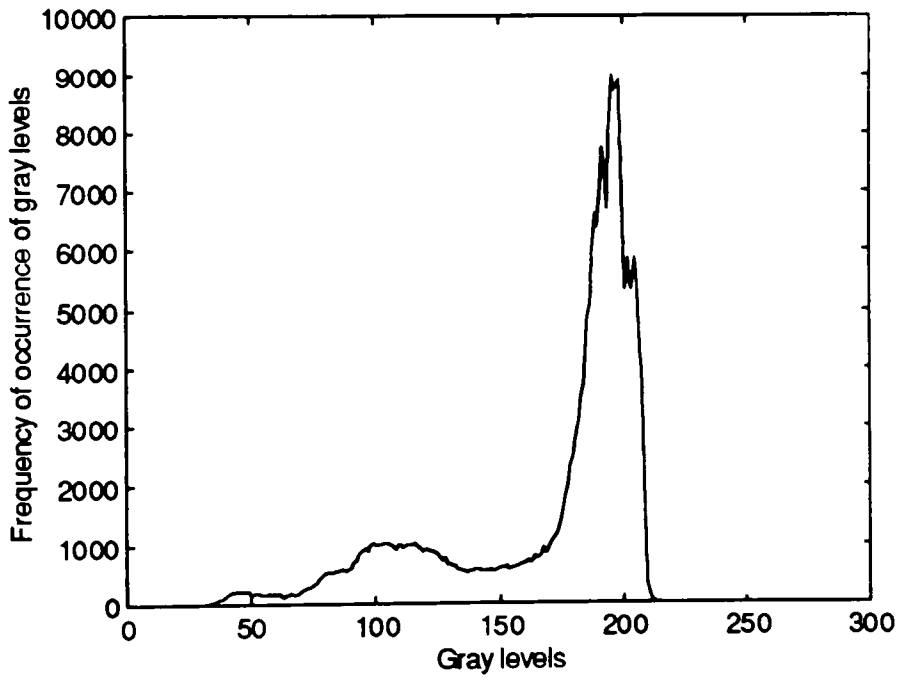
(a) Lena with FGN (C=2), MSE=502

(b) Restored image, MSE=86

Figure 4.5 Noisy Lena with FGN (C=2) and the restored image using the adaptive noise smoothing filter (5x5)



(a) Airplane image



(b) Histogram of the Airplane image

Figure 4.6 Airplane image and its histogram

(a) Airplane with FGN (C=4), MSE=2243

(b) Restored image, MSE=221

Figure 4.7 Airplane image with FGN (C=4) and the restored image using the adaptive noise smoothing filter (5x5)

The degraded image r(k,l) has pixel values between 0 and 255. For the transformed image to have the same range, $\alpha = \sqrt{255}$ is chosen. So the transformation

$$r'(k,l) = \sqrt{255 r(k,l)}, \tag{4.17}$$

makes the film-grain noise signal-independent, additive and approximately Gaussian. The inverse transformation is given by [4, 11]

$$r(k,l) = \frac{[r'(k,l)]^2}{255}. \tag{4.18}$$

Before the homomorphic transformation is applied, the standard deviation of the noise term in the FGN model is $\sigma(k,l) = C\sqrt{f(k,l)}$. After the transformation the standard deviation of the transformed signal-independent noise is constant and is given by [11]

$$\sigma' = \frac{C\sqrt{255}}{2}. \tag{4.19}$$

Now with the signal-dependent noise transformed into signal-independent noise, we can apply statistical processing designed for signal-independent noise to the transformed image. We have used the James-Stein estimator given by Equation (4.8) where the variance of the noise is calculated using Equation (4.19).

4.7 Noise Suppression Results Using the Combined HT and Local-Statistics Processing

We have used the combination of the HT and the James-Stein (J-S) estimator on the noisy Lena image with C=4. A window size of 5x5 was used for the James-Stein estimator. The restored image is shown in Figure 4.8(a). We can see that the restored image has small black patches or spots. This "salt and pepper" kind of noise can be removed using a median filter. We have used a 3x3 median filter on the estimated image to remove the artifacts. A median filter generally smoothes the edges, however in this case as the size of the filter is 3x3, the edges are not smoothed. In the case of less amount of noise, this problem with artifacts is not encountered. For the noisy Lena with C=2, the transformation and the estimation gave a restored image without any artifacts or black

(a) Restored image using HT and J-S estimator, MSE=328

(b) Restored image after using median filter on the image in (a), MSE=165

Figure 4.8 Restored images from the noisy Lena image with FGN (C=4)

patches, as seen from Figure 4.9. Thus a median filter was not needed. For the case of noisy Lena with C=2 the MSE is reduced from 503 to 92. Restored images for the noisy Airplane with C=4 are also shown in Figure 4.10.

4.8 Comparison of the Two Approaches to Film-Grain Noise Suppression

Tables 4.3, 4.4 and 4.5 show the restoration results for the two approaches. The adaptive filter and the J-S estimator results were obtained using 5x5 windows and the median filter was implemented using a 3x3 window. From the results we see that in the case of strong noise (C=4), the MSE obtained using the transformation method is significantly greater than the MSE obtained using the adaptive filter. When the transformation method is combined with the median filter, however, errors are reduced. In the lower noise case, we did not need to use the median filter, as the black spots were not there, and the difference of the MSEs obtained from the two schemes was not large. Still the adaptive filter result is slightly better. Thus in general, we conclude that the adaptive noise smoothing filter gave a better restoration for images degraded by film-grain noise. For the same level of FGN (C=4), the adaptive noise smoothing filter reduced the MSEs of the noisy Lena and Airplane images by factors of 9.4 and 10.15, respectively. The transformation method reduced the MSEs of the noisy Lena and Airplane images by factors of 5.7 and 6.5, respectively. So both methods gave almost the same amount of noise reduction for the two different images.

Table 4.3 Noise suppression results for Lena with film-grain noise (C=4)

Image type	MSE	NMSE	LOGMSE	RMSSNR (dB)
Noisy image	1869	0.1015	0.016	9.94
Adaptive	198	0.0107	8.168E-4	19.69
HT & J-S	328	0.0178	0.0025	17.50
HT, J-S & Median	165	0.009	7.5518E-4	20.48

Figure 4.9 Image restored from the noisy Lena image with FGN (C=2) using HT and J-S estimator, MSE=92

(a) Restored image using HT and J-S estimator, MSE=343

(b) Restored image after using median filter on the image in (a), MSE=193

Figure 4.10 Restored images from noisy Airplane image with FGN (C=4)

Table 4.4 Noise suppression results for Lena with film-grain noise (C=2)

Image type	MSE	NMSE	LOGMSE	RMSSNR (dB)
Noisy image	503	0.0273	0.0019	15.64
Adaptive	86	0.0047	3.65E-4	23.32
HT & J-S	92	0.005	4.23E-4	23.01

Table 4.5 Noise suppression results for Airplane with film-grain noise (C=4)

Image type	MSE	NMSE	LOGMSE	RMSSNR (dB)
Noisy image	2243	0.0737	0.007	11.33
Adaptive	221	0.0073	4.49E-4	21.39
HT & J-S	343	0.0113	0.001	19.48
HT, J-S & Median	193	0.0063	3.67E-4	21.98

4.9 Compression Results

After noise suppression in the degraded images, we next studied the performance of the compression scheme on the restored images. We used JPEG on different restored images and compared the results with those obtained from compressing the noisy images.

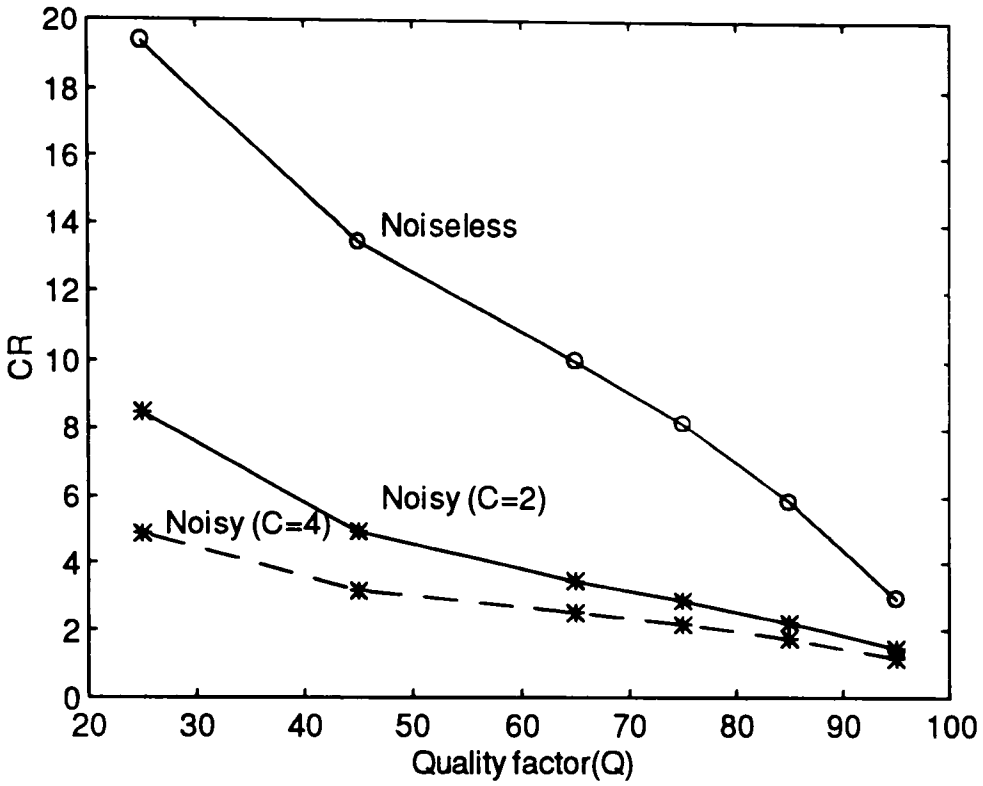
4.9.1 Compression in Signal-Independent Noise

The Lena image with SIN noise of variance 1000 yields a compression ratio of 2.866 at a quality factor of 65. After noise suppression, we obtained compression ratios of 5.52, 7.37 and 8.16, while using JPEG on the images restored using the J-S estimator with 3x3, 5x5 and 7x7 windows, respectively. Now good image quality is also desirable along with high compression ratios. We need to be careful in deciding which restored image is best suited for the compression. The image restored with a 3x3 window yields the lowest compression ratio as it has the least amount of noise suppression. So it is not the best of the results. On the other hand, the restored image with a 7x7 window yields

the highest CR as it has the highest amount of noise suppression. Still this is not the optimum result, because of the loss of significant edge information during restoration. The restored image with 5x5 window has significant noise suppression while preserving the edges and yields a high CR. Thus this image has the optimum noise reduction for the purpose of increasing the compression ratio.

