



Publisher homepage: www.universepg.com, ISSN: 2707-4625 (Online) & 2707-4617 (Print)

<https://doi.org/10.34104/ijmms.021.022042>

International Journal of Material and Mathematical Sciences

Journal homepage: www.universepg.com/journal/ijmms

International Journal of
Material and
Mathematical Sciences



Enhance the LZW Compression Ratio Through the Use of Image Preprocessing Techniques for Gray Scale Images

Negesse Tadesse^{1*}, Abebe Alemu¹, and Abay Teshager¹

¹Dept. of Information Technology, Faculty of Informatics, University of Gondar, Gondar, Ethiopia.

*Correspondence: ngstds@hotmail.com (Negesse Tadesse, Dept. of Information Technology, Faculty of Informatics, University of Gondar, Gondar, Ethiopia).

ABSTRACT

Compression ratios of encoding algorithms degrade due to signal distortion, additive noise, and hacker manipulation. Large file size costs too much disk space, difficult to analyze, and high bandwidth to transmit over the internet. In this case, compression is mandatory. LZW is a general dictionary-based lossless compression algorithm. It is fast, simple, and efficient when it includes lots of repetitive data or monochrome images. Images with little data repetition and too much-blurred signal, the compression ratio of the LZW algorithm downgraded. Besides this, the execution time of the LZW compression algorithm increases dramatically. To preprocess and analyze the image information the researcher uses LZW encoding algorithm, bit plane slicing technique, Adaptive Median Filter, and MATLAB image processing toolbox. The MATLAB public grayscale image, salt & pepper, Gaussian locavore blurred, and Bayern pattern image data sets are used. Those images dataset is used to test the normal LZW encoding algorithm and the proposed encoding algorithm compression ratio step by step. The noised dataset, the filtered datasets, and bit plane dataset images are processed and recorded quality and compression ratio parameters. The enhanced encoding algorithm average compression ratio is better by far from the normal LZW encoding algorithm by 160%. Not only has the compression ratio, but demising also improved the algorithms execution time. And the image quality metrics measurement of mean square error, peak signal to noise ratio, and structural similarity index measurement are 0, 99, and 1 respectively. This implies the enhanced encoding algorithm could decompress fully without scarfying image quality. The LZW encoding algorithm developmental environment specifies to select tiff and gif image formats. In addition, the LZW encoding algorithm functions are not available in the MATLAB image processing toolbox. The researcher challenged to write a MATLAB script for each personal function. Still, there is room to extend the compression ratio of the LZW encoding algorithm using the image masking technique.

Keywords: Bit-plane-slicing, Lempel-Ziv-Welch coding, Lossless image, LZW, Image nose, and Median filter.

INTRODUCTION:

Image compression is the process of converting data files into smaller files for efficiency of storage and transmission. Data compression treats information in digital form that is, as binary numbers represented by bytes of data with very large data sets (Sangeetha and Betty, 2017). Fox example, a single small 4"×4" size color picture, scanned at 300 dots per inch (dpi)

with 24 bits/pixel of true color, will produce a file containing more than 4 megabytes of data. Although increasing the bandwidth is a possible solution, the relatively high cost makes this less attractive. Therefore, compression is a necessary and essential method for creating image files with manageable and transmittable sizes (Xiao, 2008). In order to be useful, a compression algorithm has a corresponding

decompression algorithm that, given the compressed file, reproduces the original file. There have been many types of compression algorithms developed. These algorithms fall into two broad types, lossless algorithms and lossy algorithms (Al-Khafaji and Bassim, 2019). A lossy (Hussain *et al.*, 2008) algorithm, as its name implies, loses some data. More data can be stored in the memory space if the researcher get our images compressed and transmission will be faster because of the reduced size of the image (Mishra and Singh, 2017).

Data compression algorithms are used in these standards to reduce the number of bits required to represent an image or a video sequence or music (Roy *et al.*, 2018). Such techniques allow one to store an image with much less memory than it would normally require, hence allowing it to be transmitted more quickly (Prashanth and Singh, 2015). Digital image processing simply means the processing of images using a digital computer. Image compression (Kaur and Kaur, 2013) is an application of data compression that encodes the original image with few bits. In lossless compression, there is no loss of any information in the image when the image is decompressed (Gonzalez and Wood, 2019). In the second stage the distinction between the predicted value and the actual intensity of following pixel is coded utilizing diverse encoding techniques (Khan *et al.*, 2018). Lossless algorithm such as Huffman coding, which belongs to Entropy Encoding sub-family is a most used compression method on which based a lot of compression algorithm in particular JPEG (De Luca *et al.*, 2019) where it is possible compress an image opening it in binary mode and reading a single byte like ASCII symbol and after apply Huffman Encoding to generate compression version of raw image.

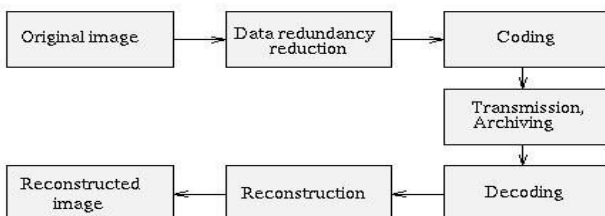


Fig 1: Stages in image processing: object recognition (Abu Taleb *et al.*, 2010).

An efficient bit-level lossless grayscale image compression based on adaptive source mapping enhances the required memory size, execution time, and hardware and software complications for compression/

decompression techniques (Al-Dmour *et al.*, 2016). The proposed linear filters are nearly optimal in a Wiener sense, and in fact outperform many more complex nonlinear filters (Malvar *et al.*, 2004). For this reason, many of the techniques developed for monochrome images can be extended to color images by processing the three component images individually (Gonzalez *et al.*, 2014). The high quality binarized image can give more accuracy in character recognition as compared original image because noise is present in the original image (Puneet and Garg, 2013). The default size of table is 256 to hold pixel values from 0 to 255 for 8 bits (Prabhakar and Ramasubramanian, 2013). LZW compression became the first widely used universal image compression method on computers.

Statement of the problem

Even though it depends on the image format, LZW image compression algorithm enhanced by using RLE and achieve 2.4: 1 average compression ratio (Husseen *et al.*, 2017). By taking information contained in the satellite multispectral images a hybrid lossless method that combining both the LZW and the Arithmetic coding gives better performance than other existing lossless methods in both quality and compression factors (Boopathiraja *et al.*, 2018). Using the new data hiding scheme (namely ODH-LZW), the LZW secret information hiding accuracy is enhanced. The scheme has significantly increased the data hiding capacity specifically by 109.6–203.6% and 28.1–381% for text and gray-scale image data, respectively over state-of-the-art methods (Kumar *et al.*, 2019). LZW technique has its tremendous performance (Badshah *et al.*, 2015) in image compression is better than all other even lossless compression algorithms if the image data is monochrome. Two basic improvements of standard LZW algorithm reduced memory requirement of LZW and reduced total number of searches in order to decode any code during decompression using MSED technique is registered (Bhattacharyya *et al.*, 2017). Recently the LZW encoding decoding time becomes more efficient with the help of GPU Parallel processing (Soobhee *et al.*, 2017). The LZW algorithm compression ratio degraded because of described and un-described reasons. With the help of image watermarking techniques, the compression ratio for gray scale image is summarized below **Table 1**.

Table 1: An average CR of LZW algorithm for RGB image using image watermarking

Algorithm	Sample 1	Sample 2	Sample 3	Sample 4	Sample 5	Sample 6	Average
LZW	0.087	0.088	0.035	0.021	0.036	0.061	0.054

Due to different noises, image distortion and lack of pattern redundancy the compression ratio and searching performance of the LZW algorithm is keep behind every lossless algorithm (Jin, 2019). This would happen because of the absence of a meaningful pattern matching in some images. In future we can improve the image quality and better compression ratio value by changing the value of image compression parameters (Senapati *et al.*, 2018). The compression ratio of the LZW algorithm using different digital image techniques has been enhanced. Noise removal using an adaptive median filter and wavelet signal denoiser and bit plane slicing are the proposed preprocessing techniques to change the image compression parameters. Bit plane slicing creates an eight plane of different pixel for a gray scale image and classified them into least significant bit, centered bit and most significant bit. Such pattern creates a meaningful repetitive pattern for the algorithm. It enhances the execution time of the algorithm.

Research questions

- 1) To identify the best preprocessing techniques to enhance the LZW algorithm compression ratio?
- 2) What is the LZW algorithm best execution time for gray scale image?
- 3) Which full-reference image quality metrics show us the local quality score?

Objectives of this study

General objective - The main objective of this study is to enhance the LZW compression ratio through the use of image preprocessing techniques for gray scale images.

Specific objective

- 1) Prepare dataset images to test both the normal and enhanced LZW algorithm
- 2) To show the application of the bit slicing technique and denoising in the development of an enhanced image compression scheme.
- 3) Identify the right image quality metrics for full referenced images comparisons
- 4) Develop a prototype for the enhanced LZW algorithm

Significance of the study

This paper will be significant to all digital image processor academic research community. In digital image processing the researcher always face storage space, transmission bandwidth, memory and retrieving time problem. Photographers, the researcher's managers, medical professionals and all those whose concern is quality are directly benefited. Since LZW image compression is a lossless algorithm, images will be decoded without any loss of information. For example, CT scan in medicine uses such outputs.

Scope of the study

This paper aims only on enhancing the compression ratio of the LZW algorithm for gray scale images using the proposed preprocessing techniques. Mainly, the researcher goes through LZW algorithm and compress 8-bit depth TIFF images to show some compression ratio enhancement. The images datasets are collected only from the MATLAB public image library.

Limitations

The researcher only selects Bayer pattern images for the testing of the newly build model. This is because it's possible to reconstruct three-dimensional Bayer pattern images from two-dimensional slices.

Digital Image Compression - There are many data compression algorithms which have been developed in the literature and, to date many researches are being carried out to come up with newer, better techniques. These algorithms can be lossy or lossless and have been developed for different applications. Some of the algorithms are developed for general use. They can be used to compress data of different types (Lina, 2009). While some of the algorithms are developed to compress efficiently a particular type of files (Ghadirli *et al.*, 2019). In this study, the researchers are concerned only with lossless image compression types.

