

A Post-Processing Algorithm for Detection & Removal of Corner Outlier

¹Anudeep Gandam

¹M.Tech Student (ECE)
DAVIET Jalandhar, Punjab

²Jagroop Singh Sidhu

²Asst. Prof (ECE)
DAVIET Jalandhar, Punjab

ABSTRACT

The reconstructed images from JPEG compression produce noticeable image degradation near the block boundaries, in case of highly compressed images, because each block is transformed and quantized independently. The blocking effects are classified into three types of noises: staircase noise, grid noise and corner outlier out of which major thrust is laid on corner outlier in this paper. A post-processing algorithm is proposed to reduce the blocking artifacts of JPEG decompressed images. The proposed post-processing algorithm, which consists of three stages, reduces the blocking artifacts efficiently. A comparative study between the proposed algorithm and other post-processing algorithms based on various performance indices is made.

General Terms

JPEG Compression, Corner outlier et.al.

Keywords

Blocking artifact, JPEG Compression, post-processing, adaptive Filters, PSNR, MSE, MSSIM

1. INTRODUCTION

Image compression is a very important issue for many applications in the field of visual communications. The purpose of image compression is to reduce the storage and transmission costs while maintaining the image quality. Therefore, many efficient image-coding techniques have been developed for various applications, and some of them are used as international standards for image communication.

The Joint Photographic Experts Group (JPEG) has been recommended as a standard compression scheme for continuous-tone still images [1]. JPEG uses an 8X8 pixel-block discrete cosine transform (BDCT) for packing information into a few transform coefficients. This block DCT scheme takes advantage of the local spatial correlation property of images and also saves processing time [2]. However, it is well known that this individual processing of each block induces visually annoying blocking effects, in particular, when a high quantization parameter is used for high compression.

There are three kinds of blocking effects in JPEG decompressed images. One is the staircase noise along the image edges, another is the grid noise in the monotone area, and the last one is the corner outlier in the corner point of the 8X8 DCT block, as shown in Fig. 1. When an 8 X8 block includes an image edge, the edge is degraded such that the block boundary looks like the edge. This artifact is called *staircase noise*. A slight change of image intensity along the 8X8 block boundary is easily noticeable

in the monotone area; this change is called *grid noise*. A *corner outlier* is visible at the corner point of the 8X8 block, where the corner point is either much larger or much smaller than neighboring pixels.

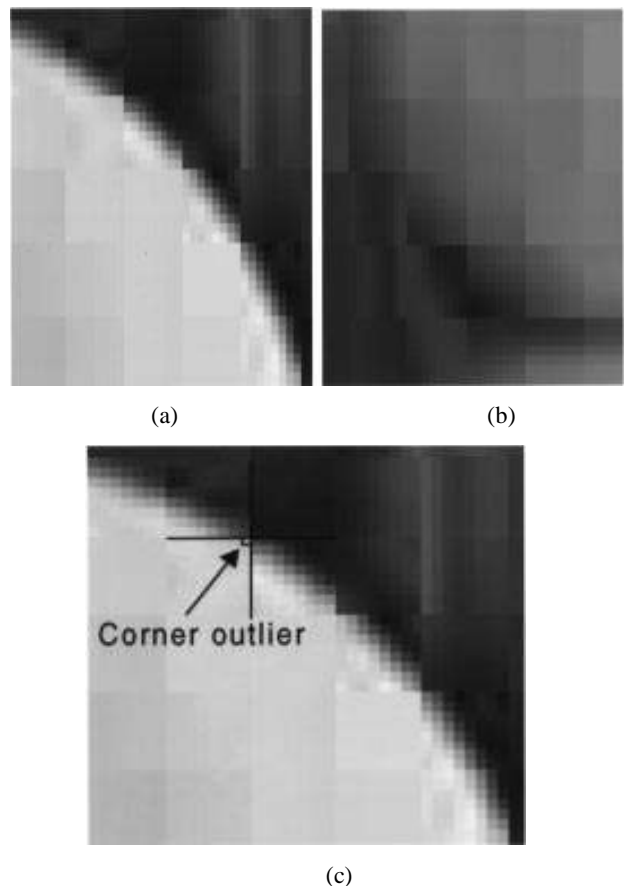


Fig1: Typical blocking effects of JPEG decompressed images.

