CHAPTER 1

INTRODUCTION

1.1 Introduction

Image compression is to reduce the redundancy of image and to transmit and to store data in a efficient form. Compressing an image is significantly different than compressing raw binary data. Image compression is of two types- lossy and lossless image compression. Lossy image compression is a type of compression where there exists a loss of data during the compression. While in lossless image compression, the image is compressed without any loss of data visually. Compression of an image is done where the redundant pixels present in the image are removed and compressed separately to reduce the size of the image. The popularity of visually lossless for JPEG2000 is increased because of lossless encoding mode, high compression efficiency, high visual quality, absence of block based artifacts, scalability, and error resiliency.

Stereoscopic 3D imaging has been applied in diverse fields such as aerial stereo photography, stereoscopic surgery, and digital cinema. Accordingly, it has received considerable attention over the last few decades. Recently, consumers are viewing many different types of stereoscopic content in television and gaming applications due to inexpensive consumer-grade 3D displays becoming widely available. Stereoscopic imaging requires double the amount of data compared to 2D images. Therefore, efficient data compression techniques are especially critical in these applications.

The lossless image is achieved by using the lossless compression technique. In lossless compression JPEG2000 is the coding technique developed scalability and editability feature than JPEG. In this technique Discrete Wavelet Transform, Wavelet compression converts image into a series of wavelets that can be stored more efficiently than blocks.

The goal of the JPEG-2000 is to develop a new image compression system for all kinds of still images (bi-level, grayscale, color, multi-component) with different characteristics (continuous-tone, text, cartoon, medical, etc), for different imaging models (client/server, real

time transmission, image library archival, limited buffer and bandwidth resources, etc) and preferably with in a unified system.

The Visually lossless compression offers the potential for using higher compression levels without noticeable artifacts. The human visual system (HVS) is the end user of most image information. Therefore, any imaging system that reflects human image processing needs should be designed with the characteristics and behavior of the HVS taken into consideration. This will ensure that only information that is relevant for the HVS is stored. One important property of the HVS is that the human eyes selectively understand the image by frequency and orientation. The sensitivity of human eyes to frequencies and orientations is represented by the contrast sensitivity function.

With consumer-grade 3D displays becoming widely available, stereoscopic 3D imaging has received increased attention over the last few years. Since a left eye and a right eye image are contained in a stereo pair, stereoscopic imaging requires double the amount of data compared to 2D images. Therefore, efficient data compression techniques are especially critical in these applications.

1.2 Objective of the Project

The main aim of the project is to develop a methodology for visually lossless JPEG2000 compression of monochrome stereo images. The primary goal of this project is the Compression of Monochrome stereoscopic 3D Images with minimal quantization error. Generally, the sensitivity of the human visual system to quantization distortion produced by the JPEG2000 image compression standard is investigated and a visual distortion model is then proposed and incorporated into a JPEG2000. To dart a scary number of measurements in the course of subjective experiments, a model for visibility thresholds is established. The left image and right image of a stereo pair are then firmed together by means of the visibility thresholds obtained. The VTs are obtained by quantization distortion that is based on distribution of wavelet coefficients and the dead-zone quantizer.

1.3 Organization of the report

In chapter 1, introduction of the project, objectives of the project are presented. Chapter 2 discusses about the literature survey. Chapter 3 reveals the overview of the project. Chapter 4 discuss the simulation results. The Conclusion and the future work are given in Chapter 5.

CHAPTER 2

LITERATURE SURVEY

2.1 Image Compression

Image compression and decompression reduce the data content necessary to describe the image. Most of the images contain lot of redundant information, compression removes all the redundancies. Because of the compression the size is reduced, so efficiently stored or transported. The compressed image is decompressed when displayed. Image compression schemes can be broadly classified into two categories, there are

a) Lossless Image Compression

b) Lossy Image Compression

a) Lossless Image Compression

Lossless image compression schemes exploit redundancies without incurring any loss of data. Thus, the data stream prior to encoding and after decoding is exactly the same and no distortion in the reconstruction quality is observed. Lossless image compression is therefore exactly reversible. Lossless compression is achieved through the exploitation of statistical redundancy. For example, if we transform the image into a string of symbols prior to encoding and then assign shorter code words to more frequently occurring symbols and longer code words to less frequently occurring symbols, then we can achieve compression and at the same time, the encoding process can be exactly reversed during decoding, since there is an one-to-one mapping between the symbols and their codes.

b) Lossy Image Compression

Lossy image compression schemes incur loss of data and hence suffer a loss of quality in reconstruction. Like lossless image compression, the image is first transformed into a string of symbols, which are quantized to a discrete set of allowable levels. In this scheme, it is possible to achieve significant data compression, but quantization being a many-to-one mapping is irreversible and exact reconstruction is never possible. The loss in reconstruction quality is acceptable to our visual sensitivity, we may accept this scheme in the interest of achieving very significant degree of compression.

2.2 Compressed Image File Formats

There are hundreds of file formats available, mostly using file formats are PNG, JPEG, and GIF formats used. These graphic formats are separated into two main families of graphics there are raster and vector. In addition to straight image formats, metafile formats are portable formats which can include both raster and vector information. In raster images, image file size is definitely correlated to the number of pixels in an image and the bits per pixel of the image. Images can be compressed in many ways.

Compression uses an algorithm that stores an exact representation of the original image in a smaller number of bytes that can be expanded back to its uncompressed form with a corresponding decompression algorithm. Considering different compressions, it is common for two images of the same number of pixels and color depth to have a very different compressed file size. Taking into the consideration exactly the same compression, number of pixels, and color depth for two images, different graphical complexity of the original images may also result in a very different file sizes after compression due to the nature of compression algorithms.

With some compression formats, images that are less complex may result in a smaller compressed file sizes. This feature sometimes results in a smaller file size for some lossless formats than lossy formats. For example, simple images may be lossless compressed into a GIF or PNG format and result in a smaller file size than a lossy JPEG format. Vector images, unlike raster images can be any dimension independent of file size. File size increases only with the addition of more vectors.

2.2.1 JPEG/JFIF Format

JPEG (Joint Photographic Experts Group) is a compression method. The compressed images are stored in the JFIF (JPEG File Interchange Format) file format. JPEG compression is lossy compression. The JPEG/JFIF file name extension is JPG or JPEG. Every digital camera can save images in the JPEG/JFIF format, which supports 8-bit grayscale images and 24-bit color images. JPEG applies lossy compression to images, which can result in a significant reduction of the file size. The amount of compression can be specified, and amount of compression affects the visual quality of the result.

2.2.2 JPEG2000 Format

JPEG is a compression standard enabling both lossless and lossy storage. The compression can improve the quality and compression ratios, but also require more computational power to process. JPEG2000 also add the features that are missing in the JPEG. It is currently used in the professional movie editing and distribution.

2.2.3 EXIF Format

The EXIF (Exchangeable image file format) format is a file standard similar to the JFIF format with TIFF extensions: it is incorporated in the JPEG used mostly in cameras. Its purpose is to record and to standardize the exchange of images with image metadata between digital cameras and editing. The metadata are recorded for individual images and include such things camera settings, time and date, speed, exposure, image size, compression, name of camera, color information.

2.2.4 TIFF Format

The TIFF (Tagged Image File Format) is a flexible format that normally saves 8 bits or 16 bits per color for 28 bit and 48 bit total respectively. TIFF can be lossy and lossless. It offer

relatively good lossless compression for bi level images. Some digital cameras can save in TIFF format for lossless storage. TIFF image format is widely accepted as a photograph file standard in the printing business.

2.2.5 RAW Format

The RAW (Raw Image Format) format mainly available in digital cameras, quite than to specific format. These formats usually use a lossless or nearly lossless compression, and produce file sizes smaller than the TIFF formats. The raw formats used by cameras are not standardized or documented. Most camera manufacturers have their own decoding or developing their own raw file format.

2.2.6 GIF Format

The GIF (Graphics Interchange Format) is limited to an 8-bit palette, or 256 colors. This makes the GIF format is suitable for graphics with relatively few colors such as simple diagrams, shapes, logos and cartoon style images. The GIF format supports the animation and it is used to provide image animation effects. It is also uses a lossless compression that is more effective when large areas have a single color, and ineffective for dithered images. Unlike JPEG, the GIF file format uses lossless compression. Assuming that only 256 colors are used, this enables the size of the image to be reduced without dropping the quality of the image. This enables the original image to be retrieved by uncompressing the reduced image. Due to the 256 color restriction, the GIF format is unsuitable for photographs.

GIF compression is ideal for images that contain sharp transitions, such as those in diagrams. The colors are stored in a palette or table which allocates actual color values to a color number. When GIF was created, very few people had the ability to view images in more than 256 colors; this is why the 256 color limitation seemed acceptable. Also for simple graphics, such as cartoons and drawings, 256 colors is usually adequate. To enable binary transparency, one of the colors in the palette can be set as transparent.

2.2.7 PNG Format

The PNG (Portable Network Graphics) file format was created as the free, open free successor to GIF. PNG provides a transferable format for storing, transmitting and displaying bitmapped (raster) images. It can compress and decompress images without loss of quality. PNG supports both color and grey-scale images. The PNG file format supports 8bit palette images with optional transparency for all palette colors and 24bit true color or 48bit true color with and without alpha channel. Compared to JPEG, PNG excels when the image has large, uniformly colored areas.

The lossless PNG format is best suited for pictures. The lossy formats like JPEG, are best suited for photographic images, because in this files are usually smaller than PNG files. PNG provides a patent free replacement for GIF can also replace many common uses of TIFF. Indexed color, grayscale, and true color images are supported, plus to optional alpha channel. PNG is robust, providing both full file integrity checking and simple detection of common source errors. The PNG can store gamma and chromaticity for improved color matching on heterogeneous platforms.

2.3 JPEG2000

JPEG2000 is a new image compression standard being developed by the Joint Photographic Experts Group (JPEG), part of the International Organization for Standardization (ISO). It is designed for different types of still images (bi-level, gray-level, color, multi component) allowing different imaging models (client/server, real-time transmission, image library archival, limited buffer and bandwidth resources, etc), within a unified system. JPEG2000 is intended to provide low bit rate operation with rate-distortion and subjective image quality performance superior to existing standards, without sacrificing performance at other points in the rate-distortion spectrum. The development of the JPEG2000 system was not just to provide higher compression efficiency compared to the baseline JPEG system. Rather, it was to provide a new image representation with a rich set of features, all supported within the same compressed bit-stream, which can address a variety of existing and emerging compression applications. The block diagram of JPEG2000 as shown in Fig.2.1.

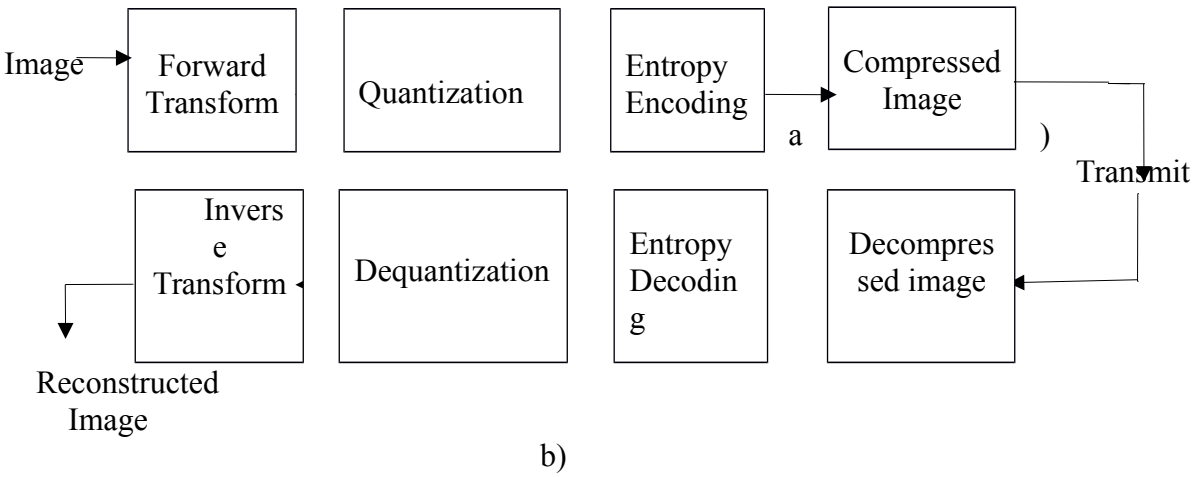


Fig. 2.1: Block diagram of the JPEG2000 a)Encoder b)Decoder

The procedure is as follows:

- The source image is decomposed into components.
- The image and its components are decomposed into rectangular tiles. The tile-component is the basic unit of the original or reconstructed image.
- Mainly, discrete wavelet transform is used for each tile.
- The wavelet transform is applied on each tile. The tile is decomposed in different resolution levels.
- These decomposition levels are made up of sub bands of coefficients that describe the frequency characteristics of local areas (rather than across the entire tile-component) of the tile component.
- Markers are added in the bit stream to allow error resilience.
- The code stream has a main header at the beginning that describes the original image and the various decomposition and coding styles that are used to locate, extract, decode and reconstruct the image with the desired resolution, fidelity, region of interest and other characteristics.

- The optional file format describes the meaning of the image and its components in the context of the application.

JPEG2000 provides four types of scalability

1. Resolution Scalability

   As the stream progresses you first get low resolution data, then medium resolution, etc.

2. Distortion (or SNR) Scalability

   As the stream progresses you first get low SNR version, then medium SNR etc.

3. Spatial Scalability

   As the stream progresses you first get a specific spatial region, then an expanded spatial region, etc.

4. Component Scalability

   As the stream progresses you first get gray-scale image, then color etc.

JPEG2000 addresses areas where current standards fail to produce the best quality of performance, such as:

- Low bit rate compression performance (rates below 0.25 bps for highly-detailed gray-level images)
- Lossless and lossy compression in a single code stream.
- Seamless quality and resolution scalability, without having to download the entire file. The major benefit is the conservation of bandwidth.
- Large images: JPEG is restricted to 64k x 64k images (without tiling). JPEG2000 will handle image sizes up to (2^{32}-1).
- Single decompression architecture: JPEG has 44 modes, many of them are application specific and not used by the majority of the JPEG decoders.

- Error resilience for transmission in noisy environments, such as wireless and the Internet.
- Computer generated imagery: JPEG is optimized for natural images and performs badly on computer generated images.
- Compound documents: JPEG fails to compress bi-level (text) imagery.
- Region of Interest coding
- Improved compression techniques to accommodate richer content and higher resolutions
- Metadata mechanisms for incorporating additional non-image data as part of the file
- JPEG2000 will be able to handle up to 256 channels of information, as compared to JPEG, which is limited to only RGB data. Thus, JPEG2000 will be capable of describing complete alternate color models, such as CMYK, and full ICC (International Color Consortium).

The most important features are

1. Superior low bit-rate performance:

This standard should offer performance superior to the current standards at low bit-rates (e.g. below 0.25 bps for highly detailed gray-scale images). This significantly improved low bit-rate performance should be achieved without sacrificing performance on the rest of the rate-distortion spectrum. Network image transmission and remote sensing are some of the applications that need this feature.

2. Lossless and lossy compression:

It is desired to provide lossless compression naturally in the course of progressive decoding. Examples of applications that can use this feature include medical images, where loss is not always tolerated, image archival applications, where the highest quality is vital for preservation but not necessary for display, network applications that supply devices with different capabilities and resources, and pre-press imagery. It is also desired that the standard should have the property of creating embedded bit stream and allow progressive lossy to lossless build-up.

3. Progressive transmission by pixel accuracy and resolution:

Progressive transmission that allows images to be reconstructed with increasing pixel accuracy or spatial resolution is essential for many applications. This feature allows the reconstruction of images with different resolutions and pixel accuracy, as needed or desired, for different target devices. World Wide Web, image archival and printers are some examples.

4. Region-of-Interest Coding:

Often there are parts of an image that are more important than others. This feature allows users to define certain ROI's in the image to be coded and transmitted with better quality and less distortion than the rest of the image.

5. Random code stream access and processing:

This feature allows user defined ROI's in the image to be randomly accessed and/or decompressed with less distortion than the rest of the image. Also, random code stream processing could allow operations such as rotation, translation, filtering, feature extraction and scaling.

6. Robustness to bit-errors:

It is desirable to consider robustness to bit-errors while designing the code stream. One application where this is important is transmission over wireless communication channels. Portions of the code stream may be more important than others in determining decoded image quality. Proper design of the code stream can aid subsequent error correction systems in alleviating catastrophic decoding failures.

7. Open architecture:

It is desirable to allow open architecture to optimize the system for different image types and applications. With this feature, a decoder is only required to implement the core tool set and a parser that understands the code stream. If necessary, unknown tools could be requested by the decoder and sent from the source.

8. Content-based description:

Image archival, indexing and searching is an important area in image processing. Standards like MPEG-7 (Multimedia Content Description Interface) are addressing this problem currently. Content based description of images might be available as part of the compression system.

9. Side channel spatial information (transparency):

Side channel spatial information, such as alpha planes and transparency planes are useful for transmitting information for processing the image for display, printing or editing. An example of this is the transparency plane used in World Wide Web applications.

10. Protective image security:

Protection of a digital image can be achieved by means of watermarking, labeling, stamping and encryption. Labeling is already implemented in Still Picture Interchange File Format (SPIFF) and must be easy to be transferred back and forth to JPEG2000 image files.

11. Continuous-tone and bi-level compression:

It is desired to have a coding standard that is capable of compressing both continuous-tone and bi-level images. If feasible, this standard should strive to achieve this with similar system resources. The system should compress and decompress images with various dynamic ranges for each color component. Examples of applications that can use this feature include compound documents with images and text, medical images with annotation overlays, and graphic and computer generated images with binary and near to binary regions, alpha and transparency planes, and facsimile.

2.3.1 Quantization

The objective of quantization is to reduce the precision and to achieve higher compression ratio. For instance, the original image uses 8 bits to store one element for every pixel; if we use fewer bits such as 6 bits to save the information of the image, then the storage quantity will be reduced, and the image can be compressed. The shortcoming of quantization is that is a lossy operation, which will result into loss of precision and recoverable distortion.

Quantization refers to the process of approximating the continuous set of values in the image data with a finite set of values. In general the input to the quantizer is the original data,

and the output always one among the finite number of levels. The quantizer is a function whose set of output values are discrete, and usually finite. This is a process of approximation, and a good quantizer is one which represents the original signal with minimum loss or distortion.

There are two types of quantization- scalar and vector quantization. In scalar quantization, each input symbol is treated separately producing the output, while the vector quantization the input symbols are clubbed together in groups called vectors, and processed to give the output. This clubbing of data and treating them as a single unit increases the optimality of the vector quantizer, but at the cost of increased computational complexity. A quantizer can be specified by its input partitions and output levels. If the input range is divided into levels of equal spacing, then the quantizer is termed as uniform quantizer, and if not it is termed as a non-uniform quantizer. A uniform quantizer can easily specified by its lower bound and the step size. Also, implementing a uniform quantizer is easier than a non-uniform quantizer.

The dequantizer is one which receives the output levels of a quantizer and converts them into normal data, by translating each level into normal data, by translating each level into a reproduction point in the actual range of data. The first in compressing an image is to separate the image data into separate blocks. Depending on the importance of data it contains, each block is allocated a portion of the total bit budget, such that the compressed image has the minimum possible distortion. This procedure called bit allocation.

2.3.2 Entropy Coding

After the data has to be quantized into a finite set of values, it can be encoded using an entropy coder, to give additional compression. The main objective of entropy coding is to achieve less average length of the image. Entropy coding assigns code words to the corresponding symbols according to the probability of symbols. In general the entropy encoders are used to compress the data by replacing the symbols represented by equal length codes with the code words whose length is inverse proportional to corresponding probability. The amount of information present in the data, and an entropy coder encodes the given set of symbols with the

minimum number of bits required to represent them. Two of the most popular entropy coding schemes are, Arithmetic coding and Huffman coding.

