IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

NOVEL APPROACH FOR IMAGE COMPRESSION USING MODIFIED SVD

Dr. H S Prasantha (0000-0003-4739-517X)
Professor
Department of Electronics & Communication Engineering
Nitte Meenakshi Institute of Technology, Bangalore, Karnataka, India

Abstract: Image compression aims at reducing the number of bits required to represent an image by removing the spatial and spectral redundancies as much as possible. The focus of the research work is only on still image compression. The lossy compression methods which give higher compression ratio are considered in the research work. The paper discusses the image compression using SVD (Singular value Decomposition) method on various images of any type and resolution. A novel method is proposed to preprocess the SVD which reduces the computation complexity compared (in terms of time or clock cycles) to SVD. The proposed method termed as Modified SVD (MSVD) which is compared with the actual SVD and other variation of SVD based on different parameters.

***Index Terms* – SVD-Singular Value Decomposition, compression ratio, modified SVD, block size, image compression**

1. INTRODUCTION

Image compression is a type of data compression applied to digital images, to reduce their cost for storage or transmission. Its aim is to reduce the number of bits required to represent an image by removing the spatial and spectral components to lowest possible level. The redundancy and irrelevancy that is present in the image will be reduced during image compression. It aims in reducing the size of multispectral image using image compression techniques. The objective is to reduce the transmission and storage requirements of image data. The two types of compression which describes image compression are Lossless compression and Lossy compression

The different transform coding techniques used for image compression includes Discrete Cosine Transform (DCT), Haar transform, Singular Value Decomposition (SVD), Slant transform, Hadamard transform, Kahrunen Loeve Transform (KLT), etc (Dr.Edel Garcia 2006), (Andrew.B.Watson 1994). (Sindhu.M 2009), (Shivali.D.Kulkarni 2008), (Mr.T.Sreenivasulu Reddy 2007), (Sathish.K.Singh 2010). The suitability of the transform is due to energy compaction property (kamrul Hasn Talukdar 2007). Also, suitability of the transforms is due to subjective quality of the decompressed images in terms of PSNR (Peak signal to Noise Ratio) and quality index, computation time and energy compaction property.

A variation to the SVD based image compression technique is proposed to compress the given input image. The variation can be viewed as a preprocessing step in which the input image is permuted as per a fixed, data independent permutation, after which it is fed to the standard SVD algorithm (Abhiram Ranade 2007). The DCT is used to transform the highly correlated blocks of the YCbCr components, while the SVD is used to transform the low correlated blocks (Y. Wongsawat, 2004). A good compromise between the quality and the compression rate factors that are achieved when processing images by the DCT technique are discussed (A. Messaoudi 2005).

Image Compression is minimizing the size of an image without degrading the quality of the image to an unacceptable level. The reduction in file size allows more images to be stored in a given amount of disk or memory space. It also reduces the time required for images/video to be sent over the internet or downloaded from web pages. The SVD is a fundamental concept in science and engineering, and one of the most central problems in numerical linear algebra. It is also known as principal component analysis (PCA) in statistics and the Karhunen-Loeve (KL) or Hotelling expansion in pattern recognition. The beauty of the SVD is that it provides a robust method of storing larger images as smaller square ones. This is accomplished by representing the original image with each succeeding non-zero singular values. To reduce the storage size even further, one may approximate a “good enough” image with using even fewer singular values. SVD, one of the most useful tools of linear algebra is a factorization and approximation technique which effectively reduces any matrix into smaller invertible and square matrix. SVD is preferred over DCT, Haar and other transforms is due to the suitability of SVD even though matrix is not invertible and non square. Also DCT, Haar transforms are linear where as SVD is nonlinear transformation.