(a) Staircase noise. (b) Grid noise. (c) Corner outlier.

1.1 Definition of the Corner Outliers

Consider the original and its reconstructed images illustrated in Fig. 2, where a diagonal edge goes through a block-corner. The edge occupies large areas in blocks **B** and **C**, whereas it occupies very small areas (d_0 in Fig. 2(a)) in block **D**. If block **D** is flat except d_0 , the AC coefficients of DCT of block **D** are mainly related to d_0 , and their values are small. Since most small AC

coefficients are truncated by quantization in very low bit-rate coding, the area of d_0 , which represents the edge in block **D**, cannot be reconstructed as shown in Fig. 2(b). As a result, the visually annoying stair-shaped artifact is produced around the block-corner. Such artifacts are called corner outliers.

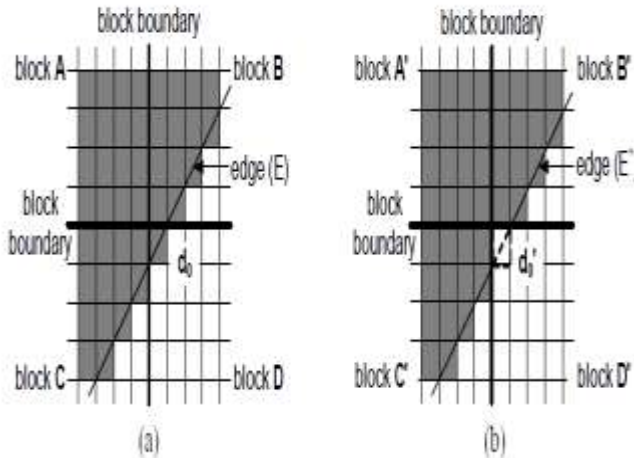


Fig 2: Conceptual illustration of a corner outlier, (a) an edge going through a block-corner in an original frame, (b) an corner outlier located in d_0 in a reconstructed frame.

In this work, a new post-processing algorithm based on signal adaptive filtering along with a corner outlier detection/replacement scheme is proposed. The proposed algorithm reduces the blocking effects, such as grid noise, staircase noise and corner outlier, without degradation of image details.

In this paper, Section 2 explains the corner outlier detection/replacement scheme of the proposed algorithm. In Section 3, proposed results are compared with the Lee algorithm [3] and JPEG Compression using various performance indices e.g. PSNR, MSE and MSSIM. Finally, conclusions are given in Section 4.

2. PROPOSED ALGORITHM FOR DETECTION & REMOVAL OF CORNER OUTLIERS

Several post-processing algorithms to reduce blocking artifacts of block coded images have been proposed [1]. In nonlinear space-variant filtering [2], the image is divided into monotone area or edge area using edge information obtained from detecting edge by gradient operator. The approach given in [3] uses a two-dimensional (2D) low pass filter to reduce grid noise in monotone area and a directional one dimensional filter to reduce staircase noise along the edges. In adaptive filtering [4], the coefficients of 2D filter are chosen by edge information of the image. Weighted or un-weighted filtering is performed at the pixels in the monotone area or near the edge but no filtering is done at edge pixels [5].

