Deblocking Filter for Block-Based ICA Transform Coding

Developed By

Bushra Asghar

317-FAS/MSCS/F06

Supervised By

Mr. Asim Munir

Department of Computer Science

Faculty of Basic & Applied Sciences

International Islamic University, Islamabad

2008

بسم الله الرحمن الرحيم

In The Name of

ALLAH ALMIGHTY

The Most Merciful, The Most Beneficent

Department of Computer Science,

International Islamic University, Islamabad.

Dated: 2nd July, 2008

Final Approval

It is certified that we have read the thesis, titled "Deblocking Filter for Block-Based ICA Transform Coding" submitted by **Bushra Asghar** Reg. No. **317-FAS/MSCS/F06.** It is our judgment that this thesis is of sufficient standard to warrant its acceptance by the International Islamic University, Islamabad, for the Degree of **Master of Science in Computer Sciences.**

Committee:

1. External Examiner

Dr. Anwar Majeed Mirza

Professor,
Department of Computer Science,
National University of Computer and Emerging Sciences,
FAST,
Islamabad.

2. Internal Examiner

Miss Fareeha Anwar

Lecturer,
Department of Computer Science,
Faculty of Basic and Applied Sciences,
International Islamic University,
Islamabad.

3. Supervisor

Mr. Asim Munir

Assistant Professor,
Department of Computer Science,
Faculty of Basic and Applied Sciences,
International Islamic University,
Islamabad.

A dissertation submitted to the

Department of Computer Science,

Faculty of Basic and Applied Sciences,

International Islamic University, Islamabad, Pakistan

as a partial fulfillment of the requirements

for the award of the degree of

Master of Science in Computer Sciences

This project is lovingly dedicated

To

The Holiest Man Ever Born,

Holy Prophet Hazrat Muhammad (صلى الله عيه وسلم)

&

To

My Beloved Parents

Whose affection has always been the source of encouragement for me and whose prayers have always been a key to my success.

&

To

My Honorable Teachers

Who have been a beacon of knowledge and a constant source of inspiration,

for my whole life span.

Declaration

I, Bushra Asghar D/O Muhammad Asghar Ali, hereby declare and affirm that this software neither as a whole nor as a part thereof has been copied out from any source. It is further Pdeclared that I have developed this software and accompanied report entirely on the basis of my personal efforts, made under the sincere guidance of my teachers. If any part of this project is proven to be copied out or found to be a reproduction of some other, I shall stand by the consequences.

No portion of the work presented in this report has been submitted in support of an application for other degree or qualification of this or any other University or Institute of learning.

Bushra Asghar
317-FAS/MSCS/F06

Acknowledgement

This project and all my efforts are fruitful only due to Allah Almighty, the Most Merciful and Beneficial Who gave me strength to complete this task to the best of abilities and knowledge.

I would to take this opportunity of thanking my supervisor **Mr. Asim Munir**, who gave all this knowledge, guidance and support to boost my confidence and learning. He was always there to solve problems and helped me through this project. I am thankful to him for this.

Last but not the least; I would like to acknowledge **Mr. Zafar Javeed Shahid,** who provided me support with this technical help. I would like to admit that I owe all my achievements to my truly, sincere and most loving parents, brother , sisters and friends who mean the most to me, and whose prayers have always been source of determination for me.

Bushra Asghar

Project in Brief

Project Title:	Deblocking Filter for Block-Based ICA Transform Coding
Undertaken By:	Bushra Asghar
Supervised By:	Mr. Asim Munir
Tool Used:	Matlab 7.0, Rational Rose 98 Enterprise Edition
System Used:	Pentium® IV
Operating System Used:	Microsoft Windows XP
Starting Date:	1st September, 2007
Completion Date:	20th May, 2008

Abstract

The main subject of the present thesis is to remove the blocking artifacts in ICA decompressed blocky images. Different deblocking filters have been studied and two techniques are chosen as base techniques. **Deblocking Filter for Low Bit Rate MPEG-4 Video** proposed a deblocking filter for MPEG-4 video coding, with three different modes, smooth, complex and intermediate, of filtering. **Low Complexity Deblocking Method for DCT Coded Video Signals** proposed a deblocking filter for DCT coded video signals using five different modes, smooth, complex, intermediate, steep and corner, of filtering. We implemented these deblocking filters to remove the blocking artifacts of ICA decoded images at different quantization parameter (QP).

Experimental results show that the subjective and objective quality of the ICA decoded images is improved without over smoothing the image details. Objective quality of the images is measured in terms of signal-to-noise ratio (SNR) and peak signal-to-noise ratio (PSNR). In Lena ICA decompressed image at 21 QP, SNR improvement is of the order of 0.225656 dB and PSNR improvement has the order of 0.355978 dB.

TABLE OF CONTENTS

Chapter No **Contents** **Page No**

Chapter No **Contents** **Page No**

Chapter No **Contents** **Page No**

CHAPTER 1
INTRODUCTION

1. Introduction

Like many other recent technologies, compression of digital image or video becomes one of the enabling technologies for each of the aspect of multimedia revolution due to ever growing internet and fast communication over the network. For fast and real time data transmission, we have to compress (encoding) and decompress (decoding) a digital image or video to reduce the amount of data.

In the process of data compression, one of the major problems we are facing at the end results are the blocking artifacts produced that reduces the visual quality of the image or video. Blocking is a square or rectangular distortion area in an image and artifacts are the visual distortion in an image. Before going further into the detail of blocking, we define digital image and digital video as follow:

"Digital image is two-dimensional discrete valued function f(x,y), where x and y are the spatial coordinates and the amplitude of f at any pair of coordinates(x,y) is called the intensity or gray level of the image at that point, where x, y and f are all finite integers."

" Digital video is the representation of spatio-temporally sampled frames (still images) in digital form and each sample having brightness (luminance) and color (chrominance) where spatial are digital values on a rectangular plane and temporal is series of still images or frames at regular interval."

1.1 Blocking Artifacts

"In Data compression when block coding techniques are used, especially at a low bit rate (or high compression), decoded image suffers from square or rectangle like visual distortion especially at the block boundary (4x4, 8x8, 16x16) called blocking artifacts".

Blocking artifacts can be produced across 4x4, 8x8, or 16x16 block boundary depending on the compression used for image or video. Like in JPEG (Joint Photographic Experts Group), an image is divided into 8x8 blocks and then on each block Discrete Cosine Transform (DCT) is applied independently. After transformation, DCT coefficients are quantized and then entropy encoded. At a low bit rate, relationship between blocks is lost and only one DC and few AC coefficients are present. As a result blocking artifacts are produced. The blocking effect in JPEG images can be classified into three categories [17]

(a) Stair case noise along the image edges,
(b) Grid noise in the monotone areas, and
(c) Corner outliers in the corner points of the 8x 8 DCT blocks.

Traditional Transform based coders like H.261, H.243, MPEG-1/2 and MPEG-4 are based on block based Discrete Cosine Transform (BDCT). BDCT is used in most of coding standards because of its fast algorithm but at high compression blocking artifacts are found. In MPEG-1/2 and MPEG-4, compression is performed on 8x8 block so blocking is produced across 8x8 block boundaries. In H.264, blocking is produced across 4x4 block boundary.

Figure 1.1 (a) shows original "Lena.bmp" image and figure 1.1 (b) shows a decompressed blocky "Lena" image. It is clearly visible from the given images that the image after decompression having clear blocking effect especially in smooth regions near cheeks and shoulder.

a) Original Lena image (512x512)

b) Decompressed Blocky Lena image (512x512).

Figure 1.1 Illustration of blocking effect

1.1.1 Reasons of blocking artifacts

At decoding side when the image is decompressed especially in BDCT based techniques, image suffers from blocking artifacts due to many reasons. Quantization Parameter (QP) is one of the major reasons. When high quantization parameter is used to achieve high compression ratio, blocking artifacts are introduced.

According to Iain E. G. Richardson [4], there are two causes of blocking artifacts:

a) Over-quantization of the DC coefficient and
b) Suppression or over-quantization of low frequency AC coefficients.

The DC coefficient corresponds to the average (mean) value of each 8 x 8 block. In areas of smooth shading over quantization of DC coefficients means that there is a large change in level between neighboring blocks. When two blocks with similar shades are quantized to different levels, the reconstructed blocks can have a larger 'jump' in level and hence a visible change of shade. This is most obvious at the block boundary, appearing as a tiling effect on smooth areas of the image.

A second cause of blocking is over-quantization or elimination of significant AC coefficients. Where there should be a smooth transition between blocks, a 'coarse' reconstruction of low frequency basis pattern leads to discontinuities between block edges.

Another reason of blocking artifacts in video is motion compensated prediction. Motions compensated blocks are generated by copying interpolated pixel data from different locations of different reference frames. As a result discontinuities are formed on the edges of copied blocks as there is never a perfect match for the data, these discontinuities causes blocking artifacts.

1.2 Deblocking Filters

"A deblocking filter is applied to blocks in decoded images or video to improve visual quality at a given compression ratio and also improves the compression ratio for a given image quality by smoothing the sharp edges which can form between blocks when block coding techniques are used."

Deblocking filter is applied to each decoded macroblock to reduce blocking distortion. In MPEG-1/2 no built-in deblocking filter is present, in MPEG-4 optional in-loop filter is present while post filtering is suggested. In H.261 optional in-loop filtering is present; in H.263 deblocking filtering is not present. While in H.264 in-loop deblocking filtering is present (in-loop filtering means the deblocking filter is applied at the encoding side as well as at the decoding side). Table 1.1 summarizes the different standards and their deblocking filters.

1.3 Aim of Deblocking Filter

The aim of deblocking filter is to improve the visual appearance of decoded images by smoothing the block edges. The filtered image improves the compression performance as the filtered image is often a more faithful reproduction of the original image than a blocky, unfiltered image.

Standard	Deblocking Filter
H.261	Optional in-loop filter
MPEG-1	No filter
MPEG-2	No filter, post-filter processing often used
H.263	No filter
MPEG-4	Optional in-loop filter, post-filter processing suggested
H.264	Mandatory in-loop filter, post-filter processing may also used

Table 1.1 Deblocking Filters for various standards [18]

1.4 Techniques of Deblocking

In block-based transform, especially at low bit rates the noise caused by the coarse quantization of transform coefficients is visible in the form of a blocking artifact. According to Iain E.G Richardson and Peter List et al. in [4] and [12] respectively, there are two main approaches of integrating deblocking filter into image or video codec:

a) In-Loop Filtering
b) Post Filtering

1.4.1 In-Loop Filtering

In in-loop filtering, deblocking filter operates with in the coding loop, on both encoder and decoder side. In video codec it is useful as it improves accuracy of motion compensated prediction for next encoded frame.

Advantages

Some of the advantages of in-loop filtering are as follows:

1. Applying in-loop filtering within loop code, guarantee to improve the quality of reference frame at a certain level expected by the producer.
2. Secondly, in the decoder, there is no need of extra frame buffer. Here filtering can be carried out macroblock-wise during the decoding process, and the filtered output stored directly to reference frame buffer.
3. Thirdly, it improves both subjective and objective quality of decoded image by reducing the complexity at decoder side.

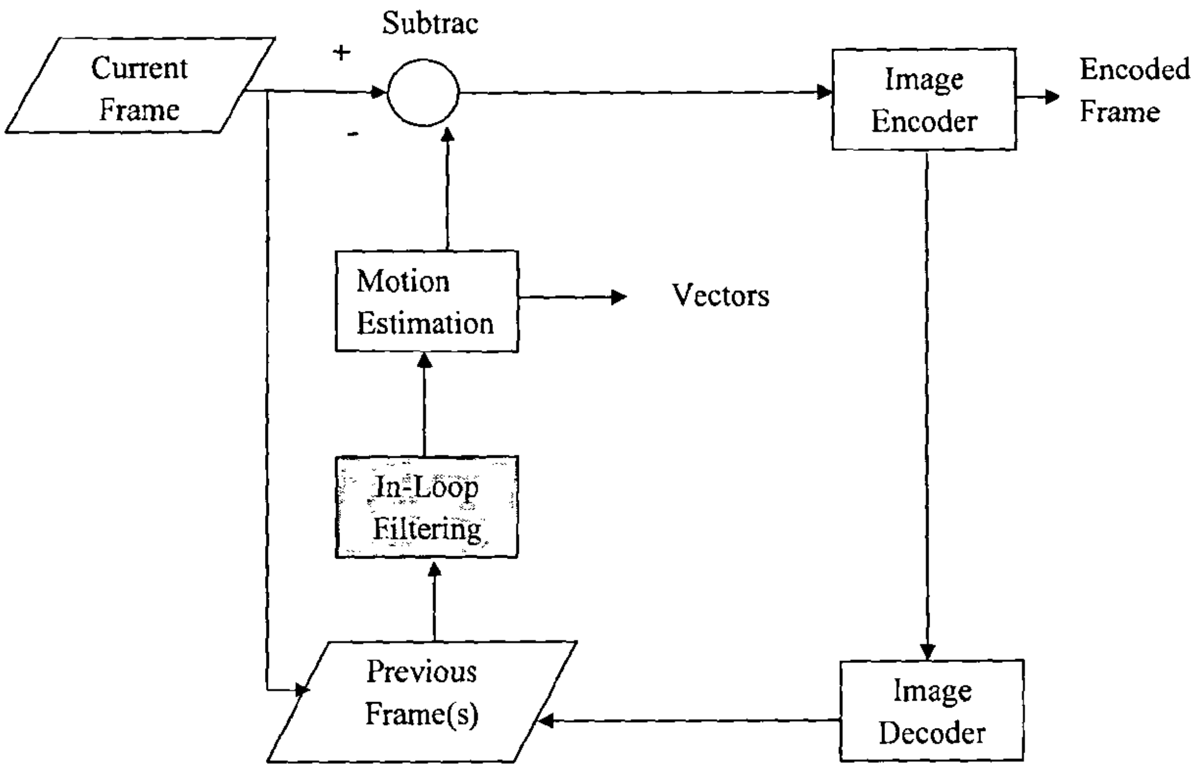


Figure 1.2 In-loop filter Encoder [4]

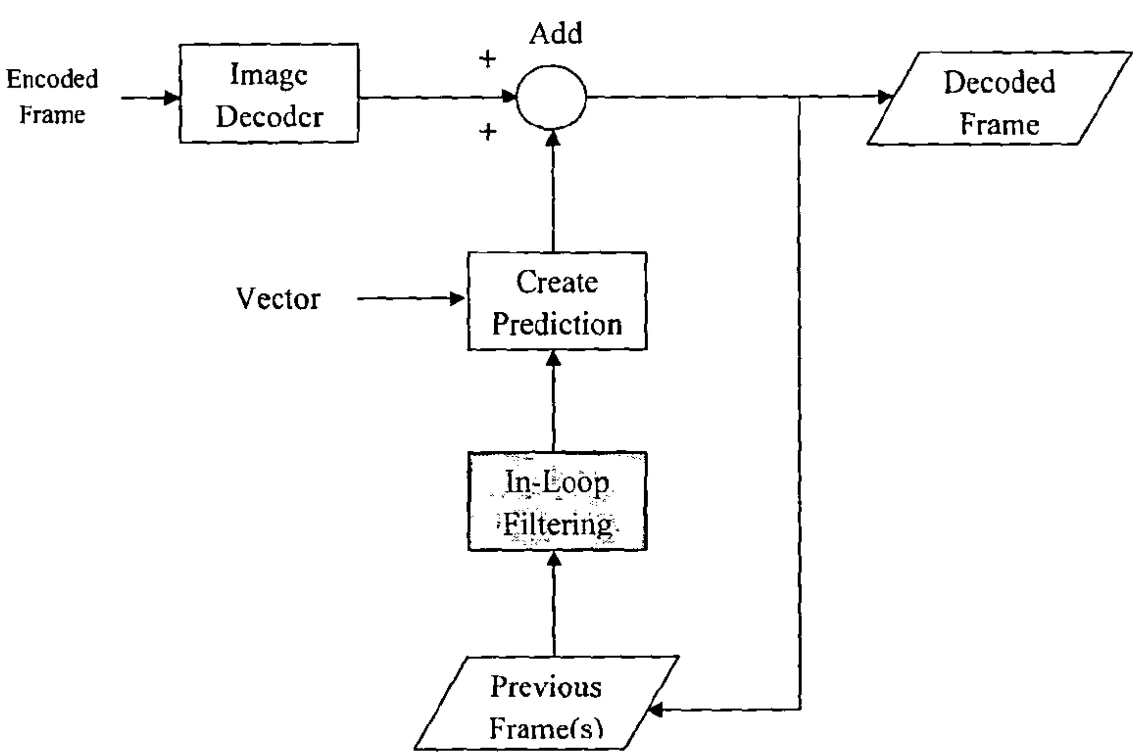


Figure 1.3 In-loop filter Decoder [4]

Disadvantages

1. In this approach the major disadvantage is encoder and decoder must use identical filter so limits interoperability between codecs.
2. Second disadvantage is the computational complexity even though filtering is implemented without multiplication or division operators.