4.9.2 Compression in Film-Grain Noise

In this section, the results obtained using JPEG compression on the images degraded by film-grain noise and the restored images are presented. Figure 4.11(a) shows the compression ratio (CR) versus quality factor (Q) plot for the noiseless Lena and Lena with two different levels of FGN. For a quality factor of 65, the CRs obtained for Lena without noise, with high noise (C=4) and low noise (C=2) are 10, 2.5 and 3.46, respectively. The CR is lower as the noise level is higher for the same quality factor, which is expected. For the airplane image CRs obtained are 10.2 and 2.38 without and with FGN (C=4), respectively, for a Q=65. We see that for same level of FGN, the reduction in compression was by factors of 4 and 4.29 for the Lena and Airplane images, respectively, for Q=65. Pre-compression noise suppression in the images increases the achievable CR compared to the direct compression of the noisy images. Figure 4.11(b) shows the plot of the CR versus quality factor for the noiseless, noisy and restored images. The noisy image used here is the noisy Lena with C=2 and the restored image is the one obtained using the adaptive noise smoothing filter shown in Figure 4.5. The plot shows that the CR is increased from 3.46 to 8 for a Q of 65. For a Q of 25, CR is increased from 8.46 to 17.39. The Lena image restored using the transformation method yielded CRs of 7.8 and 17.095 for Q-factors of 65 and 25, respectively. So compression ratios obtained from the two types of restored images are almost equal. For high level of FGN (C=4), the CRs obtained by compressing the Lena images restored using the adaptive filter and the transformation method are 6.578 and 5.077, respectively, for a Q-factor of 65. Compressing the restored Lena image (C=4) after HT, J-S and median filtering we obtained a CR of 8.45. So the images restored from the noisy images with a high level of



(a) CR versus Q for noiseless Lena and Lena with FGN

(b) CR versus Q plot for noiseless, noisy (FGN, C=2) and Restored Lena images

Figure 4.11 CR versus Q plot for different Lena images

FGN using the combination of the transformation method and median filtering yielded the highest compression ratios. This is due to fact that combination of the HT, J-S and median filtering smoothed the noise effectively. For the Airplane image, the CR increased to 6.54 from 2.38 after noise suppression using the adaptive filter. So the improvement of CRs for the two images are almost the same.

Figure 4.12 shows the bits per pixel (bpp) and the (log scale) MSE plot for the noiseless, noisy (C=2) and the restored images. Table 4.6 shows the MSE and PSNR for different images at different quality factors. We see that at a quality factor of 95 the PSNR and MSE are almost same for the images, because at such a high quality factor the quantization effects are not prominent. At the lower quality factors, the PSNRs for the noisy images are very low. At the low values of Q, the quantization of DCT coefficients is significant. In the case of the noiseless image, the loss of high frequency information is not significant, as it does not contain significant high frequency information. The noisy image has a high frequency contributions from the noise. Then use of JPEG cuts off some of the high frequency noise information during compression, because the quantization steps for the higher frequency DCT coefficients are coarser compared to those for the lower frequency DCT coefficients. When the PSNR is calculated between the input and the output images to a compression algorithm, it gives a measure of fidelity or how close the output image is to the original input image. The compression algorithm is effectively eliminating some of the noise information that was in the input uncompressed noisy image. So this noise reduction is the main reason for a greater difference existing between the uncompressed noisy image and the compressed image compared to the noiseless case. This is the main cause for low PSNR in the case of noisy image compression, not the loss of significant information. As long as the original noiseless image information, which is contained mainly in the low frequency regions, is preserved, the performance of the compression scheme can be considered satisfactory.

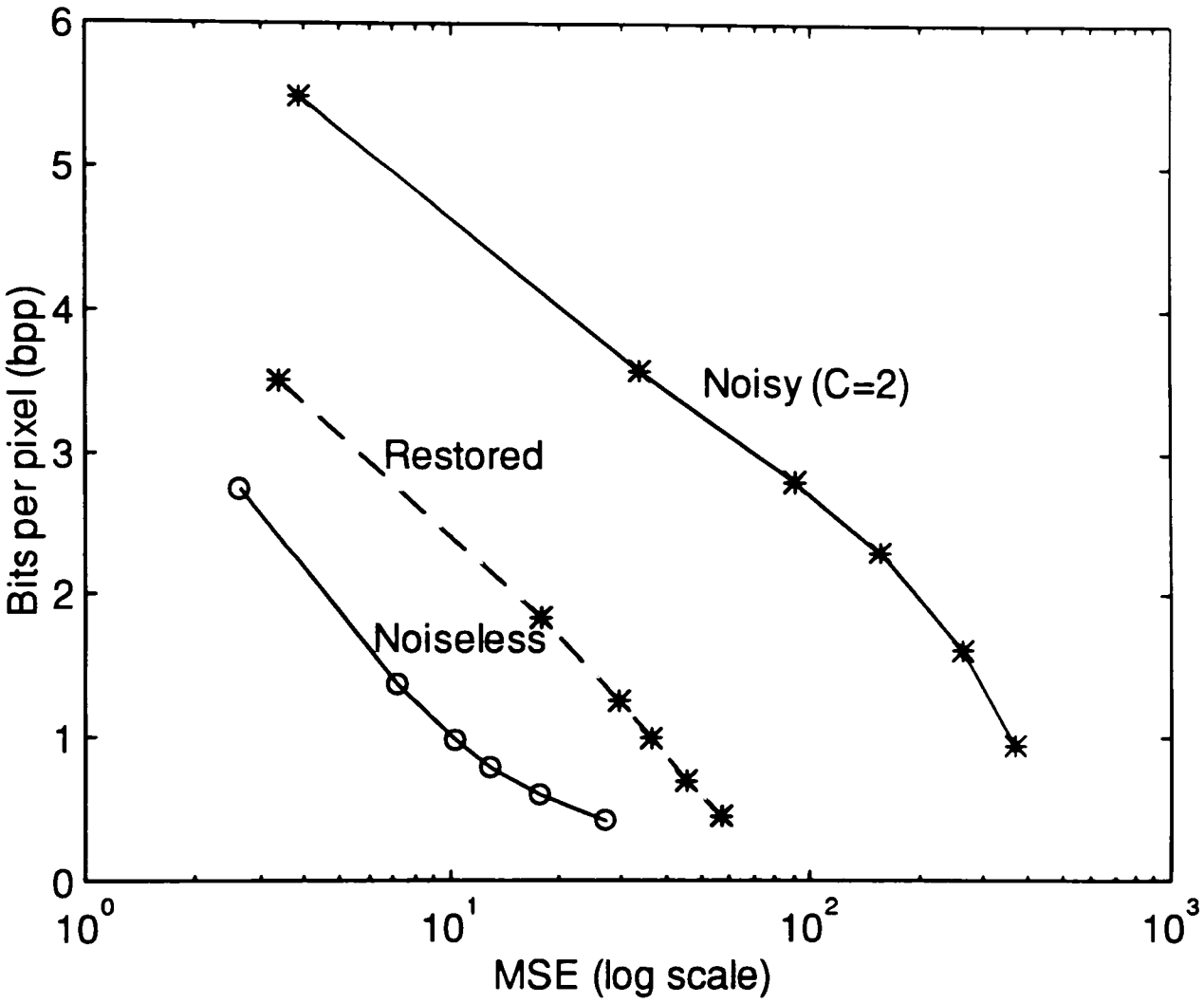
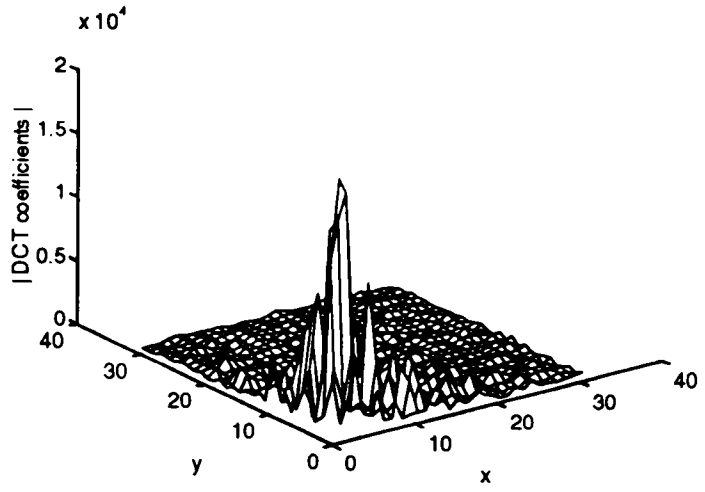


Figure 4.12 Plot of bpp versus MSE for different Lena images

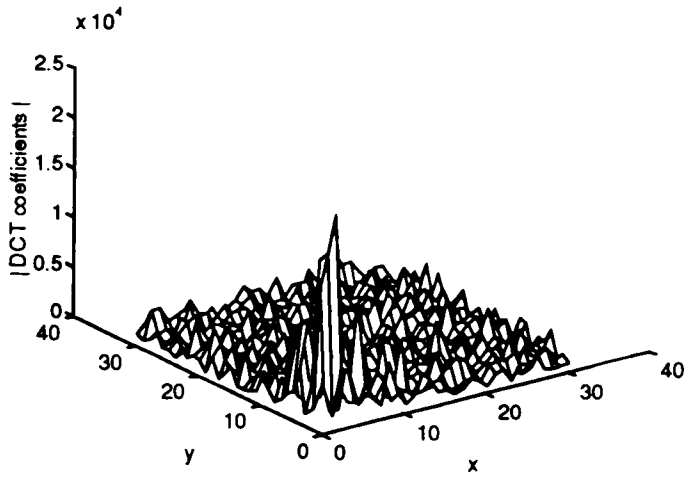
Table 4.6 Compression results for different quality factors for noisy (FGN, C=2) and the restored Lena image