In [6], a 2D multiple notch filter is applied to reduce the blocking artifacts. Both nonlinear space-variant filtering and adaptive filtering require classification of image blocks, which is based on

available edge information extracted from the received blocky image. Hence, the performance of space-variant or adaptive filtering scheme degrades. In [7], the blocking artifacts are reduced by processing the image in DCT domain itself. An approach for reducing the blocking artifact in frequency domain is presented in [8]. Mean squared difference of slope (MSDS) is used as a measure the blocking artifacts. It is shown in [9] that the expected value of the MSDS increases after quantizing the DCT coefficients. This approach removes the blocking effect by minimizing the MSDS, while imposing linear constraints corresponding to quantization bounds. Lakhani et al. [10] also used MSDS to reduce blocking effects. They have given a different solution for minimizing the MSDS. Trianta fyllidis et al. [11] have proposed another method of minimizing MSDS, which involves diagonal neighboring pixels in addition to horizontal and vertical neighboring pixels. Then, for each block affected by blocking artifacts, DC and AC coefficients are recalculated for artifact reduction. For texture and edge regions, an edge preserving smoothing filter is applied. In the algorithm, blocking artifacts are modeled as 2-D step function.

Jeon and Jeong [12] proposed a method for post-processing that gives minimum discontinuity of pixel values over block boundaries by compensating the loss of a coefficient's accuracy in the transform domain. In [13], Hsia et al. proposed transform-domain algorithm to effectively classify the characteristics of blocks and estimate the strength of the blocky effect. An adaptive finite impulse response (FIR) filter to effectively remove the blocky effect is also proposed. Chen et al. [14] introduced a DCT-domain post filtering approach to reduce blocking artifacts, where the post-filter made use of the DCT coefficients of shifted blocks in order to obtain a close correlation between the DCT coefficients at the same frequency. The filtering adapts according to the local activity of each block to achieve simultaneous artifact reduction and detail preservation. However, the computational cost is rather high.

A fast DCT-domain algorithm extracts all parameters needed to detect and estimate the blocking artifacts, by exploiting several properties of the human vision system. Using the estimate of blockiness, a novel DCT-domain method is then developed which adaptively reduces the detected blocking artifacts. Luo and Ward [15] proposed a technique, which preserve the edge and texture information. This adaptive approach performs the blocking artifact reduction in both the DCT and spatial domains. For smooth regions, the continuity of original pixel levels in the same block and the correlation between the neighboring blocks is used to reduce the discontinuity of the pixels across the boundaries. For texture and edge regions, an edge preserving smoothing filter is applied.

2.1 Algorithm to Detect Corner Outlier

An algorithm is developed by Y. L. Lee [3] which is used here to detect corner outlier as shown below:-

Algorithm for detection of Corner Outlier

If $(|A - C| > |B - D|)$

If $(|A - B| > |C - D|)$

If $(|A - C| > m \&\& |A - B| > m \&\& |A - D| > m)$

Return (point A)

Else

If ($|A - C| > m \&\& |C - D| > m \&\& |C - B| > m$)

Return (point C)

Else

If ($|C - B| > |A - D|$)

If ($|B - D| > m \&\& |A - B| > m \&\& |C - B| > m$)

Return (point B)

Else

If ($|B - D| > m \&\& |A - D| > m \&\& |C - D| > m$)

Return (point D)

Above algorithm is used to detect the corner outlier. In order to remove the corner outlier the algorithm in the next section is proposed.

2.2 Corner Outlier Reduction

The JPEG decompressed image is classified into two areas, i.e., an *edge area* and a *monotone area*. In general, thresholding is a typical approach to image classification. A global edge map is obtained by thresholding with global threshold value T_g , which is experimentally determined in [3] and is given by:

$$T_g = 10Q_f + 80$$

Where

Q_f is the quantization factor of JPEG compression

The corner outlier detection and replacement algorithm uses a 2X2

window for the cross point of the block boundaries in the decompressed image. This detection and replacement process is performed in the JPEG decompressed image. Fig 3(a) describes a simple coordination for corner outlier detection, where A; B; C, and D are corner points of 8X8 blocks.