1.4.2 Post Filtering

In post filtering, filtering is applied after decoding. As filtering is applied only after decoding so the designer having complete flexibility and wide range of filtering technique. In MPEG-4, there is optional post deblocking filtering. There are two types of post filtering:

1.4.2.1 Decoder-dependent Filters

Here decoded parameters are used for deblocking filtering.

Advantages

This filter mostly gives better performance than the decoder-independent because coding parameters gives extra information about original image.

Disadvantages

Filter is closely related to decoding parameter and must depend on decoder.

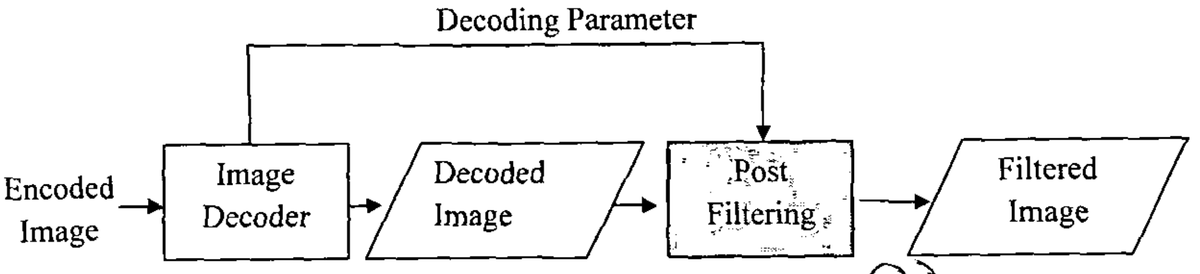


Figure 1.4 Decoder-dependent Filter [4]

1.4.2.2 Decoder-independent Filters

Here filter is applied independent of decoder and without any knowledge of decoding parameters.

Advantages

This filter gives the maximum flexibility.

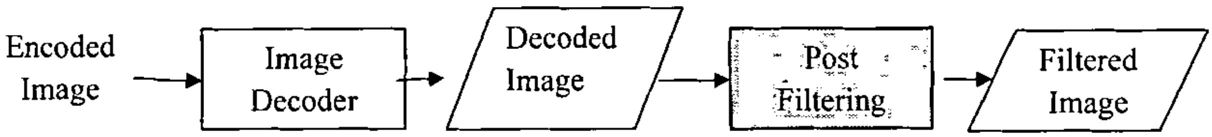


Figure 1.5 Decoder-independent Filter [4]

Disadvantages

Filter performance generally not as good as decoder-dependent filters.

1.5 Image Compression

"Minimizing the byte size of a graphic file with or without loss of values to an acceptable level is called Image compression; it is the application of data compression on digital image."

1.5.1 Basic methodology of Image Compression

The basic methodology of image compression is shown by the figure 1.6 as follows:

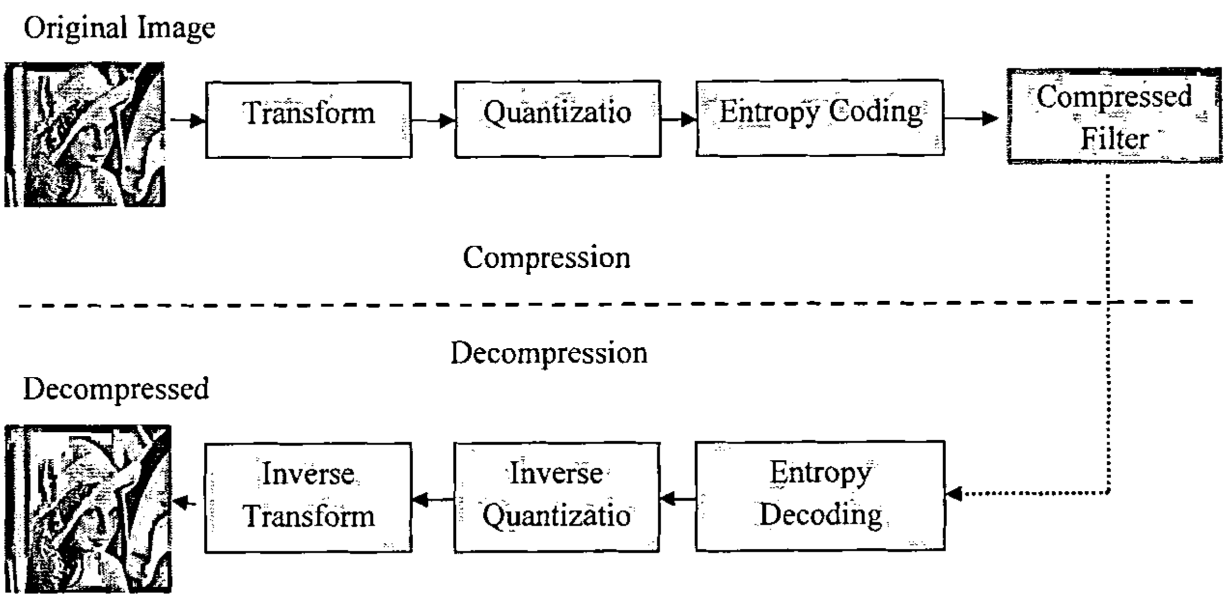


Figure 1.6 Basic Technique of Image Compression

To compress an image, firstly an image is compressed into a suitable form, like in JPEG; image is transformed by Discrete Cosine Transform (DCT) and in JPEG 2000 by Discrete Wavelet Transform (DWT). After that quantization and entropy coding is performed where transformed coefficients of higher frequencies are discarded as human eye is less sensitive to higher frequencies. In such a way by yielding higher compression ratio, a compressed file is created.

In decompression, exact inverse procedure is applied. In some techniques the decompressed image is irreversible, where the original image is not exactly reconstructed (as some of the coefficients are discarded) and in decompressed images we found distortion (like blocking effect). The reason of such distortion is coefficients quantization or entropy coding.

1.5.2 Types of Image Compression

Image compression can be lossy or lossless. Figure 1.7 shows the classification of Image Compression.

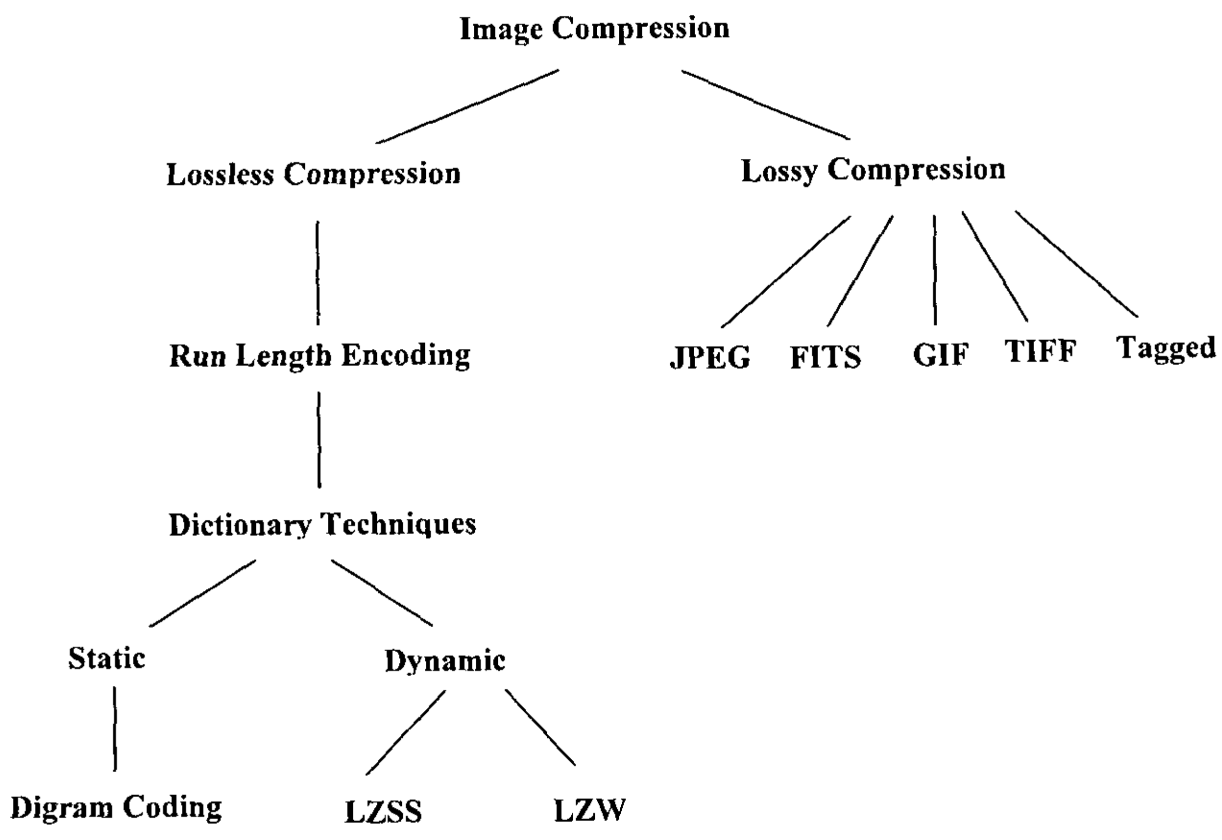


Figure 1.7 Types of Image Compression

a) Lossless image compression

In Lossless images compression exact image is reconstructed after decompression, without any loss. This technique is used in medical images, bank records, text articles etc, where no loss of image is compromised.

- Methods of lossless image compression are:
- Lempel-Ziv encoding techniques like LZSS, LZW
- Huffman coding
- Arithmetic coding

b) Lossy image compression

In lossy image compression exact image is not retrieved after compression; some data is lost in the process of compression. This technique is used in natural images like photos, multimedia data (audio, video, still images), internet telephony etc, where minor losses are acceptable.

Methods of lossy image compression

- JPEG
- TIFF
- Tagged
- GIF
- FITS

There are two basic lossy compression schemes:

- Lossy Transform Codecs
- Lossy Predictive Codecs

In lossy transform codecs, sample picture or sound are transformed into new basic shapes after segmentation then quantized and entropy encoded while in lossy predictive codecs, previous or subsequent data is used to predict current picture or sound. Then the error between the original and predicted data is quantized and encoded.

1.5.3 Advantages of Image Compression

- By reducing the size we are capable to store more data on a specified amount of a disk.
- It makes easy to send or receives data from internet/web pages.

1.5.4 Disadvantages of Image Compression

Due to compression, visual quality of images degraded causes

- blocking artifacts
- loss of information

1.6 Independent Component Analysis (ICA)

"Independent component analysis (ICA) is a computational method for separating a multivariate signal into additive subcomponents supposing the mutual statistical independence of the non-Gaussian source signals. It is a special case of blind source separation."[5]

In simple we can say, "Independent Component Analysis (ICA) is a statistical technique for decomposing a complex dataset into independent sub-parts". [6]

For the observed multivariate data of large databases, ICA defines a generative model. In the model, the data variables are assumed to be linear mixtures of some unknown latent variables, and the mixing system is also unknown. The latent variables are nongaussian and mutually independent, they are called the independent components of the observed data. Independent components also called sources or factors by ICA.
Superficially ICA is related to principal component analysis (PCA) and factor analysis. ICA is a much more powerful technique, when the classic methods completely fail; ICA is capable of finding the underlying factors or sources.

1.6.1 Examples

Some of the major examples of ICA are as follows:

- Mixtures of simultaneous speech signals that have been picked up by several microphones
- Brain waves recorded by multiple sensors
- Interfering radio signals arriving at a mobile phone
- Parallel time series obtained from some industrial process

1.6.2 Applications of ICA

The major applications of ICA are

- Blind Source Separation
- Feature Extraction
- Blind Deconvolution

- Blind Deconvolution
- Image Compression
- Separation of Artifacts in Magnetoencephalography (MEG) Data
- Finding Hidden Factors in Financial Data
- Reducing Noise in Natural Images
- Telecommunications
- Biomedical signal processing
- Data analysis in such areas as economics, psychology, and other social sciences density Data estimation
- Data regression

1.6.1.1 Blind Source Separation

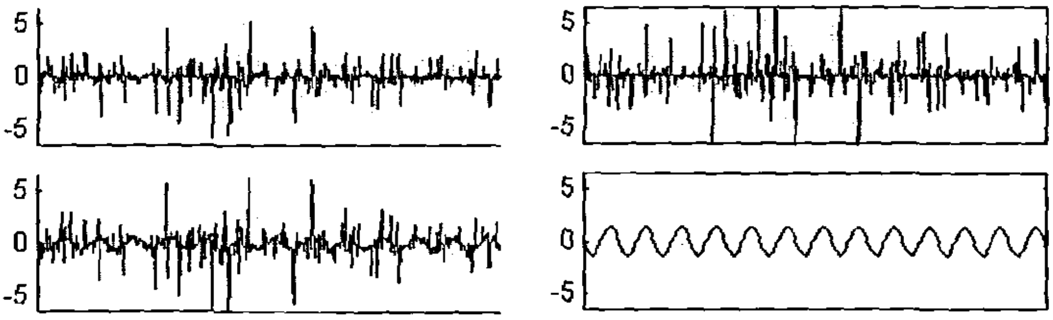
If we have two linear mixtures of two source signals that are independent of each other, i.e. observing the value of one signal does not give any information about the value of the other. The BSS determine the source signals given only the mixtures.

We can describe it in mathematical notation, to model the problem, as follows:

X=AS

Where
S is two-dimensional random vector containing the independent source signals
A is two-by-two mixing matrix
X contains the observed (mixed) signals.

Figure 1.8, (a) shows the mixture signals and (b) shows the separated signals. That is, at a given time instant, the value of the top signal is the first component of x, and the value of the bottom signal is the corresponding second component. The first step in many ICA algorithms is to whiten the data; means to remove any correlations in the data, i.e. the signals are forced to be uncorrelated. After whitening, the separated signals can be found by an orthogonal transformation of the whitened signals. Then appropriate rotation is applied. The source signals (components of S) in this example were a sinusoid and impulsive noise as shown in figure 1.8 (b).



a) Mixture Signals b) Separated Signals

Figure 1.8: An Illustration of Blind Source Separation

1.6.1.2 Feature Extraction

Another application of ICA is feature extraction. The use of ICA for feature extraction is motivated by the theory of redundancy reduction. The obtained features are closely connected to those offered by wavelet theory and Gabor analysis.

1.6.1.3 Telecommunications

Telecommunications is another emerging application area of great potential. An example of a real-world communications of ICA application is where separation of the user's own signal from the interfering other user's signals in CDMA (Code-Division Multiple Access) mobile communications.

1.6.3 Algorithms of ICA

Some of the algorithms used in ICA technique after preprocessing are

- Jutten-Herault algorithm
- Non-linear decorrelation algorithms
- Algorithms for maximum likelihood or infomax estimation
- Non-linear PCA algorithms
- Neural one-unit learning rules
- Neural (adaptive) algorithms
- The FastICA algorithm
- Tensor-based algorithm

1.7 Image Compression by ICA

Few attempts have been made to compress a digital image by ICA, like in [14], FastICA algorithm and Matching Pursuit (MP) algorithm is used to perform ICA image compression. Artur J. Ferreira et al. used four classes (natural, faces, fingerprints and synthetic) of images to compress them with ICA transform coding technique [20]. When the image is compressed by ICA, blocking artifacts are produced due to quantization and lowers the visual quality of the image.

Fig. 1.9 (a) and (b) shows original Lena image and decompressed Lena image with ICA transform coding technique. The result of fig. 1.9 presents that decompressed Lena image having visible blocking artifacts.

a) Original Lena Image (512 x 512)

b) Lena image decompressed by ICA Transform coding at QP = 21 and SNR = 9.68536

Figure 1.9 Illustration of Image Compression by ICA

1.8 Scope

The focus of research is to study existing different deblocking filters based on BDCT transformation and find a suitable deblocking filter for the compressed images by ICA transform coding technique. When image is decompressed by ICA, blocking artifacts are produced. There are many techniques to remove the blocking artifacts and these are

implemented on BDCT based (JPEG, MPEG1/2/4, H.26x) images and videos but no known technique is present to remove the blocking artifacts on ICA compressed image.
Some major tasks of the research are as follows:

- Study existing Deblocking Filters
- Find a suitable deblocking filter for ICA compressed images.
- Implement deblocking Algorithm on ICA compressed images.
- Test results on different images decompressed at different Quantization Parameters (QP).