The arithmetic encoder used by JPEG 2000 standard is a binary adaptive MQ coder. The basis of the binary arithmetic coding is a recursive probability interval subdivision process. Since it is a binary arithmetic encoder, there are only two sub-intervals. With each decision, the current probability interval is subdivided into two sub-intervals. If the value of decision is 1 then it means that it is More Possible Symbol (MPS). Otherwise, the value of decision is 0 and it means that it is Less Possible Symbol. The data distribution is shown in Fig. 2.2.

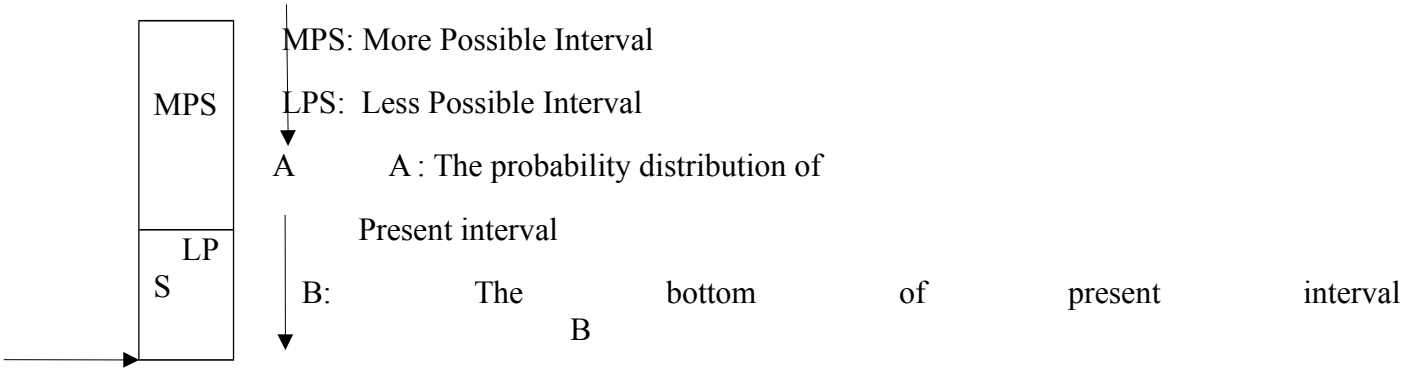


Fig. 2.2: The probability distribution of the MPS and LPS

The Huffman coding is an entropy encoding algorithm used for lossless data compression. This coding refers to the use of a variable length code table for encoding a source symbol where the variable length code table has been derived in a particular way based on the estimated probability of occurrence for each possible value of the source symbol. Huffman coding uses a specific method for choosing the representation of each symbol, resulting in a prefix code that expresses the most common source symbols using shorter strings of bits than are used for less common source symbols.

Huffman coding is based on the probability of occurrence that is pixels in image. The principle is to use a lower number of bits that encode the data that occurs more frequently. Codes are stored in a code book which may be constructed for each image or set of images. Huffman coding is a technique used to compress files for transmission it uses statistical coding. The more frequently used symbols have shorter code words.

The steps for performing Huffman coding is

- Scan text to be compressed and tally occurrence of all characters.
- Sort or prioritize characters based on number of occurrences in text.
- Build Huffman code tree based on prioritized list.
- Perform a traversal of tree to determine all code words.

The final JPEG2000 bit stream may consist of multiple layers.

We summarize the main differences:

1. **Transform module: wavelet versus DCT**

JPEG uses 8 * 8 Discrete Cosine Transform (DCT), while JPEG2000 uses a wavelet transform with lifting implementation. The wavelet transform provides not only better energy compaction (thus higher coding gain), but also the resolution scalability. Because the wavelet coefficients can be separated into different resolutions, it is feasible to extract a lower resolution image by using only the necessary wavelet coefficients.

2. **Block partition: spatial domain versus wavelet domain**

JPEG partitions the image into 16* 16 macroblocks in the space domain, and then applies the transform, quantization and entropy coding operation on each block separately. Since blocks are independently encoded, annoying blocking artifacts becomes noticeable whenever the coding rate is low. On the contrary, JPEG2000 performs the partition operation in the wavelet domain. Coupled with the wavelet transform, there is no blocking artifact in JPEG2000.

3. **Entropy coding module: run-level coefficient coding versus bitplane coding**

JPEG encodes the DCT transform coefficients one by one. The resultant block bitstream cannot be truncated. JPEG2000 encodes the wavelet coefficients bitplane by bitplane (i.e., sending all zeroth order bits, then first order, etc. The generated bitstream can be truncated at any point with graceful quality degradation. It is the bitplane entropy coder in JPEG2000 that enables the bitstream scalability.

4. Rate control: quantization module versus bitstream assembly module

In JPEG, the compression ratio and the amount of distortion is determined by the quantization module. In JPEG 2000, the quantization module simply converts the float coefficient of the wavelet transform module into an integer coefficient for further entropy coding. The compression ratio and distortion is determined by the bitstream assembly module.

2.4 Discrete Wavelet Transform

The DCT-based image compression algorithms such as JPEG have provided satisfactory quality, it still leaves much to be desired. Thus, the new DWT-based image compression algorithms such as JPEG 2000 became increasingly popular. DWT (Discrete Wavelet Transform) is an application of subband coding; thus, before introducing DWT, we briefly describe the theory of subband coding.

In subband coding, the spectrum of the input is decomposed into a set of band limited components, which is called subbands. Ideally, the subbands can be assembled back to reconstruct the original spectrum without any error. The objective of subband coding is to divide the spectrum of one image into the low pass and high pass components Fig. 2.3 shows the block diagram of two band filter bank. At first, the input signal will be filtered into lowpass and highpass components through analysis filters. After filtering, the data amount of the lowpass and highpass components will become twice that of the original signal; therefore, the lowpass and highpass components must be downsampled to reduce the data quantity. At the receiver, the received data must be upsampled to approximate the original signal. Finally, the upsampled signal passes the synthesis filters and is added to form the reconstructed approximation signal. After subband coding, the amount of data does not reduce in reality. However, the human perception system has different sensitivity to different frequency band. For example, the human eyes are less sensitive to high frequency-band color components, while the human ears is less sensitive to the low-frequency band less than 0.01 Hz and high frequency band larger than 20 KHz. We can take advantage of such characteristics to reduce the amount of data. Once the less sensitive components are reduced, we can achieve the objective of data compression.

$y_0(n)$

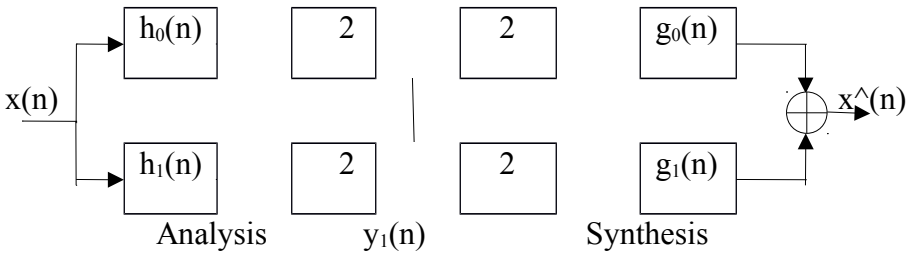


Fig. 2.3: Block diagram of two band filter bank

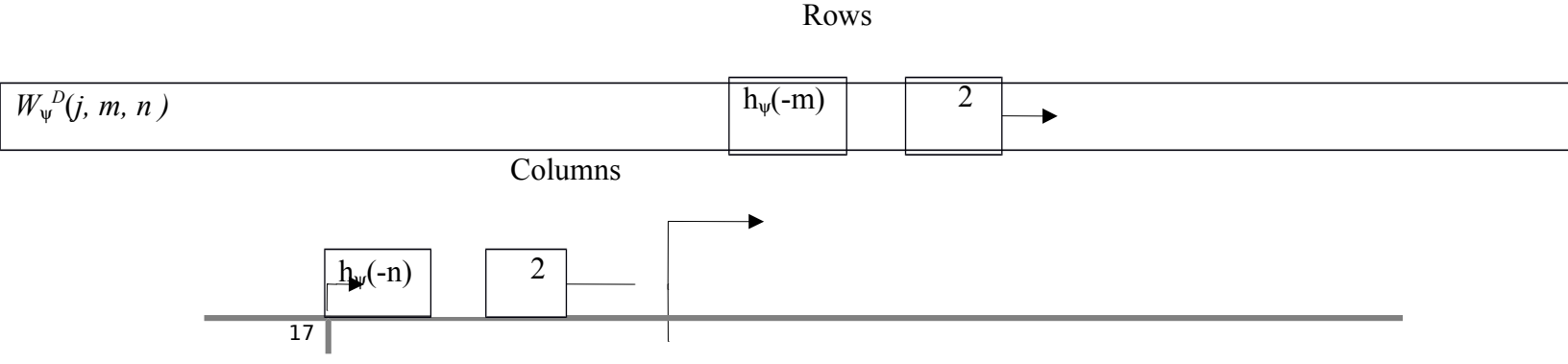
In two dimensional wavelet transform, a two-dimensional scaling function, $\phi(x, y)$, and three two-dimensional wavelet function, $\psi^H(x, y)$, $\psi^V(x, y)$, $\psi^D(x, y)$ are required. Each is the product of a one-dimensional scaling function $\phi(x)$ and corresponding wavelet function $\psi(x)$.

$$\phi(x, y) = \phi(x)\phi(y) \qquad \psi^H(x, y) = \psi(x)\phi(y)$$

$$\psi^V(x, y) = \phi(y)\psi(x) \qquad \psi^D(x, y) = \psi(x)\psi(y)$$

Where ψ^H measures variations along columns, ψ^V responds to variations along rows and ψ^D corresponds to variations along diagonals.

Similar to the one-dimensional discrete wavelet transform, the two-dimensional DWT can be implemented using digital filters and samplers. With separable two-dimensional scaling and wavelet functions, we simply take the one-dimensional DWT of the rows of $f(x, y)$, followed by the one-dimensional DWT of the resulting columns.



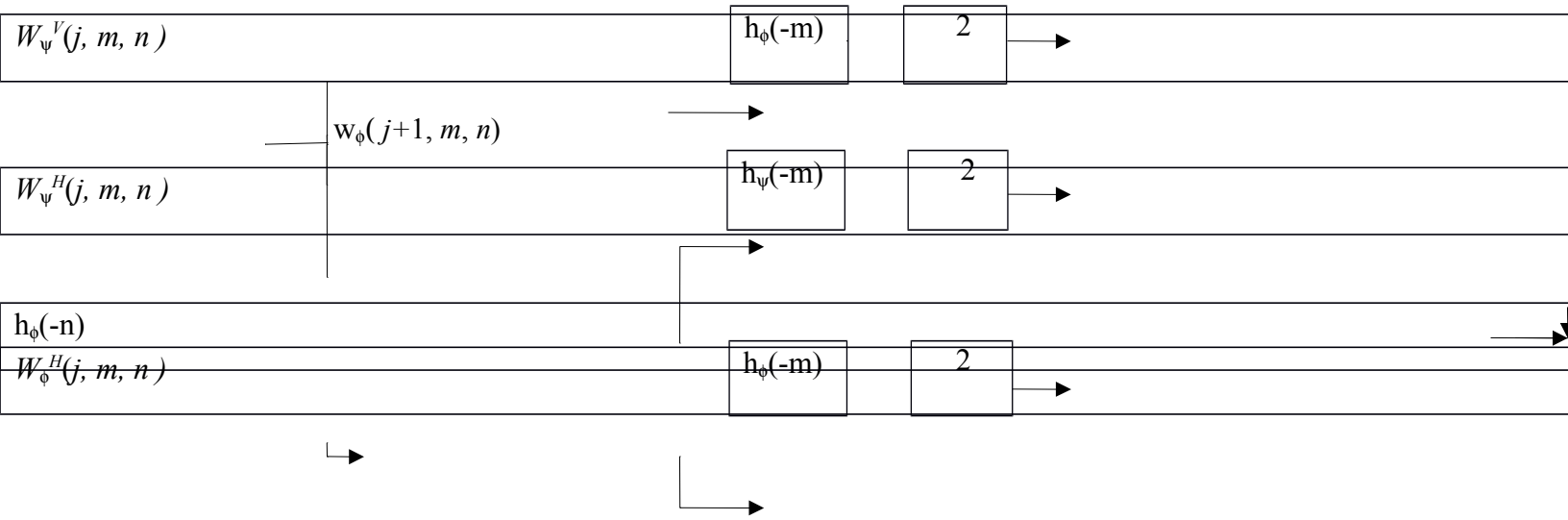


Fig. 2.4: The analysis filter bank of the two-dimensional DWT

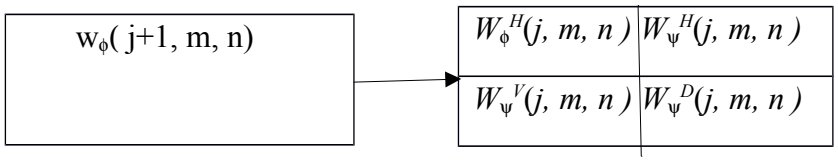
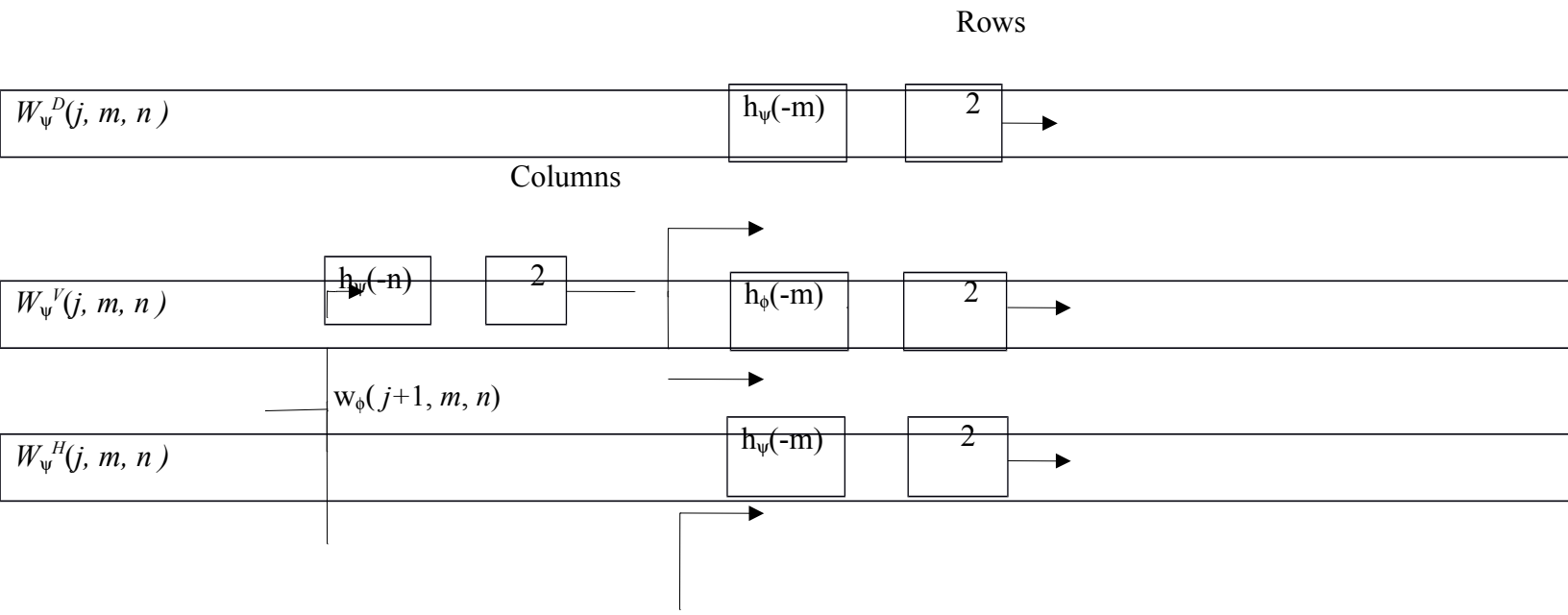


Fig. 2.5: Two-scale of two-dimensional decomposition



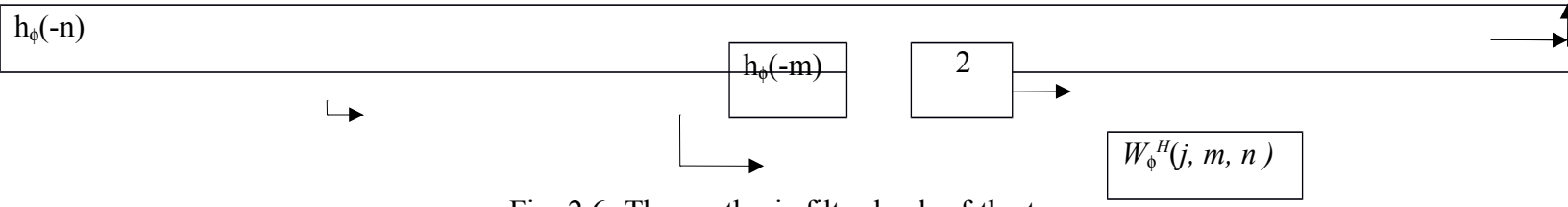


Fig. 2.6: The synthesis filter bank of the two-dimensional DWT

As in the one-dimensional case, image $f(x, y)$ is used as the first scale input, and output four quarter-size sub-images $W_\phi, W_\psi^H, W_\psi^V, W_\psi^D$ as shown in the middle of Fig. 2.5. The approximation output $W_\phi^H(j, m, n)$ of the filter banks in Fig. 2.4 can be tied to other input analysis filter bank to obtain more sub images, producing the two-scale decomposition as shown in the left of Fig. 2.5. Fig. 2.6 shows the synthesis filter bank that reverses the process described above.

2.5 Contrast sensitivity function

The Contrast Sensitivity Function (CSF), which models the varying sensitivity of the human eye as a function of spatial frequency, has been widely exploited in many applications such as quality assessment, perceptual image/video compression, and watermarking. The CSF shows some variations according to the age and visual acuity of the subject, and the viewing conditions. However, it is generally known that the sensitivity has a maximum at 2-6 cycles/degree and is limited at both high and very low frequencies. This is mainly due to the frequency-selective responses of the neurons in the primary visual cortex and low-pass optical filtering.

The contrast sensitivity function for stereoscopic 3D images in the presence of crosstalk. The CSF describes the sensitivity of the Human Visual System (HVS) to different spatial frequencies (in cycles/degree) and has been a popular avenue of investigation with respect to perceptually based compression of 2D images. Peak contrast sensitivity has been found to lie between 2 and 6 cycles/degree. More recently, the CSF has been modeled using the discrete wavelet transform. The uniform noise was added to each wavelet subband (one at a time) of an 8-bit constant 128 grayscale image to generate a stimulus image. Visibility thresholds (VTs) were

measured for each subband by which the level of the noise was adjusted to the point where it just became imperceptible to a human viewer.

The HVS is the end user of most image information. Therefore, any imaging system that reflects human image processing needs should be designed with the characteristics and behavior of the HVS taken into consideration. This will ensure that only information that is relevant for the HVS is stored. One important property of the HVS is that the human eyes selectively understand the image by frequency and orientation. The sensitivity of human eyes to frequencies and orientations is represented by the contrast sensitivity function.

Current models of the human visual system can be split into two different categories. These are neuro-biological models and models based on properties of human vision. Low level processes of the optic nerve and the eye are estimated and then a neuro-biological model is produced. These models tend to be very complex and are therefore not very useful in real-world applications. Aspects of human vision relevant to picture quality are predicted by psychophysical models. These model the human visual system sensitivity to light, also known as luminance masking or lightness non-linearity. The response of the HVS depends on local variations of luminance (i.e. relative luminance and contrast), and it varies very little in relation to the absolute luminance value for an image.