The detection procedure is shown at the bottom of the next page, where m is the threshold value, which is 20% of T_g in this work. From this detection procedure, the return point is a corner outlier. A detected corner outlier and adjacent pixels is replaced by the weighted average as shown in Fig. 3(b) For example, if a pixel of A to D is detected as a corner outlier

The pixels of A, A_1 , and A_2 will be replaced with proposed values of a , a_1 , and a_2 , respectively, as follows:

$$a = \text{int}[(5*A+B+C+D)/8]$$

$$a_1 = \text{int}[(2*A_1+A_2+a)/4]$$

$$a_2 = \text{int}[(2*A_2+A_1+a)/4]$$

The pixels of B, B_1 , and B_2 will be replaced with proposed values of b , b_1 , and b_2 , respectively, as follows:

$$b = \text{int}[(5*B+C+A+D)/8]$$

$$b_1 = \text{int}[(2*B_1+B_2+b)/4]$$

$$b_2 = \text{int}[(2*B_2+B_1+b)/4]$$

The pixels of C, C_1 , and C_2 will be replaced with proposed values of c , c_1 , and c_2 , respectively, as follows:

$$c = \text{int}[(5*C+B+A+D)/8]$$

$$c_1 = \text{int}[(2*C_1+C_2+c)/4]$$

$$c_2 = \text{int}[(2*C_2+C_1+c)/4]$$

The pixels of D, D_1 , and D_2 will be replaced with proposed values of d , d_1 , and d_2 , respectively, as follows:

$$d = \text{int}[(5*D+C+A+B)/8]$$

$$d_1 = \text{int}[(2*D_1+D_2+d)/4]$$

$$d_2 = \text{int}[(2*D_2+D_1+d)/4]$$

Where $\text{int} []$ is a round-off operation.

The complete post-processing algorithm can be summed up in figure 4 which depicts various stages namely JPEG compression, detection and removal of corner outlier and then filtering the output image using 2-D median filter which removes noises other than corner outlier. The proposed 3X3 Median filter improve PSNR, MSE and MSSIM indices.