1.9 Thesis Layout

The aim of the thesis is to reduce the blocking artifacts in ICA decompressed blocky images. This thesis is organized as follows. In chapter 2, literature review is present, describing about the pioneers of the work and related existing techniques. Chapter 3 focuses on our research methodology for reducing the blocking artifacts in ICA decompressed images. Implementation and experimental results are present in chapter 4 and 5 respectively. Lastly, conclusion and future work are covered in chapter 6.

CHPATER 2
LITERATURE SURVEY

2 Literature Survey

In still images like JPEG blocking is considered to be appeared always on block boundaries while in video compression, previous frame may propagate its blocking effect to next frame other than the block boundaries. Different authors categorize deblocking techniques into different ways like:

a) Ling Guan et al. [3], describe various spatial domain post filtering techniques such as low-pass filtering (LPF), projection onto convex sets (POCS), maximum a posteriori (MAP) filters, and adaptive low-pass filters (ALPFs).

b) Shen-Chuan Tai et al. say there are three major groups to reduce the blocking artifacts.

   1. The first group uses low-pass filtering, where the primary problem is the over-smoothing,

   2. The second group involves in statistical estimation in which maximum posterior probability (MAP) technique is used to reduce the artifacts.

   3. The final group involves set-theoretic reconstruction methods. The last method reconstructs the image by projecting onto convex sets (POCS) [10] [11].

c) Ken Kin-Hung Lam says deblocking algorithm can be classified into regression-based algorithm, wavelet-based algorithms, anisotropic diffusion based algorithm, weighted sum of pixels across block boundaries based algorithm, iterative algorithms based on projection on convex sets (POCS), and adaptive algorithms [19].

2.1 Coding Artifact Reduction Based on Local Entropy Analysis: by Ling Shao, Ihor Kirenko, 2007 [7]

Many deblocking algorithms have been proposed in the last decade. According to Ling Shao et al. [7], deblocking methods can be classified into two types

- Pixel-domain filtering
- Frequency-domain filtering

Pixel domain filtering applies low-pass filtering at block boundaries to remove coding artifacts and preferred for real time applications. The proposed technique is used to reduce coding artifacts from MPEG-4 encoder at different bit rate.

Methodology:

In this paper, an adaptive blocking artifact reduction algorithm is proposed. First of all the image is divided into N x N blocks then on each block histogram using the luminance intensity is constructed. After that local entropy of a region is calculated to get the information about the respective block by using Probability Density Function (PDF).Then on the basis of entropy, the blocks are classified into three main regions:

- Detailed regions
- Intermediate regions and
- Smooth regions

The region with high entropy having more information in it so classified as detailed region. The region with low entropy is classified as smooth region.

In detailed region, most of the information is present so weak low-pass filtering is applied to preserve the sharpness of the image. If the block is in intermediate region low-pass filtering is applied and if the block is in smooth region then strong low-pass filtering is applied as in this region blocking artifacts are more visible than other regions.

Observation

Blocking artifacts are more visible in flat regions while in detailed regions they are barely noticeable, due to two main reasons:

1. An encoder usually allocates more bits for detailed regions than for flat regions, so few artifacts appear in detailed regions.
2. The second reason is due to the masking effect. This effect makes the artifacts less noticeable for human eyes in detailed regions.
   So strong filtering is applied in flat regions.

Advantage

- Deblocking filters is applied according to the local pixel values of the block due to which sharpness of the image is preserved and excessive blurness is avoided.

- In smooth region, strong filtering is applied as in this region blocking artifacts are more visible than other regions. Weak filtering is performed in complex region to retain the sharpness of the image.

Disadvantage

- The proposed algorithm gives better result of reducing blocking artifacts at higher bit-rates.

- Local Entropy of a block is calculated and deblocking filters are applied within the block. In most of the cases, blocking artifacts appeared on the block boundary, so if the deblocking filters are applied on the block boundary then more blocking artifacts are removed.

2.2 An Adaptive Fuzzy Filter for Coding Artifacts Removal in Video and Image: by WU Jing, YE Xiu-qing, GU Wei-kang, 2007 [8]

WU Jin et al. [8] propose a new adaptive post-filtering algorithm to remove coding and ringing artifacts in block-based coder. For de-blocking, the blocking strength and the maximum quantization parameter of the image is identified to determine the filtering range and fuzzy filter respectively.

Methodology

In many compression standards using block-based processing, like JPEG, MPEG, H.261, H.263, and H.264, the most obvious coding artifacts are the blocking and ringing artifacts. The blocking artifacts shows grid noise on block boundaries in smooth areas, and the ringing artifacts show oscillations near edges, which is due to abrupt truncation of high frequency components. The proposed algorithm first apply de-blocking filtering by using fuzzy filter and then de-ringing is applied by edge detection , ringing detection and fuzzy filter implementation.

De-blocking

The proposed de-blocking algorithm, detects the blocking artifacts, classify the block into smooth and texture region and then 1 D deblocking fuzzy filter is applied, on all the 8×8 block boundaries first across the horizontal edges then across the vertical edges. Luminance component and chrominance component are dealt in the same way.

De-ringing

After deblocking, de-ringing is applied by using three main operations, edge detection, complementary ringing detection and de-ringing filtering. Edge detection is the first step of de-ringing algorithm as ringing artifacts always arises along the object edges. For it, fuzzy edge detection algorithm is used. Complementary ringing detection procedure detects the ringing artifacts that are masked by blocking artifacts. The last step is de-ringing filtering that is applied by 2 D adaptive fuzzy filter, to remove ringing artifacts.

Advantage

Processing of deblocking filtering is fast and satisfied the real time applications.

Disadvantage

The de-ringing algorithm is less speedy as compares to deblocking filtering.

2.3 A Simplified Human Vision Model Applied to a Blocking Artifact Metric: by Hantao Liu and Ingrid Heynderickx, 2007 [9]

Hantao Liu et al. [9] propose a deblocking filter which is based on two basis characteristics of Human Visual System (HVS), namely texture masking and luminance masking. This model is applied to a blockiness metric. Performance of blockiness metric is determined by comparing it to the same blockiness metric having different HVS-based models.

Methodology

According to Hantao Liu et al. masking is the reduction in the visibility of one image component (the target) due to the presence of another (the masker), and it is strongest when both components have the same or similar frequency, orientation, and location. The proposed algorithm uses two fundamental properties of HVS:

1. Luminance masking: In this masking, averaged background luminance surrounds the stimulus.

2. Texture masking: In this masking, spatial non uniformity in the background luminance.

In texture masking, first step is the texture detection, to calculate the local background activity, second step is thresholding where active background regions are captured and the third step is visibility transform function (VTF), to obtain a visibility coefficient (VC) for texture masking. In luminance masking there are two steps. First step is local luminance detection to calculate the local averaged luminance background. Second step is visibility transform function (VTF) to obtain a visibility coefficient (VC) based on the HVS characteristics for luminance masking. After texture and luminance masking visibility coefficients are combined into a single visibility coefficient.

Findings

- Mostly, blocking in a dark surrounding is less visible than the one in a bright surrounding.
- In 8 bits gray-scale images, distortion is most visible for a surrounding with an averaged luminance value between 70 and 90 (centered approximately 81).

2.4 Low Complexity Deblocking Method for DCT Coded Video Signals: by S.-C. Tai, Y.-R. Chen, C.-Y. Chen and Y.-H. Chen, 2006 [10]

S.-C. Tai et al. [10] propose a post-processing deblocking filter to remove the blocking artifacts in low bit-rate. At low bit-rate, there is only one DC and few AC coefficients present in a block, due to this relationship between blocks are lost and artificial discontinuities between block boundaries are appeared that make the image uncomfortable for human eye.

Methodology

The proposed technique is for 8x8 block boundaries, as many DCT based images and videos are suffered from blocking artifacts across 8x8 block boundaries. Difference between the two pixels across the block boundary is calculated to determine about the existence of real edge (if the difference is greater than a specified threshold value) or a blocking artifact (if the difference is smaller than a specified threshold value). If blocking artifacts exit then the block is divided into five different modes according to the respective pixel values of the block. The modes are

1. Smooth mode
2. Intermediate mode
3. Complex mode
4. Steep mode
5. Corner mode

In smooth mode filtering is applied on 6 pixels, 3 pixels on each side of the block boundary, here human eye is more sensitive than the complex region so stronger filtering is applied. In complex mode, as human eye is not sensitive so filtering is applied only on one pixel across each block boundary, the pixels inside the block boundary are not updated.

Blocking artifacts may also enhance the real edges across a block boundary. Steep mode is used to remove the enhanced edges. After applying all above four filters some gaps will occur on non-horizontal or non-vertical edges, these points decrease the visual quality of the image so these points are removed by corner mode filtering.

Advantage

- The proposed deblocking filter, keeps in mind that the real edge should not be filtered.
- It removes the enhanced edges (that are produced due to blocking artifacts).
- As a last step, proposed algorithm also fills the gaps that are created during the reduction of blocking artifacts.

Disadvantage

The resulted value of PSNR and blocking measurement (M) is not as good as the H.264 loop filter has.

2.5 On the Use of Independent Component Analysis for Image Compression: by Artur J. Ferreira and M´ario A. T. Figueiredo, 2006 [20]

Artur J. Ferreira et al. [20] describe the use of Independent Component Analysis for Image compression on four classes of images: natural images, face images, finger prints and synthetic images. Two standards, Signal to Noise Ratio (SNR) and Picture Quality Scale (PQS) are used for the assessment of results.

Methodology

In the proposed algorithm, Principle Component Analysis (PCA) is used to find out the mean removal and whitening of the test images. After that, Fast ICA algorithm is used to obtain complete (M = N) and over-complete (M > N) bases from whitened images. Generally ICA bases are non-orthogonal as the histogram shows the relative angles are above 40o. For these non-orthogonal bases, two approaches are considered:

- Orthogonalization methods. These methods yield orthogonal bases where standard orthogonal projections are used for coding.
- Matching Pursuit (MP) algorithms. Standard MP, orthogonal matching pursuit (OMP), and high resolution MP (HRMP) algorithms are used. These are iterative methods for finding decompositions on non-orthogonal bases.

For image coding, bases are used as data-independent transform as they are not transmitted along with the coded image; they are initially transmitted to both encoder and decoder. Bases are obtained by two approaches:

- Basis dimension adjustment to obtain a basis B, which can be incomplete, complete or over- complete
- Orthogonalization, obtaining an orthonormal basis C.

Transform based coding architecture is used where for non-orthogonal basis, two method are used for transmission of coefficients

- With indexes, where only the non-zero coefficients and corresponding indexes are send.
- Without indexes, where all coefficients either non-zero or not are send.

In orthogonal approach, indexes are not used and following two methods are used for sending coefficients

- The first L coefficients
- The largest L coefficients, setting all others to zero.

Findings

- When the face image is decompressed by ICA, better SNR and PQS is obtained as compared to JPEG especially at a low bit rate and closer values are obtained as compared to JPEG 2000.
- On finger print images, the proposed variable size coder gives SNR and PQS values almost same as WQS and close to JPEG 200.
- Blocking artifacts are present in an ICA decompressed images.
- ICA decompressed image is less blocky than a JPEG decompressed image.
- Natural images are coded with HRMP algorithm and HRMP algorithm gives slightly better results than MP algorithm.

2.6 Deblocking filter for Low Bit Rate MPEG-4 Video: by Shen Chuan Tai, Yen-Yu Chen, and Shin-Feng Sheu, 2005 [11]

Shen-Chuan Tai et al. [11] propose post processing deblocking filter for low bit rate video compression. Post processing does not change the existing standard of coding or decoding so it is most practical solution.

Methodology

The proposed algorithm is based on region classification across 8x8 block boundary on the basis of activity across each boundary. The block is classified into smooth, complex and intermediate region. In smooth region, as human eye more sensitive to blocking artifacts so deblocking filter update six pixels, three pixels across each block boundary. In complex region, activity is high, so strong filtering is not appropriate as it may over blur the real edge. In this mode, total four pixels, two across each block boundary is updated. In intermediate region, deblocking filter balance the strong filtering in smooth region and weak filtering in complex region, only by updating one pixel across each block boundary.

Advantage

- The decision mode is independent of the blocking artifact's position.
- Strong filtering is applied in smooth region as in this region blocking artifacts are more visible.

Disadvantage

The proposed algorithm, does not concentrate on the removal of enhanced edges that are produced during the process of deblocking and also does not fill the gap that are created by applying the respective algorithm.

2.7 Adaptive Deblocking Filter: by Peter List, Anthony Joch, Jani Lainema, Gisle Bjøntegaard, and Marta Karczewicz, 2003 [12]

Peter List et al. [12] describe the adaptive deblocking filter for H.264/MPEG-4 AVC coding standard. The filtering is applied on the macro block basis, first across vertical edges then across horizontal edges. Filtering in both directions must be conducted before moving to the next macro block.

Methodology

A Boundary-Strength (BS) parameter is calculated across every 4x4 block boundary, and assigns an integer value between 0-4. Filtering is performed across block boundary according to the value of BS. If the value is 4, special strong mode filtering is applied and if the value is 0, no filtering is applied. The adaptive deblocking filter is applied on following levels

- On the slice level
- On the block-edge level
- On the sample level

Findings

- Luminance as well as chrominance filtering is applied.
- Visual quality is improved with 9% increase in bit-rate saving and in PSNR.

2.8 Removal of DCT Blocking Artifact Using DC and AC Filtering: by Ismaeil R. Ismaeil and Rabab K. Ward, 2003 [13]

Ismaeil R. Ismaeil et al. [13] introduce a post-filtering algorithm that reduces the blocking artifacts in smooth background area of the image. In DCT based image compression, especially at a low bit rate, image is divided into 8x8 blocks. On each block, DCT transformation is applied and high frequency DCT coefficients are quantized to zero, due to which blocking artifacts are produced between block boundaries.

Methodology

In the proposed algorithm, firstly the blocks are detected that have a DC coefficient, then the average of the DC coefficient block and the DC coefficients of the neighboring four 8x8 DCT blocks are calculated. After that, low frequency AC coefficients are predicted from the DC coefficient within a 3x3 array of blocks, using least mean square algorithm. The predicted AC coefficients help to remove the blocking artifacts especially in smooth background area of the image.

Findings

- In the proposed algorithm, averaging of the DC coefficients is introduced. Due to averaging blocking artifacts may be produced if neighboring blocks have a DC coefficient that is different from the DC coefficients of the subject.

- Prediction of AC coefficients is often wrong at the edges, so not carried out near block boundary.

2.9 Class adapted image compression using Independent Component Analysis: by Artur J. Ferreira and M´ario A. T. Figueiredo, 2003 [14]

Artur J. Ferreira et al. [14] describe image compression by Independent Component Analysis technique, on two classes of images: faces and fingerprints and compared the results according to standards Signal to Noise Ratio (SNR) and Picture Quality Scale (PQS).

Methodology

In the proposed algorithm, first step is to apply Principle Component Analysis (PCA). After mean removal and sphering by PCA, Hyvarinen's FastICA algorithm is used to estimate complete (M = N) and over-complete (M > N) bases and then greedy iterative matching pursuit (MP) algorithm is applied to decompose the image.

The proposed coding architecture is transform-based where transform coefficients are obtained by MP over the (complete, over-complete, or incomplete) ICA basis. Two methods are used to encode the coefficients, first, sending only the non-zero coefficients and the corresponding indexes, second, sending all coefficients either being zero or not. Fixed (8x8) and variable (16x16, 8x8, 4x4) block size are considered. In variable block size, each 16x16 or 8x8 block is splitting into four sub-blocks and then the adaptive arithmetic coder is used to decompose these blocks.

Findings

- In all the ICA basis vectors, most of the angles are above 40o, so these bases are non-orthogonal.
- In two coding methods, sending coefficients without indexes, 5 bits per coefficients are used while in sending coefficients with indexes 5 bits for the first coefficients and 4 bits for the remaining ones are used.
- When the face image is decompressed by ICA, better SNR and PQS is obtained as compared to JPEG, at low bit rates.
- For finger print images, wavelet scalar quantization (WSQ) gives better SNR and PQS than ICA decompressed images.
- The variable block-size coder, with a complete basis yield ~2 dB lower distortion than fixed block-size coder.