The SVD involves the decomposition of an image represented as matrix A in to U, S and V matrices where $UU^T = I, VV^T = I,$ I is a identity matrix, the columns of U are orthonormal eigenvectors of AA^T, the columns of V are orthonormal eigenvectors of $A^T A$ and S is a diagonal matrix containing the square roots of Eigen values from U or V in descending order. The columns of U are the left singular vectors, S has singular values and is diagonal and V^T has rows that are the right singular vectors. The SVD represents an expansion of the original data in a coordinate system where the covariance matrix is diagonal. Decomposition of the image into U, S and V matrix is computationally very complex and for reconstruction multiplication of U, S and V matrix is also very complex. Hence method is proposed to minimize the complexity of the algorithm. A modified method proposed for preprocessing the SVD can be estimated in terms of the memory requirements and the computation time required in comparison with the SVD. The novel method of preprocessing reduces the computational complexity and also provides easy way of implementing SVD with reduced block size.

Different Image File Formats: The proposed method and actual SVD are applied to on images with different file formats. The different file formats and their descriptions are

1.1 RAW: RAW is an image output option available on some digital cameras. Though lossless, it is a factor of three of four smaller than TIFF files of the same image. The disadvantage is that there is a different RAW format for each manufacturer, and so you may have to use the manufacturer's software to view the images. (Some graphics applications can read some manufacturer's RAW formats.)

1.2 Bitmap File Format (BMP): The BMP file extension normally represents that the file is an image. BMP files are normally used on Windows machines though many other operating systems have programs to view such files.BMP files are made up of pixels, tiny dots that represent color in an image. Some BMP files may only contain black-and-white images (1-bit), others images that have up to 16 colors (4-bit), other 256 colors (8-bit), some 65,536 colors (16-bit), and others 16 million (24-bit). The more colors that can be represented in a bitmap the larger the file. You can shrink a bitmap's size to some extent and still retain much of the image quality. However, if you increase a bitmap's size, no new detail is automatically added. Thus, the more you resample a bitmap upward, the blockier and more jagged it will look as the pixels turn into larger blocks of the same color. This results in 'pixelated' low-quality images. A bitmap file is a raster (or pixel) based format that only supports the RGB color space and bit depths of 1, 4, 8, or 24 bits per channel. These attributes make bitmap images unsuitable for use in a high-end print production workflow.

Even though bitmap images are in the RGB color space, they are not supported by any Web browsers or Web coding languages. Therefore, they are not suitable for use as images in a Web application. Bitmap images are best used for their intended purpose, as a system support on a PC Windows-based computer.

1.3 Portable Network Graphics (PNG): The PNG compression algorithm is one of the best that can be found. Unlike standard JPEG images, PNG compression involves no loss of image data, so there is no smudging or blurring. In some cases, filtering can compress the image data even more. Each horizontal line in an image can have one of five filter types associated with it. Whether specifying a filter helps in a particular case depends on the image content. In most cases the default setting is best (adaptive filtering).

PNG format offers binary transparency equivalent to GIF transparency. It also has a more impressive option-variable transparency, which is referred to as "alpha transparency", "alpha-channel transparency", or simply the "mask channel". Between all-or-none it allows 254 levels of partial transparency. Nearly all the latest browsers support PNG's variable transparency, including WebTV and Microsoft Internet Explorer for Macs. Pre-IE7 versions of Internet Explorer are the exception, but background colors (transparent in other browsers) can be selected and added to alpha images. The image below shows some examples:

1. 24 bit PNG with alpha transparency.
2. 8 bit PNG with binary transparency.
3. First example in pre-IE7 versions of Internet Explorer for Windows.
4. 8 bit GIF with binary transparency.

PNG format allows all kinds of extra information to be stored inside image files. The two most potentially helpful features for web images are gamma correction and embedded text.

Gamma correction is the ability to correct for differences in how computer systems interpret color values. PNG format allows the gamma value of the computer which created an image to be embedded into the image file. Browsers and image software which support the feature can extract the gamma value and make a correction if the host computer uses a different gamma setting.

