Ilkogretim Online - Elementary Education Online, 2021; Vol 20 (Issue 5): pp. 7333-7340
http://ilkogretim-online.org
doi: 10.17051/ilkonline.2021.05.830

ANALYSIS OF APPROXIMATED COMPRESSED IMAGE USING SOBEL, HIGH PASS AND LOW PASS FILTERS

K.Pooja Sharma, Master of Engineering in Communication Systems, S.A Engineering College, Poonamallee, Chennai-77
G.Kavya, Professor, Head of Department, Department of Electronics and Communication Engineering, S.A Engineering College, Poonamallee, Chennai-77
N. Anusha, Professor, Department of Electronics and Communication Engineering, S.A Engineering College, Poonamallee, Chennai-77

ABSTRACT - In most of the applications less accurate data is used. The Approximate Computing which is a computation technique which returns a possibly inaccurate result rather than a guaranteed accurate result, and can be used for applications where an approximate result is sufficient for its purpose. In this proposed system, there are two methods: 1) Designing an approximate compressor. Even though an approximate compressor is used, there might be a loss of data and noise can also be added while compressing. 2) Filters will be used to remove that noise and will also be analyzing the performance of the compressed image with respect to Sobel Filter, Low Pass Filter and High Pass Filter.

KEYWORDS: **Approximate Compressor, Low Pass Filter, High Pass Filter, Sobel Filter.**

I. INTRODUCTION

Computing has entered the era of approximation, in which hardware and software generate and reason about estimates. Navigation applications turn maps and location estimates from hardware GPS sensors into driving directions; speech recognition turns an analog signal into a likely sentence; and search turns queries into information [1]. These complex systems require sophisticated algorithms to deliver good enough answers quickly, at scale, and with energy efficiency, and approximation is often the only way to meet these competing goals. Approximate computing has been proposed as an approach for developing energy-efficient systems [2], saving computational resources and presenting better execution times, and has been used in many scenarios, from big data to scientific applications. It can be achieved from a multitude of ways, ranging from transistor-level design to software implementations, and presenting different impacts on the integrity of the hardware and the quality of the output. Many systems, however, do not take precision and accuracy as an essential asset. Those are the ones that can profit from this computational paradigm. Even on systems where quality and accuracy are essential, the mere definition of a good quality result can be malleable. The Fig 1 represents the block diagram of the proposed system.

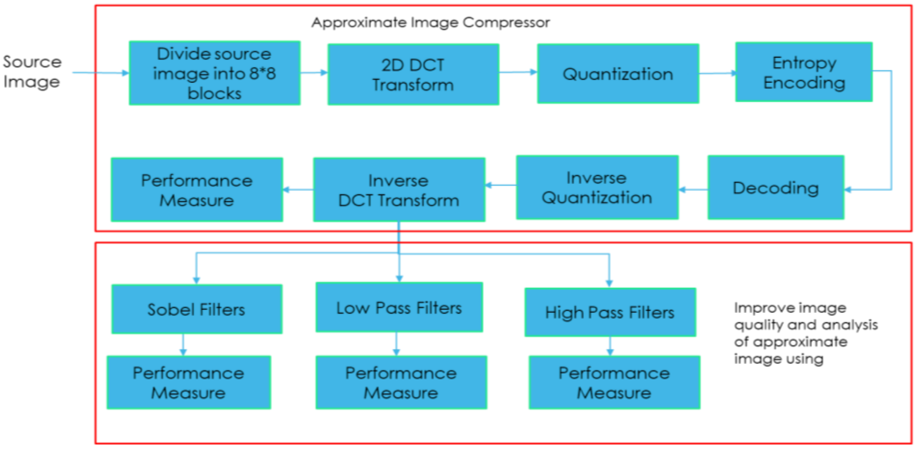


Fig 1: Block Diagram for Proposed system

II. APPROXIMATE COMPRESSORS

The approximate compressors show a significant reduction in Area, power consumption and delay compared with an exact design.