- When the image is decompressed by ICA, blocking distortion is produced.

2.10 Deblocking Filter With Two Separate Modes in Block Based Video Coding: by Sung Deuk Kim, Jaeyoun Yi, Hyun Mun Kim, and Jong Beom Ra, 1999 [16]

Sung Deuk Kim et al. [16] proposed a deblocking filter to remove the blocking artifacts at a low bit rate video coding using two filtering modes: smooth region mode and default mode which are selected according to the pixel behavior around the block boundary.

Methodology

Firstly, the proposed algorithm selects a proper mode (smooth mode, default mode), on the basis of local image characteristics of the region. In smooth mode, strong filtering is applied inside the block as well as on the block boundaries, as smooth regions are more sensitive to Human Visual System (HVS). In the default mode, filtering is performed only on the block boundary on the basis of the frequency information. To get the feature information of the pixels, four point DCT is used as a frequency analysis tool.

Findings

- In the smooth region, to prevent the real edges from over-smoothing, filtering is not performed if the difference between the maximum and minimum block boundary values is larger than the Quantization Parameter (QP) of the respective block.
- In default mode, high frequency anti-symmetric component is a major factor of the blocking artifacts. If the magnitude of this component is greater than a QP then filtering is not performed to retain the image details.
- In the proposed algorithm, each mode contributes in the improvement of PSNR.

2.11 Motivation

Independent Component Analysis (ICA) is an emerging technology, having its applications in different areas. Data compression is also one of the major application areas of ICA. But the primary problem, that appears, when the image is compressed by ICA transform coding technique is the production of blocking artifacts.

Artur J. Ferreira et al. [14] [20] compresses image using ICA transform coding and gives better PSNR and PQS as compared to JPEG but the resultant image is still blocky. To remove the blocking artifacts that are produced due to image compression by ICA, there is a need of an efficient deblocking filter.

Different deblocking algorithms exist for images or videos based on block-based discrete cosine transform (BDCT) [7]-[13], [16]-[18]. BDCT is most widely used technique to compress images or video but the main problem with the BDCT compressed images are the blocking artifacts that shows artificial discontinuities between adjacent blocks due to independent processing of blocks without the consideration of inter-block pixels.

Our task is to remove the blocking artifacts produced by ICA compressed images with the study of existing deblocking algorithm and finding a suitable algorithm for ICA compressed blocky images.

The major objectives of the research are as follows:

- Study existing deblocking algorithms for BDCT images
- Carryout the analysis of two existing deblocking techniques only for images i.e., deblocking filter for low bit rate MPEG-4 video by Shen-Chuan Tai et al. [11] and low complexity deblocking method for DCT coded video signals by S.-C. Tai et al. [10].
    - Shen-Chuan Tai et al. [11] uses three mode filtering (smooth, intermediate and complex) to remove the blocking artifacts in MPEG-4 decoded video stream.
    - S.-C. Tai et al. [10] uses five mode filtering (smooth, intermediate, complex, steep and corner) to remove the blocking artifacts in DCT coded video signals.
- Removing the blocking artifacts of ICA decompressed Images using above mentioned techniques.
- Improving the PSNR of the blocky image.

CHPATER 3
RESEARCH METHDOLOGY

3 Research Methodology

Our research work is the study of different existing deblocking filters for various BDCT based images. We also give the analysis and comparison of two existing deblocking filters to remove the coding artifacts from images, one is deblocking filter for low bit rate MPEG-4 video using three modes of filtering and the second is deblocking method for DCT coded video signal using five modes of filtering. After the analysis of these two techniques, we apply these algorithms on blocky ICA decoded images to remove the coding artifacts. In these two techniques, we use same ICA decoded images, with different Quantization Parameter (QP), to get the comparative analysis and to conclude the results.

3.1 Deblocking Filter for MPEG-4 Video Coding

The first method we choose is the "deblocking filter for low bit rate MPEG-4 video" [11]. This scheme is proposed by Shen-Chuan Tai, Yen-Yu Chen and Shin-Feng Sheu, Circuits and Systems for Video Technology, IEEE Transactions on Volume 15, Issue 6 June 2005 Page(s): 733 – 741.

3.1.1 Reasons for selecting the technique

Many deblocking algorithms exist to remove the blocking artifacts. We choose this technique as it is a post processing technique and according to Shen-Chuan Tai et al. [11], "Post processing appears to be the most feasible solution because it does not require any existing standards to be changed."

The second main reason to choose this technique is that the proposed algorithm apply deblocking filter according to the local pixel value characteristics. According to the local pixel values, three different modes of filtering, smooth filtering, complex filtering and intermediate filtering, is applied. Due to three different modes, sharpness of the image is retained and image is not over-blurred, thus giving better visual appearance of the blocky image and improving the PSNR value.

3.1.2 Methodology

Shen-Chuan Tai et al. [11] propose a deblocking filter with three modes of filtering to remove the blocking artifacts. First of all, image is divided into 8x8 blocks then on each block boundary, activity is calculated. On the basis of activity value, block is divided into three different modes i.e. smooth mode, complex mode and intermediate mode. In each mode, different filters are applied to remove the blocking artifacts and improve the visual quality of the image.

Smooth filtering is applied on six pixels, three on each side of the block boundary by keeping in mind that the image should not be over-blurred. Complex filtering is performed on four pixels (two pixels across each block boundary); to retain the sharpness of the image and intermediate filtering is also applied only on two boundary pixels to balance the strong filtering in smooth region and weak filtering in complex region. Figure 3.1 shows the flow chart of the Shen-Chuan Tai et al. [11] propose deblocking filter.

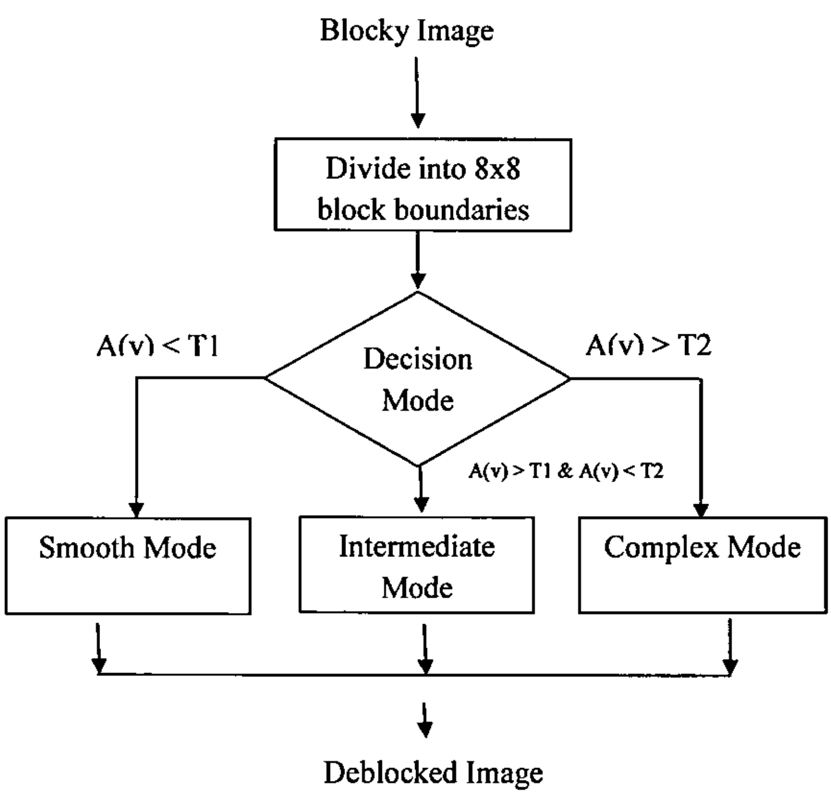


Figure 3.1 Flow chart of Shen-Chuan Tai et al. [11] Proposed Deblocking Filter

3.1.3 Comments

While implementing the proposed technique, some missing elements are observed. Like, the author only informs to calculate the activity of two pixels across a block boundary. The algorithm does not describe how to find out the activity of 8 rows across a block boundary. We modify activity formula, by taking the average of 8 activities (across each row). We also keep in mind that the real edge should not be affected with the implementation of deblocking algorithm. For it, we apply the check that if the offset is greater than 2* QP than smooth filtering is not performed and if offset is greater than QP then complex filtering is not performed where QP is the quantization parameter of the respective block.

The proposed technique is for low bit rate MPEG-4 video but we have implemented it on ICA decoded images to remove the blocking artifacts.

3.2 Deblocking Filter for DCT Coded Video

The second method is "low complexity deblocking method for DCT coded video signals". This method is proposed by Tai, S.-C., Chen, Y.-R. Chen, C.-Y. and Chen Y.-H, Vision, Image and Signal Processing, IEE Proceedings, Volume 153, Issue 1, 9 Feb. 2006 Page(s): 46 – 56.

3.2.1 Reasons for selecting the technique

The proposed technique is chosen as it is a post processing technique and it applies deblocking filtering according to local pixel values of the blocky image. Another reason for choosing this methodology is that the algorithm uses five different modes of filtering, smooth mode, intermediate mode, complex mode, steep mode and corner filtering.

Human eye is more sensitive to blocking artifacts in smooth region so in smooth mode, stronger filtering is applied as compared to complex mode. Due to blocking artifacts, edges near the block boundaries are enhanced. Steep mode is used to remove these enhanced edges. Corner mode filtering is used to remove the gaps occurring near the non-horizontal or non-vertical edges of the blocky image.

3.2.2 Methodology

S.-C. Tai et al. [10] propose a deblocking filter to remove the blocking artifacts in DCT coded video signals using five different modes of filtering. The proposed algorithm first divides the blocky image into 8x8 blocks and then on each block boundary (between two blocks) offset is measured. On the basis of offset, decision is taken that either it is a real edge or a blocking artifact. If it is a real edge then no filtering is performed and if it is a blocking artifact then a decision is taken to which region these blocks belongs to, either smooth, complex or intermediate region.

In the smooth and intermediate regions, filtering is performed only if the respective offset is less than 2QP, where QP is the Quantization Parameter of the block. If the offset is greater than 2QP then steep mode filtering is performed that remove the enhanced edges in the blocky image. In complex mode, if the offset is less than QP then complex mode filtering is performed otherwise steep mode filtering is performed. After filtering is performed in smooth, complex, intermediate and steep mode, corner mode filtering is performed on the corners four blocks to remove the non-horizontal and non-vertical gaps and improves the visual quality of the image. Figure 3.2 shows the flow chart of the Shen.-C. Tai et al. [10] propose deblocking filter.

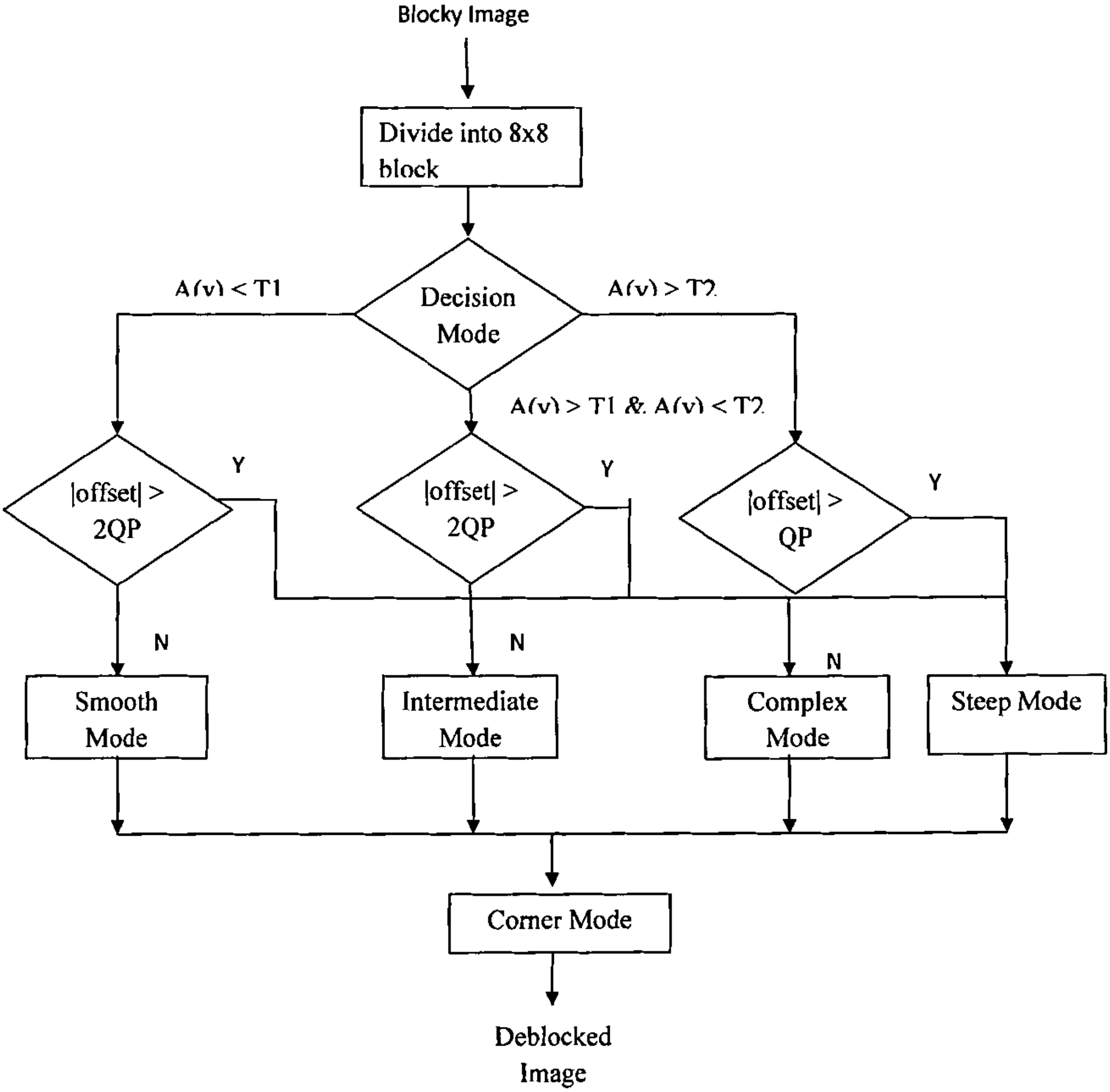


Figure 3.2 Flow chart of S.-C. Tai et al. [10] Proposed Deblocking Filter

3.2.3 Comments

In the Methodology, only one offset (of two pixels) determine that the edge of 8x8 block boundary is real edge or a blocking effect. In many cases it may happen that the 8 offsets across one block boundary may differs. So in that scenario, one block boundary may be classified as a smooth, complex, intermediate and steep mode simultaneously and classification creates the ambiguity.

The Methodology is for DCT coded video signals but we have implemented it on ICA decoded images.

3.3 Proposed Scheme

The scheme we have followed in our research is as follows:

- **Load Blocky Image**
  The first step is to load a blocky image decompressed by Independent Component Analysis (ICA) transform coding.

- **Divide Image into NxN blocks**
  The blocky image is divided into 8x8 block boundaries to remove the blocking effect across the block boundary. We also apply the deblocking filtering across 16x16 block boundaries but the result of applying 8x8 deblocking filter is better than that of16x16 block boundaries.

- **Mode Decision**
  After dividing the image into NxN block boundaries, we find out the activity across the block boundary. On the basis of activity we apply the deblocking filters.

- **Apply Deblocking Filters**
  Mode decision classifies the blocks into different regions on the basis of the value of activity. Different types of deblocking filters applied according to the two different algorithms i.e. smooth filtering, complex filtering, intermediate filtering, steep filtering and corner filtering.

- **Display the Deblocked Image**
  After the implementation of deblocking filters, the recovered image is displayed.

- **Calculate the SNR value**
  The value of SNR (Signal to Noise Ratio) is used as a standard to measure the recovery of the blocky image. The more SNR value, the more blocking effect is remove.

- **Comparison**
  The results of two techniques are observed and a comparison is made between them.

- **Conclusion**
  Finally, we conclude our work on the basis of our comparative analysis.

3.4 Key Points

Some key points that we kept in our mind are as follows:

- We keep in mind that while applying deblocking filter and removing the blocking effect the real edge should not be removed.
- Human eye is more sensitive to smooth region so stronger filtering is applied in smooth region to remove the blocking effect.
- To retain the sharpness of the image, weak filtering is applied in complex region.
- We prefer post-processing; as it does not change the existing standard and it work independently after image is decompressed by a decoder. In post-processing, filtering is applied after encoding.