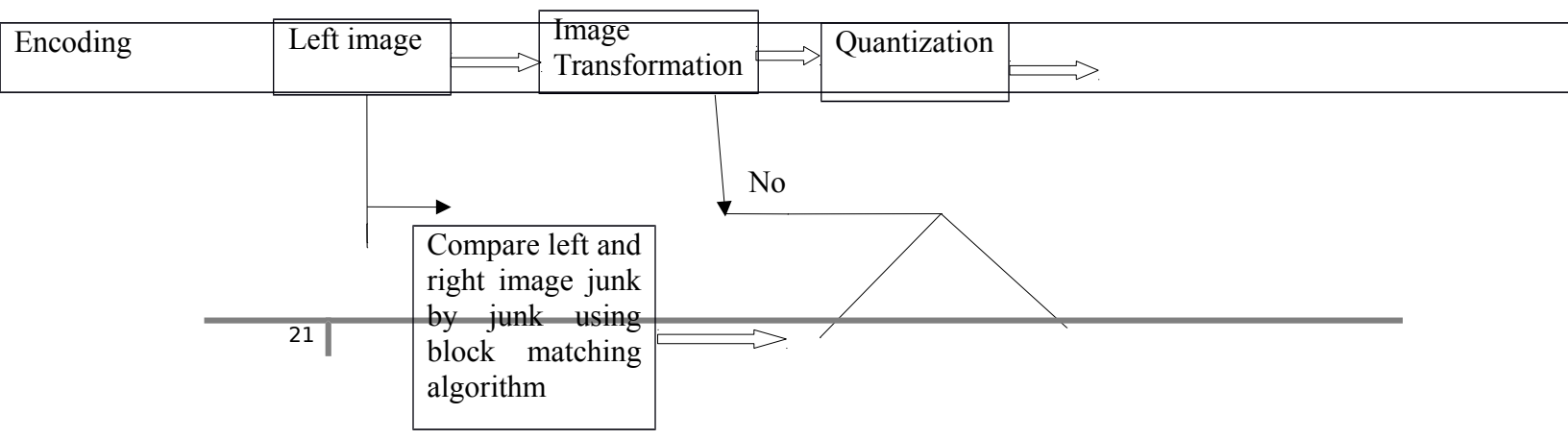
Some of the most important human visual system properties include:

- Sensitivity to contrast changes rather than just luminance changes.
- Varying sensitivity to artifacts and errors at different spatial frequencies. This can be modeled by band limited contrast, which estimates the visibility threshold for stimulus at different spatial frequencies.
- Higher level perceptual factors, such as attention, eye movements and how different types of coding artifacts are unacceptable.
- Stereoscopic rivalry between the left and right images.
- Masking, this refers to reduced inability to detect a stimulus on a spatially or temporally complex background.

2.6 Stereo image compression

There are many different image compression algorithms. These algorithms, primarily designed for monoscopic compression, can be applied to stereoscopic images. Stereoscopic compression can be achieved by either compression across both views simultaneously, independently compressing the left and right views separately. Compression algorithms can be divided into two distinct categories. They are either lossless or lossy. Lossless algorithms do not change the content of the file. Lossy algorithms, such as JPEG, are able to achieve a better compression ratio by selectively getting rid of some of the information within the file. Stereo is the ability to perceive three dimensions by merging two slightly different views of the same scene.

In stereo image compression, in order to simulate the parallax between the left and right eyes of a human observer, left and right images of a scene are taken from two different positions. These two images make up a stereo pair (stereo image). The stereo image compression as shown in Fig. 2.7. In a stereo viewing system, the left and right images of a stereo pair are presented to the left and right eyes of the observer, respectively. The human brain fuses the two images to create the perception of 3D. Typical display technologies for stereoscopic 3D viewing employ passive or active glasses, although other technologies are currently available. For passive systems, the left and right images are displayed sequentially on a display which polarizes them clockwise and anti-clockwise, respectively. Polarized glasses enable only the correct image to pass through to each eye. For active display technologies, active shutter glasses are synchronized to the display. The left and right lenses of these glasses switch to an ON state and a DARK state alternately to allow only the correct signals to be passed through to each eye.



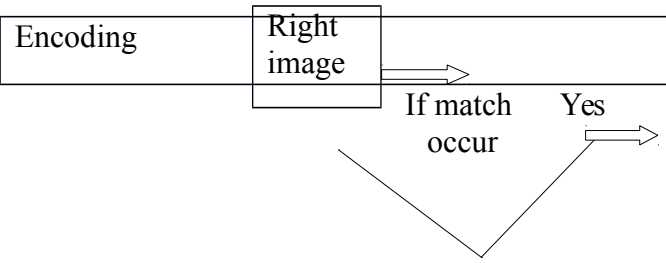


Fig. 2.7: General concept of stereo image compression

A stereo image is produced by taking photographs of the same scene from two slightly different positions. The distance between these positions is called stereo base. Many cameras have ability to stereoscopically image by sequentially taking two images. Parallax the apparent displacement of objects in the stereo scene, is achieved by horizontally shifting the camera between photographs with stereo images provide an improved sense of presence, and have been found to be operationally useful in responsibilities requiring remote manipulation or judgment of spatial relationships.

A conventional stereo system with a single left-right pair needs twice the raw data as a monoscopic imaging system. As a result there has been increasing attention given to image compression methods specialized to stereo pairs. The general concept of stereo image compression method, first compress the reference image only, after that the block matching algorithm is applied between the right and left images to identify the disparity values and they are encoded separately rather than compressing the whole right image. Stereoscopic compression can be achieved by either compression across both views simultaneously, or independently compressing the left and right views separately.

2.6.1 Stereo image compression using DCT

In stereo image compression using Discrete Cosine Transform (DCT) image transformation, the reference image is the left image of stereo pair and that image is transformed by 2D forward discrete cosine transform. The resultant image matrix image is quantized using the quantization matrix and the compressed using Huffman coding. The second part of the encoder involves compressing the right image. Since the two images are similar to each other, disparity vectors between the two images are estimated. The resultant vectors are displaced pixel values in second image that only compressed into a bit stream.

2.6.2 Stereo image compression using DWT

Stereo pair image compression using Discrete Wavelet Transform (DWT) is being increasingly used for image coding as the DWT can decompose the signals into different subbands with both time and frequency information. The main features like progressive image transmission, coding of region of interest and further manipulations in compressed image can be achieved. The separable 2D wavelet transformation can be implemented by applying a two level decomposition of the 1D DWT in the horizontal and vertical dimensions respectively. At first step the right image is estimated from left image. The step is called disparity estimation or compensation. After the subtraction, the left image and the right image are transformed. Finally these images are coded. Stereo images are highly correlated. The left and right images are differing only slightly.

2.6.3 Stereo image compression using EBTC

Stereo pair image compression using Extended Block Truncation coding (EBTC) algorithm is used. Block truncation coding is a recent technique is used for stereo images. It is one bit adaptive moment preserving quantizer that preserves certain statistical moments of small blocks of the input image in the quantized output.

In block truncation coding algorithm, it preserves the standard mean and the standard deviation. The statistical overheads mean and standard deviation are to be coded as a part of the block. The truncated block of the block truncated code is the one bit output of the quantizer for every pixel in the block. The procedure for performing extended block truncation compression for stereo images are, in step one, the left image is divided into non-overlapping rectangular regions then select two luminance values to represent each pixel in the block. These values are the mean x and standard deviation σ. To perform the two level quantization on the block, to gain the compression, it is performed by taking x as the threshold value, and bit plane is obtained by comparing each pixel value with threshold. If the pixel value is greater than the mean, it is assigned value ‘1’ otherwise ‘0’. To reconstruct the image, elements assigned to ‘0’ are replaced with the value ‘a’ and the elements assigned to ‘1’ are replaced with the value ‘b’.

The disparity vectors between the two images are estimated. To achieve a better compression ratio by selectively getting rid of some of the information is used only stereoscopic image compression technique method in truncation coding. The truncated block of the block truncated code is the one bit output of the quantizer for every pixel in the block. Stereoscopic compression can be achieved by either compression across both views simultaneously, or independently compressing the left and right views separately.

2.6.4 Block Matching Algorithm

The encoding phase of all the stereo image pair algorithm use block matching technique to find the position of object in another block. Therefore the image is divided into a number of rectangular blocks of 8 by 8 pixels. According to algorithm used for motion estimation, a block with in the search range is compared with the source. Block matching uses a value called block distortion measure to rate the similarity between two blocks.

The basic idea is to sum up the differences of pixels luminance of pixels located at the same position in the two blocks. The blocks which are different in the pixel values only undergo the process of compression. The other blocks of the image which are same as the reference image are not included in the steps involved in the compression of the reference image but just copied with the bit code of encoded form of the reference image. The block matching algorithm is used to reduce the complexity of the compression process.

2.6.5 Disparity Estimation

The correspondence between two images can be determined by either matching features or by operating on or matching of small patches of gray values. Future matching requires as a preprocessing step the extraction of the appropriate features from the images, such as object edges and corners. After obtaining the features, the correspondence problems are solved first for the spatial locations at which the features occur, from which next the full disparity field can be deduced by instance interpolation or segmentation procedures.

To employ disparity information in stereoscopic pair in image processing applications, the relation between the contents of the left view image and right view image has to be established, yielding the disparity field. The disparity field is indicated for each point in the left view image the relative shift of the corresponding point in the right view image, and vice versa. Since some parts of one view image may not be visible in the alternate view image due to occlusion, not all points in the image pair can be assigned a disparity vector.

The feature based disparity estimation is especially useful in the analysis of scenes for robot vision applications. Disparity field estimation by operating directly on the image gray value information is not unlike the problem of motion estimation. The first difference is that disparity vectors are approximately horizontally oriented. Deviations from the horizontal orientation are caused by the convergence of the camera axes and the differences between the camera optics. Usually vertical disparity components are either ignored or rectified. A second difference is that disparity vectors can take on a much larger range of values within a single image pair.

Furthermore the disparity field may have larger discontinuities associated with objects neighboring in the planar projection but having a very much different depth. In those regions of the stereoscopic image pair where one finds large discontinuities in the disparity field due to abrupt depth changes, large regions of occlusion will be present. Estimation methods for disparity fields must therefore be able not only to find the correspondence between information in the left view and right view images, but must also able to perceive and handle discontinuities and super nature of virtual reality beyond the range.

Most disparity estimation algorithms used in stereoscopic communications rely on matching small patches of gray values from one view to the gray values in the alternate view. The matching of this small patch is not carried out in the entire alternate image, but only within a relatively small search region to limit the computational complexity. Standard methods typically use a rectangular match block of relatively small size pixels. The relative horizontal shift between a match block with in the search region of the alternate image that results in the smallest value of criterion function used, is then assigned as disparity vector to the center of that match block.

Often used criterion functions are the sum of squares and the sum of the absolute values of the differences between the gray values in the match block and the block being considered in the search region. In more recent years the technology has advanced and there are now many different ways to view stereoscopic 3D images. These technologies will be detailed independently and include,

1. Anaglyph
2. Shutter Glasses
3. Polarized Glasses
4. Auto-stereoscopic Displays

1. Anaglyph

An anaglyph stereoscopic 3D image the viewer must wear glasses that contain two differently colored lenses. Red/cyan glasses are often used, but red/green and blue/yellow are also available. This disparity simulates the distance between our two eyes, which provides two views of the same scene, therefore providing us with the perception of depth, or binocular stereopsis.

2. Shutter Glasses

The lenses of shutter glasses are darkened one after the other to allow light to reach one eye then the other. The glasses are synchronized to the display or projector so that when the image for the left eye is being shown the right eye is darkened. This is then followed by the left eye lens darkening when the image for the right eye is displayed. This means that the correct eye only receives the correct view. The switch between the two images and the switching of which lens is darkened is done very rapidly, ideally at 60 times a second. This produces a flicker free 3D image. This method has become recently more popular with the reduction in the weight of the shutter glasses and introduction of high quality 3D projectors running at 120Hz.

3. Polarized Glasses

For passive systems, the left and right images are displayed sequentially on a display which polarizes them clockwise and anti-clockwise, respectively. Polarized glasses enable only the correct image to pass through to each eye. For active display technologies, active shutter glasses are synchronized to the display. The left and right lenses of these glasses switch to an ON state and a DARK state alternately to allow only the correct signals to be passed through to each eye.

Un-polarized normal light travels in electromagnetic waves. These waves vibrate in many different axes including the horizontal and vertical. Polarized stereoscopic systems project light with different polarizations for each stereoscopic pair. For example, light might be transmitted in the horizontal axis for the left eye image and the vertical axis for the right eye image. Polarized glasses are then worn so that the viewer only sees the image with the correct polarization to pass through the filters on the glasses.

4. Auto-stereoscopic Displays

Auto-stereoscopic three dimensional displays automatically direct the left and right views to the appropriate eye. Multi-view displays present more than two views at the same time in multiple viewing boxes, for example, there may be eight, nine or even twelve views. The user sees two of these views at the same time. The views are presented so that a valid stereo pair is formed and displayed to each of the viewers. This gives a large viewing area and enables multiple viewers to see the effect at the same time. These displays often have a lot lower resolution per view in comparison to the Twin view displays, because the underlying display is split into multiple views. They are, however, relatively cheap to produce. Switching between 2D and 3D is rarely a feature of Multi-view displays.

Typical display technologies for stereoscopic 3D viewing employ passive or active glasses, although other technologies are currently available. For passive systems, the left and right images are displayed sequentially on a display which polarizes them clockwise and anti-clockwise, respectively. Polarized glasses enable only the correct image to pass through to each eye. For active display technologies, active shutter glasses are synchronized to the display. The left and right lenses of these glasses switch to an ON state and a DARK state alternately to allow only the correct signals to be passed through to each eye.

2.7 Comparing Stereoscopic 3D Displays

The following specific technical features of displays should be considered.

1. Total Display Resolution

Stereoscopic displays are designed to display one view to each eye. The total resolution for a display is the sum of all the pixels in all the views and determines the required bandwidth and computational effort necessary to display the images.

2. Resolution per View

The resolution per view is a very important variable when comparing the human perception of 3D displays. When using a stereo 3D display there is still a need for a high resolution. This is because the display pixels are split between the number of views. Images on a 3D display can often appear to look better than on a 2D display with an equal total screen resolution. This is because the information received from the two views is integrated, by the human brain, into a single image.

3. Stereoscopic Resolution

Stereoscopic resolution is the number of depth voxels (intervals) within the range of +/- 100mm. It can be calculated for each of the displays by finding the screen disparity generated in this range. The total of these values is then divided by the stereoscopic pixel width of the display. The stereoscopic resolution of displays with a lower resolution per view is also low. The human eye is able to perceive at least 240 voxels over a range of +/- 100mm.

4. Viewing Freedom

The viewing freedom is the distance over which a viewer is able to move without losing the 3D effect. When using the screens for desktop applications the viewing freedom is less

important. The viewers usually position themselves in the centre of the screen. Viewing freedom becomes of greater importance in applications such as public information displays where users are passing by a display rather than sitting down to use it.

5. Crosstalk

Crosstalk is a critical factor determining the image quality of stereoscopic displays. Also known as ghosting or leakage, high levels of crosstalk can make stereoscopic images hard to fuse and lack fidelity; hence it is important to achieve low levels of crosstalk in the development of high-quality stereoscopic displays. When using stereoscopic displays, crosstalk is the resulting effect of each eye seeing an image of the unwanted perspective view. Ideally in a stereoscopic system each eye should only see its assigned image. Perfect stereoscopic images free from crosstalk can be created using a stereoscope. This is because each image is optically separate from the other. Crosstalk occurs when one eye can see a signal that is intended to be seen only by the other eye. Since these unintended signals are usually dim, It can lead not only to degradation in the perceived quality of 3D images, but also to discomfort in some individuals. Crosstalk is caused by leakage between the left and right channels.

Unfortunately stereoscopic displays can be imperfect and often crosstalk can be seen. Pommeray states that when using stereoscopic displays, there are two main opportunities for crosstalk to occur. These are departures from the ideal shutter in the eyewear and CRT phosphor afterglow. Stereographic state that they have produced electro-optical shutters that are so good that no unwanted image will be perceived in the incorrect eye from incomplete shutter occlusion. It is a key performance characteristic for 3D displays. It is created when light leaks between the left and right viewing windows. In an ideal world the crosstalk would be zero, however in practice it is often a lot more. It is difficult to know the precise crosstalk of a display because the majority of manufacturers do not quote it for their products. For applications where depth judgment is critical, the use of a display with a high crosstalk is unlikely to be acceptable.

6. Brightness and Contrast

The brightness and contrast of a 3D display is not comparable to an equivalent 2D

display. In parallax barrier displays, the brightness is reduced as light is blocked by pixels. With displays with two barriers, the rear barrier can be mirror coated on the side facing the illuminator to recirculation light.

CHAPTER 3

VISUALLY LOSSLESS JPEG2000 COMPRESSION

3.1 Visibility Thresholds

The Visibility Thresholds for a JPEG2000 codeblock of a 2D image is defined as the largest quantization step size for which quantization distortion remains invisible. It is modeled as a function of the wavelet coefficient variance σ^2 within the codeblock, as well as the orientation $\theta \in \{LL, LH, HL, HH\}$, and level k of the subband to which the codeblock belongs. The stereoscopic 3D images require careful consideration of the crosstalk effect. The crosstalk effect is considered here to be a luminance leakage from one channel to the other channel. When crosstalk occurs, signals intended for the left eye can be seen by right eye. Therefore, VTs for the left image must be chosen so that any quantization distortion is invisible to both the left and right eyes. To find two VTs $t^{l}$and t^{ll}, the first VT, t^{l}, is chosen so that the resulting distortion is invisible to the left eye, while the second VT, t^{ll}, ensures that the distortion is invisible to the right eye. The final VT t is then taken as the minimum of $t^{l}$and t^{ll}. Thus, t^{l} and t^{ll} are functions not only of σ^2, θ and k in the left image, but also of the gray level I in both the left and right images. Mathematically, the VT for a given codeblock in the left image at level k and orientation θ is then

$$t_{\theta,k,l}(\sigma^2_{\theta,k,l}, I_{\theta,k,l}, I_{\theta,k,r}) = \min\{ t^{l}_{\theta,k,l}(\sigma^2_{\theta,k,l}, I_{\theta,k,l}, I_{\theta,k,r}), t^{ll}_{\theta,k,l}(\sigma^2_{\theta,k,l}, I_{\theta,k,l}, I_{\theta,k,r})\} \quad \text{——} \quad (1)$$

Where $\sigma^2_{\theta,k,l}$ is the variance of wavelet coefficients within the codeblock and $I_{\theta,k,l}$ is a gray level for the left image representative of the spatial region associated with the codeblock. Similarly, $I_{\theta,k,r}$ is a gray level representative of the same spatial region.

For a given σ^2, there are 2 thresholds × 16 subbands ×256^2 combinations of I, for a total of more than 2×10^6 thresholds to be measured. When σ^2 is varied, the total number of thresholds to be measured increases proportionally.

The nominal thresholds $T_{\theta,3,l}(I_{\theta,3,l}, I_{\theta,3,r})$ measured for level $k = 3$ and variance $\sigma^2 = 50$ via

$$T_{\theta,3,l}(I_{\theta,3,l}, I_{\theta,3,r}) = \min \left\{ t^{I}_{\theta,3,l}(50, I_{\theta,3,l}, I_{\theta,3,r}),\ t^{II}_{\theta,3,l}(50, I_{\theta,3,l}, I_{\theta,3,r}) \right\} \quad (2)$$

The value of $\sigma^2 = 50$ is chosen as being near the middle of the range in which the VTs vary most as a function of variance. The nominal thresholds $T_{\theta,3,l}(I_{\theta,3,l}, I_{\theta,3,r})$ attempt to model the effect of crosstalk caused by different intensities in the left and right images for different orientations, but fixed variance and transform level. The nominal thresholds are then scaled by a factor $S_{\theta,k,l}(\sigma^2_{\theta,k,l})$ which attempts to model the effects of orientation, variance, and transform level , but with no crosstalk present. To achieve a state of no crosstalk, the intensities in the left and right images are set to be equal.