1. Image will be in smaller size.
2. Image can be transmitted faster.

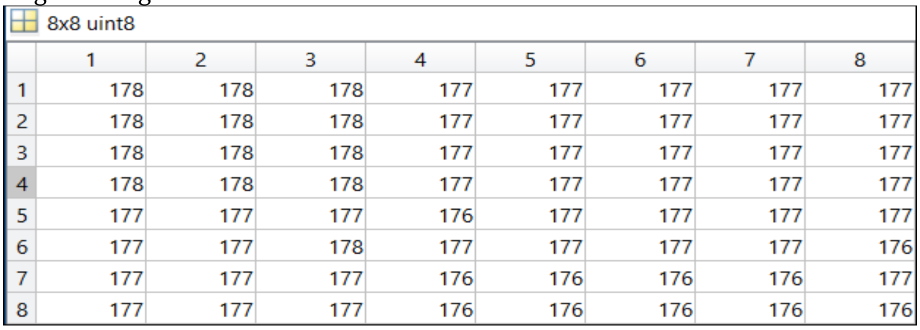
Compressor adders have lesser critical delay than normal adders [3], so that they are suitable in multiplier architectures. In multiplication operation, compressors are used for the reduction of partial products to decrease the delay and power dissipation. In image processing operations multiplication is the common operation. In these applications accurate computing is not essential and it can produce meaningful results even it has low accuracy. Approximate computing is used in these image processing and multimedia applications since it is less complex and has low power consumption. Compressor outputs are approximated to produce approximated compressor. It is an important application used in linear algebra. The need to minimize the amount of digital information stored and transmitted is an ever-growing concern in the modern world. Approximate DCT algorithm is used for compressing the image. For image processing, the Joint Photographic Experts Group method is the widely used lossy method while the Moving Picture Experts Group (MPEG) method is the widely used lossy method for video processing. Both the standards use the Discrete Cosine Transform algorithm as basic processing step. As per [4] on low quality SVD compressed images are distorted and the low quality DCT compressed images were also distorted. Another difference is DCT had a much higher compression ratio than the SVD compression, with the result images having similar image quality. The final compression ratio for DCT was about 8 times higher.

2.1 DCT ALGORITHM:

The discrete cosine transform represents an image as a sum of sinusoids of varying magnitudes and frequencies. The DCT has the property that most of the visually significant information about the image is concentrated in just a few coefficients of the DCT. For this reason, the DCT is often used in image compression applications. For example, the DCT is at the heart of the international standard lossy image compression algorithm known as JPEG. The DCT is performed on N x N square matrix of pixel values, and it yields an N x N square matrix of frequency coefficients. In practice, N most often equals to 8 because a larger block, though would probably give better compression, often takes a great deal of time to perform DCT calculations, creating an unreasonable trade off. As a result, DCT implementations typically break the image down into more manageable 8 x 8 blocks. DCT formula looks somewhat intimidating at first glance but can be implemented with a relatively straightforward piece of code. 2.3.

2.2 WORKING OF APPROXIMATE COMPRESSOR:

- The original image is divided into blocks of 8 * 8 or 16 * 16.



8x8 uint8

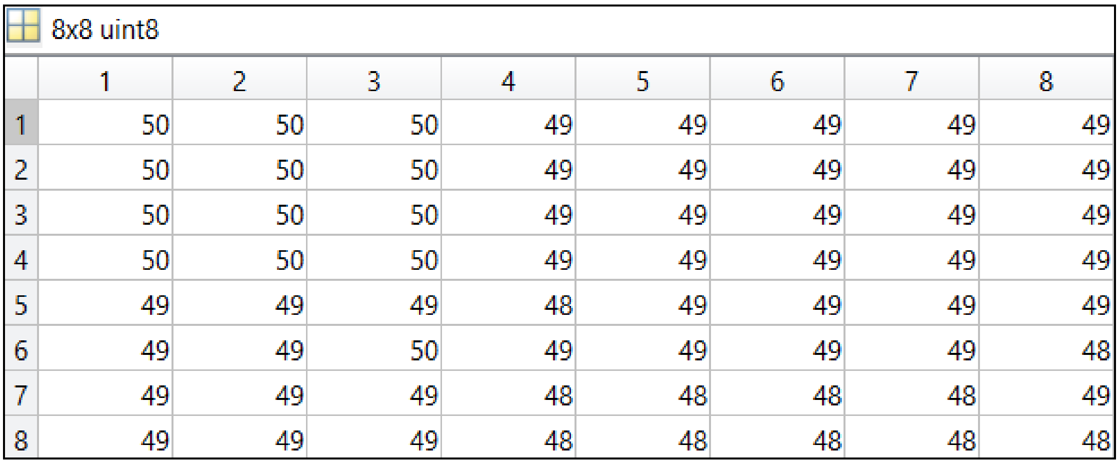
	1	2	3	4	5	6	7	8
1	178	178	178	177	177	177	177	177
2	178	178	178	177	177	177	177	177
3	178	178	178	177	177	177	177	177
4	178	178	178	177	177	177	177	177
5	177	177	177	176	177	177	177	177
6	177	177	178	177	177	177	177	176
7	177	177	177	176	176	176	176	177
8	177	177	177	176	176	176	176	176

- The pixel values of a black and white image range from 0-255 but DCT is designed to work on
- The pixel values ranging from -128 to 127. Therefore, each block is modified to work in the range.
- The below equation is used to calculate DCT matrix.