CHPATER 4
IMPLEMENTATION

4 Implementation

Deblocking filters for block based ICA transform decoded images are implemented by using two methodologies. The first method is to remove blocking artifacts in MPEG-4 by Shen-Chuan Tai et al. and the second method is deblocking filter for DCT coded video signals, [11] and [10] respectively.

4.1 Implementation of Scheme I

The first scheme is the "deblocking filter for low bit rate MPEG-4 video", proposed by Shen-Chuan Tai, Yen-Yu Chen and Shin-Feng Sheu [11]. This scheme is for MPEG-4 videos and we implement it, in Matlab 7.0, for ICA compressed images.

4.1.1 Divide Image into N x N blocks

After image reading, the first step is to divide the image into N x N block boundaries. In the proposed methodology, the image is divided into 8 x 8 block boundaries.

block_width = row / 8
block_height = col / 8
blocks = Divide Image into block_width x block_height

Where row is the total number of rows, col is the total number of columns and blocks contains all 8x8 blocks of the respective image.

4.1.2 Activity Calculation

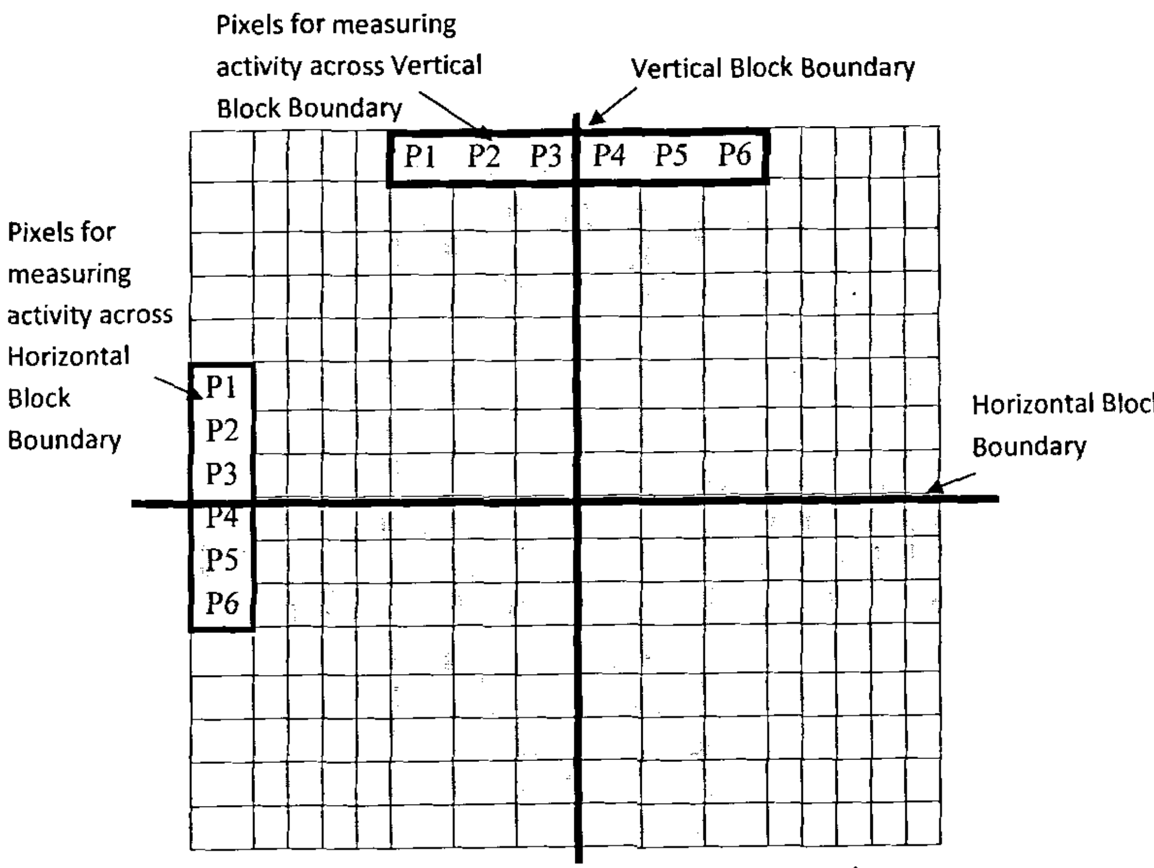
When the image is divided into 8x8 blocks, activity is calculated across each block boundary, first across horizontal then vertical block boundary. Figure, 4.1 presents the six boundary pixels (p1, p2, p3, p4, p5 and p6); three in each side of block boundary, to calculate the activity across each row. The activity across each row is calculated by the following formula:

$$A(p) = \sum_{i=1}^{5} \Phi(p_i - p_{i+1}) \quad \textbf{Eq. 4.1}$$

Where

$$\Phi(\Delta) = \begin{cases} 0, |\Delta| < S \\ 1, otherwise \end{cases} \quad \textbf{Eq. 4.2}$$

Equations 4.1 and 4.2 are applied across each horizontal and then vertical block boundary, from top left to right bottom. In Eq. 4.1, p is one-dimensional array of pixel values across each row at block boundary and in Eq. 4.2; S is threshold value, set to low, to find the difference between neighboring pixels.



Figure, 4.1 Pixels for measuring activity

After calculating the activity across each row, the activity across a block boundary is calculated by taking the average of all the eight activities in each row.

$$A(v)=\sum_{i=1}^{8} A(i) \Big/ 8 \qquad \textbf{Eq. 4.3}$$

Where A(i) is the activity across each row.

4.1.3 Mode Decision

On the basis of activity value, the blocks are classified into smooth, complex and intermediate regions. If the activity value is less than threshold (T1), the blocks are classified into smooth region. If the activity is greater than threshold (T2), the blocks are classified into complex region and if the activity lies between thresholds, T1 and T2, the blocks are classified into intermediate region.

IF A(v) < T1
Smooth Filtering
ELSEIF A(v) > T2
Complex Filtering
ELSEIF A(v) > T1 and A(v) < T2
Intermediate Filtering
ENDIF

4.1.4 Smooth Filtering

In the smooth region, we first calculate the offset by taking the difference of two block boundary pixels. If the offset is less than 2QP then blocking artifacts exist. To remove these blocking artifacts we apply smooth filtering, by updating the six boundary pixels, three on each side of block boundary. But if the offset is greater than 2QP, then no filtering is applied as the real edge exists. Smooth filtering across vertical block boundary is as follow:

Offset = p4 - p3
IF offset < (2 * QP)

Update p1, p2 and p3 as

P1 = p1 + offset / 8

P2 = p2 + offset /4

P3 = p3 + offset /2

Update p4, p5 and p6 as

P4 = p4 – offset / 2

P5 = p5 – offset / 4

P6 = p6 – offset / 8

ENDIF

Where QP is the Quantization Parameter of the compressed image and after applying smooth filtering, updated values are clipped within 0 to 255.

4.1.5 Complex Filtering

Complex filtering is applied to remove the blocking effect in detailed region, so weak filtering is applied to retain the sharpness of the image. In this mode, we first take the offset with the difference of the two block boundary pixel values. If the offset is less than QP then complex filtering is applied by updating the four boundary pixel values, two on each side of block boundary. And if the offset is greater than QP, then no filtering is performed as the real edge exits there.

Offset = p4 - p3

IF offset < QP

Update p2 and p3 as

P2 = p2 + offset /8

P3 = p3 + offset /2

Update p4 and p5 as

P4 = p4 – offset / 2

P5 = p5 – offset / 8

ENDIF

After applying complex filtering, updated values are clipped within 0 to 255.

4.1.6 Intermediate Filtering

The intermediate filtering is applied to balance the strong filtering in smooth region and weak filtering in complex region. In this mode, 3x3 low pass filtering is applied on each pixel across the block boundary. Figure, 4.2 shows a 3x3 low pass filter and fig. 4.2 presents two boundary pixels on which the filtering is applied.

The specification of intermediate filtering is given below:

$$S_1 = \sum_{\substack{i=1 \\ i \neq 5}}^{9} \alpha_i p_i$$

$$\alpha_i = \begin{cases} 1, |p_5 - p_i| < Th \\ 0, otherwise \end{cases}$$

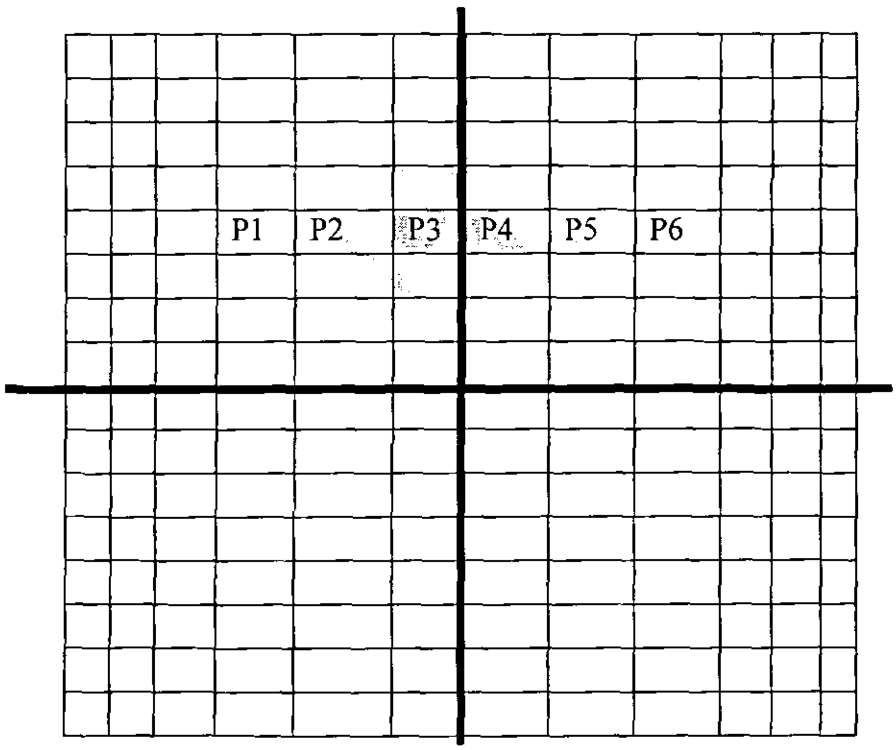
$$S_2 = \sum_{\substack{i=1 \\ i \neq 5}}^{9} \alpha_i$$

$$p_5' = (\lambda p_5 + S_1) \big/ (\lambda + S_2) \qquad \textbf{Eq. 4.4}$$

Where parameter λ controls the extent of smoothing, Pi are the pixel values of the block boundary and Th is set equal to Quantization Parameter (QP) of the image.

P1	P2	P3
P4	P5	P6
P7	P8	P9

Fig. 4.2 3x3 Low Pass Filter



Figure, 4.3 Pixels for Intermediate region

4.1.7 Calculate the SNR Value

We measure the objective quality of the deblocked image with two parameters, SNR (Signal to Noise Ratio) and PSNR (Peak Signal to Noise Ratio).

After the implementation of deblocking filters, SNR value of the deblocked image is calculated by using following formula:

$$P_S = Sum(S)^2$$
$$P_N = Sum(N)^2$$
$$SNR = 10 \times \log_{10}\left(P_S / P_N \right) \quad \textbf{Eq. 4.5}$$

Where S is the deblocked image and N is the difference between the original and deblocked image.

4.1.8 Calculate PSNR Value

PNSR value is used to calculate the objective quality of the reconstructed images. PSNR of the deblocked images are calculated by using following formula:

$$PSNR = 10 \times \log_{10} \frac{1}{M} \sum_{n=1}^{M} 255^2 / (I_1 - I_2)^2 \, dB \quad \textbf{Eq. 4.6}$$

Where M is the number of samples, I1 is the original and I2 is the reconstructed image, respectively.

4.2 Implementation of Scheme II

The second scheme is "low complexity deblocking method for DCT coded video signals", proposed by Tai, S.-C., Chen, Y.-R., Chen, C.-Y. and Chen Y.-H [10]. This method is for DCT coded videos and we implement it, in Matlab 7.0, for ICA decompressed images.

4.2.1 Divide Image into NxN blocks

We first read the blocky image and then divide it into 8x8 block boundaries by using following operations.

block_width = row / 8

block_height = col / 8

blocks = Divide Image into block_width x block_height

Where row is the total number of rows, col is the total number of columns and blocks contains all 8x8 blocks of the respective image.

4.2.2 Activity Calculation

After the image is divided into 8x8 blocks, offset is determined across each block boundary. If the offset is greater than 'Edge-Thresh' value, then a real edge exists and no operation is performed otherwise the activity across each block boundary is calculated. Edge-Thresh is calculated as by following formula:

$$Edge\_Thresh = ((QP*2)+(QP/2)) \qquad \textbf{Eq. 4.7}$$

Activity across a block boundary is calculated by using eqs. 5.1, 5.2 and 5.3.

4.2.3 Mode Decision

On the basis of activity value, the blocks are classified into smooth, complex, intermediate and steep modes. If the activity value is less than threshold (T1) and the offset value is less than 2QP then the block is classified into smooth region. And if the activity value is greater than threshold (T2) and the offset value is less than QP, the block is classified into complex mode. When the activity value is greater than T1 and less than T2, and also the offset value is less than 2QP, the block is known as intermediate region. In smooth and intermediate regions, if the offset value is greater than 2QP and in the complex region, if the offset value is greater than QP, the block is classified into steep mode.

IF A (v) <T1

IF offset < (2 * QP)
Smooth Filtering

ELSE

Steep Mode Filtering

ELSEIF A (v) >T2

IF offset<QP

Complex Filtering

ELSE

Steep Mode Filtering

ELSEIF A (v) > T1 and A (v) < T2

IF offset < (2*QP)

Intermediate Filtering

ELSE

Steep Mode Filtering

ENDIF

4.2.4 Smooth filtering

In smooth mode, we apply smooth filtering by updating the six boundary pixels, three on each side of block boundary.

Offset = p4 - p3

Update p1, p2 and p3 as

P1 = p1 + offset / 8

P2 = p2 + offset /4

P3 = p3 + offset /2

Update p4, p5 and p6 as

P4 = p4 – offset / 2

P5 = p5 – offset / 4

P6 = p6 – offset / 8

After smooth filter, pixel values are clipped within 0 to 255.

4.2.5 Complex Filtering

In complex mode, we apply complex mode filtering by updating two boundary pixels, one in each block boundary, using neighboring pixel values.

Offset = p4 - p3

Update p3:

IF | p2 – p3| < QP

P3 = (p2 + (2 * p3) + P4) /4

ELSE

P3 = p3 + offset / 4

ENDIF

Update p4:

IF | p4 – p5| < QP

P4 = (p3 + (2 * p4) + P5) /4

ELSE

P4 = p4 - offset / 4

ENDIF

4.2.6 Intermediate Filtering

Intermediate filtering is similar to complex mode filtering. The only difference lies for non-referable pixels (referable pixels are those adjacent pixels, where the difference of their gray scale values are not exceeding QP). In intermediate filtering, they are updated by offset/2 while in complex mode filtering they are updated by offset/4.