In order to measure the visibility thresholds $t^{I}_{\theta,3,l}(50, I_{\theta,3,l}, I_{\theta,3,r})$, $t^{II}_{\theta,3,l}(50, I_{\theta,3,l}, I_{\theta,3,r})$ as required by

$$S_{\theta,k,l}(\sigma^2_{\theta,k,l}) = t_{\theta,k,l}(\sigma^2_{\theta,k,l}, 128, 128)/T_{\theta,3,l}(128, 128) \quad (3)$$

Accordingly, to find $t^{I}_{\theta,3,l}(50, I_{\theta,3,l}, I_{\theta,3,r})$, $t^{II}_{\theta,3,l}(50, I_{\theta,3,l}, I_{\theta,3,r})$, a value of Δ is chosen and a left stimulus image is created with parameters, Δ , θ, $I_{\theta,3,l}$, $k = 3$, and $\sigma^2 = 50$. This image is paired with a right constant gray image with all pixels set to $I_{\theta,3,r}$.The value of Δ in the left stimulus image is then adjusted until the noise is just invisible to the left eye, with the right eye covered. The resulting value of Δ is $t^{I}_{\theta,3,l}(50, I_{\theta,3,l}, I_{\theta,3,r})$.

The thresholds $t^{l}_{\theta,3,l}(50, I_{\theta,3,l}, I_{\theta,3,r})$ are $t^{ll}_{\theta,3,l}(50, I_{\theta,3,l}, I_{\theta,3,r})$ are measured for five values of $I_{\theta,3,l}$ and nineteen values of $I_{\theta,3,r}$. Specifically, $I_{\theta,3,l} \in \{0, 60,128, 195, 255\}$ while $I_{\theta,3,r}$ can take any of the eighteen integer multiples of 15 between 0 and 255, plus the additional value of 128. That is $I_{\theta,3,r} \in \{0, 15, \ldots, 120,128, 135, \ldots, 240, 255\}$.

The measurement of $S_{\theta,k,l}(\sigma^2_{\theta,k,l})$ as given in (3). The gray levels for both the left and right images are fixed to 128 but the value of the assumed (left) codeblock variance $\sigma^2_{\theta,k,l}$ is varied. For a given choice of θ, k, and $\sigma^2_{\theta,k,l}$, the value of Δ in the left stimulus image is again adjusted to the point that the noise is just imperceptible for each eye, resulting in $t_{\theta,k,l}(\sigma^2_{\theta,k,l}, 128, 128)$ and $t^{l}_{\theta,k,l}(\sigma^2_{\theta,k,l}, 128, 128)$.

The 4th order polynomial interpolation is used to obtain values of the nominal thresholds $T_{\theta,3,l}(I_{\theta,3,l}, I_{\theta,3,r})$ for values of $I_{\theta,3,l}$ and $I_{\theta,3,r}$ is given by

$$\begin{aligned} T_{\theta,3,l}(I_{\theta,3,l}, I_{\theta,3,r}) = {} & P_{00} + I_{\theta,3,l} P_{10} + I_{\theta,3,r} P_{01} + I^2_{\theta,3,l} P_{20} \\ & + I_{\theta,3,l} I_{\theta,3,r} p_{11} + I^2_{\theta,3,r} p_{02} + I^3_{\theta,3,l} p_{30} \\ & + I^2_{\theta,3,l} I_{\theta,3,r} p_{21} + I_{\theta,3,l} I^2_{\theta,3,r} p_{12} + I^3_{\theta,3,r} p_{03} \\ & + I^4_{\theta,3,l} p_{40} + I^3_{\theta,3,l} I_{\theta,3,r} p_{31} + I^2_{\theta,3,l} I^2_{\theta,3,r} p_{22} \\ & + I_{\theta,3,l} I^3_{\theta,3,r} p_{13} + I^4_{\theta,3,r} p_{04} \qquad \text{——(4)} \end{aligned}$$

	LL3	HL3/LH3	HH3
P_{00}	-2.8563	4.3468	-3.9231
P_{10}	-0.058	-0.1092	-0.01164
P_{01}	-0.0298	-0.0411	-0.0145
P_{20}	5.126e-4	1.198e-3	1.51e-3
P_{11}	4.925e-4	6.306e-4	4.118e-4
P_{02}	1.817e-4	2.705e-4	-3.869e-5
P_{30}	-2.836e-6	-6.718e-6	-8.673e-6
P_{21}	-8.072e-7	-5.579e-7	-6.727e-7
P_{12}	-2.182e-6	-3.846e-6	-1.853e-6
P_{03}	-4.664e-7	-4.987e-7	4.085e-7
P_{40}	7.497e-9	1.518e-8	1.866e-8
P_{31}	-6.084e-9	-7.397e-9	-5.732e-9
P_{22}	8.656e-9	9.273e-9	7.864e-9
P_{13}	5.094e-10	4.483e-9	2.184e-10

P_{04}	6.647e-10	1.073e-10	-2.161e-10

The interpolation parameters P_{ij} are given in Table 3.1,

Table 3.1: Interpolation Parameters P_{ij}

For the purpose of measuring VTs, stimulus images are generated by performing the inverse wavelet transform of wavelet coefficient data containing simulated quantization distortion. Specially, the wavelet data for all subbands of an image of size 512 × 512 are initialized to 0, and then noise is added to one subband. This noise is generated pseudo-randomly according to the JPEG2000 quantization distortion model. The inverse wavelet transform is performed, and the result is added to a constant gray level image having all pixel intensities set to a fixed value $I_{\theta,3,l}$ between 0 and 255. After the addition, the value of each pixel in this stimulus image is rounded to the closest integer between 0 and 255.

The numerator part of the equation3 can be considered. That is $t_{\theta,k,l}(\sigma^2_{\theta,k,l}, 128, 128)$ is represented in two term power series as

$$t_{\theta,k,l}(\sigma^2_{\theta,k,l}, 128, 128) = a(\sigma^2_{\theta,k,l})^b + c \quad \text{———} (5)$$

subband	A	B	C
LL1	1.758e-5	2.214	1.656
LL2	-1.298	-1.088	1.309
LL3	-0.558	-0.759	0.987
LL4	1.442e-5	1.674	0.8
LL5	3.838e-8	2.789	1.044
HL1/LH1	0.003	1.323	1.904
HL2/LH2	0.034	0.671	0.892
HL3/LH3	-5.097	-1.109	1.041
HL4/LH4	-38.657	-2.094	0.95
HL5/LH5	-19.6	-1.916	0.866
HH1	4.123e-4	1.847	7.934
HH2	1.78	0.157	-0.7
HH3	4.16e-4	1.417	0.813
HH4	0.618	0.071	0.095
HH5	-4.553	-1.909	0.86

Table 3.2: Parameters for power series interpolation ($\sigma^2_{\theta,k,l} \in [0,100]$)

In order to obtain high-quality bits, the range of the variance $\sigma^2_{\theta,k,l}$ is partitioned into two segments, from 0 to 100 and from 100 to infinity, respectively. The constant parameters *a*, *b*, and *c* for each segment are provided in Tables 3.2 and 3.3.

subband	a	b	C
LL1	-346.9	-1.891	2.185
LL2	-3.251e-3	-2.362	1.362
LL3	0.002	0.689	0.932
LL4	1.181e-4	0.925	0.82
LL5	-1.202e3	-2.481	1.072
HL1/LH1	-9.122e9	-5.088	4.024
HL2/LH2	1.97e-10	2.671	1.629
HL3/LH3	-3.93e5	-3.286	1.115
HL4/LH4	-0.095	-0.435	0.96
HL5/LH5	2.858e-4	0.731	0.854
HH1	-1.658e7	-3.644	10.829
HH2	-29.434	-1.031	3.216
HH3	0.005	0.552	1.033
HH4	-0.754	-0.278	1.161
HH5	-0.469	-0.109	1.144

Table 3.3: Parameters for power series interpolation ($\sigma^2_{\theta,k,l} \in [100,\infty]$)

Spatial three-alternative forced-choice (3AFC) testing is employed to find the value of Δ for which the distortion is just invisible in each case. Three stereoscopic images are shown on the display concurrently. One is placed at the top center of the screen, and the other two are arranged at the bottom left and bottom right, respectively. A stereoscopic stimulus image (containing noise in only the left channel) is displayed randomly at one of three locations. The other two stereoscopic images contain no noise.

3.2 Visually Lossless Coding

Visually lossless compression of 8-bit monochrome stereoscopic images is adapted. In JPEG2000, a subband is partitioned into rectangular codeblocks. The coefficients of each codeblock are then quantized and encoded via bit-plane coding. Three coding passes (significance propagation, magnitude refinement, and cleanup) are performed for each bit-plane

except the most significant bit-plane which only has a cleanup pass. The maximum possible number of coding passes for a codeblock is then 3M − 2 where M denotes a number of bit-planes sufficient to represent the magnitude of all quantized coefficients in a codeblock. The actual number of coding passes included in a compressed code stream can vary from codeblock to codeblock and is typically selected to optimize mean squared error over the entire image for a given target bit rate.

The minimizing mean squared error, the method includes the minimum number of coding passes necessary to achieve visually lossless encoding of a 2D image. This is achieved by including a sufficient number of coding passes for a given codeblock such that the absolute error of every coefficient in that codeblock is less than the VT for that codeblock. This is extended to the coding of 3D images. Specifically, the coding of the left image of a stereo pair is carried out using the left VT for each codeblock of the left image as computed. The right image is then encoded using the right VT for each codeblock of the right image. To compute right VTs in the identical manner as left VTs, by simply reversing the roles of the left and right images.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Results of visually lossless JPEG2000 compression

Let us consider three images. Input image 1 taken as a cat image, input image 2 taken as a Buddha image, input image 3 taken as a owl image. Three input images are shown in Fig. 4.1.

4.1.1 Input Images

Image 1

Image 2

Image 3

Fig.4.1: Input images 1, 2 and 3

4.1.2 Slices along the Z-orientation

In three dimensional image, first consider the slices in Z-direction. In this direction slices can be considered as Z=1, Z=2, Z=3 directions for corresponding input images. The slices along the Z-direction are as shown in Fig. 4.2, Fig. 4.3 and Fig. 4.4.

Fig. 4.2: Slices along the z-orientation for image 1

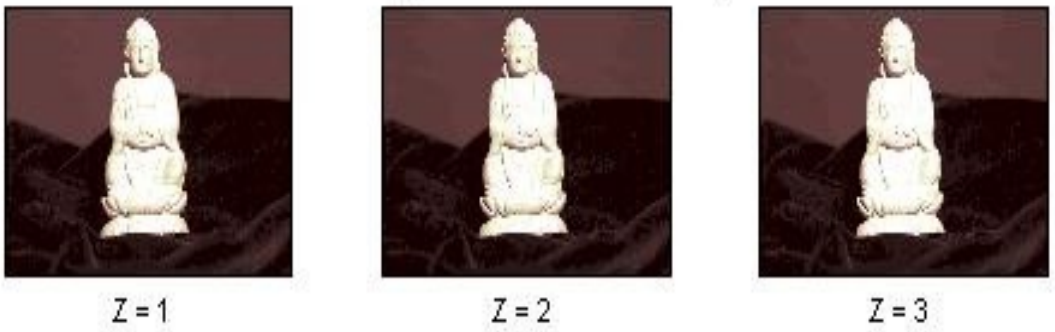


Fig. 4.3: Slices along the z-orientation for image 2

Fig. 4.4: Slices along the z-orientation for image 3

4.1.3 Slices along the Y-orientation

Let us consider the slices along the Y-direction, in the direction the slices can be considered as the Y=1, Y=2, Y=3 directions for the corresponding images. The slices are shown in Fig .4.5, Fig .4.6 and Fig. 4.7.

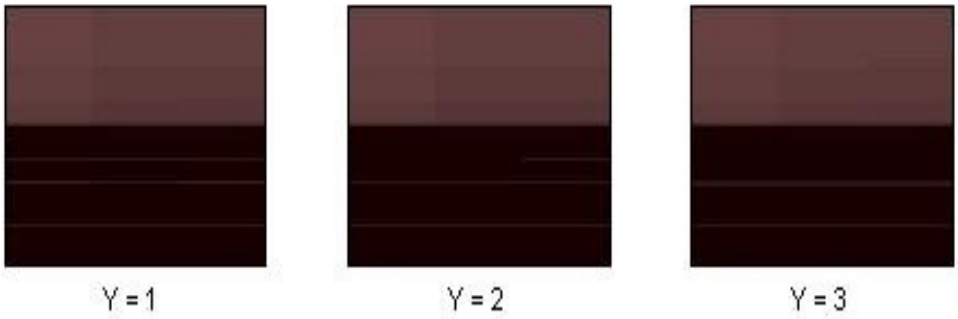


Fig. 4.5: Slices along the y-orientation for image 1

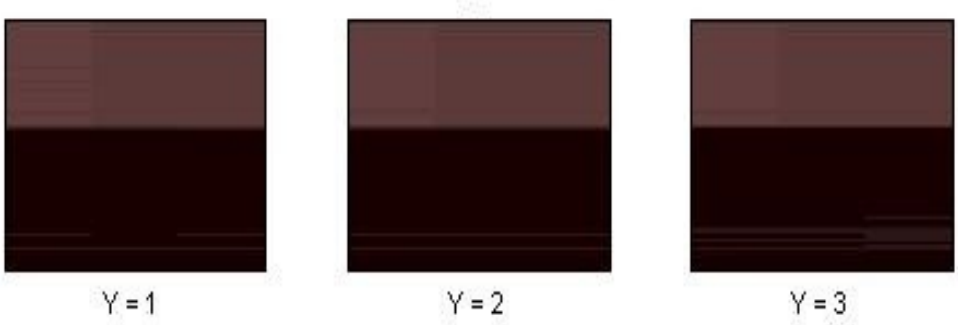


Fig. 4.6: Slices along the y-orientation for image 2

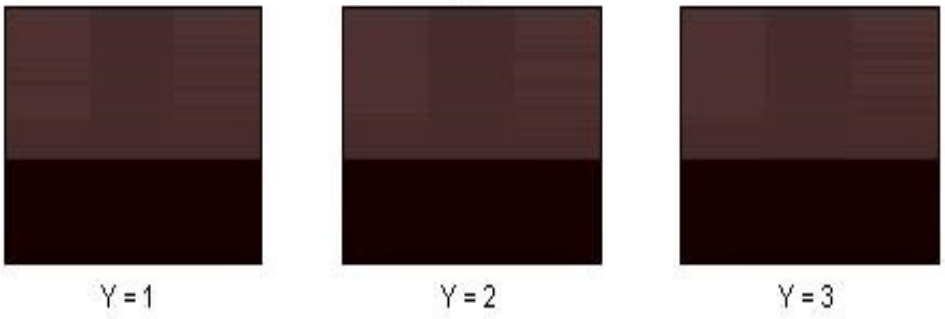


Fig. 4.7: Slices along the y-orientation for image 3

4.1.4 Approximation and Decimation at level 1

In this section consider the filters which are used to reduce the unwanted noise that present in the image. To remove the noise, using the high pass and low pass filters are used. The

high pass filter is used to pass only the high frequencies and remove all other lower frequencies. The low pass filter is used to remove the higher frequency and pass only the lower frequency. The Approximation and Decimation mainly consists of three levels: first we consider the Approximation and Decimation at level 1. In this level the approximation and decimation values can be subtracted from the slicing values, the slicing values can be either the y-direction or z-direction. Consider the approximation and decimation values can be subtracted from the z-orientation. The approximation and decimation at level 1, level 2, level 3 can be represented as a A1-Z=1, A2-Z=1, A3-Z=1 and second level A1-Z=2, A2-Z=2, A3-Z=2 and third level A1-Z=3, A2-Z=3, A3-Z=3.The approximation and decimation of all the levels as shown in the Fig. 4.8, Fig. 4.9, Fig. 4.10, Fig. 4.11, Fig. 4.12, Fig. 4.13, Fig. 4.14, Fig. 4.15 and Fig. 4.16 respectively.