Color models and color spaces - A color model is a visualization that depicts the color spectrum as a multidimensional model. A color model is an abstract mathematical model describing how colors can be represented by tuples of numbers (e.g. RGB or

CMYK) (Susstrunk, 2009). A color space is a specific organization of colors. Can have different color spaces for a specific color model; E.g., sRGB and Adobe RGB are color spaces of thereby color model. Typical examples are executable programs, text documents, and source code. Some image file formats, like PNG or GIF, use only lossless compression, while others like TIFF and MNG may use either lossless or lossy methods. Methods for lossless compression are Run length encoding, Huffman encoding, LZW coding and SCZ coding (Gupta et al., 2018). Techniques for lossless image compression (Kaur et al., 2014) Run-length encoding is considered as default method in PCX and as one of

possible in TIFF, BMP and TGA, Huffman Coding, Area image compression, Grooms Coding, DPCM and Predictive Coding, Chain codes, Entropy encoding technique, and LZW. According to their comparative study between Deflate compression algorithms and Lempel-Ziv-Welch (LZW) data compression algorithm, the deflate algorithm is efficient in both compression rate as well as the speed at which the compression is done.

Need for Compression - The main inflection of image compression is to decrease the redundancy of the image and to save or send data through a network in an efficient manner.

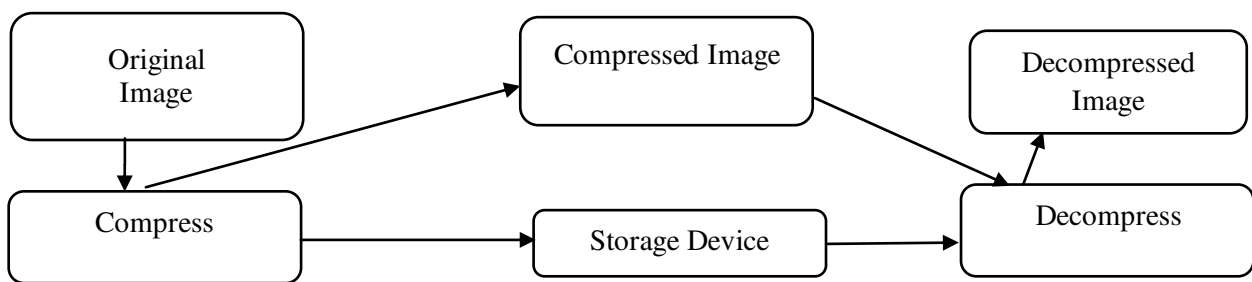


Fig 2: Flow of image compression.

Using fractal concept on the embedded LZW algorithm; compression ratio is better than the standard LZW method by 102% for gray scaled images. In future we can improve the image quality and better compression ratio value by changing the value of image compression parameters. However, in this application it is imperative to determine whether one compression standard will benefit all areas (Halder et al., 2019). Lossless compression gives lesser compression ratio (2:1) as quality of the image cannot be compromised. Lossless compression methods may be categorized according to the type of data they are designed to compress. The common lossless compression methods are Run-Length Encoding (RLE) and LZW.

Run-Length Encoding - Run-length encoding is a simple method of compressing Digitized signals can also have runs of the same value, indicating that the signal is not changing. For instance, an image of the nighttime sky would contain long runs of the character or characters representing the black background.

Huffman Encoding - This method is named after D.A. Huffman, who developed the procedure in the 1950s. A clever feature of Huffman encoding is how the variable length codes can be packed together. Imagine receiving a serial data stream of ones and

zeros. If each character is represented by eight bits, you can directly separate one character from the next by breaking off 8-bit chunks. Now consider a Huffman encoded data stream, where each character can have a variable number of bits.

The LZW Algorithm - A different approach to adaptation is taken by the popular Lempel-Ziv-Welch (LZW) algorithm. LZW also performs the researcher when presented with extremely redundant data files, such as tabulated numbers, computer source code, and acquired signals LZW is the basis of several personal computer utilities that claim to "double the capacity of your hard drive.". The LZW algorithmic basically constructed around a translation table which consists of strings, the algorithm tries to find all of the common substrings and puts a variable size code where they exist and makes them assigned to the generated table which contains previous strings that have been encountered in the message that is to be compressed (Sensing et al., 2016). The patterns emphasize the number of green sensors to mimic the human eye's greater sensitivity to green light. The DE mosaic function uses interpolation to convert the two-dimensional Bayer-encoded image into the true color image (Matlab Mathworks, 2019).

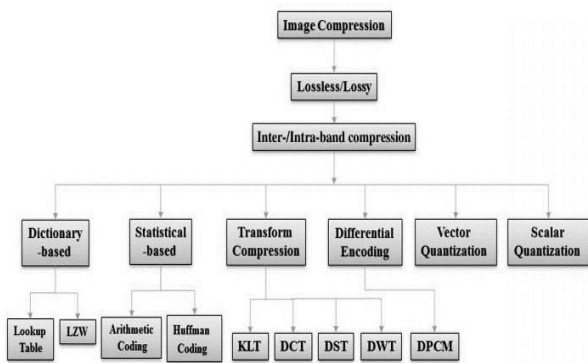


Fig 3: Block diagram of the general categorization of the compression methods (Dusselaar and Paul, 2017).

Image compression is important for web designers who want to create faster loading web pages which make the websites more accessible to others. The motivation is, uncompressed images normally require a large amount of storage capacity and transmission bandwidth. The primary goal of image compression is to minimize the number of bits required to represent the original images by reducing the redundancy in images, while still meeting the user defined quality requirements.

```

begin
  create_empty_dictionary(D)
  for each (a in ABECEDA) do
    begin
      add_to_dictionary(a, D)
    end
  S := ""
  while (!EOF) do
    begin
      readSymbol(z)
      if (find_in_dictionary(Sz, D)) then
        S := Sz
      else
        begin
          output(pointer(S, D))
          add_to_dictionary(Sz, D)
          S := z
        end
      end
    end
  end
end
    
```

Fig 4: LZW compression pseudo code.

Gray Scale Images

A grayscale image pixel tall and pixels wide is represented as a matrix of double data type of size. Element values (such as) denote the pixel grayscale intensities in with 0=black and 1=white. For processing purpose, we can drive the gray scale images from RGB or true color images. For example:



Fig 5: Lena gray (nxm).

Table 2: Gray Scale Lena Image Information Tips

Size (gray_ lean	Gray (400,400, :)	Ans. Filename	Ans. File Size	Ans. Format
Ans =512 512	182	Lena gray	Ans =102379	Ans ='pang'

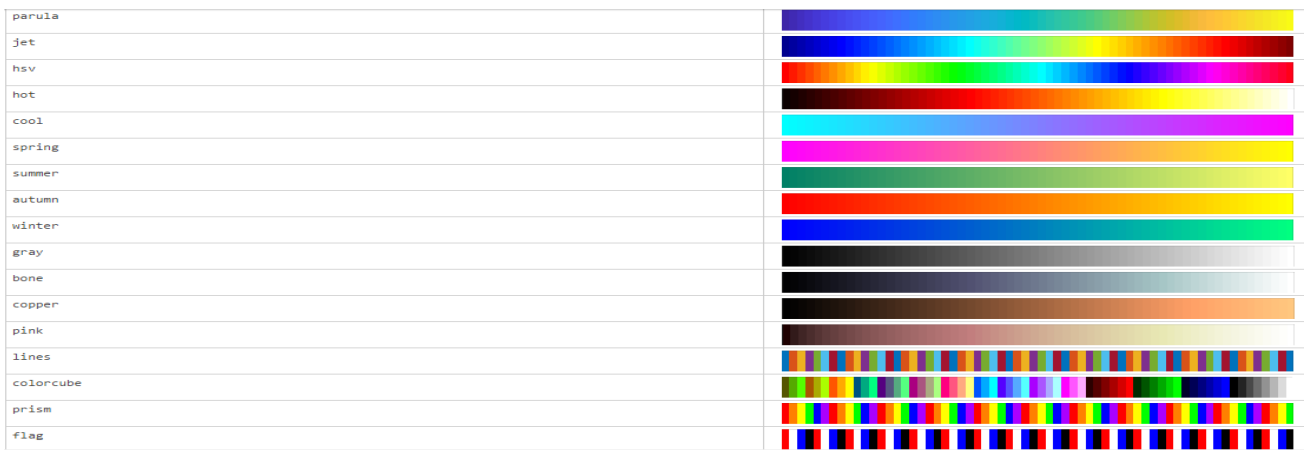


Fig 6: MATLAB Default Colormap; web (full file (do root, 'mat lab/ref/colormap.html')).

MATLAB provides us different color maps for gray scaled images. The defaults are:

So, we can display Lena gray (**Fig 6**) in summer color map as follows:

```
Gray = inroad ('lena_color.png');
```

```
I show (gray); colormap hot;
```

```
I show (gray); colormap hot;
```

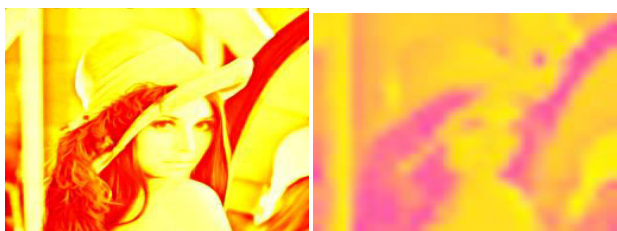


Fig 7: Hot and spring colormap respectively.

Image Preprocessing Techniques for LZW -

Image preprocessing techniques are those methods which enhances restores and maintains the quality of the image. In this section, the paper will discuss only the preprocessing techniques related to reduction of size without scarifying quality.