Quality	Noisy image			Restored image		
factor	CR	MSE	PSNR (dB)	CR	MSE	PSNR (dB)
95	1.458	4	42.31	2.287	3	42.82
85	2.237	34	32.86	4.335	18	35.61
75	2.856	91	28.56	6.31	29	33.45
65	3.467	156	26.21	8.032	36	32.50
45	4.935	263	23.93	11.369	45	31.59
25	8.462	369	22.46	17.394	57	30.57

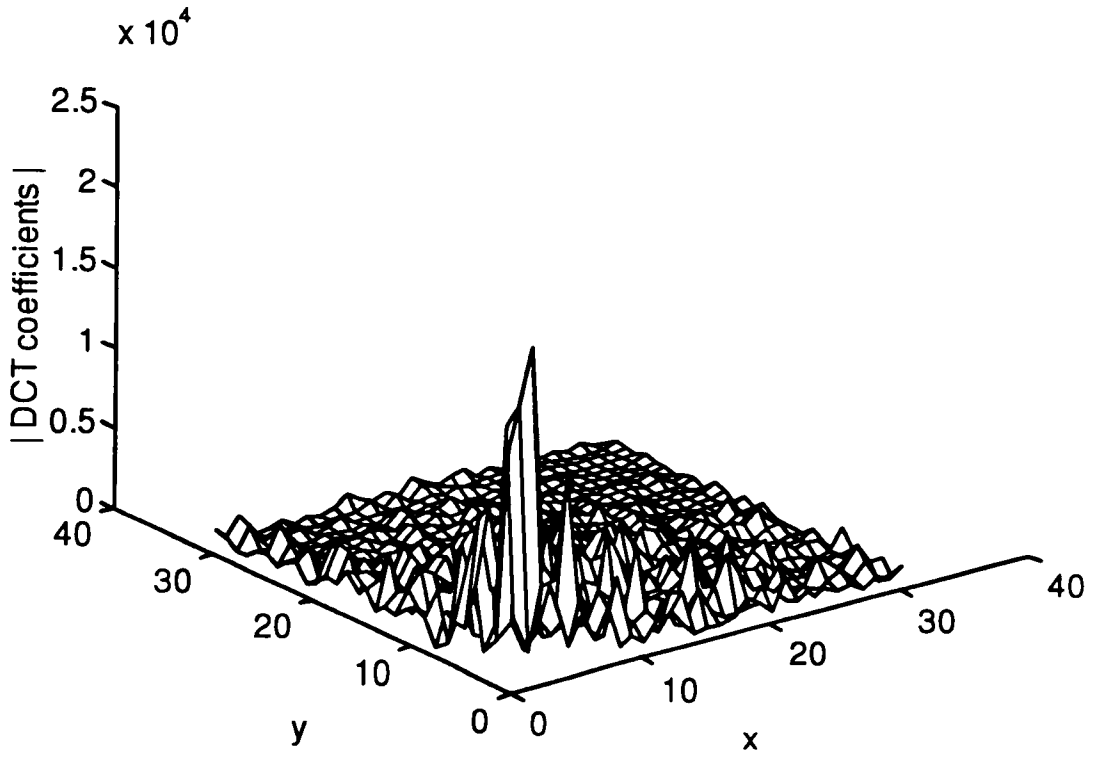
It is, however, difficult to separate the error due to the loss of the original information and the loss due to the suppression of noise by the compression algorithm by merely looking at the PSNR. For a clearer idea, we considered the forward DCT plots. We took 32x32 pixel blocks from the noiseless, uncompressed noisy and compressed noisy Lena images. The DCT coefficients are plotted in Figure 4.13. We see that the noiseless image information is mainly contained in the low frequency regions while in the compressed image only the high frequency coefficients are attenuated. To display the attenuation of high frequency coefficients clearly, the DCT of the compressed image is shown for a low quality factor (Q=25). The visual quality of the images can also be taken into consideration. As mentioned earlier there were blocking effects in the noiseless Lena image compressed at a quality factor 15 which yielded a PSNR of 32 dB. For the noisy image there is no such blocking effects in the compressed image even at a PSNR of 26 dB obtained at a Q of 65. It may thus be concluded that the low values of PSNR are mainly due to some of the noise suppression by the compression algorithm itself and not due to the loss of significant information. Figures 4.14 and 4.15 show the reconstructed images after compressing the noisy and restored images.



(a) Noiseless Lena image



(b) Uncompressed Lena with FGN



(c) Compressed Lena with FGN

Figure 4.13 DCT coefficients of 32x32 blocks of different Lena images

Figure 4.14 Lena with FGN (C=2) after compression (Q=65)

Figure 4.15 Compressed Lena image (Q=65) after restoration from FGN (C=2)

For the restored images, we see that PSNRs are higher, but still do not match the values of the noiseless case. This is due to the fact that the film-grain noise is not totally eliminated. So there is still some high frequency noise and the compression scheme reduces some of that.

In conclusion we can say that the reduction of the compression factor is the main disadvantage introduced by the presence of film-grain noise in an image. The effect of noise in increasing the errors introduced by the compression algorithm itself is rather difficult to determine for the JPEG algorithm, as discussed earlier. Thus in the presence of film-grain noise, noise suppression before compression appears to be the preferred approach.

CHAPTER V

NOISE SUPPRESSION AND COMPRESSION IN SPECKLE

Speckle noise is observed in images generated with highly coherent illumination. The presence of speckle in an image reduces the resolution of the image and the detectability of the target. Speckle noise has been shown to be signal-dependent in nature [5]. The SNR measured pointwise in a single frame speckle image is unity. Speckle is the main source of degradation in synthetic aperture radar (SAR) imagery and in ultrasonic imaging for medical applications. Thus when compressing these images, speckle effects should be taken into consideration. Many remote users are provided with a large amount of data through SAR images. Efficient compression of SAR images would significantly decrease their transmission cost. In the case of medical imaging, archiving is a present day problem, so compressing those images is also necessary.

A lot of research on the reduction of speckle noise in medical images is underway, so that the diagnosis of diseases can be facilitated. Speckle reduction is also necessary in SAR images, because the presence of speckle noise results in uncertainty in the interpretation of the image scenes. Like film-grain noise, here too, we have the choice of speckle reduction before or after compressing the images. In our work, we have investigated using noise suppression prior to compression to achieve higher compression ratios.

There are various methods of speckle reduction by modifying the imaging system. For example, partially coherent illumination is used in the imaging system instead of coherent illumination. Another method is to observe the speckle through a finite aperture and then move the aperture while observing the time-averaged image. These methods do not consider the image statistics. For noise reduction, we have considered filters that are based on the image statistics. Like film-grain noise suppression, we have used two approaches for speckle reduction. First, we have used an adaptive noise smoothing filter, one that is designed taking the signal dependence of the speckle into account. Secondly, we have used a homomorphic transformation to transform the speckle noise into signal-

independent additive noise. Then we have used the James-Stein estimator on the transformed image, followed by the appropriate inverse transformation. We have then investigated the performance of the compression scheme on both the images degraded by speckle and on the restored images.

5.1 Single Frame and Multiple Frames Speckle Images

We have presented the multiplicative model of speckle noise in section 2.2.3, which is

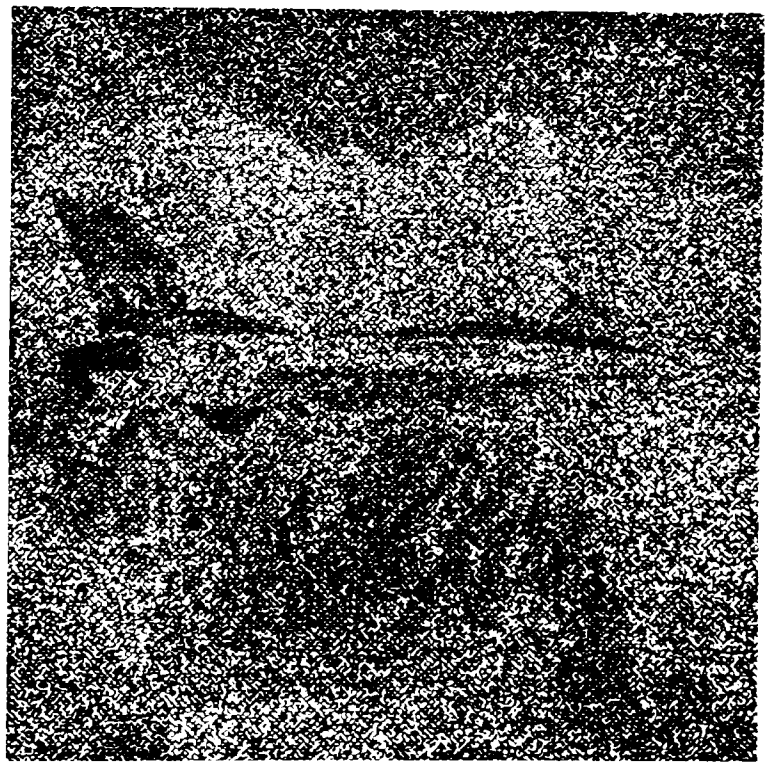
$$r(k,l) = f(k,l)n(k,l),$$

where $f(k,l)$ is the original noiseless image, $n(k,l)$ is the noise term with a negative exponential distribution, and $r(k,l)$ is the degraded image. Using this model single frame speckle images were simulated. Figure 5.1 shows the simulated single frame speckle version of both Lena and the Airplane images. The MSEs of the noisy images w.r.t. the original noiseless images are shown in the figure captions. As speckle is a strong source of noise, the MSEs are very large.

With the availability of several independent images of the same scene, a multiple frame image can be obtained by averaging the independent images. As mentioned in section 2.2.3, the speckle statistics for the multiple frame images obey a Gamma distribution, where the standard deviation decreases with the square root of the number of independent frames (looks) [14]. The SNR also increases by a factor of the square root of the number of the frames averaged. This multiple frames averaging reduces the speckle and is the most common method for speckle reduction [5]. The Lena and Airplane images with speckle and four frames averaging are shown in Figure 5.2. Note the significant reduction in MSEs when compared to the results of Figure 5.1.

We have used the noise suppression schemes on both the single frame and the four frames-averaged speckle images. For speckle reduction we took the same approach as with film-grain noise. First we have used the adaptive noise smoothing filter which takes the signal-dependence of the speckle into account. Secondly, we used the homomorphic

(a) Lena image, MSE=7303



(b) Airplane image, MSE=9715

Figure 5.1 Lena and Airplane images with speckle noise (single frame)

(a) Lena image, MSE=2424

(b) Airplane image, MSE=3909

Figure 5.2 Lena and Airplane images with speckle noise (four frames-averaged)

transformation to make the noise signal-independent and additive. Finally the James-Stein estimator was used for noise reduction.