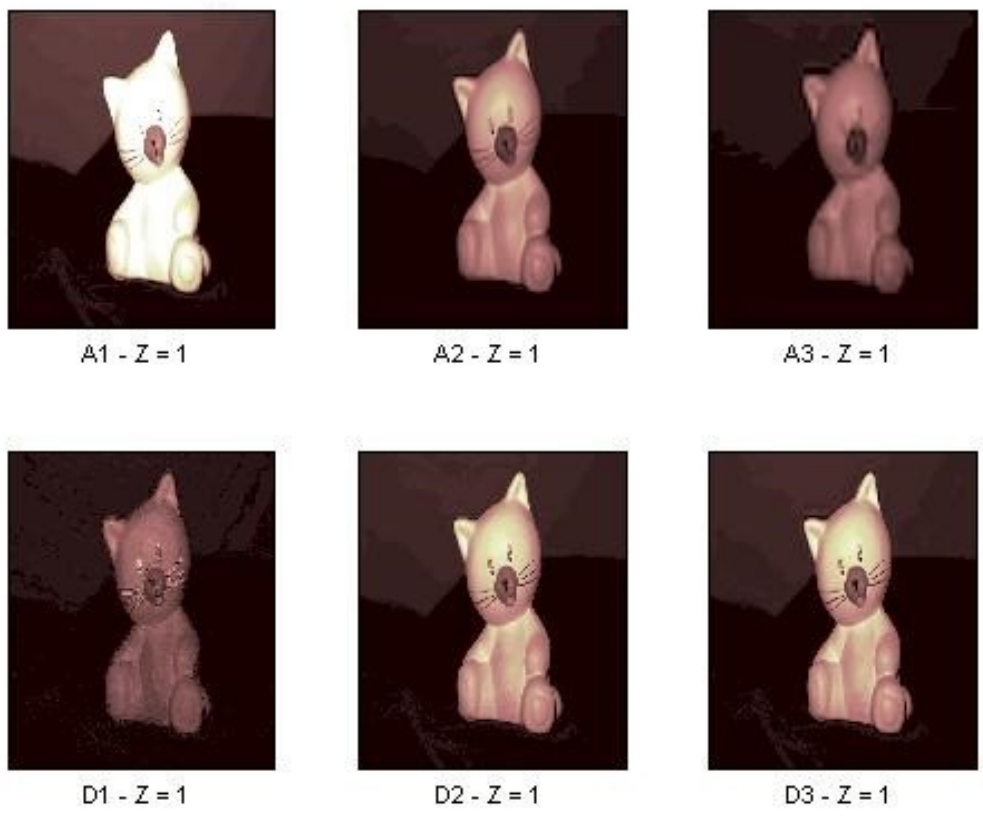


Fig. 4.8: Approximation and Decimation at level 1 for image 1

Fig. 4.9: Approximation and Decimation at level 1 for image 2

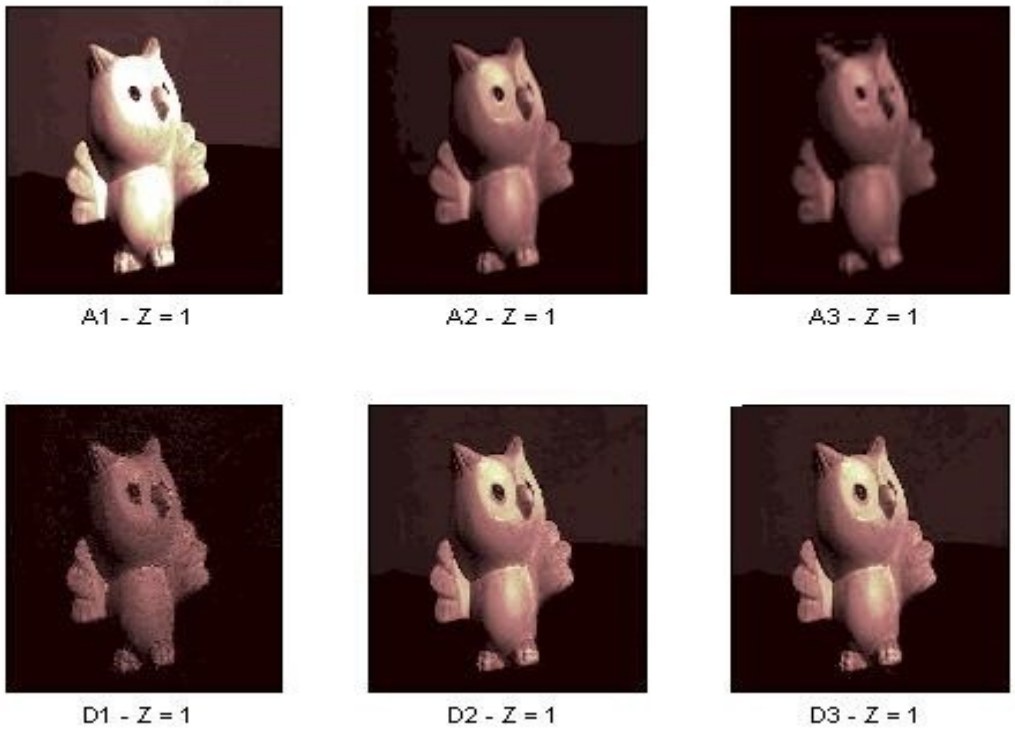


Fig. 4.10: Approximation and Decimation at level 1 for image 3

4.1.5 Approximation and Decimation at level 2

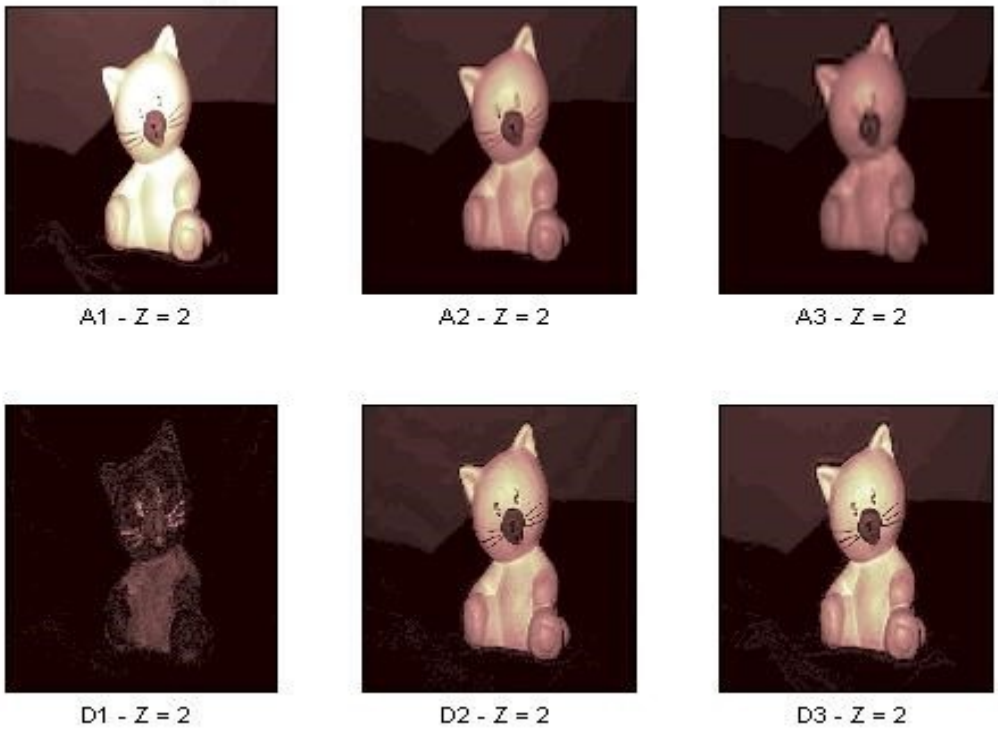


Fig. 4.11: Approximation and Decimation at level 2 for image 1

Fig. 4.12: Approximation and Decimation at level 2 for image 2

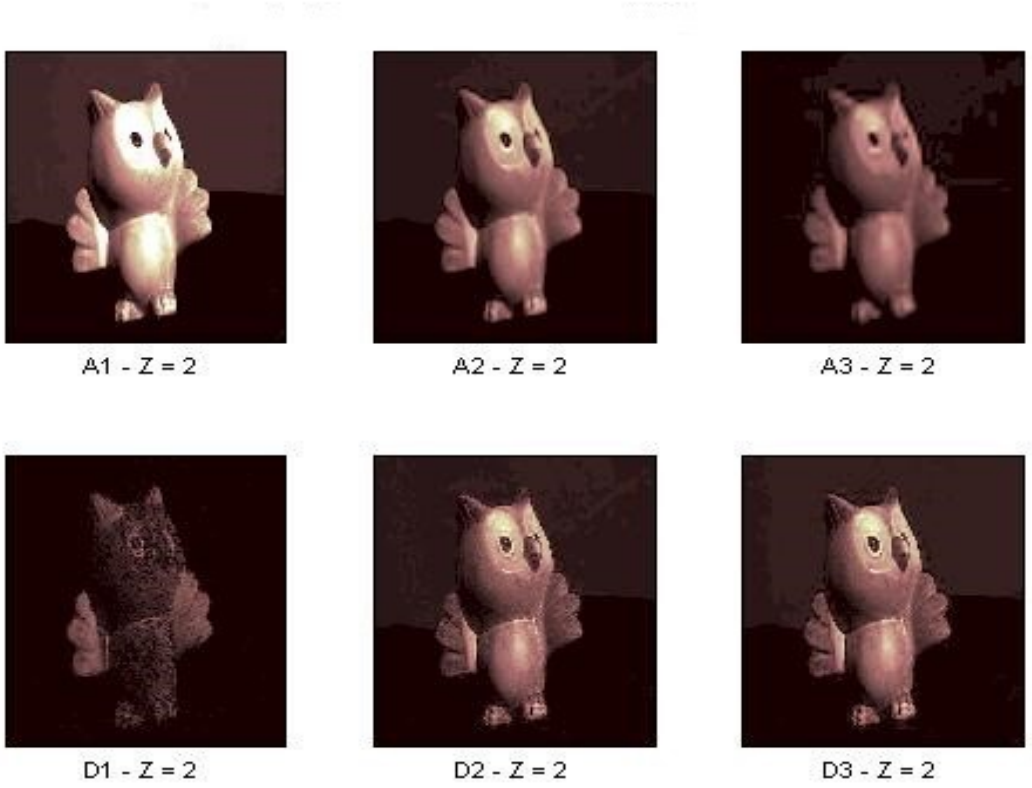


Fig. 4.13: Approximation and Decimation at level 2 for image 3

4.1.6 Approximation and Decimation at level 3

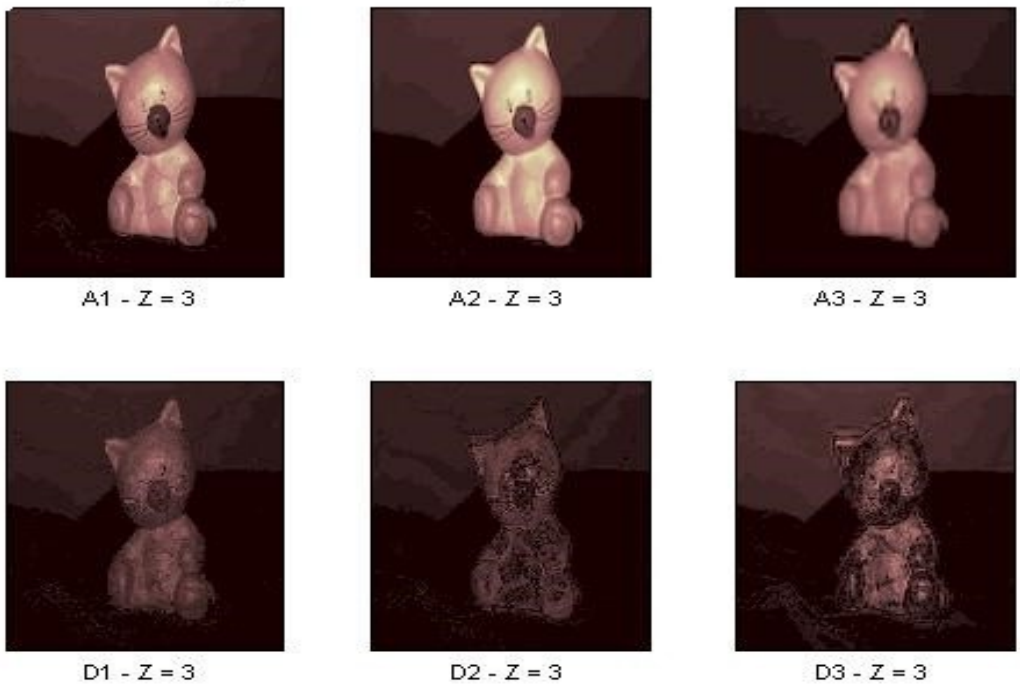


Fig. 4.14: Approximation and Decimation at level 3 for image 1

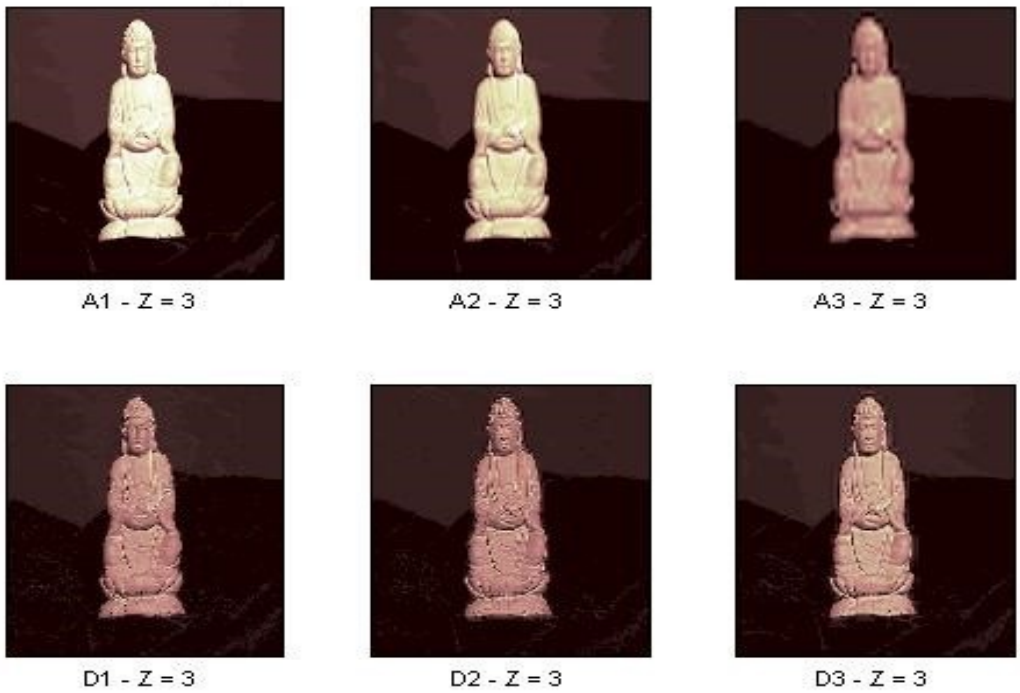


Fig. 4.15: Approximation and Decimation at level 3 for image 2

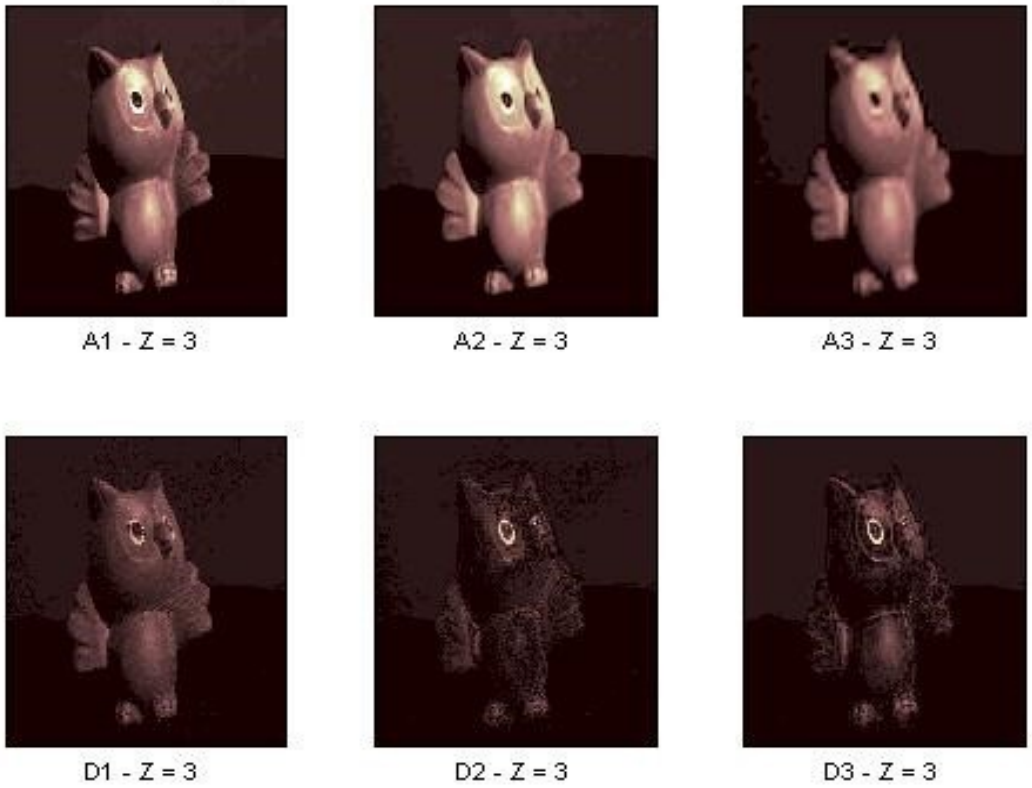


Fig. 4.16: Approximation and Decimation at level 3 for image 3

4.1.7 Three dimensional image

The 3-dimensional image mainly consists of three directions. In three dimensional image, the two 2D images together form a stereoscopic 3D pair, the amount of data for an uncompressed stereo image is doubled compared to that for an uncompressed 2D image. The three dimensional imaging system is formed from the approximation and decimation of the corresponding image. The formation of approximation and decimation images from the original input image form the three dimensional image. The three dimensional images as shown in Fig. 4.17, Fig. 4.18 and Fig. 4.19 respectively.

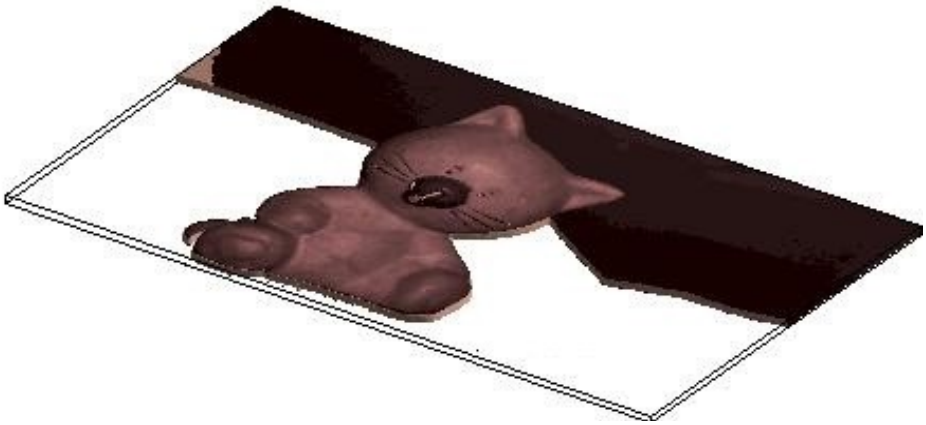


Fig. 4.17: Three dimensional image for image 1

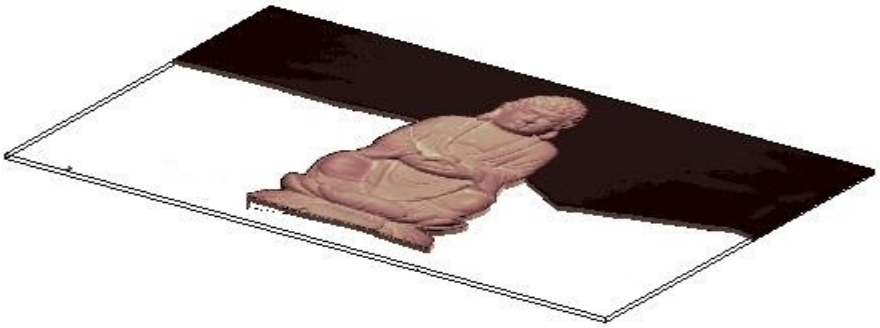


Fig. 4.18: Three dimensional image for image 2

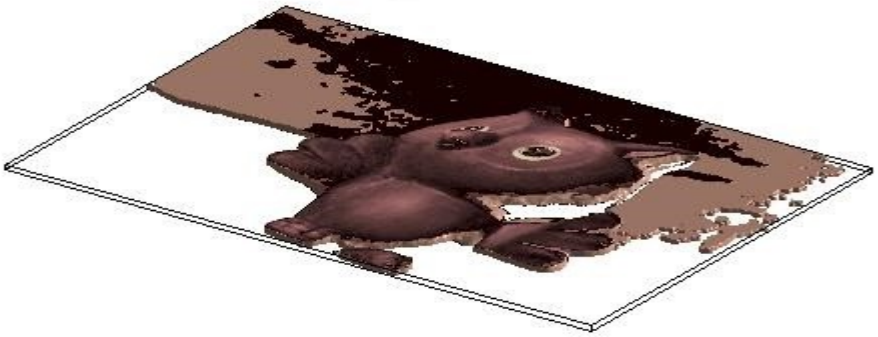


Fig. 4.19: Three dimensional image for image 3

4.1.8 Compressed image

To compress the left and right images of a stereo pair independently, it was verified that even experienced viewers were unable to perceive any coding artifacts when the original and compressed left images were viewed side-by-side as 2D images. Interestingly however, when viewed as a stereo pair, visual artifacts were readily apparent. Indeed, three separate observers examined the images. All three could easily identify significant differences between the original and compressed stereo pairs when viewed side-by-side in 3D mode. The compressed image having the reduction in the image size and also having the better quality in the image. The compressed3-D images are as shown in Fig. 4.20, Fig. 4.21 and Fig. 4.22.