Attribute Filtering - Binary Attribute Filtering, The Max-Tree Approach, The Volume Attribute and The Vision Attribute are the widely used attribute filtering techniques to improve different compression algorithms. Experiments (You *et al.*, 2018) have shown that all the filters cause an improvement of as much as 11, 10 and 20% for jpeg, jpeg2000 and LZW algorithms respectively.

Histogram - Among the characteristics found relatively frequently in computer-generated images, but those are usually not found in natural images, is intensity histogram sparseness.

Entropy Coding - This technique is to replace original data of gray scale images with particular ordered data so that performance of lossless compression can be improved more efficiently. When compressing ordered image using entropy encoder, the researcher can expect to raise compression rate more highly because of enhanced statistical feature of the input image. Scholars (Cadena *et al.*, 2017) show that lossless compression rate increased by up to 37.85% when comparing results from compressing preprocessed and non-preprocessed image data using entropy encoder such as Huffman, Arithmetic encoder, LZW.

Bi-histogram equalization with a plateau value (BHEPL) - The BHEPL is use to short processing time for image enhancement. I-Histogram Equalization with a Plateau Value (BHEPL) is similar to Brightness Preserving Bi-Histogram Equalization (BBHE). BHEPL enhancement method is the combination of two methods first one is BBHE and second one is clipped histogram. Absolute Mean Brightness Error (AMBE) is used to measure the performance of enhancement on the input images.

Digital Filter - Various filters (Cadena *et al.*, 2017) are used for medical image preprocessing such as mean filter, Gaussian filter, median filter and 2D Cleaner. The primary purpose of these filters is a noise reduction, but filter can also be used to emphasize certain features of an image or remove other features. Most of image processing filters can be divided into linear filters and nonlinear filters. Non-linear filters include order statistic filters and adaptive filters. The choice of filter is often determined by the nature of the task and the type and behavior of the data.

Watermarking - LZW has been used successfully for watermark lossless compression to watermark medical images in teleradiology to ensure less payload encapsulation into images to preserve their perceptual and diagnostic qualities unchanged. Medical image security is one of these applications based on watermarking of medical images. With the help of the LZW algorithm, the researcher can preserve image information while hiding from hacker manipulation and different security issues. Watermarking image information onto another ordinary, small size and portable format is obvious.

Contrast enhancement - Image contrast enhancement is to improve the contrast level of images, which are degraded during image acquisition. Image contrast enhancement aims to improve the contrast level of images, since the image quality can suffer due to several factors, such as contrast, illumination and noise during image acquisition procedure. Image contrast is defined as the separation factor between pattern and the brightest spot and the darkest spot in images (Chen *et al.*, 2018). A larger separation factor indicates higher contrast; on the other hand, the smaller separation factor indicates the contrast. Image contrast enhancement is useful in many real-world application areas.

Bit Plane Slicing - A bit plane is a set of bits corresponding to a given bit position in each of the binary numbers in an image. It is used to determine the adequacy of numbers of bits used to quantize each pixel in the image (Pokle and Bawane, 2017). Bit plane slicing is the conversion of image into multilevel binary image. These binary images are then compressed using different algorithm. With this technique, the valid bits from gray scale images can be separated, and it will be useful for processing these data in very less time complexity (Image and

Project, 2014). The first step is to slice the grayscale images into eight binary monochrome images by using bit-plane slicing. The generated binary images contain redundant bits. Because the number of color decreases to 2 colors black (0) and white (1).

Histogram - An image histogram is a type of histogram that acts as a graphical representation of the tonal distribution in a digital image. It plots the number of pixels for each tonal value. The Histogram Equalization block enhances the contrast of images by transforming the values in an intensity image so that the histogram of the output image approximately matches a specified histogram. This enhancement algorithm is based on plateau histogram equalization for infrared images. By analyzing the histogram of image, the threshold value is got self-adaptively.

Noise Removal - The researcher may define noise to be any degradation in the image signal, caused by external disturbance. Noise models also designed by probability density function using mean, variance and mainly gray levels in digital images (Chiranjeevi and Jena, 2016). Image noise is generally regarded as an undesirable by-product of image capture. Although these unwanted fluctuations became known as "noise" by analogy with unwanted sound they are inaudible and such as dithering. The standard model of amplifier noise is additive, Gaussian, independent at each pixel and independent of the signal intensity. In color cameras where more amplification is used in the blue color channel than in the green or red channel, there can be more noise in the blue channel. Amplifier noise is a major part of the "read noise" of an image sensor, that is, of the constant noise level in dark areas of the image.

$$P(g) = \sqrt{\frac{1}{2\pi\sigma^2} e^{-\frac{(g-\mu)^2}{2\sigma^2}}}$$

Salt-and-pepper noise - An image containing salt-and-pepper noise will have dark pixels in bright regions and bright pixels in dark regions. This type of noise can be caused by dead pixels, analog-to-digital converter errors, bit errors in transmission, etc. This can be eliminated in large part by using dark frame subtraction and by interpolating around dark/bright pixels. There are so many denoising algorithms have developed to recover a noise corrupted image. However, most of them cannot well recover a heavy noise corrupted image with noise density above 70% (Lu and Chou, 2012).

Poisson noise - Poisson noise or shot noise is a type of electronic noise that occurs when the finite number of particles that carry energy, such as electrons in an electronic circuit or photons in an optical device, is small enough to give rise to detectable statistical fluctuations in a measurement.

Speckle noise - Speckle noise is a granular noise that inherently exists in and degrades the quality of the active radar and synthetic aperture radar (SAR) images. It is caused by coherent processing of back-scattered signals from multiple distributed targets. In SAR oceanography, for example, speckle noise is caused by signals from elementary scatters, the gravity-capillary ripples, and manifests as a pedestal image, beneath the image.

Removing Noise by Adaptive Filtering - The `wiener2` function applies a Wiener filter to an image adaptively, tailoring itself to the local image variance. In addition, there are no design tasks; the `wiener2` function handles all preliminary computations and implements the filter for an input image. `wiener2`, however, does require more computation time than linear filtering. Wiener2 works best when the noise is constant-power ("white") additive noise, such as Gaussian noise (Awadala *et al.*, 2020).

Image Quality Metrics - Image quality can degrade due to distortions during image acquisition and processing. The color image includes three characteristics: chroma, sharpness and contrast. Judging the difference between image pairs, the three aspects should be included. The chroma attribute consists of lightness, saturation and hue. The sharpness represents the image edge and texture (Rippel and Bourdev, 2017). The contrast represents the distribution of image tones.

Full-Reference Quality Metrics - When the researcher care about quality of an image compressed using lossless compression algorithm, the researcher needs to make a comparison (Waliullah *et al.*, 2020). Since the researcher have an original input image and image after decompression, the researcher better use full-referenced quality metrics.

- **imams** - Mean-squared error (MSE). MSE measures the average squared difference between actual and ideal pixel values.
- **psnr** - Peak signal-to-noise ratio (pSNR). pSNR is derived from the mean square error, and indicates the ratio of the maximum pixel

intensity to the power of the distortion. Like MSE, the pSNR metric is simple to calculate but might not align well with perceived quality.

- swim - Structural Similarity (SSIM) Index. The SSIM metric combines local image structure, luminance, and contrast into a single local quality score. Because structural similarity is computed locally, swims can generate a map of quality over the image (Lu and Chou, 2012).

$$PSNR = 20 \log_{10} \left(\frac{MAX_f}{\sqrt{MSE}} \right)$$

(Peak Signal-to-Noise Equation)

Where the **MSE** (*Mean Squared Error*) is:

$$MSE = \frac{1}{mn} \sum_0^{n-1} \sum_0^{m-1} |f(i, j) - g(i, j)|^2$$

(Mean Squared Error Equation)

Legend:

f represents the matrix data of our original image. **g** represents the matrix data of our degraded image in question.

m represents the numbers of rows of pixels of the images and **I** represent the index of that row. **n** represents the number of columns of pixels of the image and **j** represents the index of that column.

Max_i is the maximum signal value that exists in our original “known to be good” image.

Table 3: Summarized LZW Image Compression literatures

Author	Objective	technique	Dimensionality	#Experiments	Compression ratio
Abu Taleb, (2010)	Compression ratio improvement	Bit plane slicing	Gray scale	One experiment with 5 samples	2.349553275
Gran Badshah <i>et al.</i> (2015)	lossless recovery	Watermarking	RGB	One experiment with 6 samples	0.077
Akoguz <i>et al.</i> (2016)	Compression ratio improvement	Huffman combination	Gray and RGB	One experiment with 5 samples	Gray: 120% RGB: 56%
Beom Kwon and Sanghoon	Error Detection Algorithm	LZ7	Both 2-D and 3-D	Two experiment	–
Aziz Makandar, (2018)	Region detection	Bit plane Slicing Edge detection	Alzheimer detection Gray scale image	Six consecutive experiments	–
Monther Aldwairi, (2019)	Providing security and privacy without sacrificing efficiency	multiple patterns matching search	True color tiff and gif images	Single experiment with 10 samples	