The 1D and 2D DCT of an M-by-N matrix:

$$T_{pq} = \begin{cases} \frac{1}{\sqrt{M}} & p = 0, \quad 0 \le q \le M-1 \\ \sqrt{\frac{2}{M}} \cos \frac{\pi(2q+1)p}{2M} & 1 \le p \le M-1, \quad 0 \le q \le M-1 \end{cases}$$

Where,
Tpq: Spatial domain function with p & q
p: row; q: column
M: Size/ data items


8x8 uint8

	1	2	3	4	5	6	7	8
1	50	50	50	49	49	49	49	49
2	50	50	50	49	49	49	49	49
3	50	50	50	49	49	49	49	49
4	50	50	50	49	49	49	49	49
5	49	49	49	48	49	49	49	49
6	49	49	50	49	49	49	49	48
7	49	49	49	48	48	48	48	49
8	49	49	49	48	48	48	48	48

- Then DCT is applied to each block by multiplying the modified block with DCT matrix on the left and transpose of DCT matrix on its right.
- Each blocks are then compressed through quantization.
- The quantized matrix is then entropy encoded.
- Then the compressed image is reconstructed through reverse process.
- Inverse DCT is been used for decompression.
- The compression ratio [5] (that is, the size of the compressed file compared to that of the uncompressed file.
- Compression ratio = (original data size) / (compressed data size).
- If the compression and decompression processes induce no information loss, then the compression scheme is lossless; otherwise, it is lossy.

2.3 JOINT PHOTOGRAPHIC EXPERTS GROUP (JPEG)

The JPEG processing is initiated by transforming an image to the frequency domain using the DCT [6], this separates images into parts of differing frequencies. Then, the quantization is performed such that the frequencies of lesser importance are discarded. This reflects the capability of humans to be reasonably good at seeing small differences in brightness over a relatively large area, but they usually this cannot distinguish the exact strength of a rapidly varying brightness variation. Compression takes place during this quantization step in which each component in the frequency domain is divided by a constant, and then rounded to the nearest integer. This results in many high frequencies components having very small or likely zero values, small values at best. Image is then retrieved during the decompression process that is performed using only the important frequencies that have been retained. The below Fig 2.3.1 shows the input Image for Approximate Image Compressor and Fig 2.3.2 shows the output Image for Approximate Image Compressor.

Fig 2.3.1 Input Image for Approximate Image Compressor

Fig 2.3.2 Output Image for Approximate Image Compressor

III. PERFORMANCE ANALYSIS OF COMPRESSED IMAGE

Here the signal to noise ratio, peak signal to noise ratio and mean square error will be measure and analysed.

3.1 SIGNAL TO NOISE RATIO (SNR):

The SNR of a system or component is defined as the ratio of signal level to the noise level. It is expressed in decibels. It is computed by dividing the signal power by the noise power. A ratio bigger than 1dB indicates that the signal is more than the noise.

3.2 PEAK SIGNAL-TO-NOISE RATIO (PSNR):

PSNR is an engineering term for the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. Because many signals have a very wide dynamic range, PSNR is usually expressed as a logarithmic quantity using the decibel scale. PSNR is most easily defined via the mean squared error (MSE).

$$PSNR = 10 \times \log_{10}\left(\frac{(2^n - 1)^2}{MSE}\right)$$

3.3 MEAN SQUARE ERROR

The MSE is a measure of the quality of an estimator—it is always non-negative, and values closer to zero are better.The MSE is the second moment (about the origin) of the error, and thus incorporates both the variance of the estimator (how widely spread the estimates are from one data sample to another) and its bias (how far off the average estimated value is from the true value).