Offset = p4 - p3

Update p3:
IF | p2 – p3| < QP
P3 = (p2 + (2 * p3) + P4) /4
ELSE
P3 = p3 + offset / 2
ENDIF
Update p4:
IF | p4 – p5| < QP
P4 = (p3 + (2 * p4) + P5) /4
ELSE
P4 = p4 - offset / 2
ENDIF

4.2.7 Steep Mode Filtering

In steep mode filtering, we updated two boundary pixel values, by using offset/4. The exact specification of steep mode filter is as follows:

Offset = p4 - p3

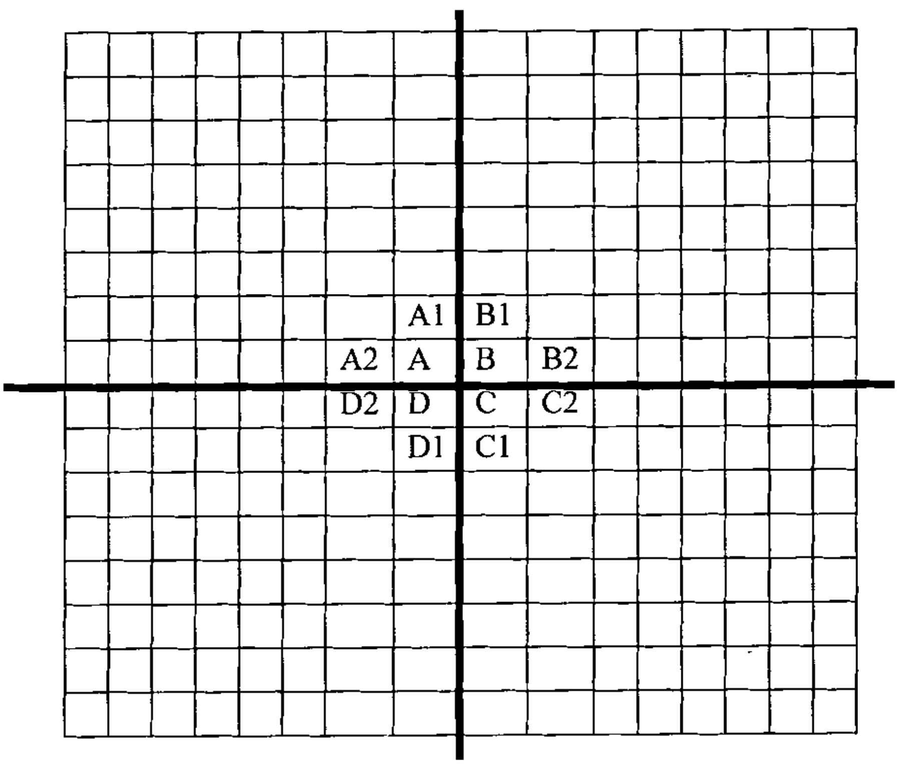
Update p3 and p4:
P3 = p3 + offset / 4

P4 = p4 - offset / 4

ENDIF

4.2.8 Corner Mode Filtering

Corner mode filtering is applied on the corner of four blocks to remove the gaps that appears along non-horizontal and non-vertical edges due to applying above deblocking filters. This is because the above algorithm only considers the block boundary and not the corners. Corner filtering is applied by updating the four corner pixel values, as shown in figure 4.4.



Figure, 4.4 Pixels A, B, C, D are defined for Corner Mode Filtering

Following steps are used to perform corner mode filtering:

IF $|A - C| > Edge\_Thresh$ and $|B - D| \leq Edge\_Thresh$

Update A or C

IF $(|D2 + B1| / 2 - A) \geq 2 * (|B2 + D1| / 2 - C)$

A = (B + 6A + D) / 8

ELSEIF (|B2 + D1| / 2 – C) ≥ 2 * (|D2 +B1| /2 –A)

C = (B + 6C + D) / 8

ENDIF

IF |B – D| > Edge_Thresh and |A – C| ≤ Edge_Thresh

Update B or D

IF (| C2 + A1| / 2 – B) ≥ 2 * (|A2 + C1| / 2 –D)

B = (A + 6B + C) / 8

ELSEIF (|A2 + C1| / 2 – D) ≥ 2 * (|C2 + A1| / 2 – B)

D = (A + 6D + C) /8

Where Edge_Thresh is defined in eq. 5.7

4.2.9 Calculate SNR value

When the deblocking filtering is applied, we display the deblocked image and calculate the SNR value of the original image with the reconstructed images, by using eq. 4.5, to check the objective quality of the images.

4.2.10 Calculate PSNR value

PNSR value of the reconstructed images is calculated by using eq. 4.6, to measure the objective quality.

CHPATER 5
EXPERIMENTAL RESULTS

5 Experimental Results

To test the performance of our deblocking filters, we have chosen three test images, "Lena.bmp", "Baboon.bmp" and "barbara.bmp". Firstly, these images are decompressed by ICA (Independent Component Analysis) Transform Coding with different Quantization Parameter and then we apply deblocking filters on these ICA decompressed blocky images to remove the blocking artifacts by preserving the detail of the images.

5.1Experimental Environment

The objective quality of the decompressed test images are measured by two standard parameters, Signal to Noise Ratio (SNR) and Peak Signal to Noise Ratio (PSNR).

SNR is a power ratio between a signal and background noise. Where SNR of an image is usually defined as the ratio of the mean pixels value to the standard deviation of the pixel values [5]. SNR usually defined in logarithmic decibel scale as

$$SNR = 10 \times \log_{10}\left(P_{Signal} \Big/ P_{Noise} \right) \tag{5.1}$$

PSNR is Peak signal to Noise Ratio which is defined as

$$PSNR = 10 \times \log_{10} \frac{1}{M} \sum_{n=1}^{M} 255^2 / \left(o_n - r_n\right)^2 dB \tag{5.2}$$

Where M is the number of samples and o_n and r_n are the gray levels of the original and reconstructed images, respectively.

Our deblocking filters depend on some predetermined parameters. In section 4.1.2, for classification of blocks into different regions, we calculated the activity across different block boundaries. The threshold S is set to 2 so that it represents quite a small difference between neighboring pixels and it is appropriate for strong filtering. After the activity calculation, two thresholds (T_1, T_2) are used to classify the 8x8 blocks. In scheme1, T_1 is set to 3 for smooth regions and T_2 is set to 4 to preserve the sharpness of the image while in scheme2 T_1 is set to 2 and T_2 is set to 3 for the same purpose.

In section 4.1.6, in intermediate filtering, we set the value of $\lambda = 8$ as it controls the extent of smoothing.

5.2 Experimental Results of "Lena.bmp" Image

We apply deblock filtering of scheme 1 and scheme 2, on "Lena.bmp" image, with different QP. Some of the results are shown below:

The test image "Lena" is compressed by ICA transforms coding at 21 Quantization Parameter (512 x 512) and then we apply deblocking filter of scheme1 and scheme2, to remove the blocking artifacts. Figs. 5.1 (a) and (b) show the original Lena image and decompressed Lena image with ICA transform coding at 21 QP. Objectionable blocking artifacts, especially in the flat regions such as on the cheek, shoulder and on the hat of the girl are observable. In complex region, blocking artifacts are also visible, particularly on the right most side of the picture.

Figs. 5.1(c) and (d) present the result of horizontal and vertical filtering on the image in Fig.. 5.1(b) using scheme1. Also, the scheme2 horizontal and vertical filtering is depicted in Fig. 5.1 (e) and (f) show an improvement in visual quality. According to scheme2, after horizontal and vertical filtering, we applied corner filtering. Fig. 5.1 (g) presents the results of corner filtering on Fig. 5.1 (f). Both the results of scheme1 and scheme2, as shown in Figs. 5.1 (d) and (g) have established successful elimination of blocking artifacts especially in the area of cheek and hat of the girl. Deblocking filter also remove the blocking artifacts in the right most side of the figure while maintaining the edge details.

Figs. 5.2 (a) and (b) show the original Lena image (512 x 512) and blocky ICA decompressed image with QP = 19 respectively. Figs. 5.2 (c) and (d) shows the result of applying deblock filtering of scheme1 and scheme2 on fig 5.2 (b). It is clear from figures that the blocking artifacts are removed from the right most corner and from the hat of the girl.

Figs. 5.3 (a) and (b) show the original Lena image (512 x 512) and blocky ICA decompressed image with QP = 17 respectively. Figs. 5.2 (c) and (d) shows the result of applying deblock filtering of scheme1 and scheme2 on fig 5.2 (b), to remove the blocking artifacts.

SNR and PSNR comparison of Lena images with different QP are shown in Table 5.1 and 5.2.

a) Original Image (512 x 512)

b) Decompressed Image with ICA Transform Coding, at QP=21

c) Image after Horizontal Deblock Filtering by using Scheme 1

d) Image after Horizontal and Vertical Deblock Filtering by using Scheme 1 (SNR = 0.262852, PSNR = 0.25079)

e) Image after Horizontal Deblock Filtering by using Scheme 2

f) Image after Horizontal and Vertical Deblock Filtering by using Scheme2

g) Image after Corner Filtering using Scheme 2 (SNR = 0.225656, PSNR = 0.355978)

Figure 5.1 Deblocking results of Lena Image at QP=21

a) Original Image (512 x 512)

b) Decompressed Image with ICA Transform Coding, at Q = 19

c) Image after Horizontal and Vertical Deblock Filtering by using Scheme 1 (SNR = 0.242744, PSNR = 0.228506)

d)Image after Horizontal, Vertical and Corner Deblock Filtering by using Scheme 2 (SNR = 0.216063, PSNR = 0.326585)

Figure 5.2 Deblocking results of Lena Image at QP=19

a) Original Image (512 x 512)

b) Decompressed Image with ICA Transform Coding, at Q = 17

c) Image after Horizontal and Vertical Deblock Filtering by using Scheme 1 (SNR = 0.21708, PSNR = 0.200198)

d)Image after Horizontal, Vertical and Corner Deblock Filtering by using Scheme 2 (SNR = 0.206238, PSNR = 0.28115)

Figure 5.3 Deblocking results of Lena Image at QP = 17

Table 5.1 and 5.2 presents the results of SNR and PSNR values with ICA decompressed Lena images at different Quantization Parameter along with images after applying deblock filtering using Scheme1 [11] and Scheme2 [10], respectively.

Original Image	Blocky Image	QP	SNR Values			SNR Gain	
			No Processing	Scheme 1	Scheme 2	Scheme 1	Scheme 2
Lena	101q_Lena	21	9.68536	9.94821	9.91101	0.262852	0.225656
	91q_Lena	19	9.84296	10.0857	10.059	0.242744	0.216063
	81q_Lena	17	9.97664	10.1937	10.1829	0.21708	0.206238
	71q_Lena	15	10.118	10.3088	10.3212	0.190799	0.20324
	61q_Lena	13	10.3173	10.4809	10.4979	0.163607	0.180653
	51q_Lena	11	10.4089	10.5355	10.5821	0.126576	0.173214

Table 5.1 SNR Values and Gain for Lena

Original Image	Blocky Image	QP	PSNR Values			PSNR Gain	
			No Processing	Scheme 1	Scheme 2	Scheme 1	Scheme 2
Lena	101q_Lena	21	29.0397	29.2905	29.3957	0.25079	0.355978
	91q_Lena	19	29.2806	29.5091	29.6072	0.228506	0.326585
	81q_Lena	17	29.4394	29.6396	29.7206	0.200198	0.281155
	71q_Lena	15	29.5901	29.7611	29.8563	0.170947	0.266192
	61q_Lena	13	29.6691	29.8105	29.8807	0.141411	0.211568
	51q_Lena	11	29.7008	29.8074	29.8452	0.106593	0.144444

Table 5.2 PSNR Values and Gain for Lena

5.3 Experimental Results of "Baboon.bmp" Image

The other test image is "Baboon.bmp", 512x512, that is compressed by ICA transform coding and then we apply deblock filtering of scheme 1 and scheme 2, with different QP. Later on, detailed results are shown in tables.

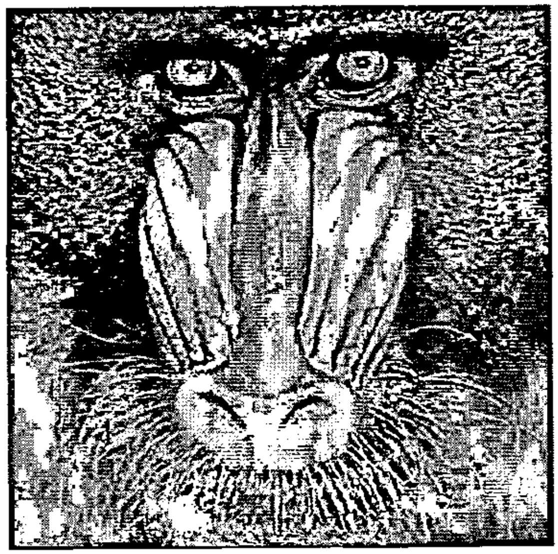
Figures, 5.4 (a) and (b) shows the original Baboon (512 x 512) and blocky ICA transform decoded image with QP = 21. In fig. 5.4 (b), blocking artifacts are clearly visible in smooth region especially at nose, at lower right and lower left boundary of the image. In complex region, like area below the eyes, blocking artifacts are also present. Figs. 5.4 (c) and (d) shows the experimental results of apply deblocking filter of scheme1, first across horizontal block boundary and then across vertical block boundary on fig. 5.4 (b), respectively. Fig. 5.4 (e) shows the result of applying scheme2 horizontal delbock filtering on fig. 5.4 (b), where fig. 5.4 (f) shows the result of scheme2 vertical deblock filtering on fig. 5.4 (e). In scheme2, after horizontal and vertical filtering, corner filtering is applied. As shown in fig. 5.4 (g) corner filtering is applied on fig. 5.4 (e). Both results of figs. 5.4 (d) and (g) reveal that the subjective and objective quality of the image 5.4 (b) is improved, by removing the blocking effect at nose, at lower right and left block boundary and under eyes region.

Fig. 5.5 (a) shows the original Baboon image (512x512). Fig. 5.5 (b) shows, image after applying ICA transform coding using QP =19 on fig. 5.5 (a) which have clear blocking artifacts. To remove these blocking artifacts we apply deblock filtering of scheme1 and scheme2 on fig. 5.5 (b), as shown in figs. 5.5 (c) and 5.5 (d), respectively. We can see that in figs. 5.5 (c) and (d) blocking effect is well reduced after performing our deblock filtering.

We also take ICA decompressed baboon image with 17 QP as our test image, as shown in fig. 5.6 (b) and then we apply our deblock filtering on it. Fig. 5.6 (c) shows the results of applying scheme1 deblock filtering and fig. 5.6 (d) shows the results of scheme 2 deblock filtering. Figs. 5.6 (c) and (d) show an improvement in the visual quality when compared with fig. 5.6 (b).

SNR and PSNR comparison for "Baboon.bmp" are shown in Tables 5.3 and 5.4, with different QP.

a) Original Image (512 x 512)



b) Decompressed Image with ICA Transform Coding, at QP=21

c) Image after Horizontal Deblock Filtering by using Scheme 1

d) Image after Horizontal and Vertical Deblock Filtering by using Scheme 1 (SNR = 0.037034, PSNR = 0.0206414)

e) Image after Horizontal Deblock Filtering by using Scheme 2

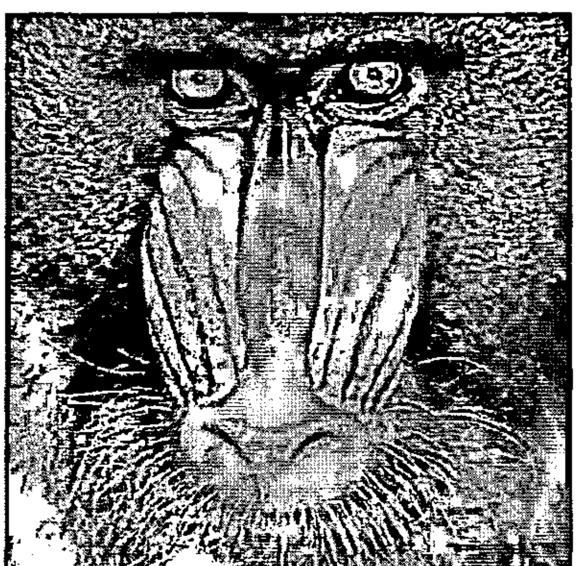
f) Image after Horizontal and Vertical Deblock Filtering by using Scheme2

g) Image after Corner Filtering using Scheme 2 (SNR = 0.0197559, PSNR = 0.0482971)

Figure 5.4 Deblocking results of Baboon Image at QP=21

a) Original Image (512 x 512)

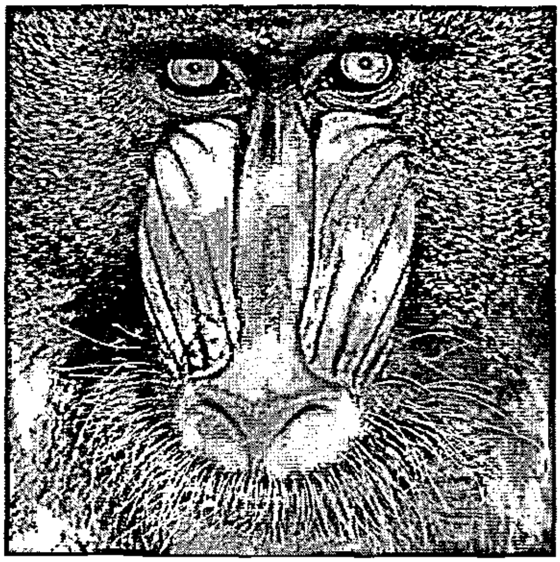
b) Decompressed Image with ICA Transform Coding, at Q = 19



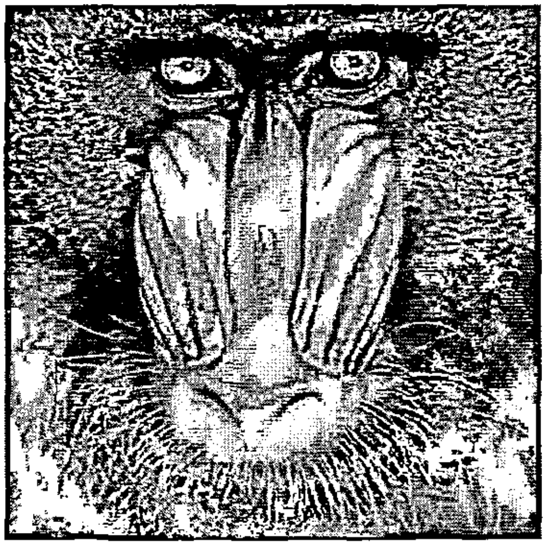
c) Image after Horizontal and Vertical Deblock Filtering by using Scheme 1(SNR = 0.035616, PSNR = 0.0183121)

d)Image after Horizontal, Vertical and Corner Deblock Filtering by using Scheme 2 (SNR = 0.0202567, PSNR = 0.0400247)

Figure 5.5 Deblocking results of Baboon Image at QP=19



a) Original Image (512 x 512)



b) Decompressed Image with ICA Transform Coding, at Q = 17

c) Image after Horizontal and Vertical Deblock Filtering by using Scheme 1 (SNR 0.0295306, PSNR = 0.0140834)

d)Image after Horizontal, Vertical and Corner Deblock Filtering by using Scheme 2 (SNR = 0.0171698, PSNR = 0.0403123)

Figure 5.3 Deblocking results of Lena Image at QP = 17

Table 5.3 and 5.4 presents the results of SNR and PSNR values with ICA decompressed Baboon images at different Quantization Parameter along with images after applying deblock filtering using Scheme1 [11] and Scheme2 [10].