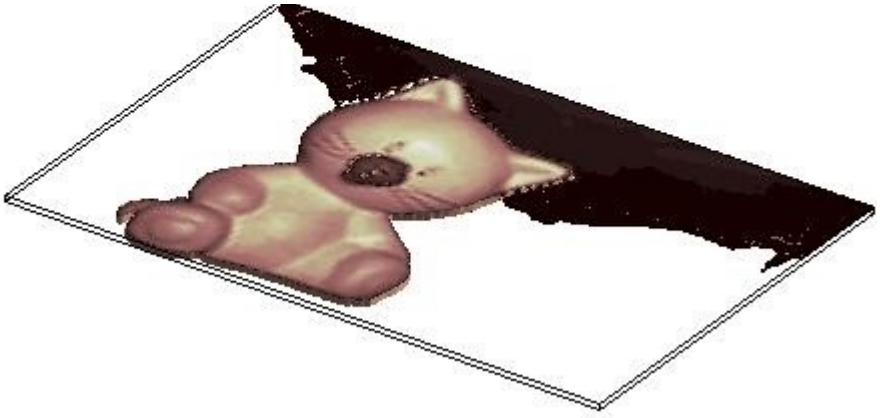


Fig. 4.20: Compressed three dimensional image for image 1

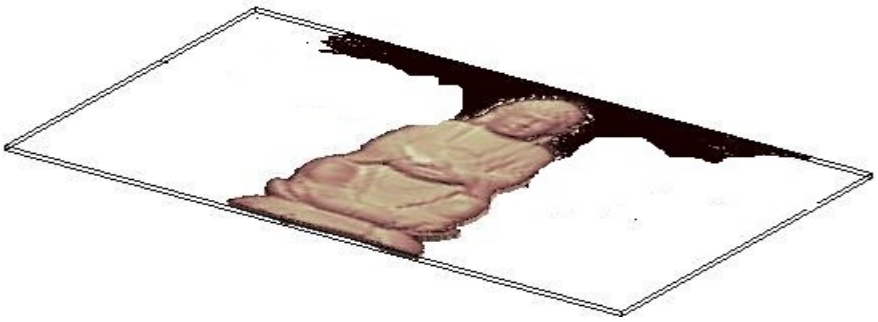


Fig. 4.21: Compressed three dimensional image for image 2

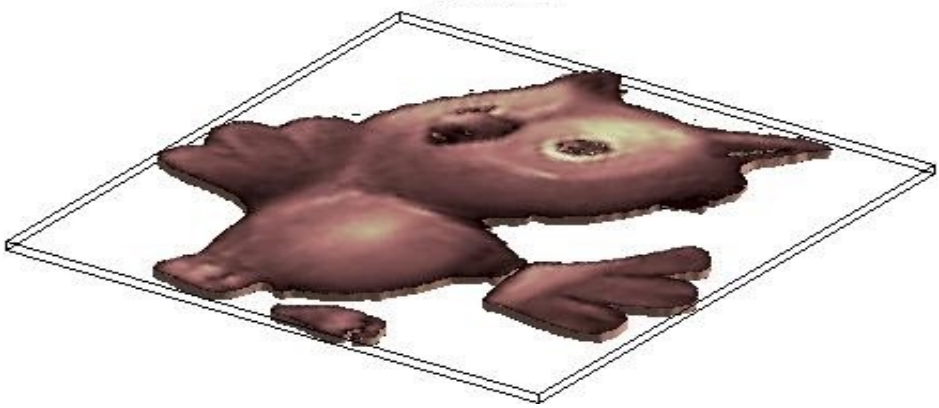


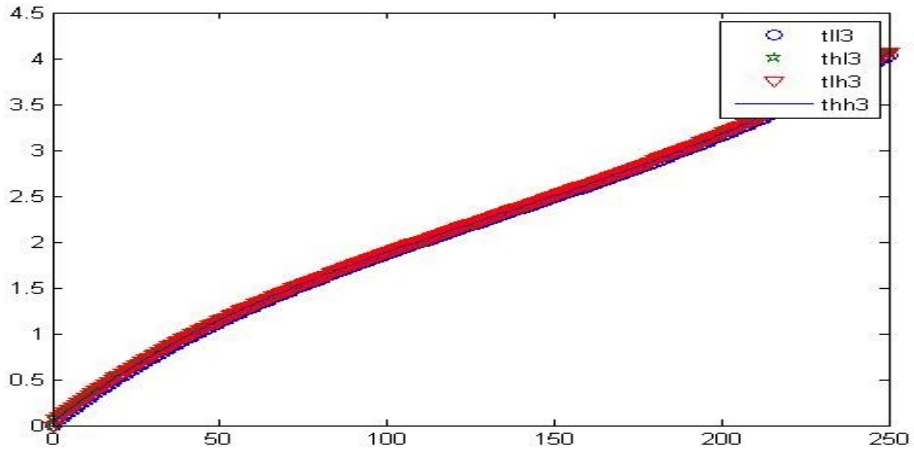
Fig. 4.22: Compressed three dimensional image for image 3

4.1.9 Visibility thresholds for input image 1

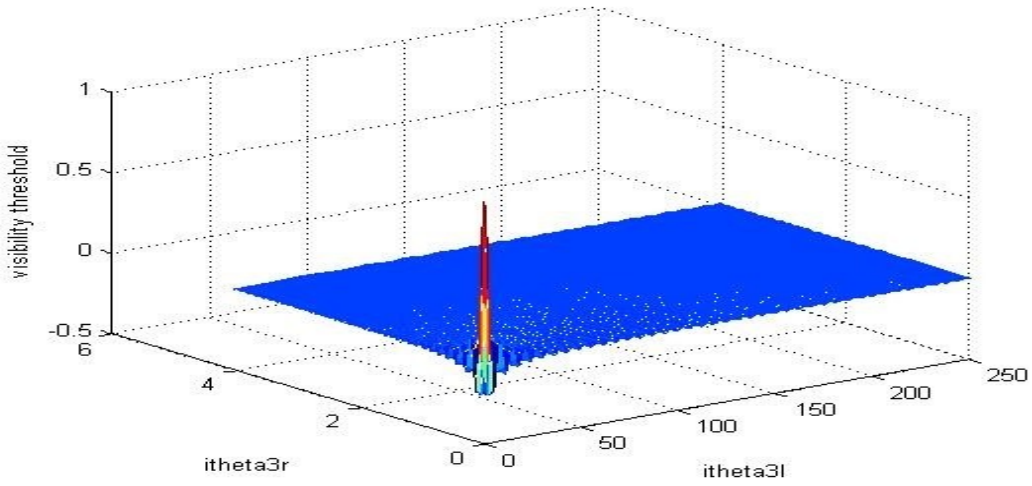
In this case shows the measured values of $t^{I}_{\theta,3,l}(50, I_{\theta,3,l}, I_{\theta,3,r})$ and $t^{II}_{\theta,3,l}(50, I_{\theta,3,l} I_{\theta,3,r})$ for the LL3, HL3, and HH3 subbands of the left image. VTs for LH subbands are similar to those for HL subbands. Thus, thresholds measured for the HL subbands are used for the LH subbands. The VTs for LL3, HL3 and HH3 are represented by circle, triangle and asterisk symbols, respectively. Each subfigure provides graphs of VTs as a function of $I_{\theta,3,r}$ for a fixed value of $I_{\theta,3,l}$. The sensitivity of the right eye to noise introduced only in the left image. In the absence of crosstalk, there would be no such sensitivity and all corresponding VTs would be infinite. The

visibility thresholds for $I_{\theta,3,l}=0$, $I_{\theta,3,l}=60$, $I_{\theta,3,l}=128$, $I_{\theta,3,l}=195$, $I_{\theta,3,l}=255$ as shown in the figures below. Consider for a moment only the blue curves, which correspond to the visibility of noise by the left eye, but plotted as a function of the background gray level in

the right image $I_{\theta,3,r}$. Consider now the red curves. These depict the sensitivity of the right eye to noise introduced only in the left image. In the absence of crosstalk, there would be no such sensitivity and all corresponding VTs would be infinite. The shape of the red curves indicate that the sensitivity to noise due to crosstalk decreases for both low and high background intensity levels in the right image.

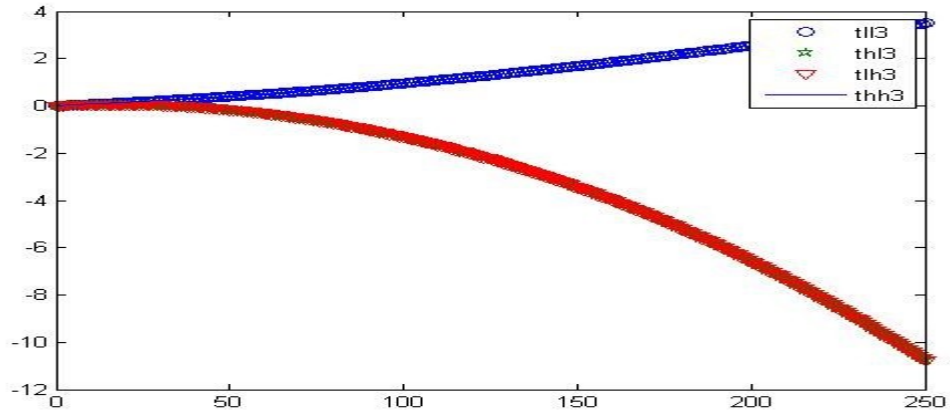


a)

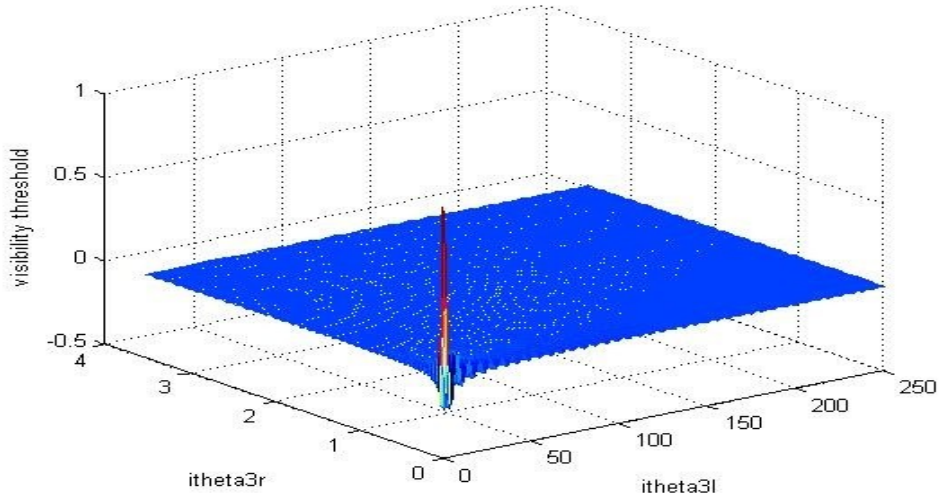


b)

Fig. 4.23 a) and b): Visibility thresholds $I_{\theta,3,l}$=0

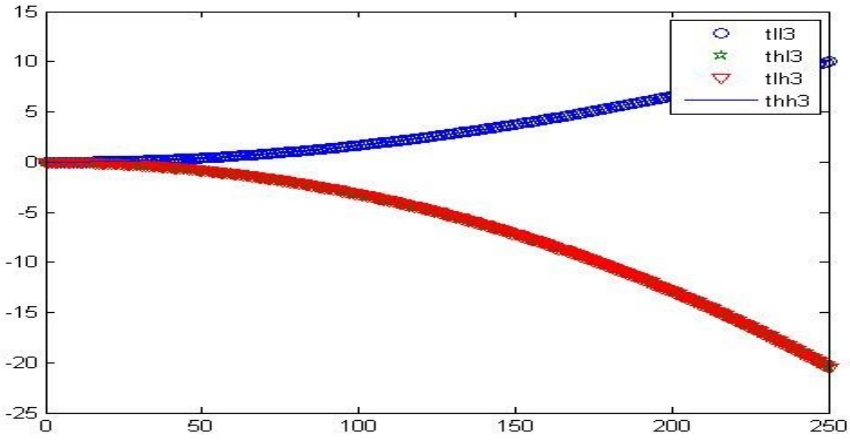


a)

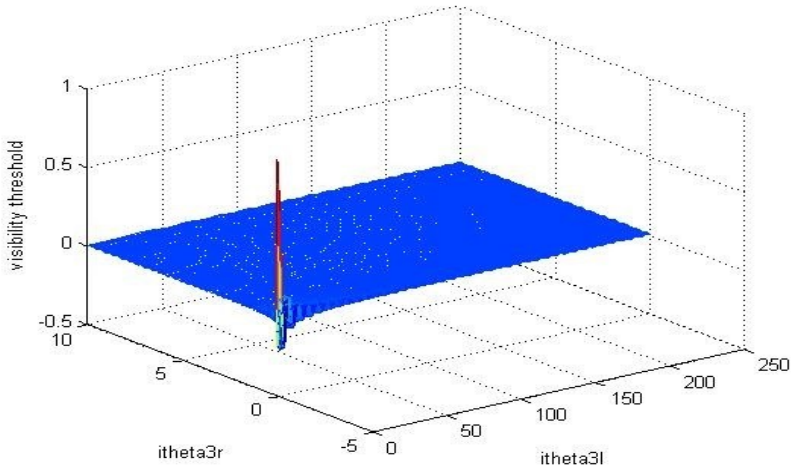


b)

Fig. 4.24 a) and b): Visibility thresholds $I_{\theta,3,l}$=60

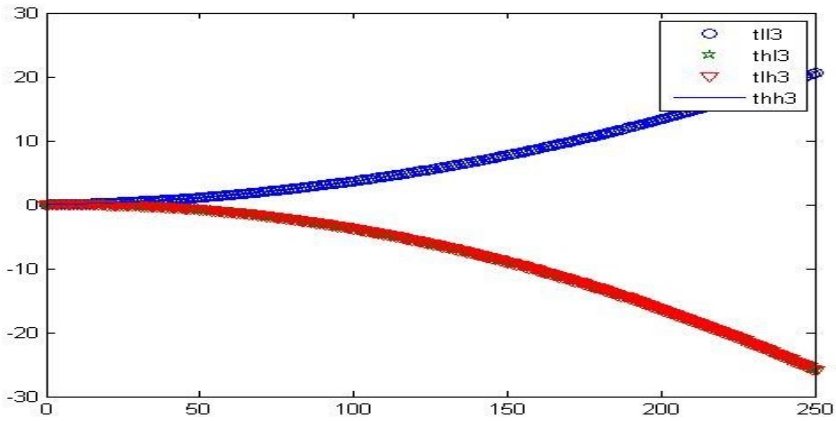


a)

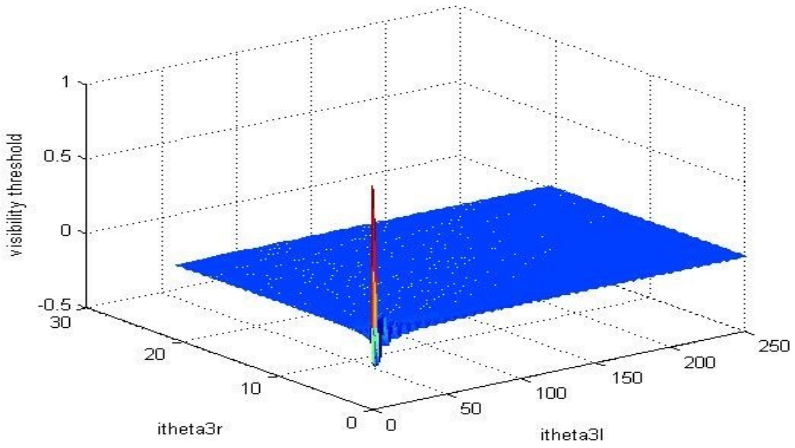


b)

Fig. 4.25 a) and b): Visibility thresholds $I_{\theta,3,l}$=128

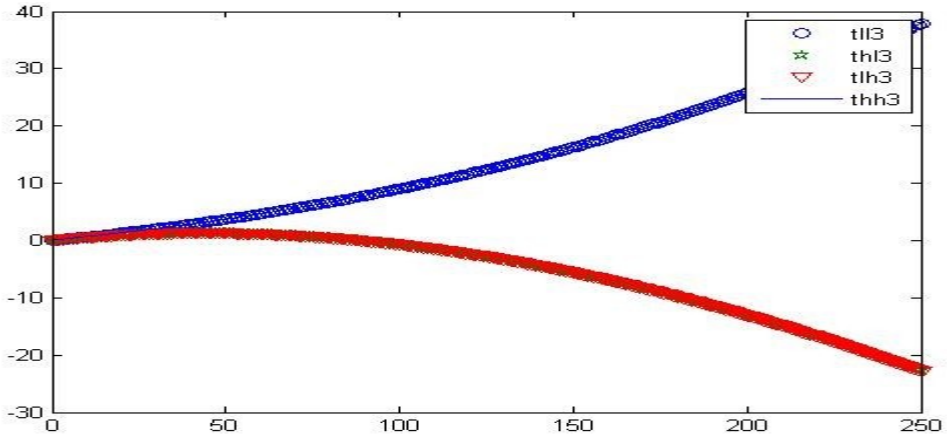


a)

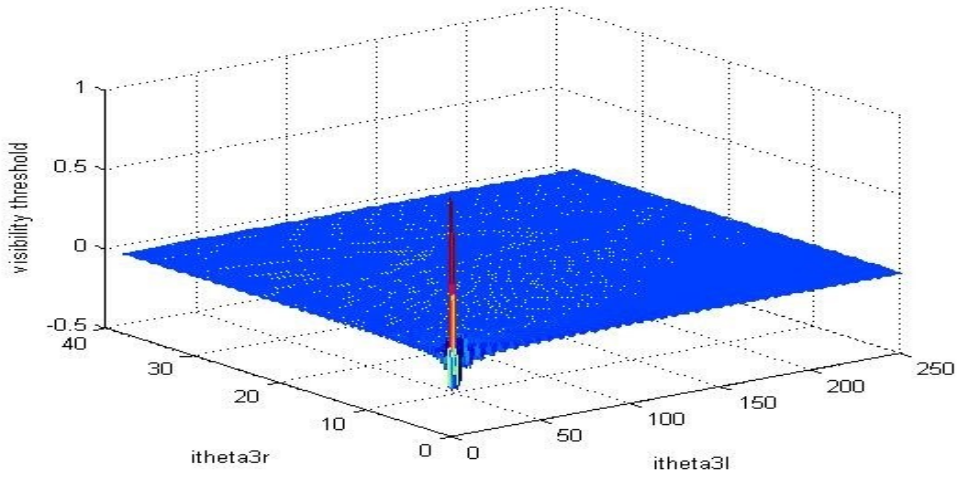


b)

Fig. 4.26 a) and b): Visibility thresholds $I_{\theta,3,l}$=195



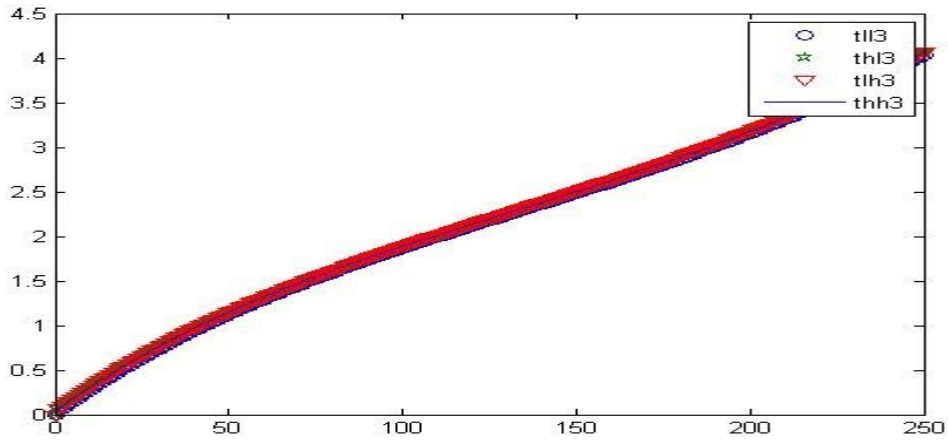
a)



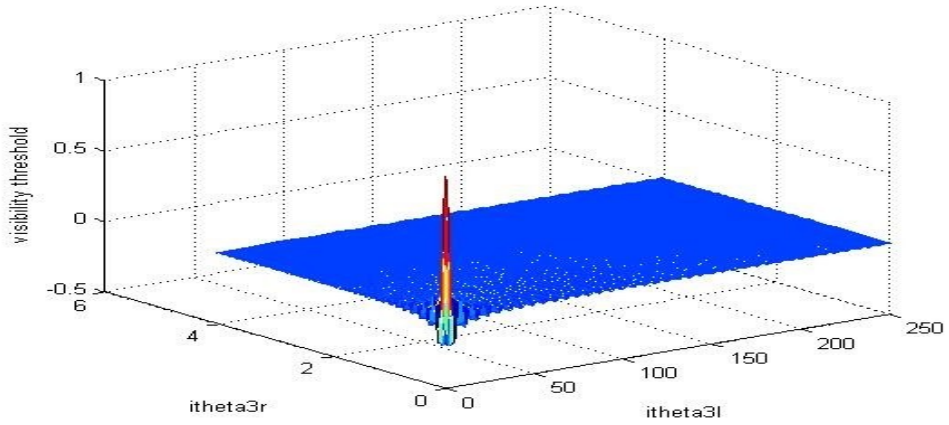
b)

Fig. 4.27 a) and b): Visibility thresholds $I_{\theta,3,l}$=255

4.2.1 Visibility thresholds for input image 2

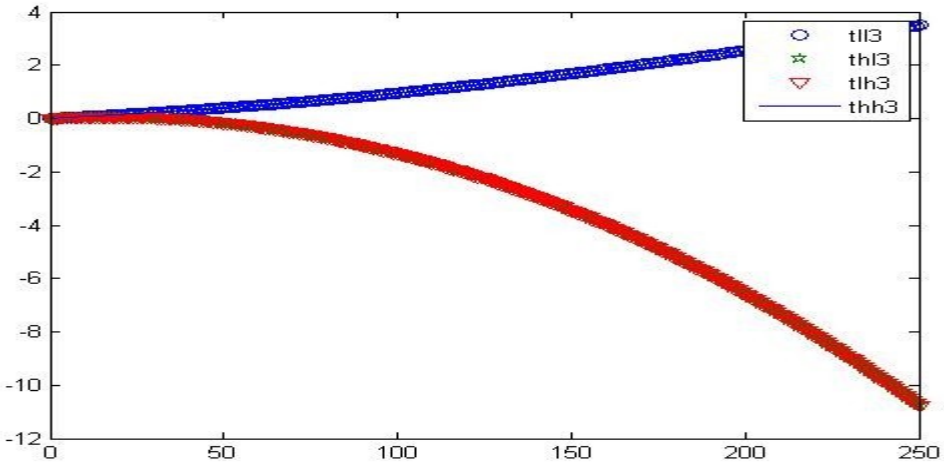


a)