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

Where,

MSE = mean squared error
n = number of data points
Y_i = observed values
$\hat{Y}_i$ = predicted values

Table 1

Performance Analysis of Compressed Image

Values	**Approximated Compressed image**
PSNR	11.02600974

SNR	18.6863
MSE	71.6539

IV. WORKING AND ANALYSIS WITH FILTERS

Most of the digital images contain noise [8]. This can be removed by many enhancement techniques. Filtering is one of the enhancement techniques which is used to remove unwanted information (noise) from the image. It is also used for image sharpening and smoothening. The image filtering is useful for many applications, including smoothing, sharpening, removing noise, and edge detection. Filter is defined by a kernel, which is a small array applied to each pixel and its neighbors within an image. In most applications, the center of the kernel is been aligned with the current pixel, and is a square with an odd number (3, 5, 7, etc.) of elements in each dimension. The process used to apply filters to an image is known as “convolution”, and may be applied in either the spatial or frequency domain.

4.1 LOW PASS FILTERING (BLURRING)

A low-pass filter is a filter that passes low-frequency signals and attenuates signals with frequencies higher than the cut-off frequency [8]. The actual amount of attenuation for each frequency varies depending on specific filter design. Smoothing is fundamentally a lowpass operation in the frequency domain. There are several standard forms of lowpass filters are Ideal, Butterworth and Gaussian lowpass filter. Fig 4.1.1 represents the input image and Fig 4.1.2 represents the output image using low pass filter.

Fig 4.1.1: Input Image for Low Pass Filter

Fig 4.1.2: Output Image for Low Pass Filter

4.2 HIGH PASS FILTERING (SHARPENING)

A high pass filter can be used to make an image appear sharper [9]. A high pass filter is a filter that passes high frequencies well, but attenuates frequencies lower than the cut-off frequency.

Fig 4.2.1: Input Image for High Pass Filter

Fig 4.2.2: Output Image for High Pass Filter

Fig 4.2.1 represents the input image and Fig 4.2.2 represents the output image using high pass filter. Sharpening is fundamentally a high pass operation in the frequency domain. There are several standard forms of high pass filters such as Ideal, Butterworth and Gaussian high pass filter. All high pass filter (Hhp) is often represented by its relationship to the lowpass filter (Hlp):

$Hhp = 1 - Hlp$

4.3 SOBEL FILTER

The Sobel operator, sometimes called the Sobel–Feldman operator or Sobel filter. The Sobel Operator [10] is a discrete type of differentiator operator that can be utilized to identify the edges by using gradient realization. At each pixel location in an image, the resultant of Sobel operator corresponds to the gradient vector or normal to this vector. This operator calculates the gradient of image intensity at each point, realizing the direction of variation in the intensity and the direction. First order partial differentiation is used by Sobel operator. Fig 4.3.1 represents the input image and Fig 4.3.2 represents the output image using sobel filter.

Fig 4.3.1: Input Image for Sobel Filter

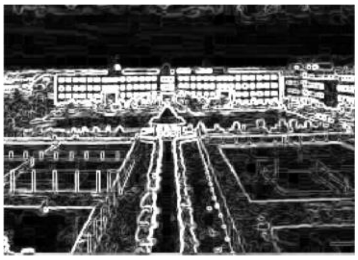


Fig 4.3.2: Output Image for Sobel Filter

V. PERFORMANCE ANALYSIS OF COMPRESSED IMAGE, SOBEL FILTER, LOW PASS FILTER, HIGH PASS FILTER

As the PSNR, SNR and MSE values of approximated compressed images are calculated, after filtering using sobel filters, low pass filters, high pass filters are also analysed. Below table 2 shows the complete analysis of 4 output images.

Table 2
Performance Analysis of Approximated Compressed Image, Sobel filter, low pass filter, high pass filter

Values	Approximated Compressed image	Sobel Filter	Low Pass Filter	High Pass Filter
PSNR	11.02600974	10.203212	16.00186648	20.0387549
SNR	18.6863	17.9312	17.5419	17.6658
MSE	71.6539	78.7734	36.4447	25.3865

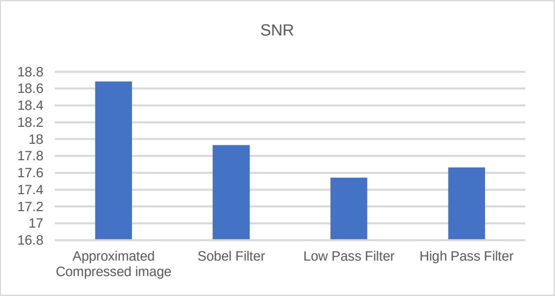


Fig 5.1: Representation of SNR values of Approximated Compressed Image, Sobel filter, low pass filter, high pass filter