PNG is of principal value in two applications:

1. If you have an image with large areas of exactly uniform color, but contains more than 256 colors, PNG is your choice. Its strategy is similar to that of GIF, but it supports 16 million colors, not just 256.
2. If you want to display a photograph exactly without loss on the web, PNG is your choice. Later generation web browsers support PNG, and PNG is the only lossless format that web browsers support.

1.4 Tagged Image File Format (TIFF): TIFF is, in principle, a very flexible format that can be lossless or lossy. In practice, TIFF is used almost exclusively as a lossless image storage format that uses no compression at all. Most graphics programs that use TIFF do not use compression. This is usually the best quality output from a digital camera. Digital cameras often offer around three JPG quality settings plus TIFF. Since JPG always means at least some loss of quality, TIFF means better quality.

A more important use of TIFF is as the working storage format as you edit and manipulate digital images. You do not want to go through several load, edit, save cycles with JPG storage, as the degradation accumulates with each new save. One or two JPG saves at high quality may not be noticeable, but the tenth certainly will be. TIFF is lossless, so there is no degradation associated with saving a TIFF file.

1.5 Graphics Interchange Format (GIF): GIF creates a table of up to 256 colors from a pool of 16 million. If the image has fewer than 256 colors, GIF can render the image exactly. When the image contains many colors, software that creates the GIF uses any of several algorithms to approximate the colors in the image with the limited palette of 256 colors available. Better algorithms search the image to find an optimum set of 256 colors. Sometimes GIF uses the nearest color to represent each pixel, and sometimes it uses "error diffusion" to adjust the color of nearby pixels to correct for the error in each pixel. GIF achieves compression in two ways. First, it reduces the number of colors of color-rich images, thereby reducing the number of bits needed per pixel, as just described. Second, it replaces commonly occurring patterns (especially large areas of uniform color) with a short abbreviation: instead of storing "white, white, white, white, white," it stores "5 white." Thus, GIF is "lossless" only for images with 256 colors or less. For a rich, true color image, GIF may "lose" 99.998% of the colors. If your image has fewer than 256 colors and contains large areas of uniform color, GIF is your choice. The files will be small yet perfect.

1.6 Joint Photographic Experts Group: JPEG is optimized for photographs and similar continuous tone images that contain many, many colors. It can achieve astounding compression ratios even while maintaining very high image quality. GIF compression is unkind to such images. JPEG works by analyzing images and discarding kinds of information that the eye is least likely to notice. It stores information as 24 bit color. Important: the degree of compression of JPG is adjustable. At moderate compression levels of photographic images, it is very difficult for the eye to discern any difference from the original, even at extreme magnification. Compression factors of more than 20 are often quite acceptable. Better graphics programs, such as Paint Shop Pro and Photoshop, allow you to view the image quality and file size as a function of compression level, so that you can conveniently choose the balance between qualities and file size. JPEG is the format of choice for nearly all photographs on the web. We can achieve excellent quality even at rather high compression settings. We can also use JPEG as the ultimate format for all digital photographs. Digital cameras save in a JPG format by default. Shooting in TIFF has two disadvantages compared to JPEG: fewer photos per memory card, and a longer wait between photographs as the image transfers to the card. Never use JPG for line art. On images such as these with areas of uniform color with sharp edges, JPG does a poor job. These are tasks for which GIF and PNG are well suited .A JPEG file is encoded by using an adjustable lossy compression approach. This means that to achieve smaller file sizes, image data is actually thrown away. In small doses, the JPEG compression approach can be very effective and efficient. However, in larger amounts, the resulting file will contain noise" and undesired artifacts in the image. Be very careful when preparing JPEG files for use in a print production workflow. The JPEG format will support the RGB, CMYK, and gray scale color spaces. The use of JPEG images is supported in HTML and Web applications. However, unlike a GIF file, all of the color information is stored in the file. There is no support for transparency in a JPEG file.