Original Image	Blocky Image	QP	SNR Values			SNR Gain	
			No Processing	Scheme 1	Scheme 2	Scheme 1	Scheme 2
Baboon	101q_Baboon	21	5.99662	6.03365	6.01638	0.037034	0.0197559
	91q_ Baboon	19	6.03778	6.07339	6.05804	0.0356116	0.0202567
	81q_ Baboon	17	6.05526	6.08479	6.07243	0.0295306	0.0171698
	71q_ Baboon	15	6.08355	6.10806	6.09541	0.0245065	0.0118592
	61q_ Baboon	13	6.0942	6.1129	6.10677	0.0186998	0.0125716
	51q_ Baboon	11	6.11021	6.12646	6.12196	0.0162461	0.0117499

Table 6.3 SNR Values and Gain for Baboon

Original Image	Blocky Image	QP	PSNR Values			PSNR Gain	
			No Processing	Scheme 1	Scheme 2	Scheme 1	Scheme 2
Baboon	101q_Baboon	21	22.6272	22.6478	22.6755	0.0206414	0.0482971
	91q_ Baboon	19	22.6341	22.6524	22.6741	0.0183121	0.0400247
	81q_ Baboon	17	22.6325	22.6466	22.6728	0.0140834	0.0403123
	71q_ Baboon	15	22.6342	22.6477	22.67	0.0134575	0.0357672
	61q_ Baboon	13	22.6478	22.6597	22.6769	0.0119074	0.029012
	51q_ Baboon	11	22.7217	22.7311	22.7232	0.00934597	0.00145764

Table 6.4 PSNR Values and Gain for Baboon

5.4 Experimental Results of "Barbara.bmp" Image

Another test image is "Barbara.bmp" (512 x 512). We also take its ICA decompressed images with different QP to and then deblocking filters are applied on it.

Figs. 5.7 (a) and (b) shows the original Barbara image (512x512) and ICA decompressed Barbara image with QP = 21. In fig. 5.7 (b), after ICA compression, blocking artifacts are clearly visible especially in smooth background region, at the face and arms of the girl. To remove these blocking artifacts we apply deblock filtering. Figs. 5.7 (c) and (d) show the horizontal filtering on fig. 5.7(b) and vertical filtering on fig. 5.7 (c) by using scheme1, respectively. Fig. 5.7 (e) shows horizontal filtering by using scheme 2 on fig. 5.7(b) and fig. 5.7 (f) shows vertical filtering by using scheme2 on fig. 5.7 (e). Corner filtering is applied on fig. 5.7 (f) as result shown in fig. 5.7 (g), by using scheme2. The resultant figs. 5.7 (d) and (g) show that the blocking artifacts are well reduced from smooth background region, from the arms and face of the girl when compared with fig. 5.7 (b).

Barbara image decompressed with 19 QP is also taken to test the deblocking filters, as shown in fig. 5.8 (b). Scheme 1 and scheme 2 deblock filtering is applied on fig. 5.8 (b), as shown in figs. 5.8(c) and (d), respectively. The results reveal that the deblocking filters successfully eliminate the blocking artifacts and improve the visual quality.

Fig. 5.9 (a) shows the original Barbara image (512 x 512). Fig. 5.9 (b) shows ICA decompressed image with 17 QP, which have clear blocking artifacts. To remove these blocking artifacts we apply deblock filtering of scheme 1 and scheme 2 as shown in figs. 5.9 (c) and (d). These resultant figures when compared with fig. 5.9 (b) show an improvement in visual quality.

SNR and PSNR comparison of Barbara images are shown in Table 5.5 and 5.5 with different QP.

a) Original Image (512 x 512)

b) Decompressed Image with ICA Transform Coding, at QP=21

c) Image after Horizontal Deblock Filtering by using Scheme 1

d) Image after Horizontal and Vertical Deblock Filtering by using Scheme 1 (SNR = 0.107416, PSNR = 0.0490489)

e) Image after Horizontal Deblock Filtering by using Scheme 2

f) Image after Horizontal and Vertical Deblock Filtering by using Scheme2

g) Image after Corner Filtering using Scheme 2
(SNR = 0.101314, PSNR = 0.0871843)

Figure 5.7 Deblocking results of Barbara Image at QP=21

a) Original Image (512 x 512)

b) Decompressed Image with ICA Transform Coding, at Q = 19

c) Image after Horizontal and Vertical Deblock Filtering by using Scheme 1 (SNR = 0.104718, PSNR = 0.041551)

d)Image after Horizontal, Vertical and Corner Deblock Filtering by using Scheme 2 (SNR = 0.102851, PSNR = 0.0786462)

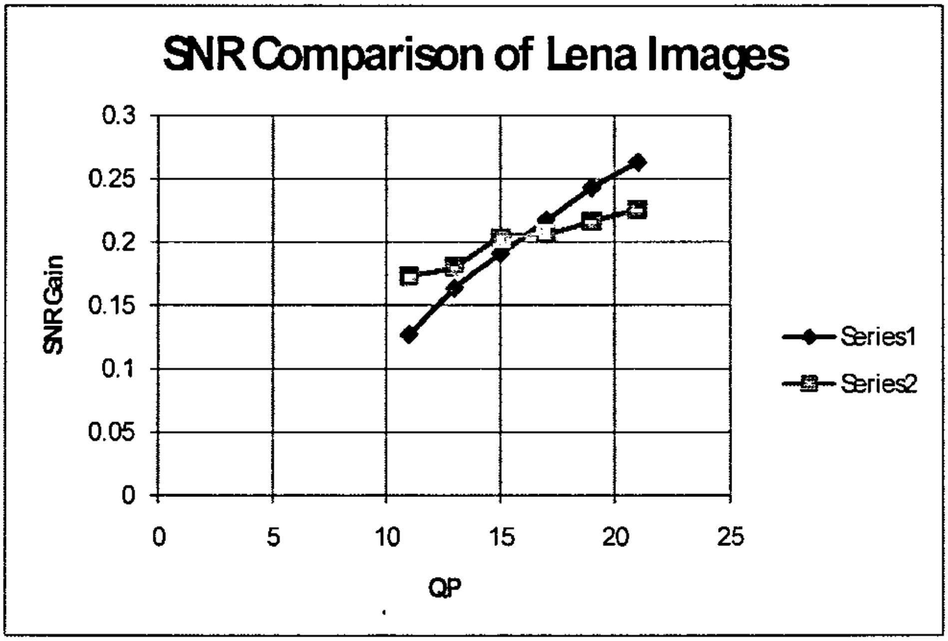
Figure 5.8 Deblocking results of Barbara Image at QP=19

5.5 Comparison of SNR Values

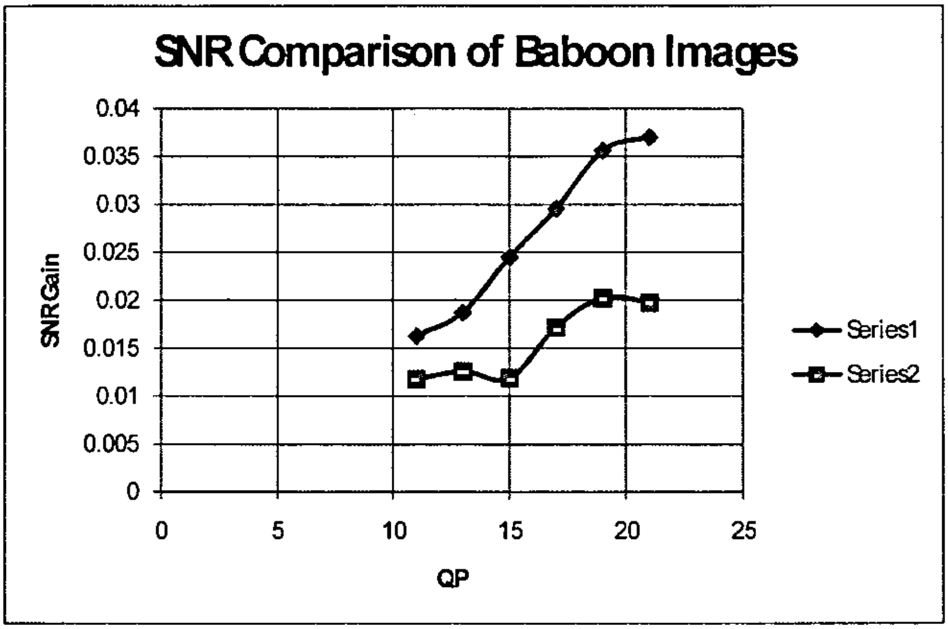
Graph 5.1 shows the SNR values comparison of ICA decoded "Lena.bmp" images with images after deblock filtering at different QP. In the graph, series 1 show the results of scheme 1 and series 2 show the results of scheme 2 of deblock filtering. From the comparison it is noticed that scheme 1 give better SNR at higher QP than scheme 2. While scheme 2 is more stable in its resultant values than scheme 1 and better performing at low QP values.

Graph 5.2 shows SNR gain comparison of ICA decoded "Baboon.bmp" images at different QP with deblocked images. It is clear from the graph that the scheme 1 give better SNR than the scheme 2, on all the values of QP where series 1 show scheme 1 and series 2 show scheme 2.

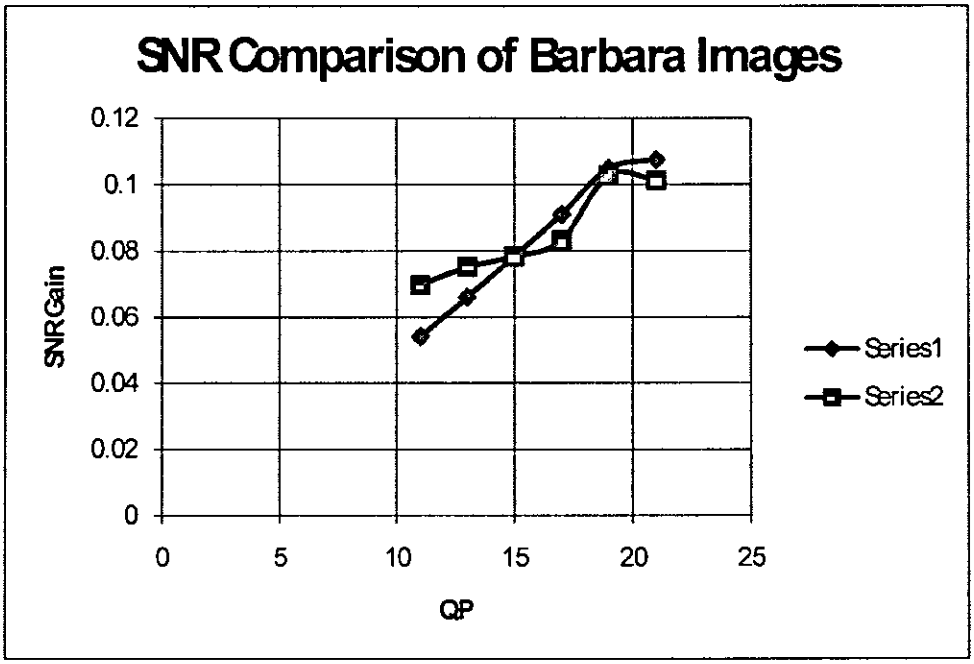
Graph 5.3 shows SNR gain comparison of Barbara images at different QP. From the graph we can say that the scheme 1 give better SNR at higher QP of decoded images and scheme 2 give better SNR at lower QP of decompressed images.



Graph 5.1 SNR Gain for Lena



Graph 5.2 SNR Gain for Baboon



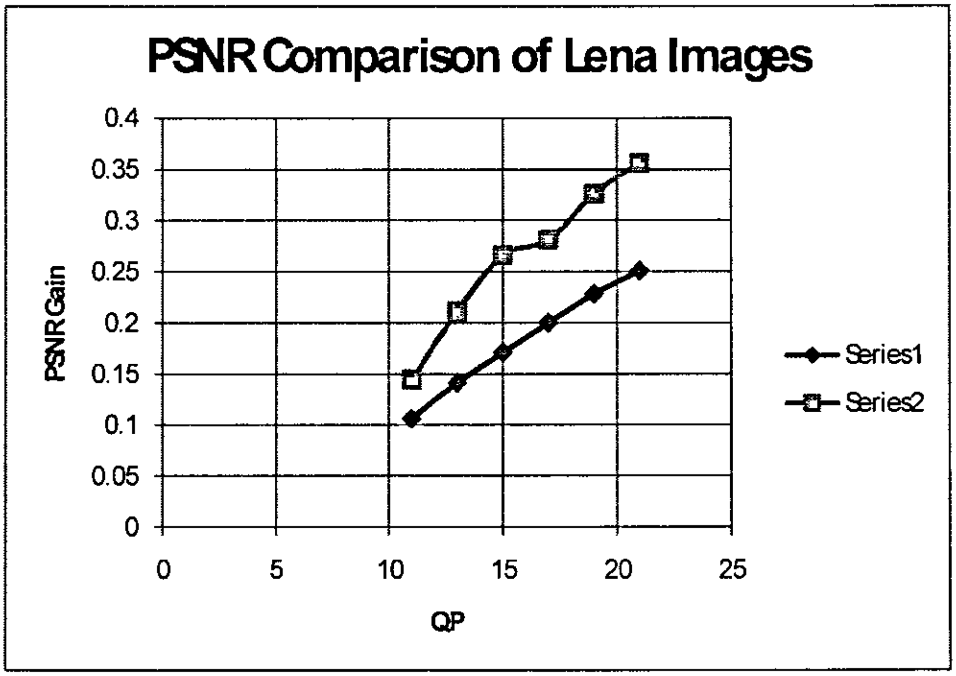
Graph 5.3 SNR Gain for Barbara

5.6 Comparison of PSNR Values

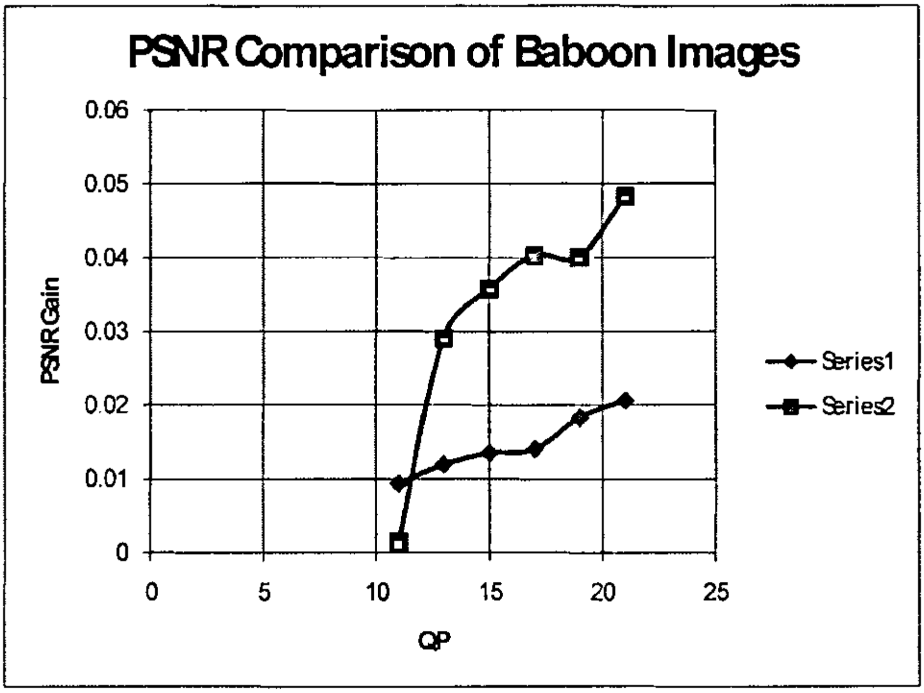
Graph 5.4 shows the PSNR values comparison of ICA decoded "Lena.bmp" images with images after deblock filtering at different QP. In the graph, series 1 show the results of scheme 1 and series 2 show the results of scheme 2 of deblock filtering. From the comparison it is noticed that scheme 2 give better PSNR values on all the images with different QP than scheme 1.