5.2 Adaptive Noise Smoothing Filter for Speckle

The general form of the adaptive noise smoothing filter can be modified for multiplicative noise [9]. As our speckle model is a multiplicative noise model, we used the adaptive noise smoothing filter for a multiplicative model for the purpose of speckle reduction. According to our noise model, the adaptive noise smoothing filter for multiplicative noise with unit mean and unit variance is the adaptive noise smoothing filter for the single frame speckle noise. For M frames-averaged speckle images, the adaptive noise smoothing filter is the adaptive filter for multiplicative noise with unit mean and a variance of $\frac{1}{M}$ [14]. As the noise $n(k,l)$ is unit mean (section 2.2.2), we can say that

$$E\left[r(k,l)\right] = m_r(k,l) = E\left[f(k,l)\right] = m_f(k,l),$$

where $r(k,l)$ and $f(k,l)$ are the speckled and the original images, respectively. For this unit mean multiplicative noise, the adaptive noise smoothing filter becomes [9, 14]

$$\hat{f}(k,l) = m_r(k,l) + \frac{V_f(k,l)}{V_f(k,l) + \sigma_n^2\left[m_r^2(k,l) + V_f(k,l)\right]}\left[r(k,l) - m_r(k,l)\right],$$

where $V_f(k,l)$ and $m_r(k,l)$ are the local variance of $f(k,l)$ and the local mean of $r(k,l)$ calculated over all the pixels inside a (2M+1)x(2N+1) window. σ_n^2 is the noise variance. $V_f(k,l)$ is given by [5, 9]

$$V_f(k,l) = \frac{V_r(k,l) - \sigma_n^2 m_r^2(k,l)}{1+\sigma_n^2},$$

where $V_r(k,l)$ is the local variance of $r(k,l)$. Now the estimated value of each pixel is given by

$$\hat{f}(k,l) = m_r(k,l) + W_S\left[r(k,l) - m_r(k,l)\right], \tag{5.1}$$

where

$$W_S = Max\left[\frac{1-\frac{m_r^2(k,l)\sigma_n^2}{V_r(k,l)}}{1+\sigma_n^2},\ 0\right]. \tag{5.2}$$

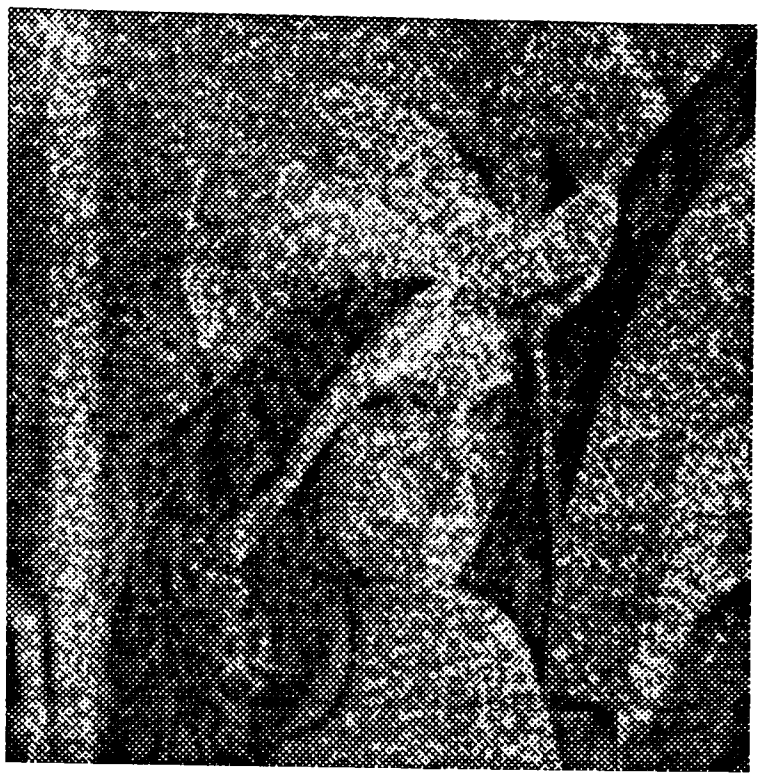
Here $m_r(k,l)$ and $V_r(k,l)$ are given by Equations (4.5) and (4.6). For an M-frame speckle image the noise variance is given by

$$\sigma_n^2 = \frac{1}{M}. \tag{5.3}$$

5.3 Speckle Reduction Results Using the Adaptive Noise Smoothing Filter

We have used the adaptive noise smoothing filter with a 5x5 window on both the single frame and the four frames-averaged speckle images. In the case of the Lena image, averaging four frames reduced the MSE from 7303 to 2424, that is by a factor of 3. For the Airplane image the MSE was reduced by a factor of 2.5. On the other hand, using the adaptive noise smoothing filter on the single frame images the MSE came down to 1191 and 2440 for the Lena and Airplane images, respectively. Using the adaptive filter on the four frames-averaged images the MSEs obtained were 960 and 2170 for the Lena and the Airplane, respectively. So the combination of the frame averaging and the adaptive noise smoothing filter reduced the MSE by factors of 7.6 and 4.48 for the Lena and the Airplane images, respectively. The amount of noise suppression was thus not the same for the two images.

We can see that the speckle reduction achieved by frame averaging brings out the details in the image which are not perceivable in the single frame speckle images. However, the apparent graininess of the image is still present. The images restored (Figure 5.3) using the adaptive noise smoothing filter on the single frame speckle images do not have same the graininess, though the edges are blurred. These restored images have a very poor visual quality. Compared to them, the restored images from the four frames-averaged speckle images (Figure 5.4) have more visible details.



(a) Restored Lena image, MSE=1191

(b) Restored Airplane image, MSE=2440

Figure 5.3 Images restored from single frame speckle images using adaptive noise smoothing filter (5x5)

(a) Restored Lena image, MSE=960

(b) Restored Airplane image, MSE=2170

Figure 5.4 Images restored from four frames-averaged speckle images using adaptive noise smoothing filter (5x5)

The restored images have a very low contrast compared to the noiseless images and the blurring is significantly noticeable. In the case of film-grain noise, the restored images had less blurring and higher contrast. This is due the fact that the speckle noise is a stronger source of degradation compared to the film-grain noise and the SNRs are lower in the images degraded by speckle. Tables 5.1 and 5.2 show different restoration results for speckle reduction.

Table 5.1 Noise suppression results for the Lena with single and four frames-averaged speckle noise using the (5x5) adaptive noise smoothing filter

Image type	MSE	NMSE	LOGMSE	RMSSNR(dB)
Noisy, single frame	7303	0.397	0.0705	4.02
Noisy, four frames	2424	0.132	0.0113	8.81
Restored, single frame	1191	0.065	0.0033	11.89
Restored, four frames	960	0.052	0.0022	12.83

Table 5.2 Noise suppression results for the Airplane with single and four frames-averaged speckle noise using adaptive noise smoothing filter (5x5)

Image type	MSE	NMSE	LOGMSE	RMSSNR(dB)
Noisy, single frame	9715	0.319	0.0621	4.96
Noisy, four frames	3909	0.128	0.0103	8.92
Restored, single frame	2440	0.08	0.0041	10.96
Restored, four frames	2170	0.071	0.0033	11.47

5.4 Combined Homomorphic and Local-Statistics Processing for the Speckle

The homomorphic transformation required to make the speckle noise signal-independent, additive and Gaussian in nature is a logarithmic transformation given by [11]

$$r'(k,l) = \beta \ln\left[r(k,l)+1\right],$$

where β is a scalar constant. The addition of one to r(k,l) is done to avoid taking the logarithm of zero. As r(k,l) is between 0 and 255, $\beta = \frac{255}{\ln(256)}$ is used so that the transformed image has the same range of values. So the transformed image is

$$r'(k,l) = \frac{255}{\ln(256)} \ln[r(k,l)+1]. \tag{5.4}$$

The inverse transform is then given by

$$r(k,l) = \exp\left[\frac{r'(k,l)\ln(256)}{255}\right] - 1. \tag{5.5}$$

After the transformation given by Equation (5.4), the noise becomes signal-independent and additive. On the transformed image we have used the James-Stein (J-S) estimator, which is the adaptive noise smoothing filter for signal-independent additive noise. The form of the James-Stein estimator given in Equation (4.8) is for additive noise with zero mean. Speckle noise, after transformation, has a non zero mean. So we need to subtract the non-zero mean m_{sp} from the estimated value. The modified form of the J-S estimator for the transformed image is thus given by

$$\hat{f}(k,l) = \left[m_{r'}(k,l) - m_{sp}\right] + W\left[r'(k,l) - m_{r'}(k,l)\right], \tag{5.6}$$

where $m_{r'}(k,l)$ is the local mean of $r'(k,l)$ given by Equation (4.5) and

$$W = Max\left[\left(1 - \frac{\sigma_{sp}^2}{V_{r'}(k,l)}\right), 0\right], \tag{5.7}$$

where $V_{r'}(k,l)$ is the local variance of $r'(k,l)$ given by Equation (4.6), m_{sp} and σ_{sp} are the mean and the variance of the transformed noise.