II. RESEARCH METHODOLOGY

The methodology involves actual SVD, Variation in SVD and modified SVD. The steps or the procedure involved in Variation in SVD and modified SVD are discussed as follows.

2.1 Variation in SVD for Image Compression (SSVD)

SSVD is a variation in SVD based image compression proposed by Abhiram Raanade, Srikanth S. M. and Satyen Kale. It involves pre-processing the image matrix prior to applying SVD. Let A again be N ×N. Suppose further that $N = n^2$ with n integer.Informally, we form $X = S(A)$ by

a) breaking A into blocks of size $n \times n$,
b) Taking the i^{th} block in rowmajor order and arranging its pixels in row major order to produce the i^th row of X. More precisely

$$X[[i/n]n + [j/n], (i \bmod n)n + j \bmod n] = A[i, j]$$

The original image matrix is permuted a fixed number of times in the pre-processing stage. SVD is applied on this permuted image. During the reconstruction process, the reconstructed image is again permuted the same number of times to get back the pixels back to their original positions.

The disadvantages of SSVD are

- It involves resizing the image into a square image.
- Its dimensions must be a perfect square.
- This increases the overhead during computation.
- SSVD is slower compared to SVD because of Preprocessing involved.

2.2 Proposed Method - Modified SVD (MSVD)

In the proposed method, instead of directly applying the SVD on the entire image, we segment the image into blocks of smaller sub images of size 64 X 64. The algorithm used to compute SVD is then applied onto these sub-images individually. To reconstruct the image, individual sets of U, S and V matrices are used to re-compute the respective 64 X 64 blocks. These blocks are then arranged re-placed in their original positions to obtain the complete image.

The use of Singular Value Decomposition (SVD) in image compression has been widely studied [1, 3, 9, 10]. If the image, when considered as a matrix, has low rank, or can be approximated sufficiently well by a matrix of low rank, then SVD can be used to find this approximation, and further this low rank approximation can be represented much more compactly than the original image. More specifically, suppose we are given an image A which we will assume for simplicity is a $N \times N$ real matrix. Then we first factor it into its SVD representation $A = U\Sigma V^T$, where Σ is a diagonal matrix with entries along the diagonal ordered in a non-increasing order, and U; V are orthogonal matrices [4]. Then a rank r approximation to A is the matrix $A_r = U_r \Sigma_r V_r^T$, where Σ_r is the top-left $r \times r$ sub-matrix of Σ, U_r consists of the first r columns of U, and V_r^T the first r rows of V^T. The SVD decomposition is interesting because U_r, Σ_r, V_r provide the best rank r approximation to A in the sense of packing the maximum energy from A. Furthermore, for compression, the decomposition is interesting because unlike A which has N^2 entries, the total number of entries in U_r, Σ_r, V_r^T are only $2Nr + r$. It often turns out that even with small r, the approximation A_r gets most of the energy of A, and is visually adequate, hence the attractiveness of the method.

In the proposed approach, we first scale up the dimensions of the given image to the next integral multiples of 64, by replicating the last row and/or column required number of times. Further, it is divided into 64×64 blocks and SVD is applied on each block independently. SVD of rank r is computed. Hence compared to 4096 pixel values of the original block, we are left with $2 \times 64 \times r + r$=129r pixel values. Thus for an image of dimension $M \times N$ where M and N are the integral multiples of 16, we need to store

$\frac{M \times N}{64 \times 64} \times 129r = m \times n \times 129r$ Pixel values compared to $M \times N$ pixel values, where, $m = M/64$ and $n = N/64$

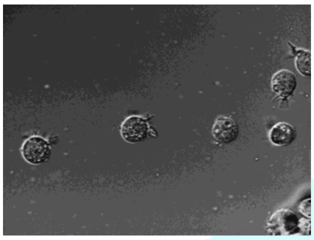
During the process of reconstruction, the values of U, S and V for each block is used to reconstruct the block and then these $m \times n$ blocks are arranged in the same fashion as they are segmented from the original image matrix. Thus the entire image is reconstructed. This reconstructed version of the original image is similar to an SVD approximation of rank R equal to

$r \times \frac{\left(\frac{m}{bl}*\frac{n}{bl}*(2*bl+1)\right)}{(M+N+1)}$ applied to the entire image at once.