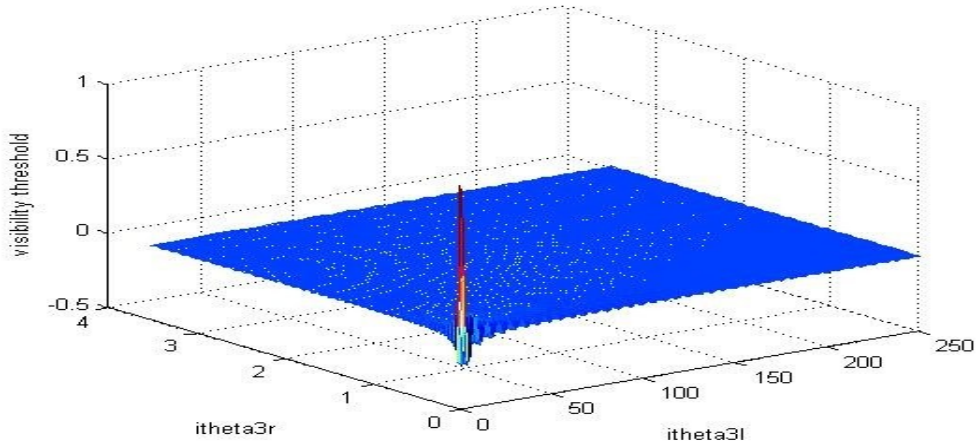


b)

Fig. 4.28 a) and b): Visibility thresholds $I_{\theta,3,l}=0$

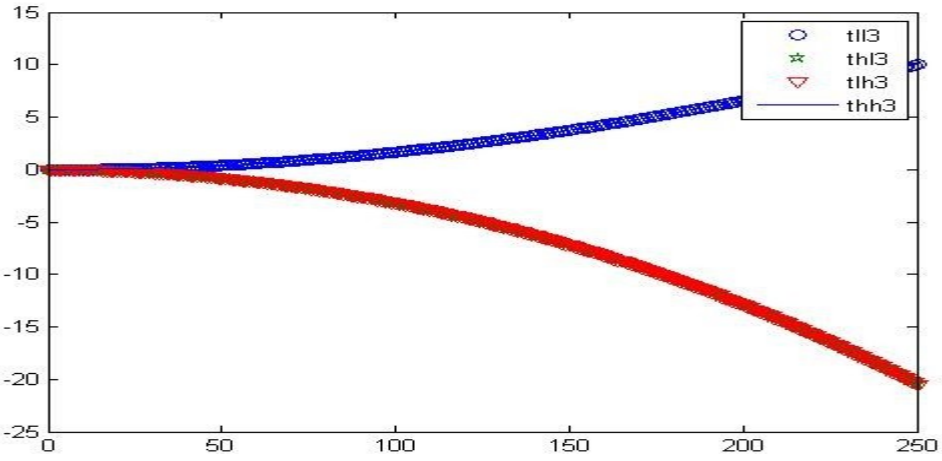


a)

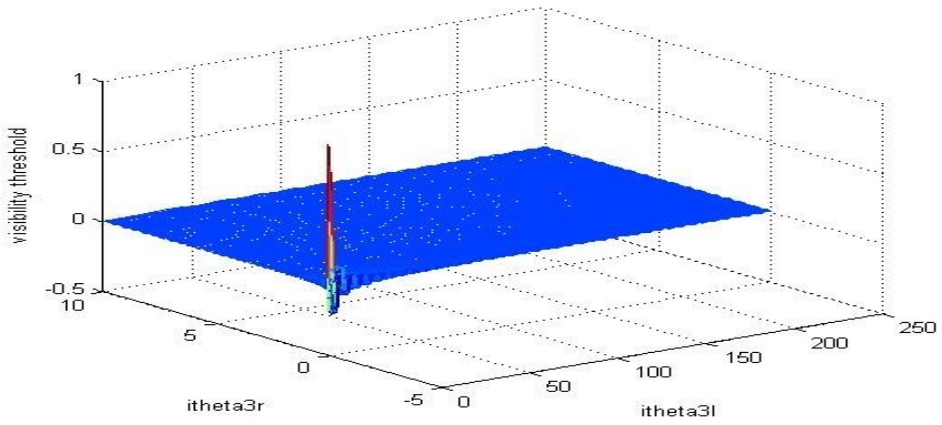


b)

Fig. 4.29 a) and b): Visibility thresholds $I_{\theta,3,l}$=60

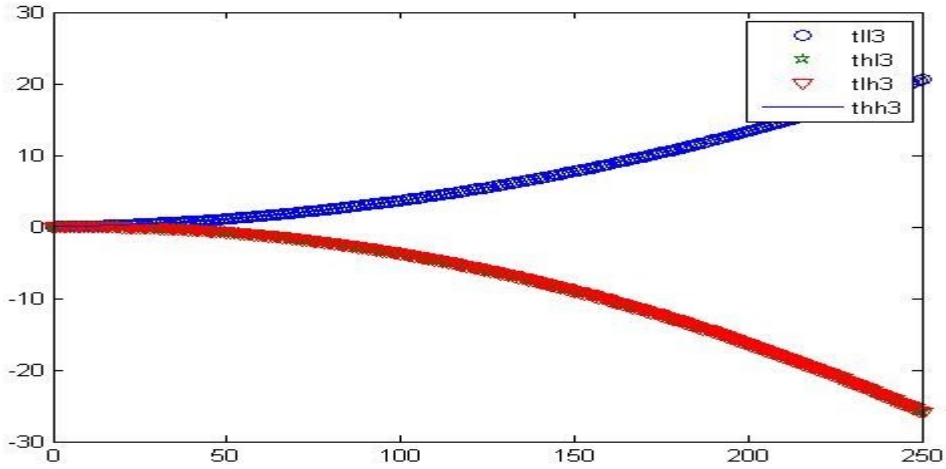


a)

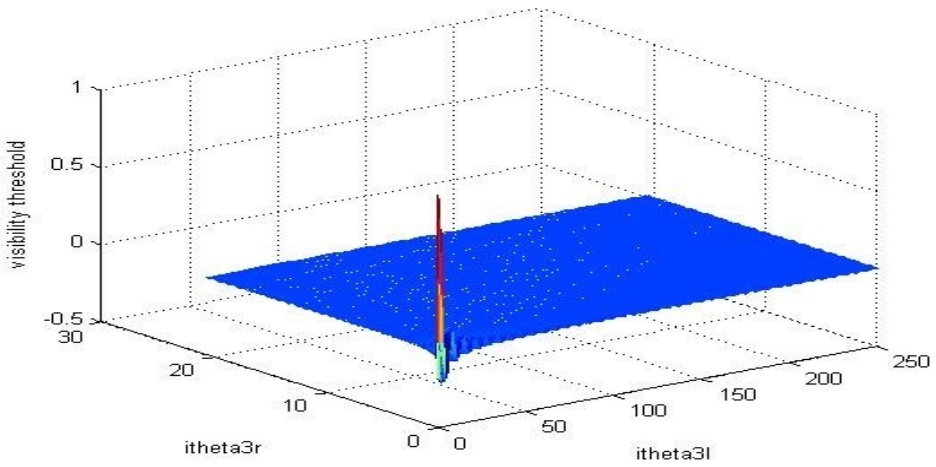


b)

Fig. 4.30 a) and b): Visibility thresholds $I_{\theta,3,l}$=128

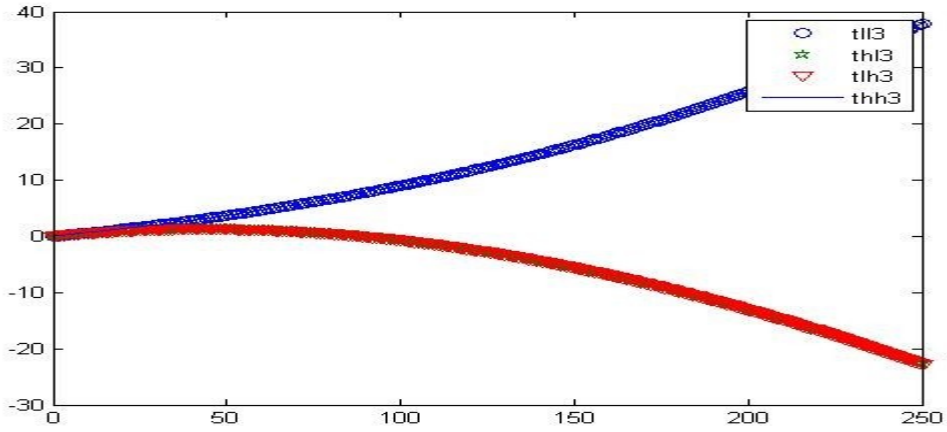


a)

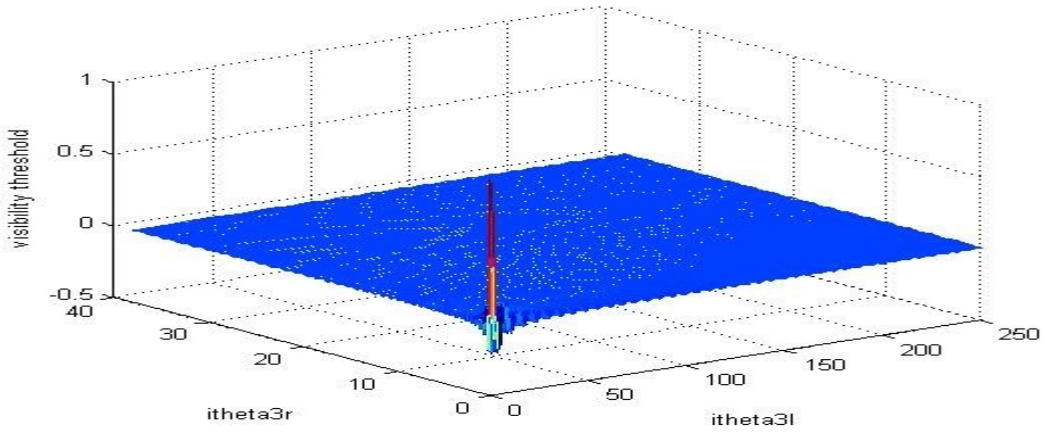


b)

Fig. 4.31 a) and b): Visibility thresholds $I_{\theta,3,l}$=195



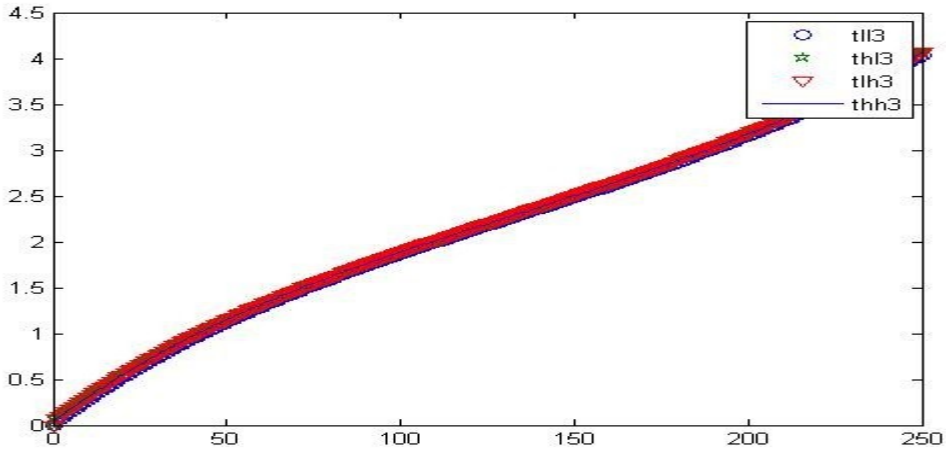
a)



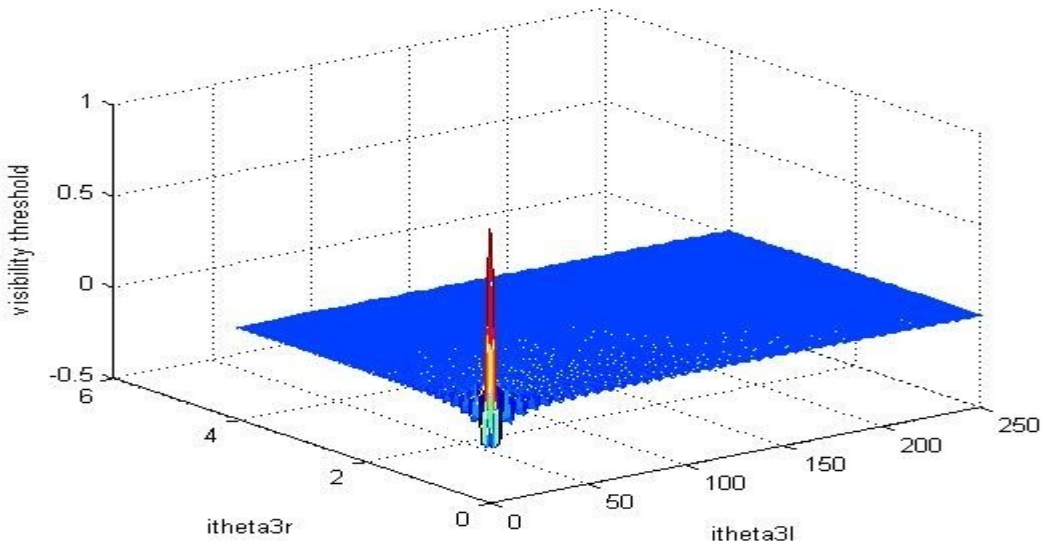
b)

Fig. 4.32 a) and b): Visibility thresholds $I_{\theta,3,l}$=255

4.2.1 Visibility thresholds for input image 3

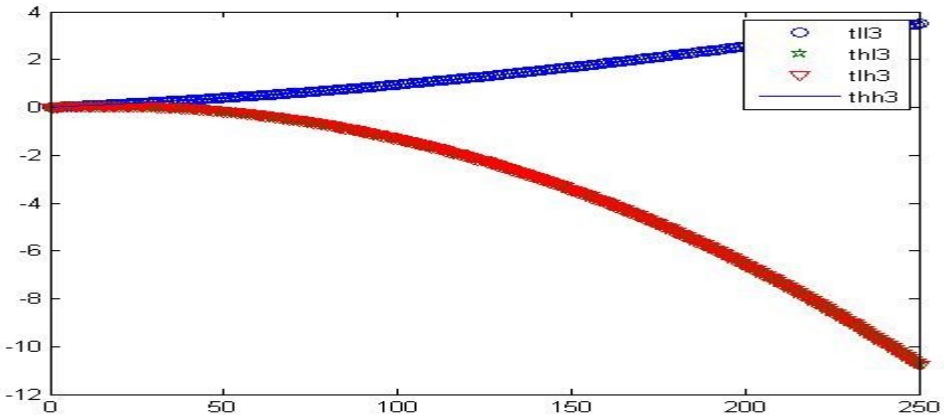


a)

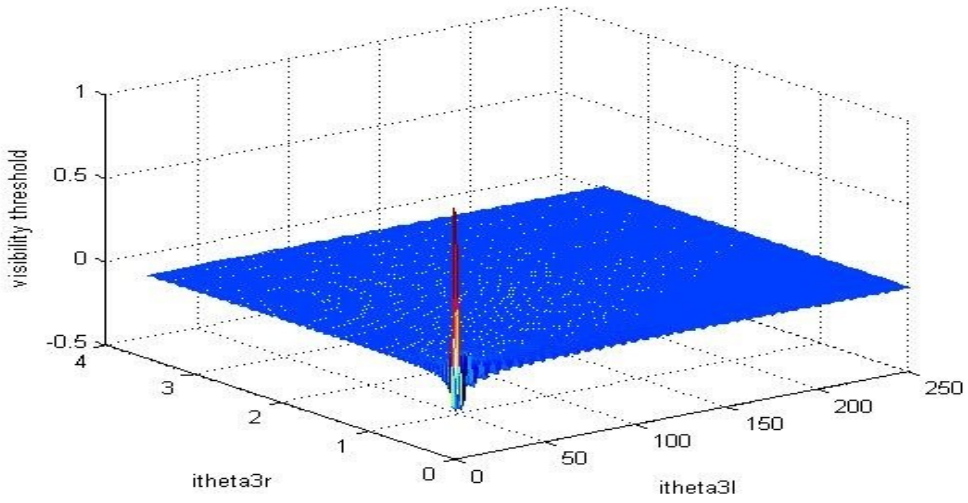


b)

Fig. 4.33 a) and b): Visibility thresholds $I_{\theta,3,l}=0$

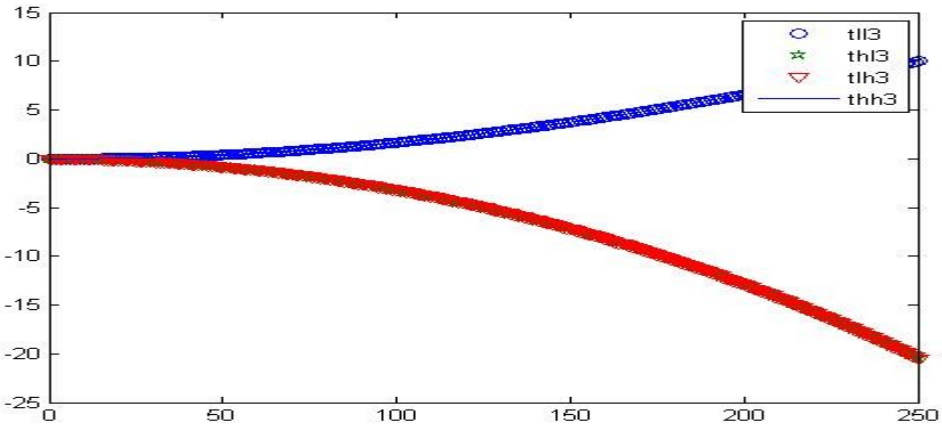


a)

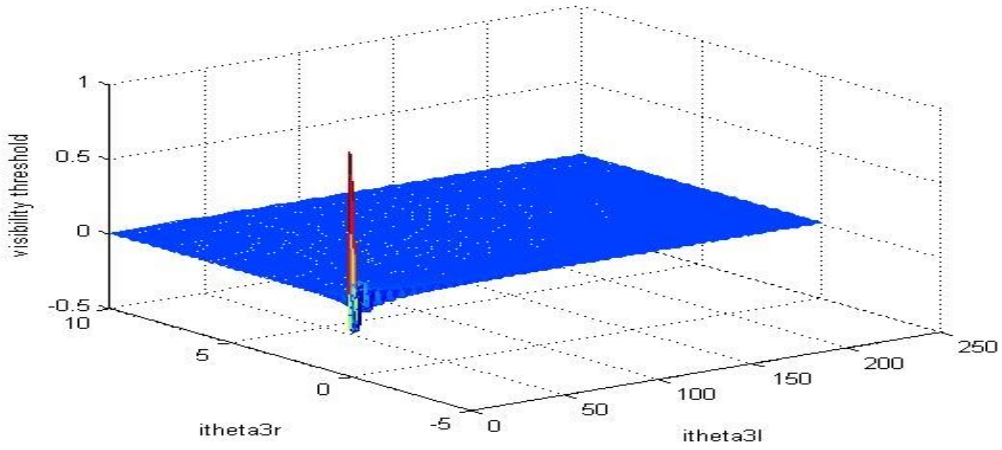


b)