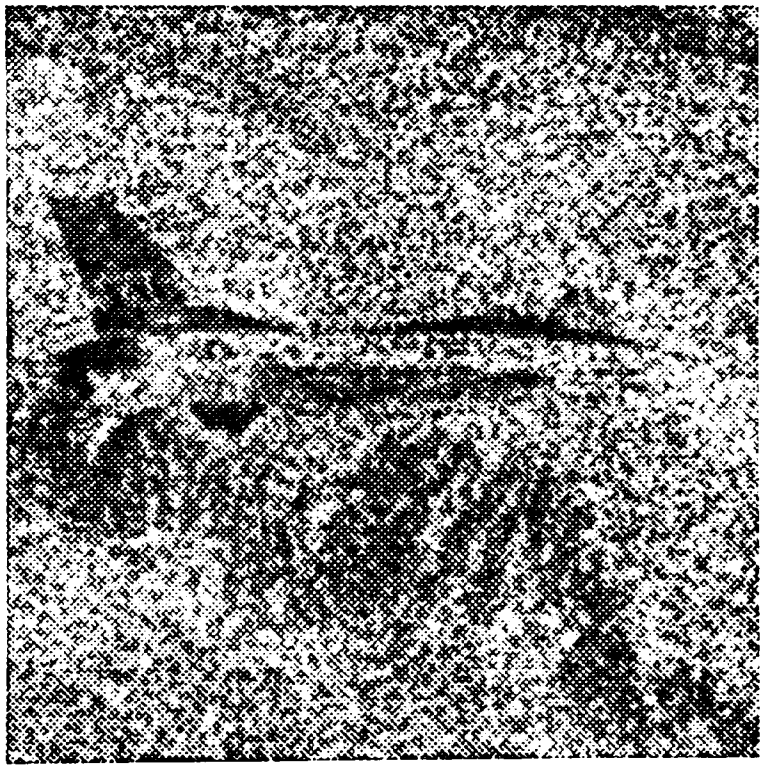
5.5 Speckle Reduction Results Using Combined Homomorphic and Local-Statistics Processing

We have used the HT and the James-Stein estimator with a 5x5 window on the single frame and the four frames-averaged speckle images. For the single frame case, we obtained an MSE of 1065 and 1547 for the Lena and the Airplane images, respectively.

The restored images are shown in Figure 5.5. The MSEs obtained by the SDN adaptive filter are 1191 and 2440 for the single frame Lena and the Airplane, respectively. The differences in the MSEs for the restored Lena images (single frame) is not large for the two schemes. For the Airplane image the MSE obtained from the transformation method (1547) is significantly smaller than that obtained (2440) from the SDN adaptive filter. For Lena, we see that though the MSE is less for the transformed method compared to the adaptive filter method for the single frame case, the LOGMSE is greater. For the Airplane image both the MSE and the LOGMSE are less for the transformed method than for the adaptive filter method. For the four frames-averaged speckle images, the MSEs obtained by the transformation method with a 5x5 window J-S estimator were 682 and 1484 for the Lena and the Airplane images, respectively. So, w.r.t. the single frame noisy images the combination of frame averaging and the transformation method reduced the MSEs by factors of 10.7 and 6.5 for the Lena and Airplane images, respectively. The restored images are shown in Figure 5.6. These images have higher contrasts compared to the images restored using the adaptive noise smoothing filter from the four frames-averaged speckle images.

Another interesting result is obtained using a 3x3 window for the James-Stein estimator. The edges of the images are well preserved with the 3x3 window. The noise was, however, not suppressed efficiently. To remove the noise, we did a second iteration of transformation and estimation on the image with a 3x3 window. From Tables 5.3 and 5.4, we see that it yielded low MSEs of 349 and 589, respectively, for Lena and the Airplane. So w.r.t to the single frame speckled images the combination of four frames-averaged method and two iterations of the transformation method reduced the MSEs by factors of 20.9 and 16.49 for Lena and the airplane images, respectively. The restored images obtained using 3x3 windows are shown in Figure 5.7.

(a) Restored Lena image, MSE=1065



(b) Restored Airplane image, MSE=1547

Figure 5.5 Images restored from single frame speckle images using HT and J-S (5x5) estimator

(a) Restored Lena image, MSE=682

(b) Restored Airplane image, MSE=1484

Figure 5.6 Images restored from four frames-averaged speckle images using HT and J-S estimator (5x5)

Table 5.3 Noise suppression results for Lena with single and four frames-averaged speckle noise using HT and J-S estimator

Image type	MSE	NMSE	LOGMSE	RMSSNR(dB)
Restored from single frame , 5x5 window	1065	0.58	0.0036	12.37
Restored from four frame , 5x5 window	682	0.037	0.0017	14.31
Restored, four frame , (3x3 , 1st iteration)	789	0.043	0.0023	13.68
Restored, four frame , (3x3 , 2nd iteration)	349	0.019	0.0011	17.22

Table 5.4 Noise suppression results for Airplane with single and four frames-averaged speckle noise using HT and J-S estimator

Image type	MSE	NMSE	LOGMSE	RMSSNR(dB)
Restored from single frame , 5x5 window	1547	0.051	0.0032	12.94
Restored from four frame , 5x5 window	1484	0.049	0.0022	13.12
Restored, four frame , (3x3 , 2nd iteration)	589	0.019	9.4E-4	17.14

5.6 Compression Results in Speckle

The SNRs of the single frame speckle images have been shown to be less than the SNRs of the multiple frames-averaged speckle images, i.e., the single frame speckle images are noisier than the multiple frames-averaged images. Thus the compression ratios (CR) are improved for the four frames-averaged speckle images compared to the single frame ones. For a quality factor (Q) of 65, the CR for the four frames-averaged images

(a) Restored Lena image, MSE=349

(b) Restored Airplane image, MSE=589

Figure 5.7 Images restored from four frames averaged speckle images using HT and J-S (3x3) estimator and two iterations

rose to 2.487 from 1.896 and to 2.606 from 1.969 for the Airplane and Lena images, respectively. Figure 5.8 shows the plots of CR versus Q for the noisy images. The restored image from the single frame Airplane image yielded a CR of 6.09 for a Q of 65. However, the images restored from the single frame case have a very low visual quality and were not good restoration results. Thus we have not considered those images to be suitable for the performance study of compression on the restored images.

We have mainly used the images restored from four frames-averaged speckled images for compression. The Lena image restored using 5x5 window for the adaptive filter and the transformation method yielded CRs of 8.8717 and 7.465, respectively, for a Q of 65. The restored Lena from the two iterations of the HT and the James-Stein estimator with 3x3 window size yielded a CR of 6.755 for a Q=65. For lower Q-factors the difference between the CRs was greater. For the adaptive filtered (5x5) Lena image the CR is 18.45 for a Q of 25 and the CR is 12.3 for the restored image using the transformation method with two iterations (3x3 window size) for the same quality factor. For the Airplane images restored using the adaptive filter and the transformation method both with 5x5 window yielded CRs of 8.6387 and 7.241, respectively, for a Q of 65. The plots of CR versus Q for the restored Lena images are shown in Figure 5.9. We see that the CRs obtained for the restored Lena image with adaptive filter (5x5) are close to the ones obtained for the restored image using HT and J-S estimator (3x3, two iterations) for high quality factors. For low values of Q, however, the restored images obtained from the adaptive filter yielded higher CRs. This is expected, as the restored images obtained using the adaptive filter is smoother and thus more edges are blurred. On the other hand, the restored images obtained from the transformation method had more edge information and the presence of some amount of unsuppressed noise made them appear grainier because of the smaller window size. Thus they had more high spatial frequency contents. Figure 5.10 shows the reconstructed images after applying compression to the restored images with a Q of 65.

Figure 5.11 shows the bits per pixel (bpp) versus MSE plot for the noisy and the restored Lena images. Due to the same reasons as was discussed in Chapter IV, the