III. IMPLEMENTATION DETAILS

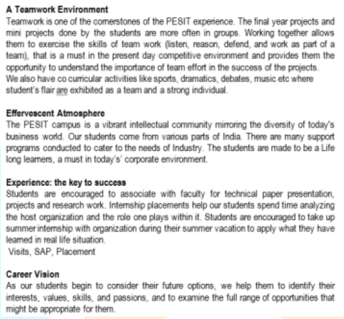
Different image file formats of different resolutions are considered for the experimentation. The different file formats such a TIFF, JPEG, PNG, BMP, etc are considered for experimentation. Extensive experimentation is done using different file formats of different resolutions and by using actual SVD, Variation in SVD (SSVD) and Modified SVD. Proposed method is considered and the experiments are conducted and the comparison of the result is done with actual SVD and variation in SVD using different comparison parameters. The different comparison parameters considered are MSE, PSNR, Normalized cross correlation, rank, compression ratio and computation time, HVS (Human Visual System) measures. Figure 1 shows the different input image considered for experimentation

TIFF IMAGE (640X 480)



JPEG IMAGE (483X475)

TEXT IMAGE (500X550)



A Teamwork Environment
Teamwork is one of the cornerstones of the PESIT experience. The final year projects and mini projects done by the students are more often in groups. Working together allows them to exercise the skills of team work (listen, reason, defend, and work as part of a team), that is a must in the present day competitive environment and provides them the opportunity to understand the importance of team effort in the success of the projects.
We also have co curricular activities like sports, dramatics, debates, music etc where student's flair are exhibited as a team and a strong individual.

Effervescent Atmosphere
The PESIT campus is a vibrant intellectual community mirroring the diversity of today's business world. Our students come from various parts of India. There are many support programs conducted to cater to the needs of Industry. The students are made to be a Life long learners, a must in today's' corporate environment.

Experience: the key to success
Students are encouraged to associate with faculty for technical paper presentation, projects and research work. Internship placements help our students spend time analyzing the host organization and the role one plays within it. Students are encouraged to take up summer internship with organization during their summer vacation to apply what they have learned in real life situation.
Visits, SAP, Placement

Career Vision
As our students begin to consider their future options, we help them to identify their interests, values, skills, and passions, and to examine the full range of opportunities that might be appropriate for them.

BMP IMAGE (600X813)

PNG IMAGE (189X250)

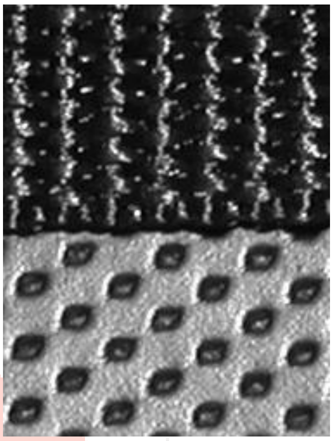
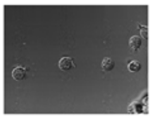
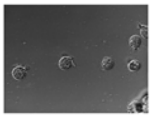
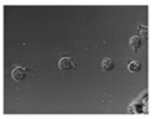
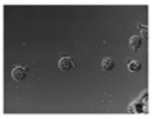
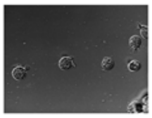
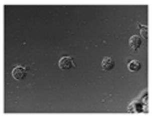


Figure 1: Input image considered for experimentation

IV. EXPERIMENTAL RESULTS AND DISCUSSIONS

	Rank	Compression Ratio	Time	MSE	PSNR	Normalized Cross Correlation	HVS
SVD	32	8.5638	1.1054	28.2566	33.6196	0.9955	
SSVD	30	7.5684	1.1054	28.4413	33.5913	0.9955	
MSVD	5	6.0553	1.552	28.4731	33.5865	0.9955	