Fig. 4.34 a) and b): Visibility thresholds $I_{\theta,3,l}=60$

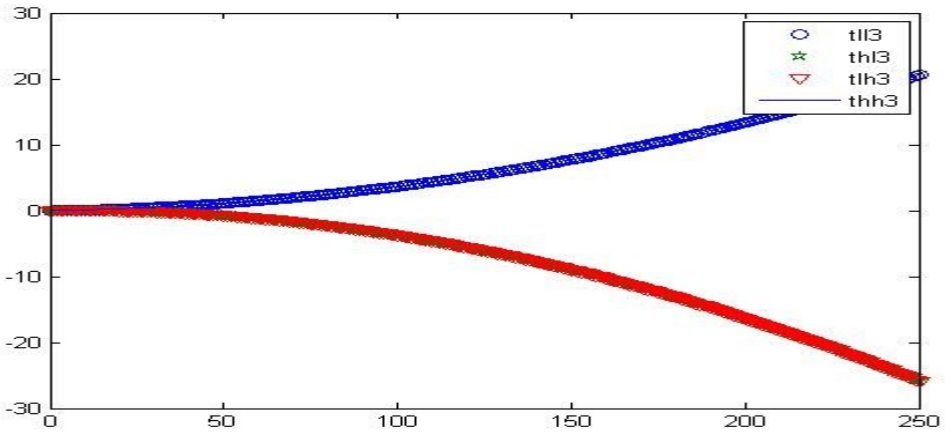


a)

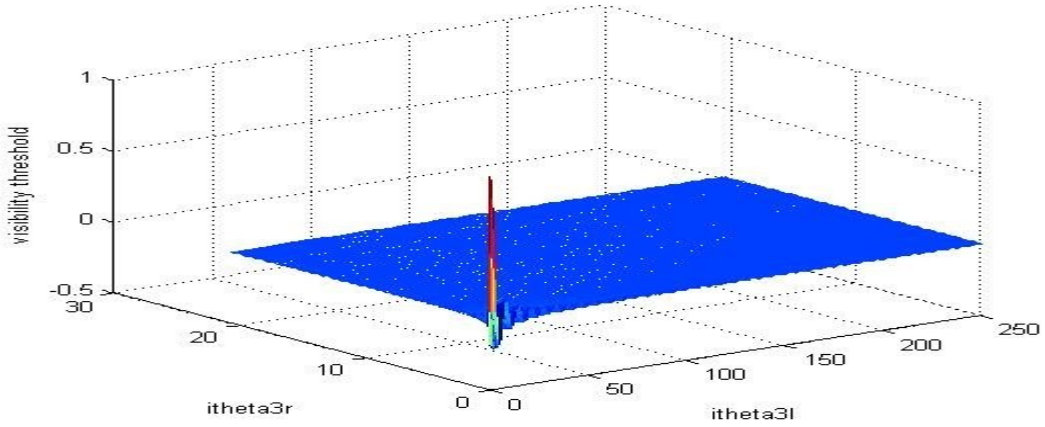


b)

Fig. 4.35 a) and b): Visibility thresholds $I_{\theta,3,l}$=128

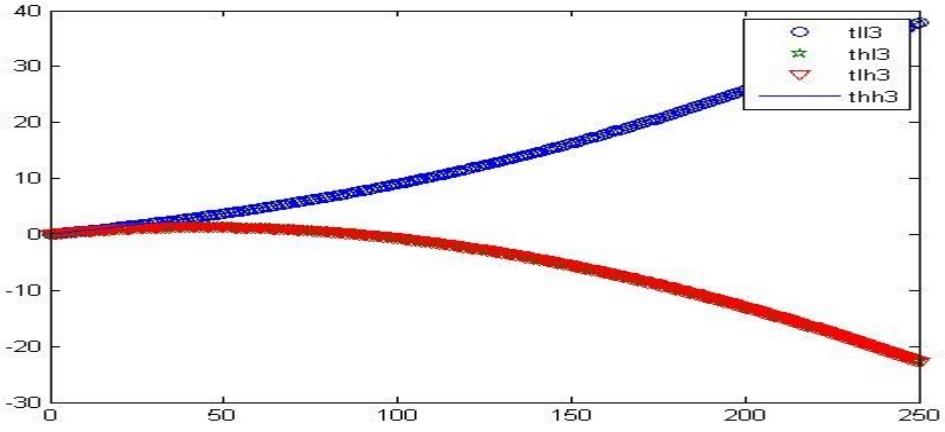


a)

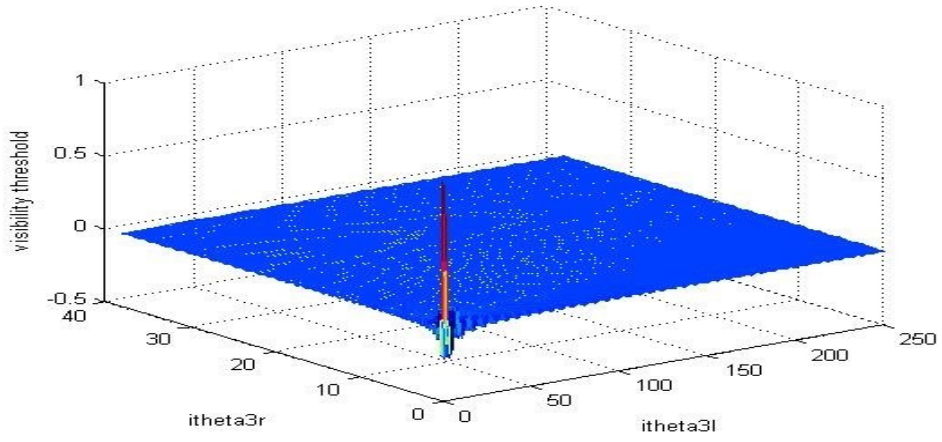


b)

Fig. 4.36 a) and b): Visibility thresholds $I_{\theta,3,l}$=195



a)



b)

Fig. 4.37 a) and b): Visibility thresholds $I_{\theta,3,l}$=255

4.2 Bitrates for visually lossless coding

The bitrates for visually lossless coding as shown in the Table 4.1.

Image	Dimension (W×H)	Error rate	Bitrates
Cat	251×251	0.3214	3.214
Buddha	251×251	0.3588	3.588
Owl	251×251	0.4830	4.830

Table 4.1: Bitrates for visually lossless coding

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

5.1 Conclusion

This work proposes a method of encoding images in a visually lossless manner using JPEG2000. Visually lossless coding shows the compression of images without any perceptible degradation in image quality. The human visual system has varying sensitivity to different color components, spatial frequencies, orientations, and underlying background images. A distortion

model is developed using the distribution of wavelet coefficients and the dead-zone quantizer employed in JPEG2000 and provides higher accuracy than the conventional model. The sensitivity of each sub-band is obtained via psychophysical experiments using random noise generated by a JPEG2000 quantization distortion model and the inverse wavelet transform. In order to hide coding artifacts caused by quantization, visibility thresholds are measured and have different values depending on the local variances within each sub-band. The proposed JPEG2000 coding scheme successfully yields compressed images, whose quality is indistinguishable to those of the original images, at significantly lower bitrates than those of numerically lossless coding and other visually lossless algorithms.

5.2 Future Scope

To improve the performance of the proposed algorithm, the distribution model for wavelet coefficients could be replaced by a more sophisticated model such as Gaussian Scale Mixture, which takes into account spatial correlation between wavelet coefficients.

REFERENCES

[1] H. Oh, A. Bilgin, and M. W. Marcellin, Visually lossless encoding for JPEG2000,IEEE Trans. Image Process., vol. 22, no. 1, pp. 189–201, Jan. 2013.

[2] H.C. Feng, M. W. Marcellin, and A. Bilgin, Visually lossless compression of stereo images, in Proc. IEEE Data Compress. Conf., Snowbird, UT, USA, Mar. 2013, p. 490.

[3] H. C. Feng, M. W. Marcellin, and A. Bilgin, Validation for visually lossless compression of stereo images,in Proc. Int. Telemetering Conf., Las Vegas, NV, USA, Oct. 2013.

[4]A. P. Dal Poz, R. A. B. Gallis, J. F. C. da Silva, and E. F. O. Martins, Object-space road extraction in rural areas using stereoscopic aerialimages, IEEE Geosci. Remote Sens. Lett., vol. 9, no. 2, pp. 654–658, Jul. 2012.

[5] D. P. Noonan, P. Mountney, D. S. Elson, A. Darzi, and G.-Z. Yang, A stereoscopic fibroscope for camera motion and 3D depth recovery during minimally invasive surgery, in Proc. IEEE Int. Conf. Robot. Autom., Kobe, Japan, May 2009, pp. 4463–4468.

[6] L. Lipton, The stereoscopic cinema: From film to digital projection, SMPTE J., pp. 586–593, Sep. 2001.

[7] D. A. Bowman, 3D User Interfaces: Theory and Practice. Boston, MA, USA: Addison-Wesley, 2005.

[8] S. Pastoor and M. Wöpking, 3D displays: A review of current technologies, Displays, vol. 17, no. 2, pp. 100–110, Apr. 1997.

[9] H. Urey, K. V. Chellappan, E. Erden, and P. Surman, State of the art in stereoscopic and autostereoscopic displays, Proc. IEEE, vol. 99, no. 4, pp. 540–555, Apr. 2011.

[10] M. Barkowsky, S. Tourancheau, K. Brunnström, K. Wang, and B. Andrén, Crosstalk measurements of shutter glasses 3D displays, in Proc. SID Int. Symp., 2011, pp. 812–815.

[11] S.-M. Jung *et al.*, Improvement of 3D crosstalk with over-driving method for the active retarder 3D displays, in SID Symp. Dig. Tech. Papers, Seattle, WA, USA, May 2010, pp. 1264–1267.

[12] S. Shestak, D. Kim, and S. Hwang, Measuring of gray-to-gray crosstalk in a LCD based time-sequential stereoscopic display, in SID Dig., Seattle, WA, USA, May 2010, pp. 132–135.

[13] C.-C. Pan, Y.-R. Lee, K.-F. Huang, and T.-C. Huang, Cross-talk valuation of shutter-type stereoscopic 3D display, in SID Dig., Seattle, WA, USA, May 2010, pp. 128–131.

[14] J. D. Yun, Y. Kwak, and S. Yang, Evaluation of perceptual resolution and crosstalk in stereoscopic displays, J. Display Technol., vol. 9, no. 2, pp. 106–111, Feb. 2013.

Appendix: - Project Code

```
clc
clear all
close all
ch=menu('Select input image','cat','budda','owl');
if ch==1
```

```
   filename='cat.tif';
elseif ch==2
   filename='budda.tif';
elseif ch==3
   filename='owl.tif';
end
X=imread(filename);
imshow(X),title('Original Image')
% X=imresize(X,[128 128 27])
% X=matfile(X,'myFile.mat');
% load myFile.mat
% X=imread('c.jpg');
% X=cat(27,size(X))
map = pink(90);
idxImages = 1:1:size(X,3);

figure('DefaultAxesXTick',[],'DefaultAxesYTick',[],...
   'DefaultAxesFontSize',8,'Color','w')
colormap(map)
for k = 1:3
   j = idxImages(k);
   subplot(3,3,k); image(X(:,:,j)); xlabel(['Z = ' int2str(j)]);
   if k==2
      title('Some slices along the Z-orientation of the original  data');
   end
end
perm = [1 3 2];
XP = permute(X,perm);
figure('DefaultAxesXTick',[],'DefaultAxesYTick',[],...
   'DefaultAxesFontSize',8,'Color','w')
colormap(map)
```

```
for k = 1:3
   j = idxImages(k);
   subplot(3,3,k); image(XP(:,:,j)); xlabel(['Y = ' int2str(j)]);
   if k==2
      title('Some slices along the Y-orientation');
   end
end
clear XP
n = 3;                % Decomposition Level
w = 'sym4';           % Near symmetric wavelet
WT = wavedec3(X,n,w);    % Multilevel 3D wavelet decomposition.
A = cell(1,n);
D = cell(1,n);
for k = 1:n
   A{k} = waverec3(WT,'a',k);   % Approximations (low-pass components)
   D{k} = waverec3(WT,'d',k);   % Details (high-pass components)
end
err = zeros(1,n);
for k = 1:n
   E = double(X)-A{k}-D{k};
   err(k) = max(abs(E(:)));
end
disp(err)
nbIMG = 3;
idxImages_New = [1 2 3];
for Mountne = 1:nbIMG
   j = idxImages_New(ik);
   figure('DefaultAxesXTick',[],'DefaultAxesYTick',[],...
      'DefaultAxesFontSize',8,'Color','w')
   colormap(map)
   for k = 1:n
```

```
        labstr = [int2str(k) ' - Z = ' int2str(j)];
        subplot(2,n,k);
        image(A{k}(:,:,j)); xlabel(['A' labstr])
        if k==2
           title(['Approximations and decimation at level 1 to 3 - Slice = ' num2str(j)]);
        end
        subplot(2,n,k+n);
        images(abs(D{k}(:,:,j))); xlabel(['D' labstr])
    end
end
figure('DefaultAxesXTick',[],'DefaultAxesYTick',[],...
        'DefaultAxesFontSize',8,'Color','w')
XR = X;
Ds = smooth3(XR);
hiso = patch(isosurface(Ds,5),'FaceColor',[1,.75,.65],'EdgeColor','none');
hcap = patch(isocaps(XR,5),'FaceColor','interp','EdgeColor','none');
colormap(map)
daspect(gca,[1,1,.4])
lightangle(305,30);
set(gcf,'Renderer','zbuffer'); lighting phong
isonormals(Ds,hiso)
set(hcap,'AmbientStrength',.6)
set(hiso,'SpecularColorReflectance',0,'SpecularExponent',50)
set(gca,'View',[215,30],'Box','On');
axis tight
title('3D image')
figure('DefaultAxesXTick',[],'DefaultAxesYTick',[],...
        'DefaultAxesFontSize',8,'Color','w')
XR = A{2};
Ds = smooth3(XR);
hiso = patch(isosurface(Ds,5),'FaceColor',[1,.75,.65],'EdgeColor','none');
```

```
hcap = patch(isocaps(XR,5),'FaceColor','interp','EdgeColor','none');
colormap(map)
daspect(gca,[1,1,.4])
lightangle(305,30);
set(gcf,'Renderer','zbuffer'); lighting phong
isonormals(Ds,hiso)
set(hcap,'AmbientStrength',.6)
set(hiso,'SpecularColorReflectance',0,'SpecularExponent',50)
set(gca,'View',[215,30],'Box','On');
axis tight
title('compressed image')

p00=2.8563;p10=-0.058;p01=-0.0298;p20=5.126e-4;p11=4.925e-4;p02=1.817e-4;p30=-2.836e-
6;p21=8.072e-7;p12=-2.182e-6;p03=-4.664e-7;p40=7.497e-9;p31=-6.084e-9;p22=8.656e-
9;p13=5.094e-10;p04=6.647e-10;

pll=[2.8563 -0.058 -0.0298 5.126e-4 4.925e-4 1.817e-4 -2.836e-6 8.072e-7 -2.182e-6 -4.664e-7
7.497e-9 -6.084e-9 8.656e-9 5.094e-10 6.647e-10];
phl3=[4.3468 -0.1092 -0.0411 1.198e-3 6.306e-4 2.705e-4 -6.718e-6 -5.579e-7 -3.846e-4
-4.987e-7 1.518e-8 -7.397e-9 9.273e-9 4.483e-9 1.073e-10];
plh3=[4.3468 -0.1092 -0.0411 1.198e-3 6.306e-4 2.705e-4 -6.718e-6 -5.579e-7 -3.846e-4
-4.987e-7 1.518e-8 -7.397e-9 9.273e-9 4.483e-9 1.073e-10];
phh3=[3.9231 -0.1164 -0.0145 1.51e-3 4.118e-4 -3.869e-5 -8.673e-6 -6.727e-7 -1.853e-6
4.085e-7 1.866e-8 -5.732e-9 7.864e-9 2.184e-10 -2.161e-10];
itheta3l=128;
Ttheta3l=0;
for itheta3r=1:250
Ttheta3l=Ttheta3l+pll(1,1)+itheta3l*pll(1,2)+itheta3r*pll(1,3)+itheta3l^2*pll(1,4)+itheta3l*ithet
a3r*pll(1,5)+itheta3r^2*pll(1,6)+itheta3l^3*pll(1,7)+itheta3l^2*itheta3r*pll(1,8)+itheta3l*itheta
3r^2*pll(1,9)+itheta3r^3*pll(1,10)+itheta3l^4*pll(1,11)+itheta3l^3*itheta3r*pll(1,12)+itheta3l^2
*itheta3r^2*pll(1,13)+itheta3l*itheta3r^3*pll(1,14)+itheta3r^4*pll(1,15);
```

```
Tthetahl3=Ttheta3l+phl3(1,1)+itheta3l*phl3(1,2)+itheta3r*phl3(1,3)+itheta3l^2*phl3(1,4)+ithet
a3l*itheta3r*phl3(1,5)+itheta3r^2*phl3(1,6)+itheta3l^3*phl3(1,7)+itheta3l^2*itheta3r*phl3(1,8)
+itheta3l*itheta3r^2*phl3(1,9)+itheta3r^3*phl3(1,10)+itheta3l^4*phl3(1,11)+itheta3l^3*itheta3r
*phl3(1,12)+itheta3l^2*itheta3r^2*phl3(1,13)+itheta3l*itheta3r^3*phl3(1,14)+itheta3r^4*phl3(1
,15);
Tthetalh3=Ttheta3l+plh3(1,1)+itheta3l*plh3(1,2)+itheta3r*plh3(1,3)+itheta3l^2*plh3(1,4)+ithet
a3l*itheta3r*plh3(1,5)+itheta3r^2*plh3(1,6)+itheta3l^3*plh3(1,7)+itheta3l^2*itheta3r*plh3(1,8)
+itheta3l*itheta3r^2*plh3(1,9)+itheta3r^3*plh3(1,10)+itheta3l^4*plh3(1,11)+itheta3l^3*itheta3r
*plh3(1,12)+itheta3l^2*itheta3r^2*plh3(1,13)+itheta3l*itheta3r^3*plh3(1,14)+itheta3r^4*plh3(1
,15);
Tthetahh3=Ttheta3l+phh3(1,1)+itheta3l*phh3(1,2)+itheta3r*phh3(1,3)+itheta3l^2*phh3(1,4)+it
heta3l*itheta3r*phh3(1,5)+itheta3r^2*phh3(1,6)+itheta3l^3*phh3(1,7)+itheta3l^2*itheta3r*phh3
(1,8)+itheta3l*itheta3r^2*phh3(1,9)+itheta3r^3*phh3(1,10)+itheta3l^4*phh3(1,11)+itheta3l^3*it
heta3r*phh3(1,12)+itheta3l^2*itheta3r^2*phh3(1,13)+itheta3l*itheta3r^3*phh3(1,14)+itheta3r^4
*phh3(1,15);
op(itheta3r+1)=Ttheta3l./100;
op1(itheta3r+1)=Tthetahl3./100;
op2(itheta3r+1)=Tthetalh3./100;
op3(itheta3r+1)=Tthetahh3./100;
end

% figure
% subplot(221)
% plot(op)
% subplot(222)
% plot(op1)
% subplot(223)
% plot(op2)
% subplot(224)
% plot(op3)
figure
```

```
plot((0:250),op,'o',(0:250),op1,'p',(0:250),op2,'v',(0:250),op3,'b')
legend('tll3','thl3','tlh3','thh3')
% axis([200 350 -2500 4000])
figure
[X,Y] = meshgrid(0:250);
R = sqrt(X.^2 + Y.^2) + eps;
Z = sin(R)./R;
mesh((0:250),op,Z)
xlabel('itheta3l')
ylabel('itheta3r')
zlabel('visibility threshold')
fprintf('Dimensions are %d %d\n',size(X,1),size(X,2));
```

SIMILARITY INDEX	12%	4%	9%
	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Vietnam Commercial University Student Paper %	3
2	Submitted to Visvesvaraya Technological University Student Paper	2
3	edoc.pub Internet Source %	2
4	Submitted to University of Pretoria Student Paper %	1
5	www.xilinx.com Internet Source %	1
6	www.iisthub.com Internet Source %	1
7	www.ijcotjournal.org Internet Source %	1

www.innofour.se

8 Internet Source

1