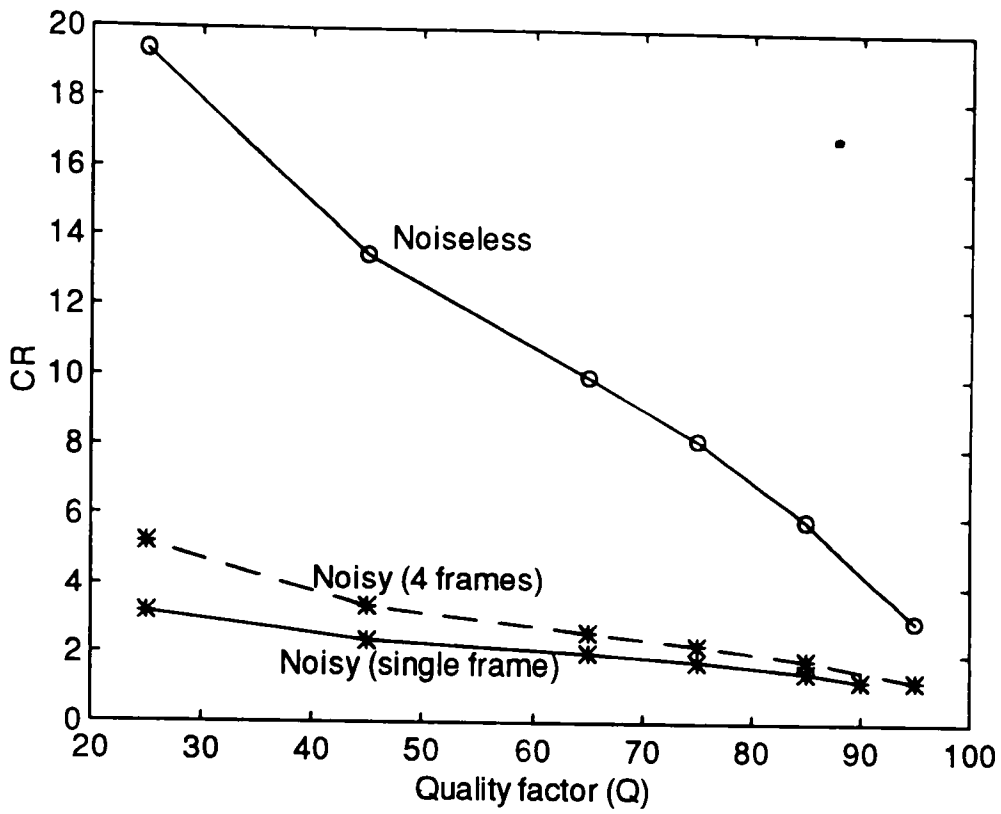


Figure 5.8 CR versus Q for the noisy Lena images with speckle

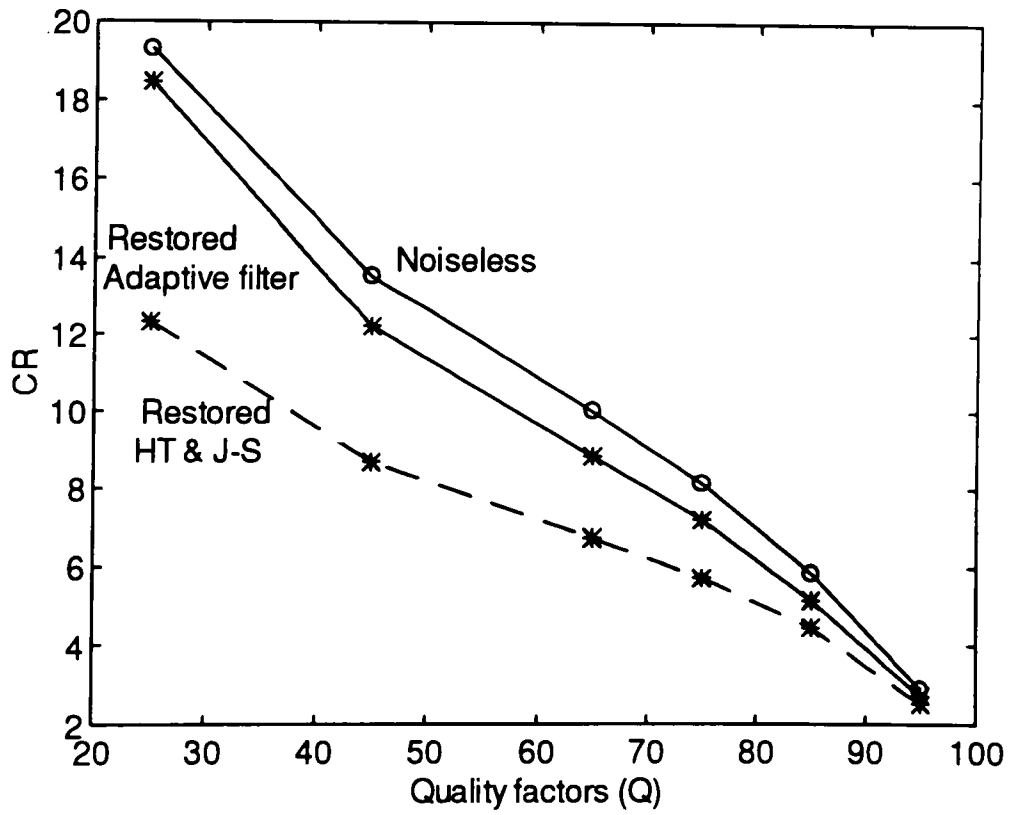


Figure 5.9 CR versus Q for the restored Lena images with speckle noise

(a) Compressed Lena with speckle noise (four frames-averaged)

(b) Compressed Lena after restoration using adaptive filter from four frames-averaged speckle image

Figure 5.10 Compressed Lena images at quality factor 65

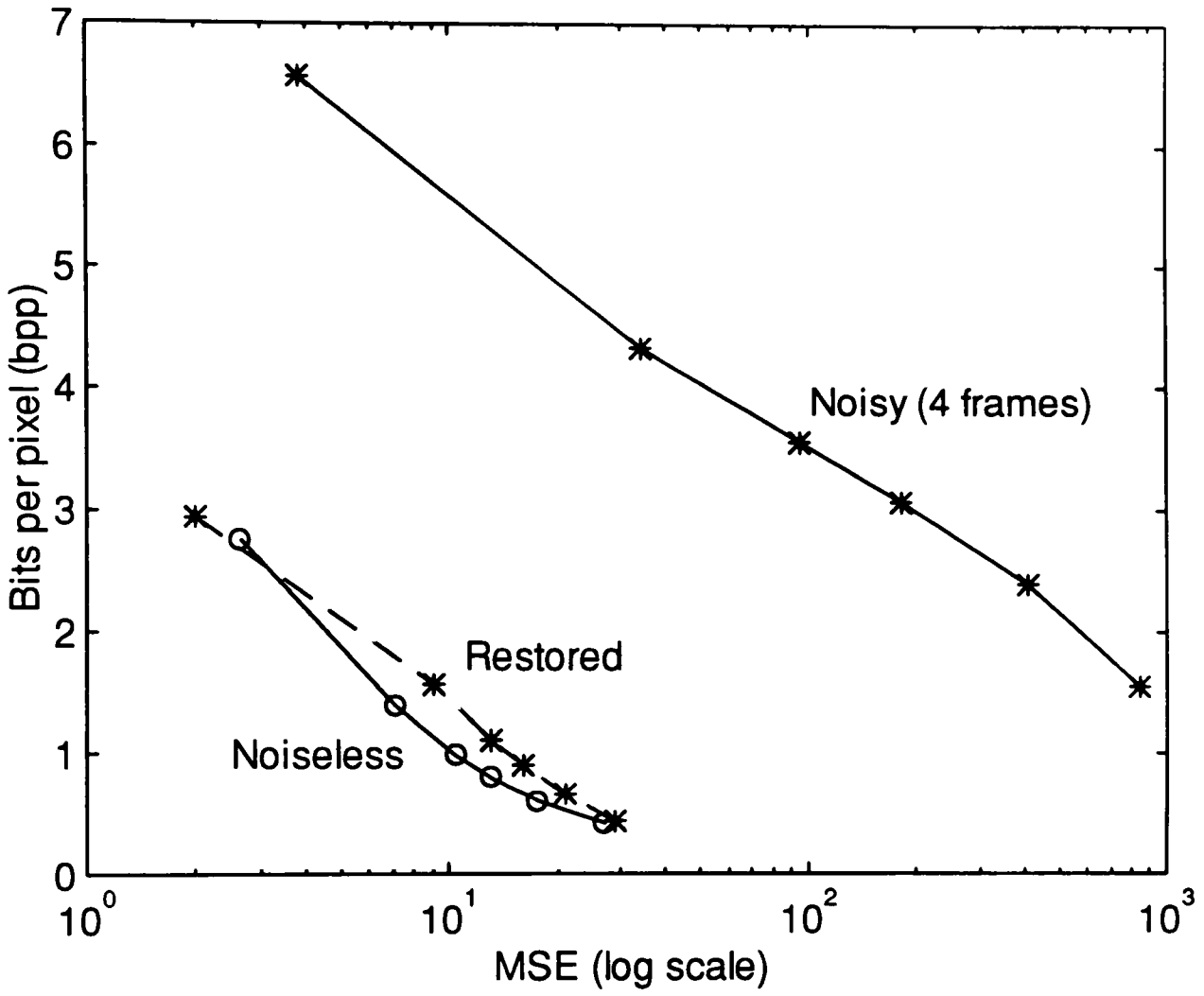


Figure 5.11 Plot of bpp versus MSE for the speckle image and the restored image using adaptive noise smoothing filter (5x5)

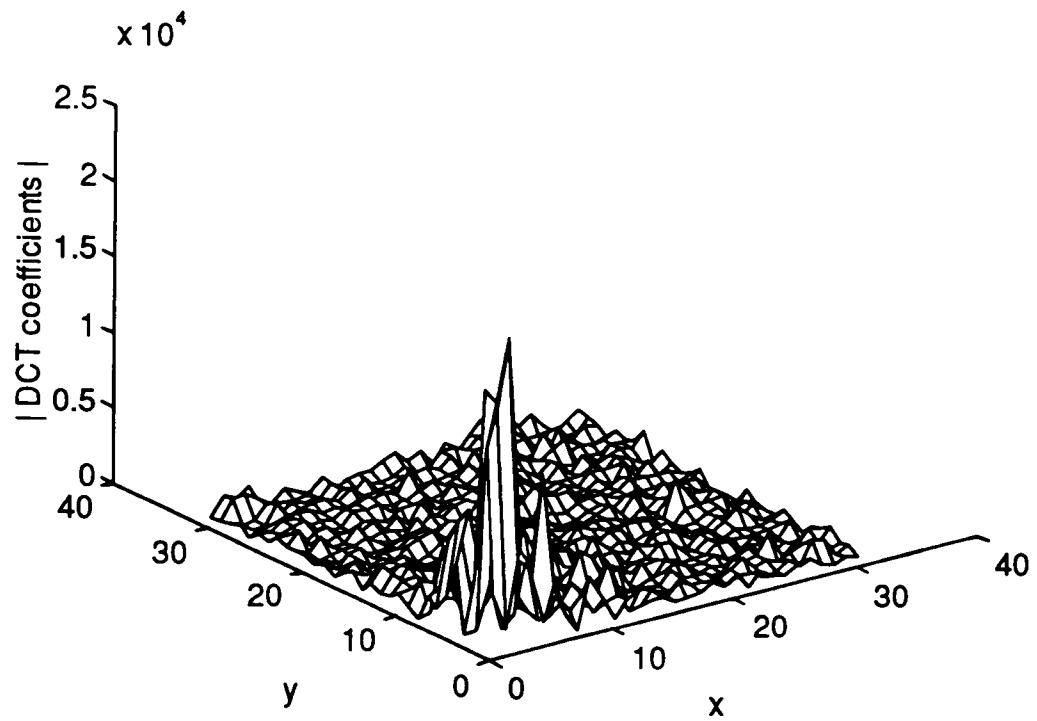
MSEs were very high for the noisy images after compression. For the restored images, the MSEs were lower and had almost the same range as for the noiseless image. From Tables 5.5 and 5.6, we see that PSNRs were also high for the restored images after compression. For Q's above 45 we obtained PSNRs of 35 dB and higher. This result deserves some comment in comparison to the results obtained in the case of film-grain noise. As we have seen in Chapter IV, the PSNRs obtained after compressing the images restored from the film-grain noise were not as high as in the noiseless case and were below 35 dB for quality factors below 85. The reason for this becomes clear if we look at the DCT plots of the restored images. Figure 5.12 shows the DCT plots for the image restored from Lena degraded by FGN (C=2) and Lena restored from the four frames-averaged speckle image, using the adaptive filter. We see that for the FGN case there is a significant amount of high frequency content that comes from the unsuppressed noise. In the case of the restored speckle image from the speckle noise the high frequency information content is reduced. Thus in the latter case the quantization of the DCT coefficients does not introduce appreciable high frequency information loss and the difference between the uncompressed restored image and the compressed image is small.