	Rank	Compression Ratio	Time	MSE	PSNR	Normalized Cross Correlation	HVS
SVD	71	3.8597	1.5677	8.212	38.9863	0.9987	
SSVD	62	3.6621	1.5677	8.122	39.0342	0.9987	
MSVD	10	2.9937	2.6249	8.2932	38.9436	0.9987	

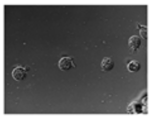
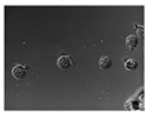
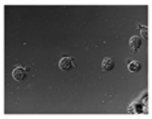
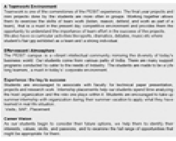
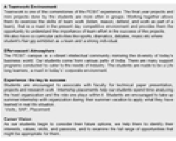
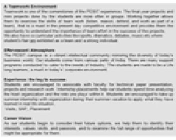
	Rank	Compression Ratio	Time	MSE	PSNR	Normalized Cross Correlation	HVS
SVD	272	1.0075	15.6555	0.4019	52.09	0.9999	
SSVD	285	0.7967	15.6555	0.4138	51.9631	0.9999	
MSVD	30	0.9942	3.883	0.3842	52.2855	0.9999	

Figure 2: The comparison result of the proposed method with other methods for the input in TIFF Format

	Rank	Compression Ratio	Time	MSE	PSNR	Normalized Cross Correlation	HVS
SVD	34	7.6958	0.7004	994.9069	18.153	0.9825	
SSVD	78	3.0578	1.7566	993.9952	18.157	0.9825	
MSVD	5	6.0125	1.2553	986.8219	18.1884	0.9826	

	Rank	Compression Ratio	Time	MSE	PSNR	Normalized Cross Correlation	HVS
SVD	235	1.1134	4.2262	8.6676	38.7518	0.9998	
SSVD	332	0.7184	8.0262	8.6111	38.7802	0.9998	
MSVD	30	0.9942	8.5474	8.7374	38.717	0.9998	

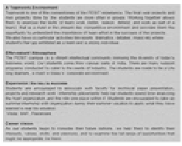
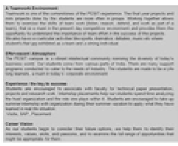
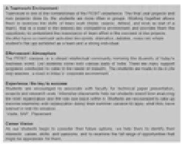
	Rank	Compression Ratio	Time	MSE	PSNR	Normalized Cross Correlation	HVS
SVD	79	3.3121	1.297	384.1646	22.2856	0.9932	
SSVD	151	1.5795	2.9696	385.1216	22.2748	0.9932	
MSVD	10	3.0063	1.8615	383.2524	22.296	0.9933	

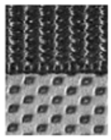
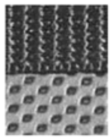
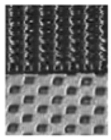
Figure 3: The comparison result of the proposed method with other methods for the input Text

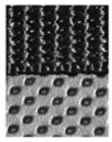
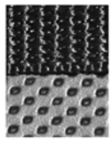
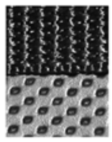
	Rank	Compression Ratio	Time	MSE	PSNR	Normalized Cross Correlation	HVS
SVD	6	57.4965	0.46	58.4741	30.4612	0.995	
SSVD	8	36.2299	0.95	52.411	30.9366	0.9955	
MSVD	1	30.1055	1.25	56.2378	30.6305	0.9954	

	Rank	Compression Ratio	Time	MSE	PSNR	Normalized Cross Correlation	HVS
SVD	59	5.8471	2.03	9.217	38.4849	0.9992	
SSVD	64	4.5287	3.1	9.2905	38.4504	0.9992	
MSVD	5	5.8098	2.4963	9.2556	38.4667	0.9992	