The Lempel-Ziv-Welch/LZW/Image compression

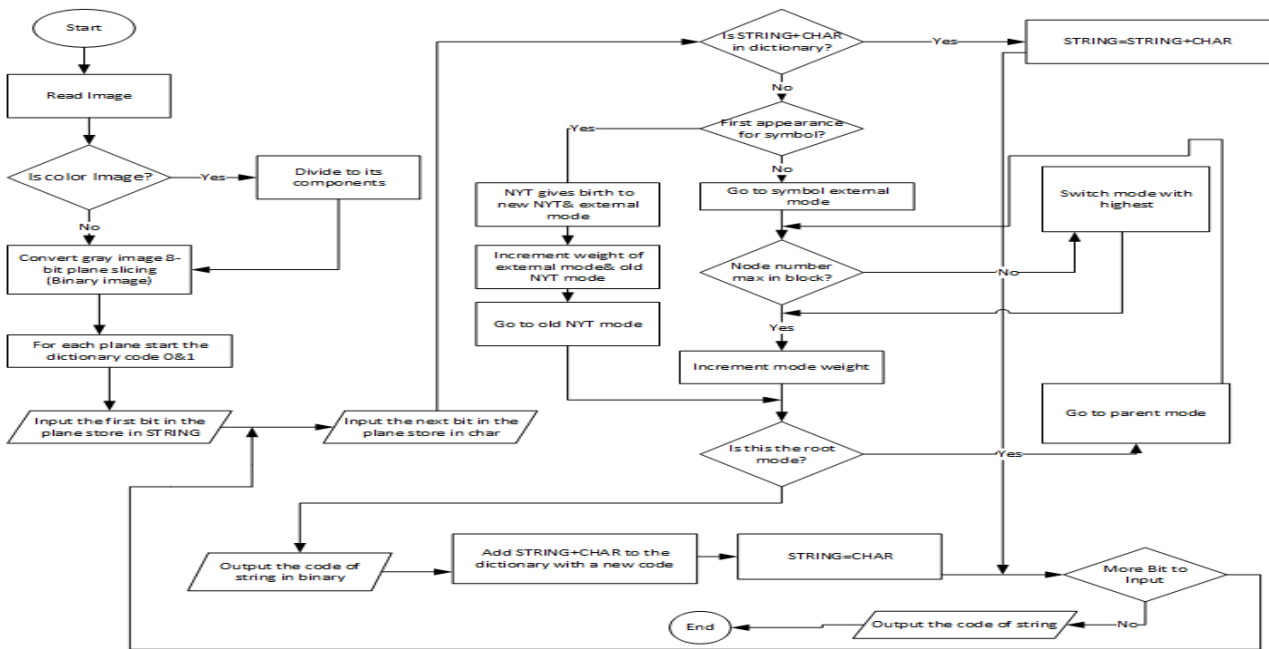
Since, LZW is a dictionary-based algorithm; it uses code table compression. The LZW method achieves compression by using codes 256 through 4095 to represent sequences of bytes. This makes MATLAB one of the most popular software programs for engineers and scientists (Had *et al.*, 2016). The compression system improves the compression of the image through the implementation of LZW algorithm. The algorithm builds a data dictionary (also called a translation table or string table) of data occurring in an uncompressed data stream (Alarabeyyat *et al.*, 2012). In our design the unique dictionary is replaced with a dictionary set consisting of several smaller dictionaries with different address spaces and word widths (Engineering, 2008). The flow image compressions of any algorithm takes an

original image, binarize it, separates determinant bit sets which represents most useful information of that particular image (Pratibha *et al.*, 2018).

MATLAB Image processing toolbox: is used to develop the prototypes using the script language and to run the built-in toolbox functions. MATLAB R2019a which is the easiest and most productive computing environment for engineers and scientists. It includes the MATLAB language, the only top programming language dedicated to mathematical and technical computing. MATLAB toolboxes offer professionally developed, rigorously tested, field-hardened, and fully documented functionality for a wide range of scientific and engineering applications. Toolboxes are designed to work together, and they integrate with parallel computing environments, GPUs, and C code generation. MATLAB

does the hard work of making your code fast. Math operations are distributed across your computer's cores, library calls are heavily optimized, and all code is just-in-time compiled. Math Works engineers continuously verifies quality by running millions of tests on the MATLAB code base every day.

ImageJ2: ImageJ is a public domain Java image processing program inspired by NIH Image for the Macintosh. It can display, edit, analyze, process, save and print 8-bit, 16-bit and 32-bit images. It can read many image formats including TIFF, GIF, JPEG, BMP, DICOM, FITS and "raw".



An adapted LZW image compression approach, (Abu Taleb et al., 2010).

Our Image Compression Flowchart - In this section the researcher did all the preprocessing tasks like separating rub color components (i.e. red, green and blue), masking each color component, finding their respective bit plane and filtering noise.

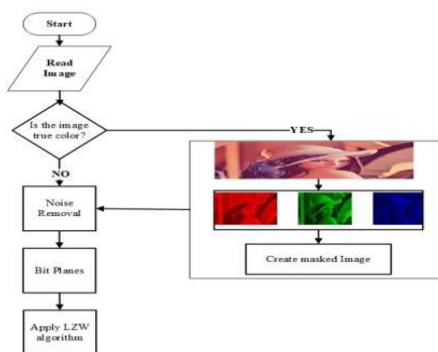


Fig 8: My workflow diagram.

This paper aims to let the LZW lossless image compression algorithm accept RGB true color images by

Algorithms: In this section the researcher briefly discusses the algorithms the researcher use. The first algorithm the researcher took is Lempel-Ziv-Welch (LZW). The second algorithm employed to enhance the LZW algorithm is bit plane slicing.

Normal LZW - The LZW algorithm is a greedy algorithm in that it tries to recognize increasingly longer and longer phrases that are repetitive, and encode them. Each phrase is defined to have a prefix that is equal to a previously encoded phrase plus one additional character in the alphabet.

dividing true color or 8-bit depth image to its' RGB components (i.e., red, green and blue). The algorithm is favorable 8-bit color of GIF and TIFF images. The compression algorithm looks like (Engineering, 2008): 1) Initialize table with single character strings; 2) String = first input character; 3) WHILE not end of input stream; 4) Char = next input character; 5) IF String + Char is in the string table; 6) String = String + Char 7 ELSE; 7) output the code for String; 8) add String + Char to the string table; 9) String = Char; 10) END WHILE; and 11) Output code for String.

Test Images in Thesis - The MATLAB public library images are taken for testing the normal LZW algorithm and the researcher's prototype demo. Since the paper concerns only on enhancing the LZW algorithm compression ratio for gray scale images, RGB to gray scale image conversion has

been made. The MATLAB image library is full of different image formats and color models. But the paper aims to enhance the LZW algorithm compression ratio with different image format, the image is rewritten to the prescribed image using the MATLAB image toolbox functions and plugins. To compare the compression ratio the LZW algorithm for noisy and denoised image, the noised images are created using the built-in functions like monies ().

RGB image representation - An RGB image can be viewed as three different images (a red scale image, a green scale image and a blue scale image) stacked on top of each other, and when fed into the red, green and blue inputs of a color monitor, it produces a color image on the screen. IN MATLAB, an RGB image is basically a M*N*3 array of color pixel, where each color pixel is associated with three values which correspond to red, blue and green color component of RGB image at a specified spatial location. So, the color of any pixel is determined by the combination of the red, green, and blue intensities stored in each color plane at the pixel's location.

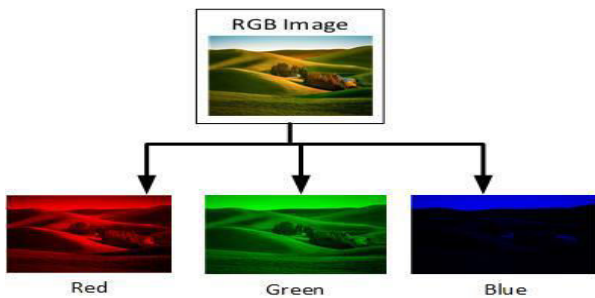


Fig 9: RGB Image color components.

Three Bit Plane Formation using Three Color Plane - Well known color spaces considered commonly by all the researches are Red, Green and Blue. Mostly color images are stored in the RGB color format. In this method the color image is dividing into R, G, and B components. Then compute three thresholds which is the mean of each color components MR, MG, MB for red, green and blue color component respectively given as[30]:

$$MR = \frac{1}{256 \times 256} \sum_{i=1}^{256} \sum_{j=1}^{256} r(i, j) \quad (1)$$

$$MG = \frac{1}{256 \times 256} \sum_{i=1}^{256} \sum_{j=1}^{256} g(i, j) \quad (2)$$

$$MB = \frac{1}{256 \times 256} \sum_{i=1}^{256} \sum_{j=1}^{256} b(i, j) \quad (3)$$

Extracting RGB Components - The foremost preprocessing technique to improve the normal LZW UniversePG | www.universepg.com

lossless image compression technique is dividing the true color or RGB image to its components. This dictionary-based algorithm will feed each color channel (red, green and blue) alone. LZW algorithm is outfitted to compress gray scale or 8-bit depth image than other lossless image compression algorithms. The easiest and effective technique to extract the color component from RGB color image is the MATLAB function; `imsplit()`.

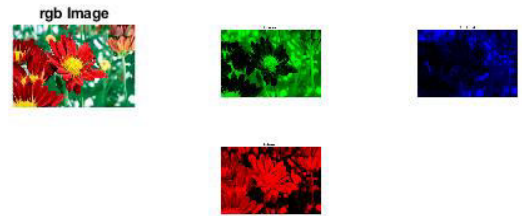
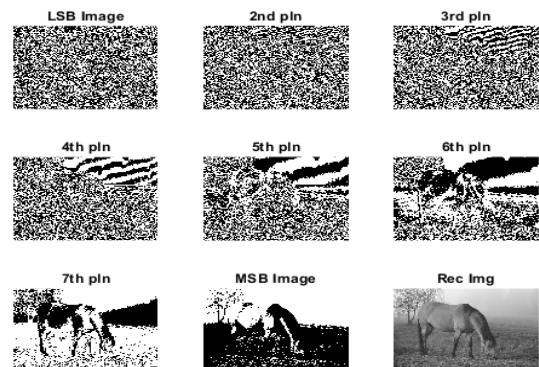


Fig 10: RGB color components (red, green and blue).