Graph 5.5 shows PSNR gain comparison of ICA decoded "Baboon.bmp" images at different QP with deblocked images. We can see that the scheme 2 give better PSNR values at higher QP while scheme 1 give better PSNR at lower QP, where series 1 show the results of scheme 1 and series 2 show the results of scheme 2.

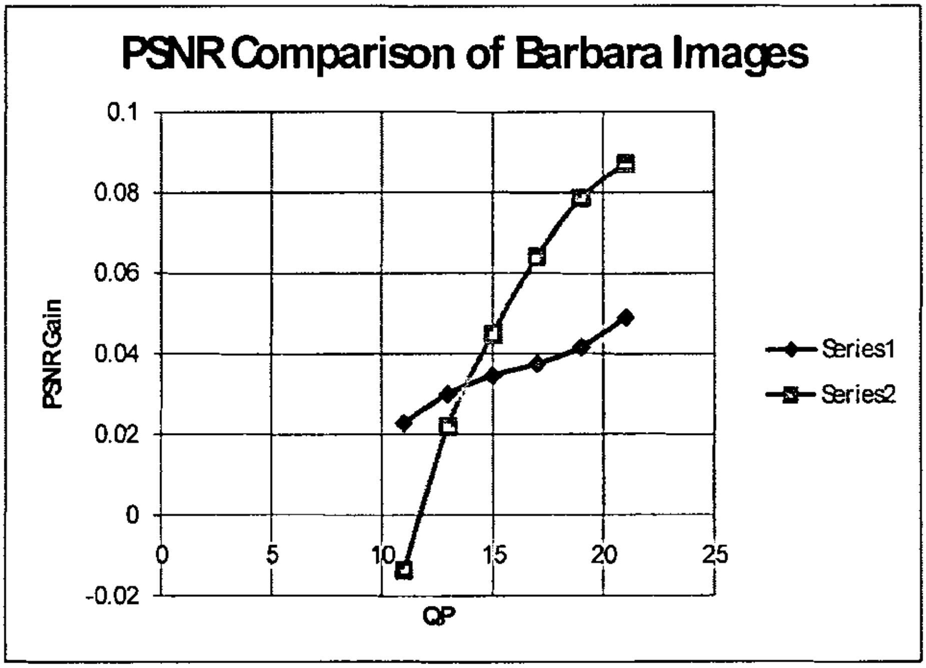
Graph 5.6 shows PSNR gain comparison of Barbara images at different QP. From the graph we can say that the scheme 2 give better PSNR values at higher QP while at lower QP in scheme 2, PSNR gain goes to negative value (negative value means no improvement in image after applying deblock filtering).



Graph 5.4 PSNR Gain for Lena



Graph 5.5 PSNR Gain for Baboon



Graph 5.6 PSNR Gain for Barbara

CHPATER 6
CONCLUSION AND FUTURE WORK

6 Conclusion and Future Work

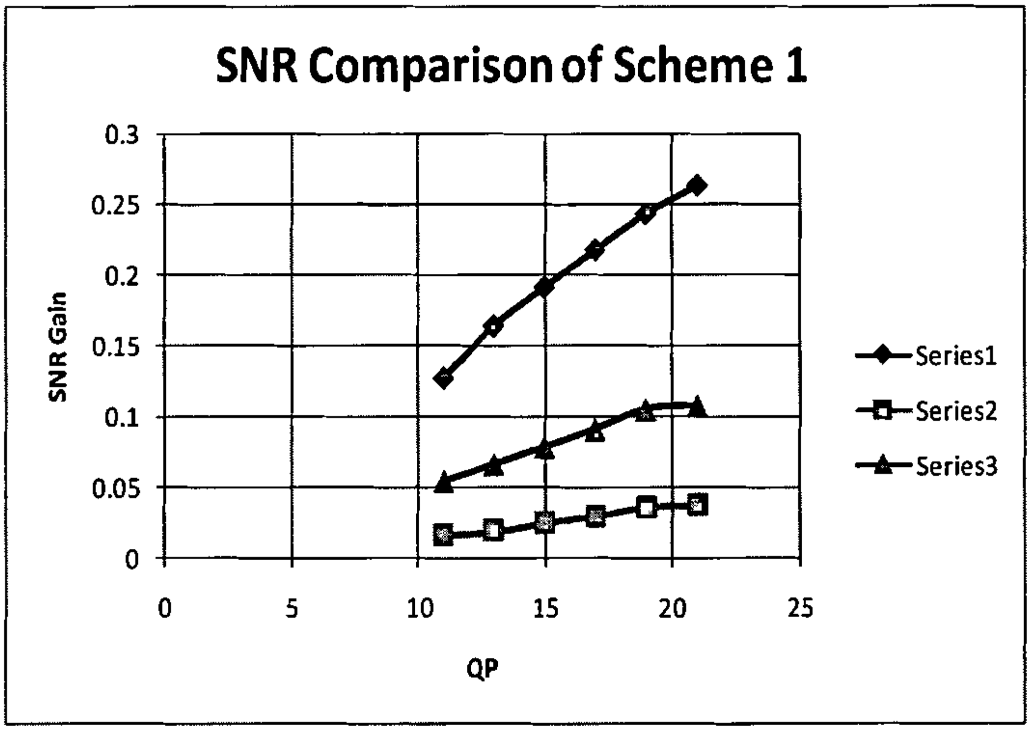
Based upon the results following conclusion is drawn and suggestions for improvements and additions are also given.

6.1 Conclusion

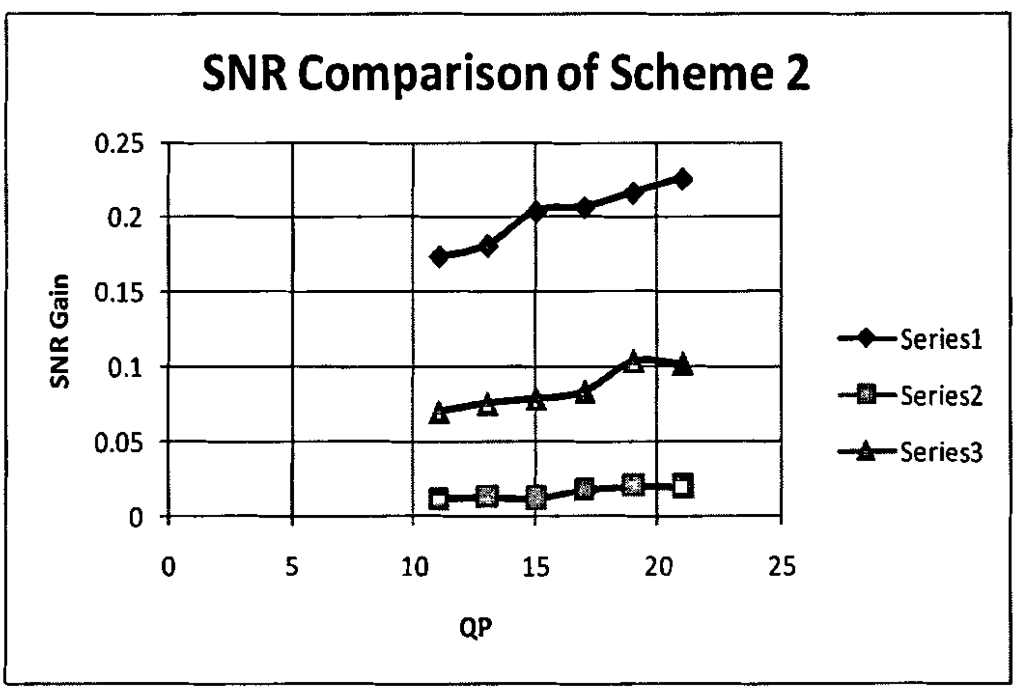
Many deblocking filters have been studied using different techniques. Two techniques [10] and [11] are selected as base techniques to remove the blocking artifacts on ICA decoded images. These techniques proposed deblocking filters for DCT and MPEG-4 coded video signal, respectively. We implemented these deblocking filters for ICA transforms decoded images using different QP. In scheme 1, image is classified into three different modes, smooth, complex and intermediate, using local pixel values of the region. While in scheme 2, five different modes of filtering are used, smooth, complex, intermediate, steep and corner mode, to remove the blocking artifacts in respective regions.

It is clearly observed from the experimental results that both methods improve visual quality of ICA decoded images by removing the blocking artifacts. Objective quality of the images is measured by PSNR and SNR standards, as shown in tables 5.1 to 5.6 and in graphs 5.1 to 5.6. It is noteworthy that PSNR of scheme2 has provided better results than scheme1 but PSNR take only the peaks of the images and used MSE (Mean Square Error) which is not good enough for image quality assessment. In graphs 6.1 and 6.2, the comparison of SNR gain is done to conclude that at high compression ratio or at a low bit rate, the results of scheme 1 is better than that of scheme 2.

In graph 6.1 and 6.2, Series 1 show SNR gain for Lena, Series 2 show the SNR gain for Barbara and Series 3 show the SNR gain for Baboon images.



Graph 6.1 SNR Comparison of Scheme 1



Graph 6.2 SNR Comparison of Scheme 2

6.2 Future Work

- Blocking artifacts on ICA decoded images can be further removed by applying different other deblocking filters.
- In this thesis only gray scale images are dealt. Deblocking filters for colored ICA decoded images can be developed. Firstly, RGB model is converted into HSI model as RGB is independent plane and HSI model is better to use.
- The scope of our work is limited for ICA decoded images, but it can be further enhanced for ICA decoded videos.
- Future work also includes optimization of the threshold and λ values.

APPENDIX A
BIBLIOGRAPHY AND REFRENCES

Bibliography and References

[1] Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing*, 2nd Edition, Prentice Hall.

[2] Iain E. G. Richardson, *H.264 and MPEG-4 Video Compression, Video Coding for Next-generation Multimedia*, John Wiley & Sons Ltd, 2003.

[3] Ling Guan, Sun-Yuan Kung, Jan Larsen, *Multimedia Image and Video Processing*, CRC Press LLC, 2001.

[4] Iain E. G. Richardson, *Video Codec Design, Developing Image and Video Compression Systems*, John Wiley & Sons Ltd, 2002.

[5] http://en.wikipedia.org/

[6] http://www.cis.hut.fi/projects/ica/icademo/

[7] Ling Shao, Ihor Kirenko, "Coding Artifact Reduction Based on Local Entropy Analysis", IEEE Transactions on Consumer Electronics, Vol. 53, pp. 691-696, May 2007.

[8] WU Jing, YE Xiu-qing, GU Wei-kang, "An adaptive fuzzy filter for coding artifacts removal in video and image", Zhejiang University Press, co-published with Springer-Verlag GmbH, Vol. 8, pp. 841-848, June 2007.

[9] Hantao Liu and Ingrid Heynderickx, "A Simplified Human Vision Model Applied to a Blocking Artifact Metric", Springer Berlin / Heidelberg, Vol. 4673/2007, pp. 334-341, August 2007.

[10] S.-C. Tai, Y.-R. Chen, C.-Y. Chen and Y.-H. Chen, "Low complexity deblocking method for DCT coded video signals", IEE Proceedings on Vision, Image and Signal Processing, Vol. 153, pp. 46- 56, February 2006.

[11] Shen-Chuan Tai, Yen-Yu Chen, and Shin-Feng Sheu, "Deblocking filter for low bit rate MPEG-4 video", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 15, pp. 733- 741, June 2005.

[12] Peter List, Anthony Joch, Jani Lainema, Gisle Bjøntegaard, and Marta Karczewicz, "Adaptive Deblocking Filter", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 13, pp. 614-619, July 2003.

[13] Ismaeil R. Ismaeil and Rabab K. Ward, "Removal of DCT blocking artifact using DC and AC filtering", 2003 IEEE Pacific Rim Conference on Communications, Computers and signal Processing, 2003. PACRIM. Vol. 1, pp. 229- 232, August 2003.

[14] Artur J. Ferreira and M'ario A. T. Figueiredo, "Class adapted image compression using Independent Component Analysis", 2003 International Conference on Image Processing, ICIP 2003, Vol. 1, pp. I- 625-8, September 2003.

[15] Artur J. Ferreira and M'ario A. T. Figueiredo, "Image compression using orthogonalized independent component bases", 2003 IEEE 13th Workshop on Neural Networks for Signal Processing, NNSP'03 , pp. 689 – 698, September 2003.

[16] Sung Deuk Kim, Jaeyoun Yi, Hyun Mun Kim, and Jong Beom Ra, "A deblocking filter with two separate modes in block based video coding", IEEE Transations on circuits and systems for Video Technology, Vol. 9, pp. 156 – 160, February 1999.

[17] G.A. Triantafyllidis, M. Varnuska, D. Sampson, D. Tzovaras, M.G. Strintzis, "An efficient algorithm for the enhancement of JPEG-coded images", pp. 529 – 534, April 2003.

[18] Gulistan Raja and Muhammad Javed Mirza, "In-loop Deblocking Filter for H.264/AVC Video", Proceedings of the 5th WSEAS International Conference on Signal Processing, Robotics and Automation, pp. 235-240 , 2006 .

[19] Ken Kin-Hung Lam, "Optimizing the Deblocking Algorithm for H.264 Decoder Implementation".

[20] Artur J. Ferreira and M'ario A. T. Figueiredo, "On the use of Independent Component Analysis for image compression", Science Direct, Signal Processing: Image Communicaiton, Vol. 21, pp. 378-389 , June2006.

[21] Sukhwinder Singh, Vinod Kumar , H.K. Verma, "Reduction of blocking artifacts in JPEG compressed images", ACM Portal, Digital Signal Processing Vol. 17 , pp. 225-243, January 2007.

APPENDIX B
GLOSSARY

Glossary

Artifact Visual distortion in an image

BMP bitmapped picture

Block Region of macroblock (8×8 or 4×4) for transform purposes

Blocking Square or rectangular distortion areas in an image

BDCT Block-Based Discrete Cosine Transform

Chrominance Color difference component

CODEC COder / DECoder pair

DCT Discrete Cosine Transform

GIF Graphics Interchange Format

HVS Human Visual System, the system by which humans perceive and interpret visual images

ICA Independent Component Analysis
PEG Joint Photographic Experts Group, a committee of ISO (also an image coding standard)

JPEG-2000 An image coding standard

Loop Filter Spatial filter placed within encoding or decoding feedback loop

MPEG Motion Picture Experts Group, a committee of ISO/IEC

MPEG-1 A video coding standard

MPEG-2 A video coding standard

MPEG-4 A video coding standard

Objective Quality Visual quality measured by algorithm(s)

PNG Portable Network Graphics

PQS Picture Quality Scale

PSNR Peak Signal to Noise Ratio, an objective quality measure

Quantize Reduce the precision of a scalar or vector quantity

QCIF Quarter Common Intermediate Format

QP Quantization Parameter

SNR Signal to Noise Ratio

Subjective Quality Visual quality as perceived by human observer(s)