	Rank	Compression Ratio	Time	MSE	PSNR	Normalized Cross Correlation	HVS
SVD	118	2.9235	3.95	5.3268	40.8662	0.9995	
SSVD	127	2.2822	6.38	5.3272	40.8659	0.9995	
MSVD	10	2.9049	4.3472	5.3222	40.8699	0.9995	

Figure 4: The comparison result of the proposed method with other methods for the input in BMP

	Rank	Compression Ratio	Time	MSE	PSNR	Normalized Cross Correlation	HVS
SVD	17	6.3168	0.2919	387.6159	22.2468	0.9693	
SSVD	20	4.6053	0.3512	394.64	22.1687	0.9685	
MSVD	5	6.1902	0.4277	396.26	22.15	0.9686	

	Rank	Compression Ratio	Time	MSE	PSNR	Normalized Cross Correlation	HVS
SVD	110	0.9762	0.6217	3.3608	42.8664	0.9997	
SSVD	120	0.7675	0.9661	3.8654	42.2589	0.9997	
MSVD	30	1.0217	0.8444	3.9021	42.2178	0.9997	

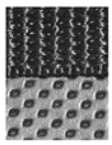
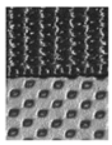
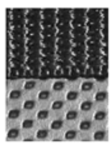
	Rank	Compression Ratio	Time	MSE	PSNR	Normalized Cross Correlation	HVS
SVD	37	2.9023	0.3564	152.5119	26.2978	0.9879	
SSVD	39	2.3617	0.409	152.9737	26.2846	0.9877	
MSVD	10	3.0951	0.5838	155.8524	26.2037	0.9877	

Figure 5: The comparison result of the proposed method with other methods for the input in PNG

	Rank	Compression Ratio	Time	MSE	PSNR	Normalized Cross Correlation	HVS
SVD	40	5.9808	0.7814	63.7213	30.088	0.9977	
SSVD	22	10.762	0.7158	63.9703	30.071	0.9977	
MSVD	5	5.5957	1.1193	64.8556	30.0113	0.9977	

	Rank	Compression Ratio	Time	MSE	PSNR	Normalized Cross Correlation	HVS
SVD	80	2.9904	1.3436	18.1396	35.5445	0.9993	
SSVD	58	4.0822	1.0721	18.3091	35.5041	0.9993	
MSVD	10	2.7979	2.023	18.1276	35.5474	0.9993	

	Rank	Compression Ratio	Time	MSE	PSNR	Normalized Cross Correlation	HVS
SVD	229	1.0447	6.053	0.2908	53.4947	1	
SSVD	225	1.0523	6.1428	0.2978	53.3914	1	
MSVD	30	0.9326	4.5306	0.291	53.4919	1	

Figure 6: The comparison result of the proposed method with other methods for the input in JPG

Figure 2-6 indicated the result obtained for the proposed method with block size of 64 X 64 and comparison of the result obtained with Variation in SVD and actual method. The different comparison parameters are tabulated for the proposed method with the Variation in SVD and actual method and are listed in the figure 2-6. PSNR OF minimum 30 dB are maintained throughout the experimentation which is generally good enough. From the result obtained, it is clear that in the proposed method the same quality in terms of PSNR can be obtained for lower ranks compared to Variation in SVD (SSVD) and actual method. Lower ranks indicated least complexity in terms of computation time. In case of SVD, challenging task is computational complexity to obtain higher ranks since higher ranks provide better quality

V. SUMMARY AND CONCLUSIONS

In the proposed approach, we partition the image matrix into blocks of equal sizes. We apply SVD to each of these blocks independently. As the blocks are considered to be independent, parallel processing can be achieved. Depending on the capability of hardware being used, multiple blocks can be processed simultaneously which results in reduction of computation time required to decompose and reconstruct the image. Even after considering different blocks independently and applying SVD on them the quality of the reconstructed image is not degraded by a large value. Due to this property the image quality assessment parameters are almost coinciding with that of the traditional approach. Also the compression ratio obtained is nearly same as that of the traditional approach.