Table 5.5 Compression results for Lena with speckle noise

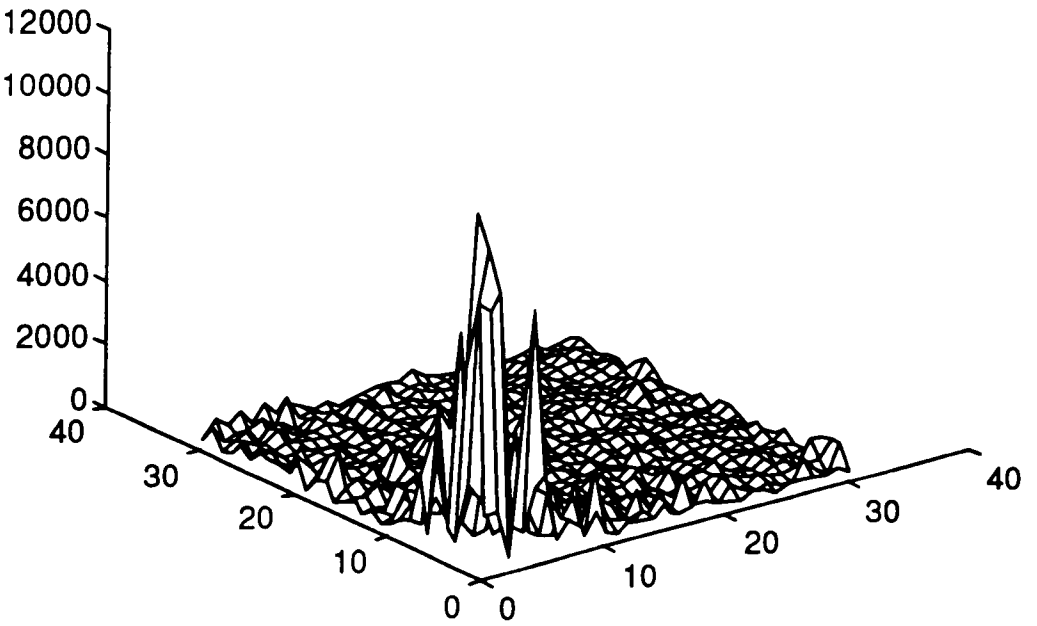
Quality	Noisy (single frame)			Noisy (four frame averaged)		
factor	CR	MSE	PSNR(dB)	CR	MSE	PSNR(dB)
95	—	—	—	1.219	4	42.29
85	1.44	30	33.29	1.843	34	32.84
75	1.745	84	28.91	2.257	94	28.39
65	1.969	158	26.13	2.606	180	25.58
45	2.373	380	22.33	3.353	410	22.01
25	3.186	1111	17.68	5.221	841	18.88

Table 5.6 Compression results for Lena restored from four frames-averaged speckle image

Quality	Adaptive filter (5x5 window)			HT & J-S (3x3) two iterations		
factor	CR	MSE	PSNR(dB)	CR	MSE	PSNR(dB)
95	2.724	2	44.55	2.527	2	45.86
85	5.161	9	38.74	4.464	6	40.44
75	7.210	13	37	5.75	9	38.37
65	8.872	16	36.13	6.755	13	37.07
45	12.194	21	34.98	8.688	20	35.20
25	18.453	29	33.5	12.3	34	32.77



(a) DCT of the image restored from the film-grain noise



(b) DCT of the image restored from the speckle noise

Figure 5.12 DCT (32x32 blocks) plots of different Lena images restored from noise

CHAPTER VI
CONCLUSIONS AND SUGGESTIONS FOR FUTURE WORK

We have studied the effects of signal-dependent noise on image compression. The pre-processing of the images (i.e., noise suppression prior to compression) has also been investigated. We have seen that the presence of noise reduces the allowable compression ratios (CR) significantly. The amount of compression reduction depends on how noisy the image is and the type of the noise present. With the JPEG compression algorithm, CRs obtained for the Lena image degraded by film-grain noise (FGN) were between 1.2 and 4.8 for quality factors between 95 and 25 when the noise level is high (C=4). For a lesser amount of film-grain noise (C=2) the CRs ranged from 1.5 to 8.5, whereas for the noiseless image the CRs were between 3 and 19 for the same quality factors. For the speckled images, the range of compression ratios obtained were from 1 to 3.2 and from 1.2 to 5.2 for the single frame and the four frames-averaged speckle images, respectively. Thus the reduction of the allowable compression is greater in the case of speckle than for film-grain noise when using the JPEG compression algorithm.

For the improvement of the allowable compression ratios, our approach was to pre-process the noisy images for noise suppression. For the images degraded by FGN, the adaptive noise smoothing filter gave better restoration results compared to the transformation method, which is the combination of a homomorphic transformation (HT) and local-statistics processing with the James-Stein (J-S) estimator. For images with a low level of FGN (C=2), both methods worked well, with the adaptive noise smoothing filter giving slightly better results. The compression ratios obtained for both types of restored images were almost equal. With a higher level of film-grain noise (C=4), the transformation method gave rise to artifacts or black patches, which were not present in the restored images obtained using the adaptive noise smoothing filter. The CRs achieved by applying compression to the restored images obtained using the transformation method were less than the CRs obtained for the adaptive noise smoothing filtered images. The

artifacts in the restored images were removed by using a median filter. The restored image and the CR obtained from it were then as good as the restored image obtained using the adaptive noise smoothing filter. The compression ratios achieved for the restored images were slightly less than the compression ratios achieved using the noiseless images; however, they were significantly higher than the compression ratios obtained from the noisy images.

In the case of the speckled images, the images restored from the single frame speckle images were not satisfactory. The restored images from the four frames-averaged speckle images were "good." In this case, the combination of the HT and the J-S estimator gave better restorations compared to the adaptive noise smoothing filter, when a 5x5 window was used for both schemes. The restored images obtained from the transformation method (HT and J-S) had more edge information and better contrast. Using two iterations of the HT and the J-S estimator with 3x3 window we obtained even a better restoration result. In the case of film-grain noise, the amount of noise reduction for different images was almost the same for the same level of noise for a particular noise suppression scheme. In the case of speckled images, the amount of noise reduction by frame averaging was different for different images.

For speckled images, the compression obtained from the restored images was significantly higher than that obtained by compressing the noisy images. The compression ratios obtained for the restored images using the adaptive noise smoothing filter (5x5) were higher than those obtained for the restored images using two iterations of the HT and the J-S estimator approach. The reason is that, the adaptive filter smoothed out the noise at the expense of removing edge information and blurring the image. On the other hand, the restored images obtained from the transformation method had more edge information and less blurring. There was, however, some unsuppressed noise present which led to lower values of compression ratios achieved. As a trade-off between the noise suppression and the preservation of original image information, the restored images obtained from the transformation method were superior. Thus we have considered it to be the restoration approach best suited prior to compression.

Hence the effects of image noise on compression depend on the type of noise present and the amount of noise. The effectiveness of pre-processing the images prior to compression is also dependent on these two factors. As the amounts of compression obtained with different images were not identical for the same quality factors, its performance is also image dependent.

We have used simulated noisy images to study the noise effects on image compression. The performance of the compression schemes on actual noisy images (such as SAR images) remains to be studied. The performance of image compression schemes other than JPEG in the presence of signal-dependent noise should also be investigated further. Mitra et al. used morphological filters for speckle reduction prior to compression and used multiresolution wavelet transform for compression [29]. The performance of the morphological predictor and the statistical estimators may be compared. The modification of the compression scheme itself for noise suppression, leading to higher compression ratios, may also be explored further.

REFERENCES

1. J. Vaaben and B. Niss, "Compressing images with JPEG," Information Display, vol. 7, no. 7/8, pp. 12-14, July/Aug. 1991.

2.. J. F. Walkup and R. C. Choens, " Image processing in signal-dependent noise," Opt. Eng., vol. 13, no. 3, pp 258-266, May/June 1974.

3. G.K. Froehlich, J. F. Walkup and R. B. Asher, "Optimal estimation in signal-dependent noise," J. Opt. Soc. Am., vol. 68, no. 12, pp. 1665-1672, Dec. 1978.

4. H. H. Arsenault, C. Gendron and M. Denis, "Transformation of film-grain noise into signal-independent additive Gaussian noise," J. Opt. Soc. Am., vol. 71, no. 1, pp. 91-94, Jan. 1981.

5. D.T. Kuan, A. A. Sawchuk, T. C. Strand and P. Chavel, "Adaptive restoration of images with speckle," IEEE Trans. on Accoust. Speech Signal Process., vol. ASSP-35, no. 3, pp. 373-382, March 1987.

6. T. Loupas, W. N. McDicken and P.L. Allan, "An adaptive weighted median filter for speckle suppression in medical ultrasonic images," IEEE Trans. on Circuits and Sys., vol. 36, no. 1, pp. 129-135, Jan. 1989.

7. W. B. Pennebaker and J. L. Mitchel, JPEG Still Image Data Compression Standard, Van Nostrand Reinhold, New York, 1993.

8. W. K. Pratt, Digital Image Processing, John Wiley, New York, 1978.

9. D. T. Kuan, A. A. Sawchuk, T. C. Strand and P. Chavel, "Adaptive noise smoothing filter for images with signal-dependent noise," IEEE Trans. Pattern Anal. Mach. Intell., vol. PAMI-7, no. 2, pp. 165-177, March 1985.

10. B. R. Hunt and T. M. Cannon, "Nonstationary assumptions for Gaussian models of images," IEEE Trans. Syst. Man. Cybern., vol. SMC-6, pp. 876-881, Dec. 1976.

11. H. H. Arsenault and M. Levesque, "Combined homomorphic and local-statistics processing for restoration of images degraded by signal-dependent noise," Applied Optics, vol. 23, no. 6, pp. 845-850, March 1984.

12. M. Tur, K. C. Chin and J. W. Goodman, "When is speckle noise multiplicative?," Applied Optics, vol. 21, no. 7, pp. 1157-1161, April 1982.

13. J. S. Lim and H. Nawab, " Techniques for speckle noise removal," Opt. Eng., vol. 20, no. 3, pp. 472-480, May/June 1981.

14. D. T. Kuan, Nonstationary Recursive Restoration of Images with Signal-Dependent Noise with Application to Speckle Reduction, USCIPI Report 1060, Dept. of Elect. Eng., University of Southern California, Los Angeles, California, August 1982.

15. C. Y. Chang, R. Kwok and J. C. Curlander, "Spatial compression of seasat SAR imagery," IEEE Trans. on Geoscience and Remote Sensing, vol. 26, no. 5, pp. 673-685, Sept. 1988.

16. S. B. Lo, B. Krasner, S. K. Mun, " Noise impact on error-free image compression," IEEE Trans. on Medical Imaging, vol. 9, no. 2, pp. 202-206, June 1990.

17. Mark Nelson, The Data Compression Book, M&T Books, New York, 1992.

18. C. A. Christopoulos and A. N. Skodras, "Adaptive DCT coding of images," Int. J. Electronics, vol. 75, no. 4, pp. 627-639, 1993.