Bit Plane Slicing - Bit plane slicing is a method of representing an image with one or more bits of the byte used for each pixel. One can use only MSB to represent the pixel, which reduces the original gray level to a binary image (Dastanova et al., 2018). The three main goals of bit plane slicing are: Converting a gray level image to a binary image, representing an image with fewer bits and corresponds the image to a smaller size, and Enhancing the image by focusing. Highlighting the contribution made to the total image appearance by specific bits. The Assumption here is that each pixel is represented by 8-bits and the image is composed of eight 1-bitplanes. Plane (0) contains the least significant bit and plane (7) contains the most significant bit. The higher order bits only (top four) contain the majority visually significant data. The remaining bit planes contribute the more subtle details. It is useful for analyzing the relative importance played by each bit of the image. The code in is the bit slicing MATLAB code I adapt from Ahmed Ayman (<https://www.mathworks.com/matlabcentral/fileexchange/>).



```

clear all, close all, clc
% load and display image
I=imread('flamingos.jpg');
imshow(I)
% Grayscale
Igray=rgb2gray(I);
figure, imshow(Igray)
% Size
[row , col] = size(Igray);
% Bitwise Separation
Inew = zeros(row,col);
mask = 1;
for a = 1:8
for i = 1:1:row
for j = 1:1:col
Inew(i,j)=bitand(Igray(i,j),mask);
end
end
mask = mask * 2;
figure, imshow(Inew)
%figure, imagesc(Inew)
%colormap gray
end
% Compressing Image
% by eliminating the least significant four bits
% then the picture will need only 4 bits
% (16 levels) instead of 8 bits (256 levels)
% i.e. half storage size :)
mask = 240;
for i = 1:1:row
for j = 1:1:col
Inew(i,j) = bitand(Igray(i,j),mask);
end
end
figure, imshow(uint8(Inew))
% The Difference
Idif = Igray - uint8(Inew);
figure, imagesc(Idif), colormap gray
    
```

Fig 11: Bit plane of a gray image.

The code displays images in eight/8/different bit intervals. As the researcher can observe and compare each output to the original gray image, the image at bit 8 or MSB is equal to the original gray image. So, the researcher takes it and apply the LZW algorithm.

LZW Pre-processing

Image binarization - These are the pre-processing steps often performed in improved LZW.

Binarization - Usually presented with a grayscale image; simply a matter of choosing a threshold value. Color reduction is done here by converting RGB true color image to gray level image. Since the bit-depth decreases from 24 bit-depth to 8-bit depth, there exists quality degradation. This is done by using rgb2ind function. This section classifies the some important local and global binarization methods that are currently used for binarization.



Fig 12: Binarized image.

Gray Scale Image Noise Removal - In this section, the give gray scale image would be distorted by adding salt& pepper noise with different intensity. Of course, the maximum threshold to be restored with linear wave let medifelt2 is 70% distortion.

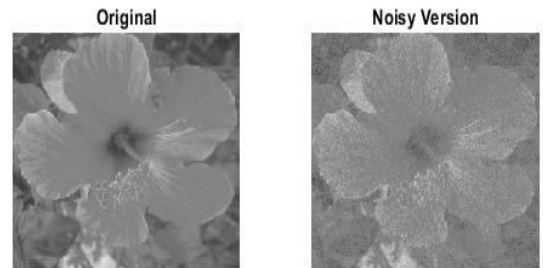


Fig 13: A gray scale flower and its Noisy version.

To see compare the local quality scores and compare the compression ratio of an image, the noised image must be filtered. To filter a noise, wdenoise2 is applied.

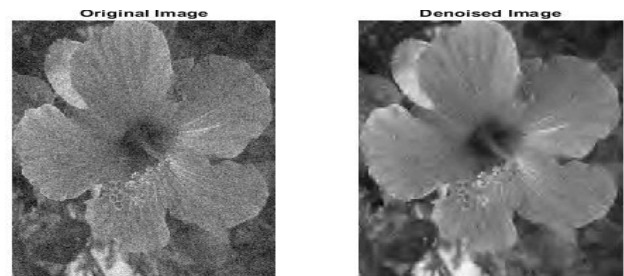


Fig 14: Denoising A Noised Image with Wdenoise2.

DE mosaicking - DE mosaicking is a process of making RGB image from the gray scale slice image. To reconstruct the first true color image from the 8-bit depth image, the researcher needs at least 5 rows by 5 column image array pixels. And the test images must be Bayer pattern sensor alignment digital images. If so, the democratic function can reconstruct the true color image using one of the sensor alignments: GBRG, GRBG, BGGR or RGGB. This will be done if the image was captured by adjusting our camera sensor BAYER pattern. Otherwise, to concatenate and convert gray scale images to RGB, color map is needed.



Fig 15: Gray Scale Image with BAYER pattern 'BGGR'

In this case a gray scale image can be converted to its true color version with a real colormap. The technique is called DE mosaicking. The figure below is an example for gray to RGB conversion.

The following MATLAB script gives us the RGB version of gray mand (Fig 15).

```
>> gray=mired('mandi.tif');
>>mishap(gray)
>> RGB = democratic(gray,'bggr');
>>mishap (RGB);
```

Histogram - An image histogram is a type of histogram that acts as a graphical representation of the tonal distribution in a digital image. It plots the number of pixels for each tonal value. By looking at the histogram for a specific image a viewer will be able to judge the entire tonal distribution at a glance. Image histograms are present on many modern digital cameras. Photographers can use them as an aid to show the distribution of tones captured, and whether image detail has been lost to blown-out highlights or blacked-out shadows. The horizontal axis of the graph represents the tonal variations, while the vertical axis represents the number of pixels in that particular tones.

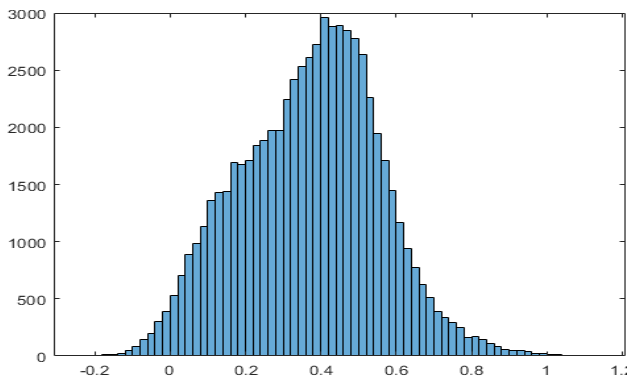


Fig 16: The Histogram of Fig 13.

Noise filtration - They are different noises which distort electronic images like transferring over the internet, image acquisition, hacker manipulation, digital camera, etc. to reduce such unwanted signals, the researcher uses noise filtering functions.

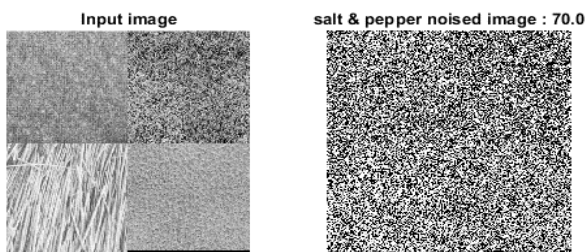


Fig 17: Salt peppers noised image.

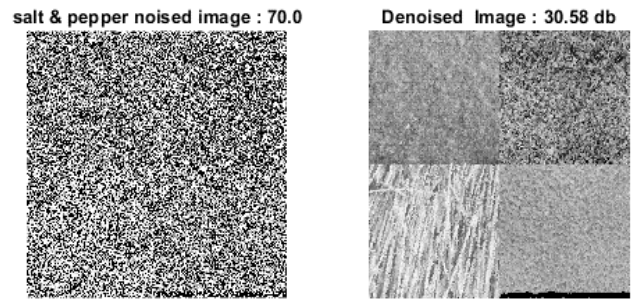


Fig 18: Filtered image using adaptive median filter.

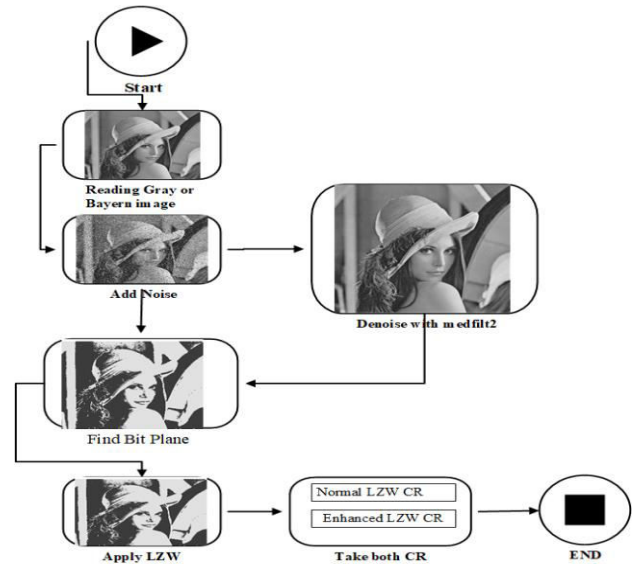


Fig 19: The Enhanced LZW Compression Algorithm Model with Lena Gray Image.

Our proposed salt-peppers noise remover is adaptive median filter.

Enhanced LZW - In this section, the normal LZW image compression algorithm would be embedded with the proposed enhancement techniques. All the code blocks are done so far, merging them yields the final proposed model. Here the model would produce a compressed gray scale image both with noise and after noise filtration. The compression ratio gained from the normal LZW and the enhanced LZW would be registered for every single image for the comparison purpose. And the quality of the image before and after compression measured using SSIM, MSE, and PSNR.

Experimental setup

The experiment was performed using Windows 10 professional, Intel Core i5-4200M CPU processor with 2.50 GHz speed. The software used for the experiment implementation is MATLAB r2019a with image processing toolbox, ImageJ and GIMP. The datasets are all gray image formats from MATLAB public library and images with BAYERN pattern.

Noised images are produced using salt & peppers and gaussian blurring model with the intensity up to 70%. Filtered or denoised image are produced using medfilt2 and wdenoise2. The 8-bit plane slicing technique is an appropriate technique to separate the bit planes for each plane. The 8-bit plane slicing output bits will be exported/write to the disk for the LZW encoding scripts. The compression ratio measurement and execution time script are included in the enhanced LZW encoding algorithm MATLAB file.