Though resizing of the original image is inevitable in the proposed approach, the amount of resizing is very much less when compared to SSVD approach. This is due to the fact that in our algorithm, the dimensions of the image must be resized to the next higher integral multiple of the block size. Also, the image need not necessarily be a square image, whereas for an image to apply SSVD, the image should be a square one and its dimensions should be a perfect square.

Nowadays majority of the images encountered are rectangular and hence the number of columns are nearly twice as that of the number of rows. If you consider SSVD approach, the image has to be resized to form a square image i.e., we are unnecessarily replicating the pixels to form a square image. This increases both time and computation complexity and also memory required to store the resized image will be more. Whereas if you consider proposed approach, the image dimensions must be resized to the next higher integral multiple of the block size. Hence both time and computation complexity are drastically reduced, and also the memory constraint is better handled in our approach as compared to the other two approaches.

REFERENCES

1. H. S. Prasantha, H. L. Shashidhara, and K. N. Balasubramanya Murthy. Image compression using SVD. In Proceedings of the International Conference on Computational Intelligence and Multimedia Applications, pages 143–145. IEEE Computer Society, 2007
2. Abhiram Ranade, *et. al.*, “A Variation on SVD Based Image Compression”, Image Vision Computation”, 25(6), 20017, pp.771-777, 2007
3. Messaoudi, *et. al.*, “A Good Compromise between Images Quality and Compression Rate of the DCT Technique”, GVIP Conference, CICC, Cairo, Egypt., 2005
4. Dr. Edel Garcia, “Singular Value Decomposition (SVD)” A Fast Track Tutorial Published on September 11 2006.
5. Niranjan Damera Venkata, “Image Quality Assessment Based on a Degradation Model”, IEEE transactions on image processing, Vol. 9, No. 4, 2000.
6. Satu Jumisko, *et .al.*, “Evaluation of Subjective Video Quality of Mobile Devices”, MULTIMEDIA '05, Proceedings of the 13th annual ACM international conference on Multimedia, 2005
7. Gunasheela K S, H S Prasantha, “Satellite image compressiondetailed survey of the algorithms”, Proceedings of ICCR in LNNS Springer, 2017, vol. 14, pp. 187-198.
8. Raghavendra.M.J, Prasantha.H.S and S.Sandya, “Image Compression Using Hybrid Combinations of DCT SVD and RLE”, International Journal of Computer Techniques, Volume 2 Issue 5-2015
9. Gunasheela K S, H S Prasantha, “Compressive sensing for image compression: survey of algorithms”, Proceedings of Emerging Research in Computing, Information, Communication and Applications, ERCICA, Springer publication, Bengaluru, 2018
10. K N Shruthi, B M Shashank, Y. SaiKrishna Saketh, H.S Prasantha and S. Sandya, "Comparison Analysis Of A Biomedical Image For Compression Using Various Transform Coding Techniques", IEEE, pp. 297-303, 2016
11. Raghavendra.M.J, Prasantha.H.S and S.Sandya, “DCT SVD Based Hybrid Transform Coding for Image Compression”, International Journal of Recent and Innovative Trends in computing and communication. 2015
12. Prasantha H.S., Shashidhara H.L, Balasubra manya Murthy, K N: Comparative analysis of different interpolation schemes in image processing. In: International Conference on Advanced Communication Systems (ICACS), India, pp. 17–24 (2007)
13. Prasantha H.S, Shashidhara H.L and K.N.Balasubramanya Murthy, “Image Scaling comparison using Universal Image Quality Index”, *International Conference on Advances in computing, control and Telecommunication Technologies (ICACCTT)*, India, pp.859-863, Dec 2009.