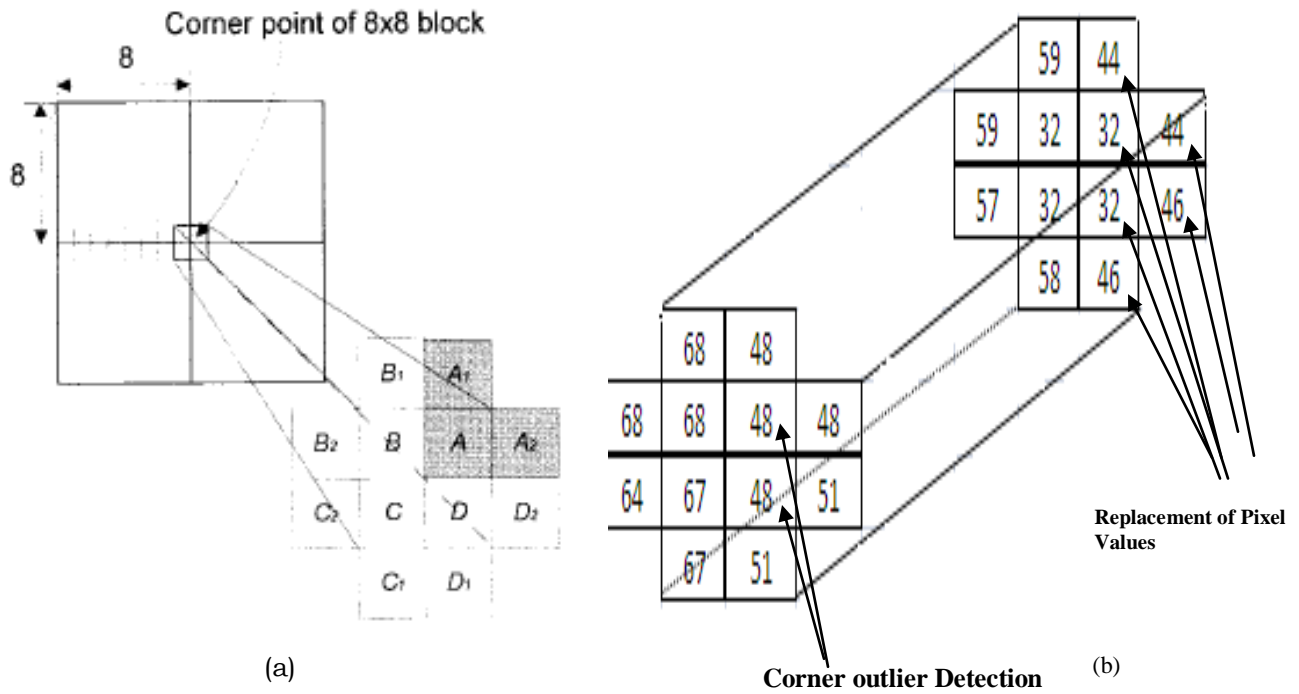


Fig.3: Corner outlier detection and replacement scheme.

(a) Coordination of the corner point. (b) Example of the detection and replacement result.

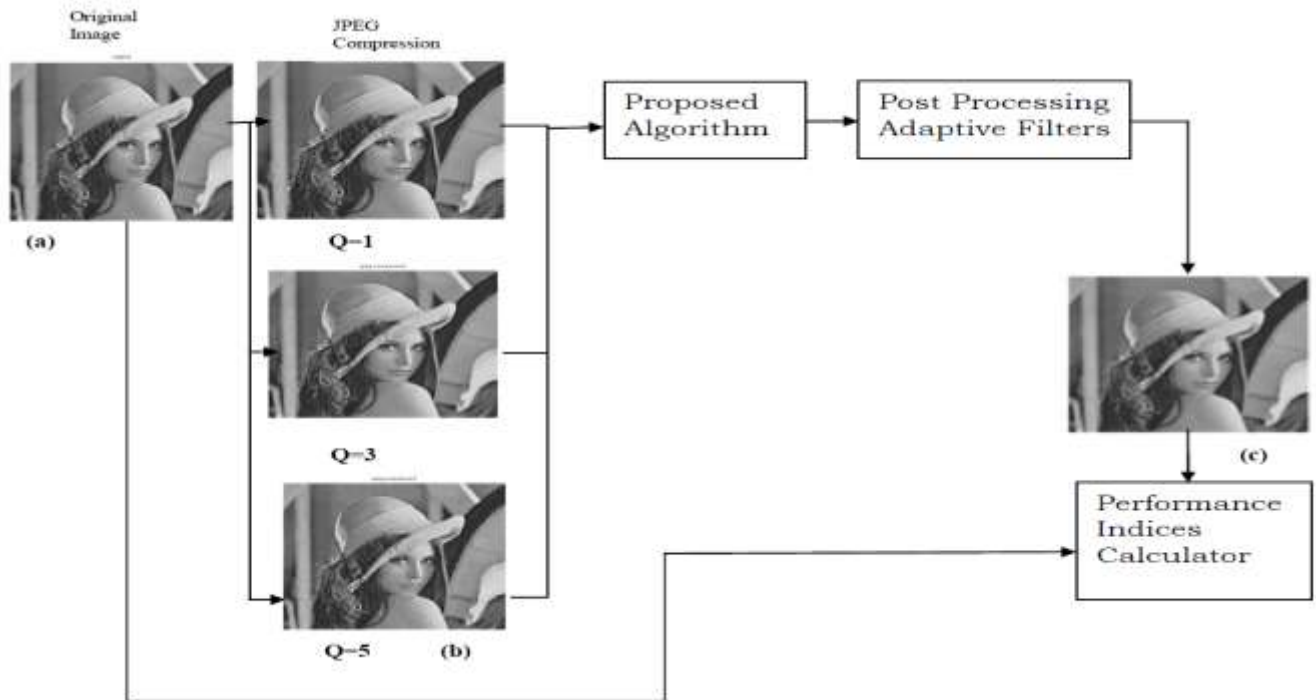


Fig 4: Various stages of Post-processing Algorithm to Detect and Remove Corner Outlier (a) Original Lena (512X512)
(b) JPEG compression of Lena(512X512) at different Q-Factor (c) Post-processed image after filtering