Split RGB Image into Its Component Channels - The horse RGB image from the Math Works laboratory is split to its color channels using `implant ()` MATLAB function as follows:

```
[r,g,b] = implant('horses.jpg');
subplot (1,3,1); mishap(g); title ('Green channel');
subplot (1,3,2); mishap(b); title ('Blue channel');
subplot (1,3,3); mishap(r); title ('Red channel');
```

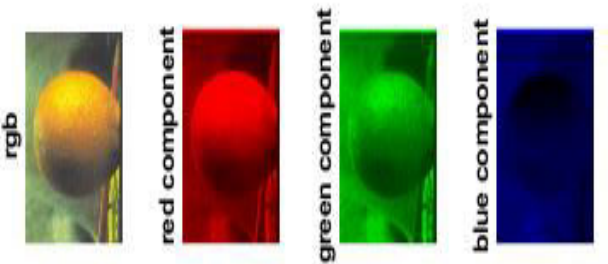
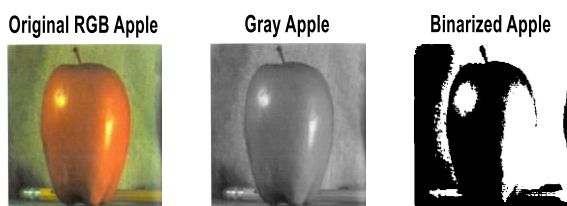


Fig 20: RGB Components of Horse.jpg image.

Having the three-color components of an RGB image alone, the researcher converts each to gray. So that, the researcher can binarize and put in a bit plane each color as 8-bit depth color image. For this task, the researcher employee `implant ()` function, but the researcher can also alternatively use `rgb2gray ()`.

Image Binarization

```
Apple = imread ('Apple.png')
Imray = rgb2gray (Apple);
Iamb = binarize (Imgray);
subplot (1,3,1); mishap (imread ('apple.png')); title ('Original RGB Apple');
subplot (1,3,2); imshow(Imgray); title('Gray Apple');
```



```
subplot (1,3,3); imshow (Imgbw); title ('Binarized Apple');
```

Gray Scale Image to Bit Plane Conversion



Fig 21: RGB Image files information.

The researcher converts an RGB image to gray level image using `RGB2GRAY` conversion model. The grayscale output g is a constrained linear combination of R , G , and B channels of the input color image I , which is:

$$3g = w_r I_r + w_g I_g + w_b I_b$$

s.t. $w_r + w_g + w_b = 1,$
 $w_r \geq 0, w_g \geq 0, w_b \geq 0,$

Where, $I_r, I_g,$ and I_b are input channels, respectively. Channel weights $w_r, w_g,$ and w_b are non-negative numbers that sum to 1.



Fig 22: RGB to Gray conversion.

Having an 8 bit gray level image, the researcher employee a MATLAB function `bitget` which can extract all 8 planes of the gray image. Most of the information concentrated at the eighth plane; called most significant (MSB).



RGB Image to Bit Plane Conversion

Since the researcher cannot enter a true color image in to an 8-bit plane, the researcher need to find the color components of red, green and blue of a given true color image as follows.

Noise Removal - Image noise is random (not present in the object imaged) variation of brightness or color information in images, and is usually an aspect of electronic noise. It can be produced by the sensor and circuitry of a scanner or digital camera. Image noise can also originate in film grain and in the unavoidable shot noise of an ideal photon detector. Image noise is an undesirable by product of image capture that adds spurious and extraneous information. The meaning of “Noise” is unwanted signal. Removing those unwanted signals enhances the quality of the image and decreases the file size. The major noises are: photon, thermal, salt noise and pepper noise.



Fig 23: Noise creation and filtration using imnoise and medfilt2 respectively.

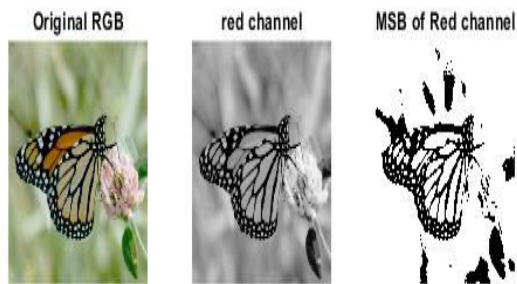


Fig 24: MSB of the red channel from the given RGB image.

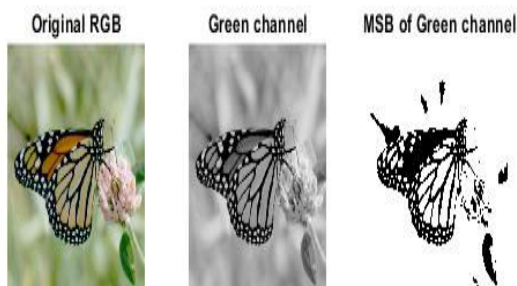


Fig 25: MSB of the green channel from the given RGB image.

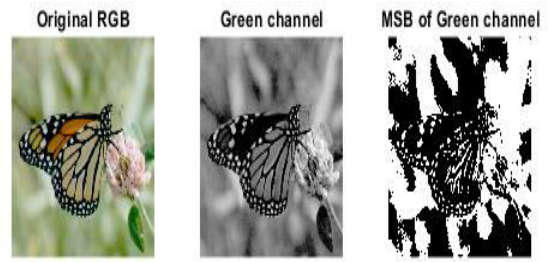


Fig 26: MSB of the blue channel from the given RGB image.

There are different noise removal techniques as per the noise type. Some of them are median, blurring, sigma filter, knn filter, Savitzky-Golay, BM3D, non-local means, K-SVD, K-LLD, Knox-Thompson, etc. Multiply each plane by a given mask and create masked red, green and blue planes. Having those two-dimensional images, the researcher concatenate to reconstruct a masked rgb image.

Then apply 2-D adaptive noise-removal filtering (i.e. wiener2) and 2-D Median Filtering (i.e. medfilt2) to each MSB plane.

Performance Measurement Parameters

After reconstruction/decompression of the image, the quality of the image and the performance of the compression algorithm have to be tested. Therefore, the amount of compression and how good the reconstructed image is similar to the original is known from the test conducted. In this thesis the test was conducted by calculating important distortion measures namely: the mean square error (MSE), peak signal-to-noise ratio (PSNR) measured in decibels (dB), the compression ratio (CR) and structural similarity index (SSIM) which are all briefly defined below.

Mean Square Error (MSE)

The MSE is the cumulative squared error between the compressed and the original image. It measures the average of the square of the error. A lower value of MSE means lesser error.

$$MSE = \frac{1}{mn} \sum_0^{n-1} \sum_0^{m-1} |f(i, j) - g(i, j)|^2 \text{ ---- (4.1)}$$

Compression Ratio (CR)

It is the measure of the reduction of the detailed coefficient of the data. In the process of image compression, it is important to know how much detailed (important) coefficient one can discard from the input data in order to sanctuary critical information of the original data. Compression ratio can be expressed as:

$$CR = \frac{\text{image compressed of size}}{\text{image original of size}} \text{-----} (4.2)$$

Peak Signal to Noise Ratio (PSNR)

PSNR is a measure of the peak error. Many signals have very wide dynamic range, because of that reason PSNR is usually expressed in terms of the logarithmic decibel scale (dB). A higher value of PSNR means higher signal to noise ratio. Values for PSNR range between infinity for identical images, to 0 for images that have no commonality. The PSNR is given as:

$$PSNR = 20 \log_{10} \left(\frac{MAX_f}{\sqrt{MSE}} \right) \text{-----} (4.3)$$

Where the MSE is *Mean Squared Error* and \max_f is the maximum possible pixel value of the image.

Structural Similarity Index (SSIM) - SSIM is the structure similarity index for measuring the similarity between the original image and compressed image. Given two image signals or pixels x and y from two images, which are aligned with each other, the SSIM between the two image signals is given as a function of three characteristics which are luminance $l(x,y)$, contrast $c(x,y)$ and structure $s(x,y)$. SSIM lies between 0 and 1.

$$SSIM(x, y) = f(l(x, y), c(x, y), s(x, y))$$

$$l(x, y) = \frac{2\mu_x\mu_y+C_1}{\mu^2_x+\mu^2_y+C_1} \text{-----} (4.4)$$

$$C(x, y) = \frac{2\sigma_x\sigma_y+C_2}{\sigma^2_x+\sigma^2_y+C_2} \text{-----} (4.5)$$

$$S(x, y) = \frac{\sigma_{xy}+C_3}{\sigma_x\sigma_y+C_3} \text{-----} (4.6)$$

Where, $\sigma_{xy} = \frac{1}{N-1} \sum_{i=0}^N (x_i - \mu_x)(y_i - \mu_y)$; where, N is the number of signal samples (pixels), and C_3 is a non-negative constant.

Using equations (4.6), (4.7) and (4.8) the SSIM is given as:

$$SSIM = \frac{(2\mu_x\mu_y+C_1)(2\sigma_{xy}+C_2)}{(\sigma^2_x+\sigma^2_y+C_1)(\sigma^2_x+\sigma^2_y+C_2)} \text{-----} (4.7)$$

Speckle/multiplicative - Speckle is a granular interference that inherently exists in and degrades the quality of the active radar, synthetic aperture radar (SAR), medical ultrasound and optical coherence tomography images. Since speckle noise is multiplicative in nature, average contrast of an image may be treated as a measure of speckle removal. Speckle index (SI) is defined as[55]:

$$SI = \frac{\sqrt{var(I)}}{E(I)}$$

And its discrete version for an image reads

$$SI = \frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^m \frac{\sigma(i,j)}{\mu(i,j)} \text{-----} (4.8)$$





Where, $m \times n$ is the size of the image, σ is the mean and μ is the standard deviation. For optimal performance, the measured value of S.I. should be low.



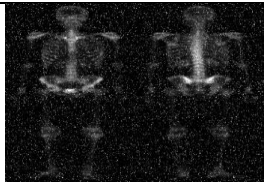
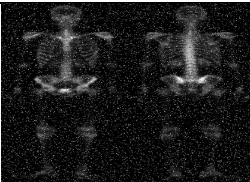
The speckle index can be regarded as an average reciprocal signal-to noise ratio (SNR) with the signal being the mean value and noise being the standard deviation.

$$Average\ SNR = \frac{1}{SI} \text{-----} (4.9)$$

Experiment One - MATLAB image processing toolbox, GIMP and image noise techniques are used. All the datasets taken from the MATLAB library are gray level and tiff format. The blurred dataset are prepared by applying a salt & pepper noise with 0.1 intensity. The compression ratio, peak signal to noise ratio, mean square error and structural similarity index measurements are taken.