19. N. M. Nasrabadi and R. A. King, "Image coding using vector quantization: A Review," IEEE Trans. on Commn., vol. 36, no. 8, pp. 957-971, Aug. 1988.

20. Y. Zhao and B. Yuan, "Image compression using fractals and discrete cosine transform," Elect. Letters, vol. 30, no. 6, pp. 474-475, March 1994.

21. D. M. Baylon and J. S. Lim, "Video compression and noise reduction using transform/subband coding and adaptive amplitude modulation," SMPTE Journal, pp. 404-406, June 1992.

22. M. Antonini, M. Barlaud, P. Mathieu and I. Daubechies, "Image coding using wavelet transform," IEEE Trans. on Image Processing, vol. 1, no. 2, pp.205-220, April 1992.

23. G K. Wallace, " The JPEG still picture compression standard," Communications of the ACM, vol. 34, no. 4, pp. 30-44, April 1991.

24. Independent JPEG Group's Documentation of JPEG software for release 5, Sept., 1994.

25. R. C. Gonzalez and R. E. Woods, Digital Image Processing, Addison-Wesley Pub. Co., New York, 1992.

26. H. H. Arsenault and M. Denis, "Integral expression for transforming signal-dependent noise into signal-independent noise," Optics Letters, vol. 6, no. 5, pp. 210-212, May 1981.

27. P. R. Prucnal, B. E. A. Saleh, "Transformation of image-signal-dependent noise into image-signal-independent noise," Optics Letters, vol. 6, no. 7, pp. 316-318, July 1981.

28. G. K. Froehlich, J. F. Walkup and T. F. Krile, "Estimation in signal-dependent film-grain noise," Applied Optics, vol. 20, no. 20, pp. 3619-3626, Oct. 1981.

29. S. Mitra, R. A. Muyshondt and S. Pemmaraju, "Hybrid high-fidelity image compression technique using multi-scale wavelets," SPIE Proc. on Wavelet Applications in Signal and Image processing III, vol. 2569, San Diego, California, July 12-14, 1995.

APPENDIX A

SAMPLE PROGRAM FOR IMAGE RESTORATION

```
%% This program is written in MATLAB
%% The program below generates a noisy image with film-grain noise (FGN), uses %%
%% James-Stein estimator for noise suppression after transforming the FGN into signal-
%% independent noise by homomorphic transformation (HT) %%%
%% For speckle image similar program works the with appropriate changes for HT, %%
%% inverse HT, transformed noise variance and mean and the noise model for noisy %%
%% image simulation %%%%
%% This program supports a square image (i.e. an NxN image) %%%
%%%  Generation of Film-Grain Noise %%%
%% kk is the FGN constant, wn is the image size, %%
%% Give the value of kk and wn %%%
wn=512;
kk=4;
%% Read the image file (here it is a binary file 512x512 raw data without any header, %%
%% each pixel value with 8 bits %%%
%% The original image is assigned to a 512x512 matrix lena %%
fid=fopen('lena.pic','rb')
lena=fread(fid,[wn,wn]);
fclose(fid);
%% nois is the 512x512 matrix of Gaussian random numbers with zero mean and unit
%% variance %%
%% flena is the image with FGN noise %%
%% randn is the random number generator %%
nois=randn(wn);
temp1=sqrt(lenap);
flena=lenap+kk*(temp1.*nois);
```

```
%% Threshold values above 255 to 255 and values less than zero to zero %%
for k=1:wn
        for l=1:wn
                If flena(k,l)>255
                flena(k,l)=255;
                elseif flena(k,l)<0
                flena(k,l)=0;
                else
                flena(k,l)=round(flena(k,l));
                end
        end
end
%%% Homomorphic transformation of the FGN %%
%% Give value of w which is the window size for the estimator %%
%% And w=2m+1 %%
 w=5;
 m=(w-1)/2;
 trans=sqrt(255*flena);
 %% r is the transformed image with zero padding %%
 r=zeros(wn+2*m);
 r(m+1:wn+m,m+1:wn+m)=trans;
 clear trans
 %% Apply James-Stein estimator on the transformed noisy image
 %% devf is the variance of the transformed image (in case of FGN)
 %% rav is the local average and v is the local variance of the transformed image calculated
 %% over wxw size window
 devf=kk*(sqrt(255))/2;
 devf=devf*devf;
 for i=m+1:wn+m
```

```
        for j=m+1:wn+m
        rav=sum(sum(r(i-m:i+m,j-m:j+m)));
        rav=rav/(w*w);
        v=r(i-m:i+m,j-m:j+m)-rav;
        v=v.*v;
        v=sum(sum(v));
        denom=w*w-1;
        v=v/denom;
%% Avoid division by zero
%% s is the matrix of the estimated pixels
                if v==0
                s(i-m,j-m)=r(i,j);
                else
                temp=1-devf/v;
                if temp>0
                q=temp;
                else
                q=0;
                end
                s(i-m,j-m)=rav+q*(r(i,j)-rav);
                end
        end
end
clear r
clear q temp rav i j v devn
%% Perform inverse transform to get the image back into original plane
%% recon is the restored image
recon=(s.*s)/255;
```

```
%% threshold image pixel values between zero and 255
for i=1:wn
        for j=1:wn
                if recon(i,j)>255
                recon=255;
                elseif recon(i,j)<0
                recon(i,j)=0;
                else
                recon(i,j)=round(recon(i,j));
                end
        end
end
clear s
%% Calculating different errors between the original image and the restored image
%% (i.e. MSE NMSE RMSSNR LOGMSE)
eror=sum(sum((lenap-recon).^2));
energy=sum(sum(lenap.^2));
mse=eror/(wn*wn);
nmse=eror/energy;
rmssnr=10*log10(1/nmse);
sq=(log10(1+lenap)-log10(1+recon)).^2;
totsq=sum(sum(sq));
logenr=(log10(1+lenap)).^2;
totenr=sum(sum(logenr));
logmse=totsq/totenr;
clear eror
clear energy
clear sq
clear totsq logenr totenr
```

APPENDIX B

SOFTWARE USED FOR COMPRESSION

There are two free software packages available for implementation of the JPEG image compression/decompression algorithm. One is the Independent JPEG Group's JPEG software and another one is the software written by the PVRG group at Stanford. As mentioned earlier we have used the fifth public release of the Independent JPEG Group's (IJG) software in our work, which implements a lossy JPEG. The Stanford group's JPEG software also supports a lossless JPEG. The official archive site for the IJG JPEG software is ftp.uu.net. The most recent version of the IJG software is available at this site in the directory graphics/jpeg. The JPEG software of the Stanford group is available from havefun.stanford.edu in the directory pub/jpeg.

The fifth release of IJG software, that we have used provides two programs for compression and decompression. The program cjpeg compresses the image file into JPEG format and the program djpeg decompresses a JPEG file back into a conventional image file format. One may also use the JPEG library in one's own programs. Different features of the IJG JPEG software that we used are presented below.

For using the cjpeg program to compress an image file the command used on UNIX-like systems is

cjpeg [switches] [imagefile] > jpegfile.

The program reads the input image file or the standard input if no input image file is specified. On most non-UNIX systems the command used is

cjpeg [switches] imagefile jpegfile.

So both the input and the output files need to be specified on the command line. The input image file to the cjpeg program must be in a file format that is supported by the IJG software. The fifth release of this software supports image file formats such as, GIF (Graphic Interchange Format), BMP, PPM (PBMPLUS color format), PGM (PBMPLUS gray-scale format), Targa and RLE(Utah Raster Toolkit format). The program cjpeg

recognizes the input file format automatically except some of the Targa-format file. JPEG files are in the defacto standard JFIF file format.

Some of the basic command switches for cjpeg are given below,

-quality N: This is used to define the quality factor desired for the image. It scales the quantization tables to adjust the image quality defined. Quality is 1 (worst) to 100 (best). The default value is 75.

-grayscale: creates monochrome JPEG file from color input. This switch should be used when a gray-scale GIF file is compressed, because cjpeg does not recognize that a GIF file only contains the shades of gray.

-targa: As cjpeg does not recognize some of the Targa file format, this switch is used to make cjpeg treat the file as Targa input.

-optimize: Performs optimization of entropy encoding parameters.

-dct int: Uses integer DCT (default).

-dct fast: Uses fast integer DCT (less accurate).

-dct float: Uses floating point DCT (most accurate, but slower).

-verbose: It enables debug printouts. More v's give more printouts.

-qtables file: Uses the quantization table given in the specified file.

To decompress a JPEG file the command used on a UNIX like system is

djpeg [switches] [jpegfile] > imagefile.

It reads from the input file specified or from the standard input if no file is specified.

On most non-UNIX systems the command is

djpeg [switches] jpegfile imagefile.

Here, both input and output files need to be specified on the command line. The user have to tell djpeg which file format to generate. Some basic command line switches for djpeg are,

-colors N: This reduces the number of colors in the output images to a maximum of N colors, so that it can then be displayed on a colormapped display or stored in a color mapped file format. As an example, for 8 bit display we need to reduce to 256 or fewer colors.

-grayscale: It forces gray-scale output even if the JPEG file is color. This is useful for viewing on monochrome displays.

-bmp: It selects BMP output file format. An 8 bit colormapped format is emitted if JPEG file is grayscale. Otherwise 24 bit full-color format is emitted.

-gif: Selects GIF format. As GIF supports no more than 256 colors, -colors 256 is assumed.

-pnm: Selects PPM/PGM format (default format). PGM is emitted if JPEG file is grayscale or -grayscale is specified, otherwise PPM is emitted.

-rle: Selects RLE output format.

-targa: Selects Targa output format.

-dct int: Uses integer DCT (default).

-dct fast: Uses fast integer DCT (less accurate).

-dct float: Uses floating point DCT (most accurate, but slower).

-verbose: It enables debug printouts. More v's give more printouts.

PERMISSION TO COPY

In presenting this thesis in partial fulfillment of the requirements for a master's degree at Texas Tech University or Texas Tech University Health Sciences Center, I agree that the Library and my major department shall make it freely available for research purposes. Permission to copy this thesis for scholarly purposes may be granted by the Director of the Library or my major professor. It is understood that any copying or publication of this thesis for financial gain shall not be allowed without my further written permission and that any user may be liable for copyright infringement.

Agree (Permission is granted.)

Rubeena Rahmaz — July 24, 1995

Student's Signature — Date

Disagree (Permission is not granted.)

_____ — _____

Student's Signature — Date