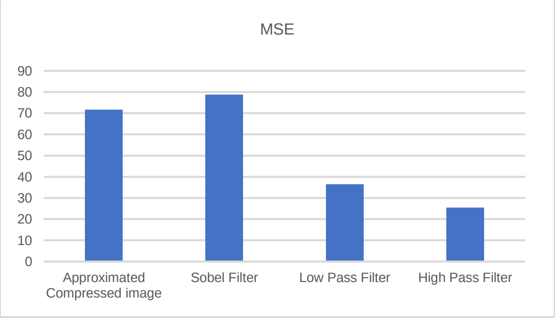


Fig 5.2: Representation of MSE values of Approximated Compressed Image, Sobel filter, low pass filter, high pass filter

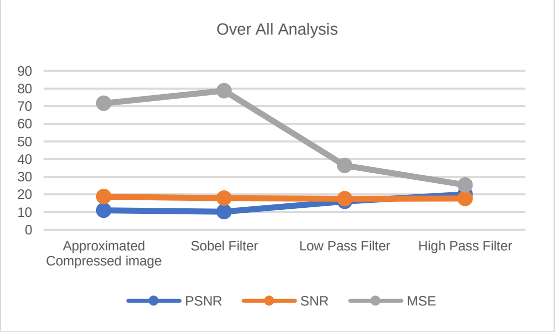


Fig 5.3: Over all representation of MSE values of Approximated Compressed Image, Sobel filter, low pass filter, high pass filter

VI. CONCLUSION AND FUTURE WORK

In this paper, the approximate compressor is been designed, and its SNR, PSNR, MSE values are calculated. After that the approximated compressed image is been obtained, it is further been filtered using Sobel filter, Low Pass Filter and High Pass Filter. Then for each of the obtained filtered image, PSNR, SNR, MSE values are been calculated and further been analyzed. The results shows that the PSNR value is found to be less for Sobel Filter. The SNR values of all the output images are found to be in good range.

The future scope of research work is, analyzing the combination of Sobel and Low pass filter, Sobel high pass pass. Other Filters can also be used for analyzing the best results. The proposed project was under mainly for digital images, but this can also be done for the medical images.

REFERENCES

[1] Qiang et al, "Approximate Computing: A Survey" IEEE Design & Test (Volume: 33, Issue: 1, Feb. 2016)

[2] Gennaro et al, "Survey on Approximate Computing and Its Intrinsic Fault Tolerance" Electronics 2020, 9, 557; DOI:10.3390/electronics9040557

[3] Swathi et al, "15-4 Approximate Compressor Based Multiplier for Image Processing" 2018 2nd International Conference on Trends in Electronics and Informatics (ICOEI)

[4] Haider el at, "Approximate DCT Image Compression Using Inexact Computing" IEEE Transactions on Computers (Volume: 67, Issue: 2, Feb. 1 2018) Page(s): 149 - 159

[5] K.N. Satone et al, "A Review of Image Compression Techniques" 2017 International conference of Electronics, Communication and Aerospace Technology (ICECA)

[6] Harshapradha et al, "Image Processing Techniques and It's Applications-Review" CSEIT194726| Published: 11Oct2019| September-October-2019[4(7): 143-148] National Conference on Communication Technology & Network Security In association with International Journal of Scientific Research in Computer Science, Engineering and Information Technology© 2019IJSRCSEIT| Volume 4| Issue 7| ISSN : 2456-3307

[7] Bouguezel, M. O. Ahmad, and M. N. S. Swamy, "Low complexity 8×8 transform for image compression, "Electronics Letters, vol. 44, pp. 1249–1250, Sept. 2008.

[8] Dhivya et al, "A Review On Enhancement And Pre-Processing Of images Using Filtering" International Journal Of Research In Computer Applications And Robotics (Vol.2 Issue.1, Pg: 99-107January 2014, ISSN 2320-7345)

[9] Aziz et al, "Image Enhancement Techniques using High pass and Lowpass Filters" International Journal of Computer Applications (109(14):21-27 January 2015 DOI: 10.5120/19256-0999)

[10] Chethan et al, "Analysis of Image Quality using Sobel Filter" 2019 Third International Conference on Inventive Systems and Control (ICISC) DOI:10.1109/ICISC44355.2019.9036369