Table 4: Compression Ratio of LZW and Image Quality Scores for Noised Image

Name of the Image	Original blurred Image	Decompressed blurred Image	PSNR	MSE	CR	SSIM
Lena			99	0	113.33	1
Cameraman			99	0	105	1

Einstein			99	0	89.47	1
Skeleton			99	0	89.47	1

Experiment Two - MATLAB image processing toolbox, GIMP and image denoise techniques are used. All the datasets taken from the above table (Experiment one). The denoised dataset are prepared by applying a medfilt2 algorithm. The compression ratio, peak signal to noise ratio, mean square error and structural similarity index measurements are taken (**Table 5**).

Experiment Three - MATLAB image processing toolbox, GIMP and bit plane slicing algorithm are used. All the datasets taken from the above table (Experiment two). The 8-bit planes of the denoised gray scale image are extracted and the MSB of each image are taken for the experimental input. The compression ratio, peak signal to noise ratio, mean square error and structural similarity index measurements are taken (**Table 6**).

Table 5: Compression Ratio of LZW after Noise Removal





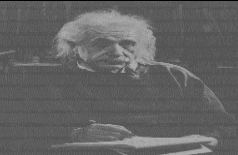
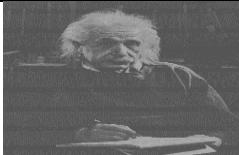
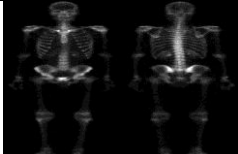
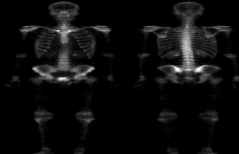




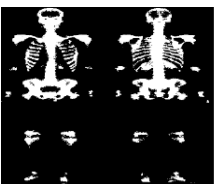
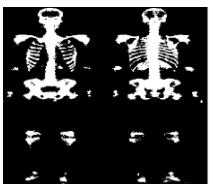
Name of the Image	Original blurred Image	Decompressed blurred Image	PSNR	MSE	CR	SSIM
Lena			5.7365	1.73	123.53	0.89
Cameraman			99	0	107.69	1
Einstein			99	0	116.67	1
Skeleton			99	0	116.67	1

Table 6: Compression Ratio and Quality Metrics of Fully Enhanced Encoding Algorithm

Name of the Image	Original blurred Image	Decompressed blurred Image	PSNR	MSE	CR	SSIM
Lena			5.7365	1.73	194.12	0.89

Camerman			99	0	161.54	1
Einstein			99	0	141.67	1
Skeleton			99	0	141.67	1

The researcher made a simple numerical comparison between the original image and the decoded one. It shows us (above **Tables 4, 5** and **6**) a totally lossless recovery of the original given image has been achieved. The compressed image fully decompressed

without scarifying image information, except the unwanted and repetitive data. For optimal performance, measured values of MSE, RMSE and should be small and PSNR should be large.

Table 7: Compression Ratio and Quality Metrics Score Summary of (Table 4, 5, 6)

Image Names	Table 4				Table 5				Table 6			
	MSE		SSIM	PSNR	MSE	CR	SSIM	PSNR	MSE	CR	SSIM	
Lena	99	0	113.33	1	5.7365	1.73	123.53	0.89	5.7365	1.73	194.12	0.89
Camera	99	0	105	1	99	0	107.69	1	99	0	161.54	1
Einstein	99	0	89.47	1	99	0	116.67	1	99	0	141.67	1
Skeleton	99	0	89.47	1	99	0	116.67	1	99	0	141.67	1

RESULTS AND DISCUSSION:

The image dataset which fulfils the LZW algorithm specification are selected from the MATLAB public library. LZW is a general compression algorithm but its compression performance increases for tiff and gif format images. Images with Bayern pattern sensor property also included because those images can be fully converted to their RGB color model with a predefined true color map input. The technique to convert Bayern pattern gray scale images to RGB is called Demosaicking. The maximum threshold image noise is applied to the gray scale images. UniversePG | www.universepg.com

The image noises applied are salt & pepper, gaussian, poisson and localvar with different blurring intensity. The gray scale images are binarize to generate the unsigned integer values matrix of the image. The 8 bit plane slicing of gray scale images are extracted from binarized matrix vector with bitget function. It gets bits of a pixel at a specific position of the image. Those image positions are categorized based of their information weight; least significant bits, centered bits and most significant bits. Full referenced image quality metrics measurements are taken for each experiment. It's understood that low

mean square error, high peak signal to noise ratio and high (i.e. 1) structural similarity index measurement parameter scores of an image indicates maximum quality.

The SSIM metric combines local image structure, luminance, and contrast into a single local quality score. IT agrees more closely with the subjective quality score, that is the real human eye perception. Because structural similarity is computed locally, ssim can generate a map of quality over the image. The compression ratio is recorded for each experiment of different samples to know how much detailed (important) coefficient one can discard from the input data in order to sanctuary critical information of the original data. The second experiment results told that image denoising and compression ratio has a positive correlation relationship. The more the image if filtered, the high the compression ratio achieved by the encoding algorithm. Take experiment one (**Table 4** and **Table 5**), the compression ratio of the proposed enhanced algorithm outreach the normal compression algorithm by 116% for filtered gray scale images. Even the execution time of the encoding algorithm is improved when the unwanted signal is reduced. The third experiment proves slicing an image into 8-bit plane enhances the compression ratio of the encoding algorithm. If we take a comparison between the normal LZW encoding and the proposed encoding algorithm, a clear difference visualized. Slicing and separating the image defining bits in the plane can create a meaningful bit pattern, so that the algorithm can put many bytes as possible with a single symbol in the codebook inside the dictionary. The total compression ratio improvement of the enhanced algorithm with both proposed techniques is recorded as 160%. Beside the compression ratio enhancement, the three image quality metrics imp-lies there is no loss of information during file size reduction. The compressed images are fully decoded to their original quality. The average values of SSIM, MSE AND PSNR are 1, 0, and 99 respectively. So, the enhanced encoding algorithm is fully lossless. Since the LZW encoding algorithm is not available in MATLAB image processing toolbox, coding for each personal function is one of the tasks which scarifies the entire thesis time. Getting a Bayern pattern image dataset is also another problem an image analyst face in the field. This is because

cameraman does not aware of the Bayern pattern sensor in their digital camera.

CONCLUSION:

Even though the proposed enhanced LZW algorithm show a dramatic improvement in compression ratio without scarifying image information or image quality, the enhancement can be extended still. In real images, different natural noises are occurring. Such natural speckle additive noises cannot be reduced using the proposed filtering techniques. Adaptive median filter and speckling denoising technique can be used for reduce natural additive noise like interferences, camera misuses, hacker manipulations. One can also improve the encoding algorithms execution time by selecting a noise reduction technique knowing the noise type. Those proposed techniques can be also used for Bayern pattern RGB or true color images. Enhancing the LZW encoding algorithm for high dimensional or true color images using bit plane slicing is our next work. This dream can be true with the help of image masking, concatenate the three color components with their bit planes.

6. ACKNOWLEDGEMENT:

I would like to express our gratitude to our students for their effortless support in completing this paper and also thanks to our colleagues for their support and advice to the successful of present study.

7. CONFLICTS OF INTEREST:

The authors declare that there is no conflict of interest to publish it.

8. REFERENCES:

- 1) Abu Taleb S., Musafa H. M. J., Khtoom A., and Gharaibeh I. (2010). Improving Lzw Image Compression, **44**. 2010. <https://adiwijaya.staff.telkomuniversity.ac.id/wp-content/uploads/sites/56/2013/10/04-Improving-LZW.pdf>
- 2) Alarabeyyat A. et al. (2012). "Lossless Image Compression Technique Using Combination Methods," *J. Softw. Eng. Appl.*, **05**(10), Pp. 752–763.
- 3) Al-Dmour A., Abuhelaleh M., Musa A., and Al-Shalabi H. (2016). "An Efficient Bit-Level Lossless Grayscale Image," **12**(2), Pp. 322–331. <http://jips.jatsxml.org/upload/pdf/jips-12-2-322.pdf>

- 4) Al-Khafaji A. P. G. and Bassim M. (2019). "Color Image Compression of Inter-Prediction Base," *Academia*, **8**, Pp. 7–15.
<https://ijcsmc.com/docs/papers/July2019/V8I7/201902.pdf>
- 5) Awadala AS, Elfaky AE, and Marouf AAS. (2020). Influence of high power Nd: YAG laser on hardness and surface properties of zirconium silicate, *Int. J. Mat. Math. Sci.*, **2**(3), 39-44.
<https://doi.org/10.34104/ijmms.020.039044>
- 6) Badshah G., Liew S., Zain J. M., Hisham S. I., and Zehra A. (2015). "Importance Of Watermark Lossless Compression In Digital Medical Image Watermarking," *Res. J. Recent Sci.*, **4**(3), Pp. 75–79.
- 7) Bhattacharyya S., Gandhi T., Sharma K., Dutta P., and Conference I. (2017). *Advanced Computational and Communication Paradigms*, **1**, 2017.
<https://www.springer.com/gp/book/9789811082399>
- 8) Boopathiraja S., Kalavathi P., and Chokalingam S. (2018). "A Hybrid Loss-less Encoding Method For Compressing Multi-spectral Images Using Lzw And Arithmetic Coding," **4**, Pp. 313–318.
- 9) Buades A., B. Coll, J. Morel, and J. M. A. (2010). "A Review of Image Denoising Algorithms, With a New One," In *Multi-scale Modeling and Simulation*, Hal-002-7114: *A Siam Interdisciplinary Journal*, **4**(2), Pp. 490–530.
<https://doi.org/10.1137/040616024>
- 10) Cadena L., Zotin A., Cadena F., Korneeva A., Legalov A., and Morales B. (2017). "Noise Reduction Techniques For Processing Of Medical Images," *Lect. Notes Eng. Comput. Sci.*, **2229**, Pp. 496–500.
- 11) Chen J., Yu W., Tian J., Chen L., and Zhou Z. (2018). "Image Contrast Enhancement Using An Artificial Bee Colony Algorithm," *Swarm Evol. Comput.*, **38**, December 2016, Pp. 287–294.
- 12) Chiranjeevi K. and Jena U. R. (2016). "Image Compression Based On Vector Quantization Using Cuckoo Search Optimization Technique," *Ain Shams Eng. J.*, **9**(4), Pp. 1417–1431.
<https://doi.org/10.1016/j.asej.2016.09.009>
- 13) Dastanova N., Duisenbay S., Krestinskaya O., and James A. P. (2018). "Bit-Plane Extracted Moving-Object Detection Using Memristive Crossbar-Cam Arrays For Edge Computing Image Devices," *IEEE Access*, **6**, Pp. 18954–18966.
<https://doi.org/10.1109/ACCESS.2018.2819986>
- 14) De Luca P., Maria V., and Ciro R. (2019). "A Study For Image Compression Using Repair Algorithm," *Semant. Sch.*, **3**, Pp. 78–88.
- 15) Dusselaar R. and Paul M. (2017). "Hyperspectral Image Compression Approaches: Opportunities, Challenges, And Future Directions: Discussion," *J. Opt. Soc. Am. A*, **34**(12), P. 2170.
<https://pubmed.ncbi.nlm.nih.gov/29240091/>
- 16) Engineering C. (2008). "Lossless Data Compression and Decompression Algorithm and Its Hardware Architecture Master of Technology in Lossless Data Compression and Decompression Algorithm and Its Master of Technology In," 2008.
- 17) Ghadirli H. M., Nodehi A., and Enayatifar R. (2019). "An Overview Of Encryption Algorithms In Color Images," *Signal Processing*, 2019.
<https://doi.org/10.1016/j.sigpro.2019.06.010>
- 18) Gonzalez, R. C., and Wood, R. E. (2019). *Digital Image Processing Concepts 2*, 4th Ed. New York: Pearson.
- 19) Gonzalez R. C., Woods R. E., and Eddins S. L. (2014). *Digital Image Processing*, 2nd Ed. Upper Saddle River, N.J: Prentice Hall.
- 20) Gupta R., Mehrotra D., and Tyagi R. K. (2018). "Comparative Analysis Of Edge-Based Fractal Image Compression Using Nearest Neighbor Technique In Various Frequency Domains," *Alexandria Eng. J.*, **57**(3), Pp. 1525–1533.
<https://doi.org/10.1016/j.aej.2017.03.038>
- 21) Had B. Welch, Wright C. H. G., and Morrow M. G. (2016). *Real Time Digital Signal Processing Using Matlab*, 3rd Edition, Boca Raton: Crc Press, 2016.
- 22) Halder A., Kundu A., Sarkar A., and Palodhi K. (2019). *Emerging Technologies In Data Mining And Information Security*, **813**. Springer Singapore, 2019.
<https://www.springer.com/gp/book/9789811315008>

- 23) Hussain A. J., Al-Fayadh A., and Radi N. (2018). "Image Compression Techniques: A Survey In Lossless and Lossy Algorithms," *Sciencedirect*, **300**, Pp. 44–69.
- 24) Hussein A. H., Al-Juboori S. S., and Mohammed R. J. (2017). "Image Compression Using Proposed Enhanced Run Length Encoding Algorithm," *Ibn Al-Haitham J. Pure Appl. Sci.*, 24(1), 2017. <https://arxiv.org/abs/1804.00589>
- 25) Image D., and Project P. (2014). "Digital Image Processing Project 3 Image Compression By Bit - Plane Slicing," Pp. 1–6.
- 26) Jin Z. (2019). "The Application Of Image Compression Technology In Library," **82**, No. *Icmete*, Pp. 211–214. <https://doi.org/10.2991/icmete-19.2019.51>
- 27) Kaur E. R., Jalandhar P., Kaur E. M., and Jalandhar P. (2014). "A Survey Of Medical Image Compression techniques," *Experientia*, **8**(4), Pp. 212–215
- 28) Kaur D. and Kaur K. (2013). "Huffman Based Lzw Lossless Image Compression Using Retinex Algorithm," **2**(8), Pp. 3145–3151.
- 29) Khan T., Beenish; Ms. Poonam; Mr. Mohammad, (2018). Matlab B Based Image Compression U Using," Pp. 1638–1642. <https://doi.org/10.31142/fjtsrd14394>
- 30) Kumar R., Chand S., and Singh S. (2019). "An Optimal High Capacity Reversible Data Hiding Scheme Using Move To Front Coding For Lzw Codes," 2019.
- 31) Lina K. J. (2009). "Lossless Image Compression," In *The Essential Guide To Image Processing*, Elsevier, Pp. 385–419.
- 32) Lu C. T. and Chou T. C. (2012). "Denoising Of Salt-And-Pepper Noise Corrupted Image Using Modified Directional-Weighted-Median Filter," *Pattern Recognit. Lett.*, **33**(10), Pp. 1287–1295. <https://doi.org/10.1016/j.patrec.2012.03.025>
- 33) Malvar H. S., He L., Cutler R., and Way O. M. (2004). "High-Quality Linear Interpolation For Demosaicing Of Bayer-Patterned Color Images," Pp. 5–8.
- 34) Matlab Mathworks, (2019). "Matlab," Image Analysis On Mosacing And Demosacking, 2019.
- 35) Mishra S. and Singh A. (2017). "Image Compression And Enhancement By Using The Lzw And Bhepl," *Int. J. Sci. Res.*, **7**(5), Pp. 235–238. <http://www.ijsrp.org/research-paper-0517/ijsrp-p6524.pdf>
- 36) Perumal S. and T. Velmurugan, (2018). "Prepro-cessing By Contrast Enhancement Techni-ques For Medical Images," **118**(18), Pp. 3681–3688.
- 37) Pokle P. B. and Bawane N. G. (2017). "Still Image Compression Using Angular Domain : Analysis And Fpga Implementation," **2**, Pp. 145–153. [https://www.iiaras.org/iiaras/filedownloads/ijsp/2017/003-0022\(2017\).pdf](https://www.iiaras.org/iiaras/filedownloads/ijsp/2017/003-0022(2017).pdf)
- 38) Prabhakar G. and Ramasubramanian B. (2013). "An Integrated and Efficient Approach For Enhanced Medical Image Compression Using Spiht And Lzw Coding," **4**(2), Pp. 1–8.
- 39) Prashanth. N And Singh A. V. (2015). "Fractal Image Compression For Hd Images With Noise Using Wavelet Transforms," *IEEE*, INSPEC Access. Number: 15487162. <https://ieeexplore.ieee.org/document/7275774/>
- 40) Pratibha, D. S. Vijay, and Dubey S. K. (2018). "A Review of Image Transmission Using Real Time Technique over Wmsn," *Sciencedirect*, **13**(5), Pp. 2477–2483.
- 41) Puneet P. and Garg N. (2013). "Binarization Techniques Used For Grey Scale Images," *Int. J. Comput. Appl.*, **71**(1), Pp. 8–11. <https://doi.org/10.5120/12320-8533>
- 42) Rippel O. and Bourdev L. (2017). "Real-Time Adaptive Image Compression," *Science direct*, 2017.
- 43) Roy S. K., Kumar S., Chanda B., Chaudhuri B. B., and Banerjee S. (2018). "Fractal Image Compression Using Upper Bound On Scaling Parameter," *Chaos, Solitons and Fractals*, **106**, Pp. 1339–1351. <https://doi.org/10.1016/j.chaos.2017.11.013>
- 44) Sangeetha M. and Betty P. (2017). "A Dynamic Image Compression Using Improved Lzw Encoding Algorithm," **2**(1), Pp. 264–270.
- 45) Senapati B., Kisan S., Biswal S. P., and Pattanaik A. (2018). "Gray Medical Image Compression Using Fractal Concepts," **3**(3), Pp. 851–854.

- 46) Sensing R. et al. (2016). "Comparison Of Open Source Compression Algorithms On Vhr Remote Sensing Images For Efficient Storage Hierarchy," **12**(7), Pp. 3–9.
<https://doi.org/10.5194/isprsarchives-XLI-B4-3-2016>
- 47) Soobhee A., Ruma K. N., and Ahsan F. (2017). "Faster Image Compression (Lzw Algorithm) Technique Using Gpu Parallel Processing," 2017.
- 48) Susstrunk S. (2009). "Color spaces, color encodings, & color image encodings," 2009.
- 49) Tushabe F. (2007). "Image Preprocessing For Compression: Attribute Filtering," *Lect. Notes Eng. Comput. Sci.*, **2167**(1), Pp. 999–1004.
http://www.iaeng.org/publication/WCECS2007/WCECS2007_pp999-1004.pdf
- 50) You K., Kwak H., and Technology I. (2018). "A Preprocessing Algorithm For Efficient Lossless Compression Of Gray Scale Images, *The Proposed Preprocessing Scheme*," Pp. 2485–2489.
<https://scienceon.kisti.re.kr/srch/selectPORSrchArticle.do?cn=NPAP08127430>
- 51) Xiao P. (2008). "Image Compression by Wavelet Transform," **2**(6), Pp. 411–413.
- 52) Waliullah GM, Bala D, Hena A, Abdullah MI, and MA Hossain. (2020). Study the BER performance comparison of MIMO systems using BPSK modulation with ZF and MMSE equalization, *Aust. J. Eng. Innov. Technol.*, **2**(5), 77-84.
<https://doi.org/10.34104/ajeit.020.06677084>

Citation: Tadesse N, Alemu A, and Teshager A. (2021). Enhance the LZW compression ratio through the use of image preprocessing techniques for gray scale images, *Int. J. Mat. Math. Sci.*, **3**(2), 22-42.

<https://doi.org/10.34104/ijmms.021.022042>