3. RESULTS

The proposed post-processing algorithm were applied to the test images such as Lena, peppers, Pentagon and house all of which were compressed with the standard JPEG code, which was

distributed by the Portable Video Research Group. Resolutions of the images were all 512X512 8-bit gray level. Comparative results are presented using the performance indices (PSNR, MSE & MSSIM) as shown in Table 1.

Table 1: Performance of the proposed algorithms on various compressed images: Comparison of PSNR, MSE & MSSIM (in dB) obtained by various algorithms

Sr. No	Image Name	Q Factor	JPEG Compression			Lee			Proposed Algorithm		
			PSNR	MSE	MSSIM	PSNR	MSE	MSSIM	PSNR	MSE	MSSIM
1	Lena	1	31.01	51.5	0.9901	32.14	39.6	0.991	32.44	36.9	0.9997
2		3	30.07	63.8	0.978	30.96	49.1	0.9809	31.63	44.6	0.9897
3		5	29.24	77.3	0.9651	29.96	60.3	0.9689	30.74	54.7	0.9692
4		7	28.49	91.9	0.952	29.09	72.9	0.9571	29.90	66.4	0.9583
5		10	27.41	117.8	0.928	27.87	94.6	0.9362	28.68	88.0	0.9364
6	Peppers	1	31.00	51.5	0.9895	32.13	39.7	0.9904	32.74	34.5	0.9904
7		3	30.10	63.4	0.9779	31.00	51.6	0.9788	31.85	42.4	0.9804
8		5	29.30	76.3	0.9656	30.03	64.5	0.9664	30.93	52.3	0.9689
9		7	28.51	91.5	0.9506	29.11	79.6	0.9515	30.05	64.1	0.9543
10		10	27.45	116.9	0.9305	27.91	105.0	0.9314	28.80	85.5	0.9354
11	House	1	30.31	60.4	0.9904	31.26	48.6	0.9917	31.64	46.0	0.9918
12		3	28.85	84.7	0.9772	29.50	72.8	0.9786	29.79	68.1	0.9787
13		5	27.80	107.7	0.9623	28.31	95.9	0.9636	28.92	83.2	0.9639
14		7	26.96	130.7	0.9486	27.37	118.9	0.9498	28.11	100.4	0.9503
15		10	25.85	168.9	0.9276	26.16	157.0	0.929	26.93	131.8	0.9307
16	Pentagon	1	29.13	79.3	0.9865	29.83	67.4	0.9873	29.86	67.5	0.9887
17		3	27.89	105.4	0.9689	28.41	93.6	0.9698	28.79	85.8	0.9774
18		5	27.04	128.5	0.9498	27.46	116.6	0.9506	28.03	102.2	0.9512
19		7	26.37	149.8	0.9298	26.73	137.9	0.9307	27.35	119.6	0.9321
20		10	25.51	182.8	0.8979	25.80	170.9	0.8988	26.43	147.8	0.9017

The proposed algorithm is tested on various images with different characteristics. The algorithm is applied on four images (as mentioned in table 1) at different quality factor (Q=1, 3, 5, 7 and 10).The results of Lena (512X512) for different performance indices are shown in Fig: 4 and Table-1.

The figure 5 shows the relationship between PSNR and Quality factor of Lena (512X512) Image using JPEG compression Method, Lee Method and Proposed Algorithm. It is very clear from the graph that there is improvement in PSNR value of Lena image with the use of proposed method over the JPEG method and Lee Method. This increase represents improvement in the objective quality of the image.

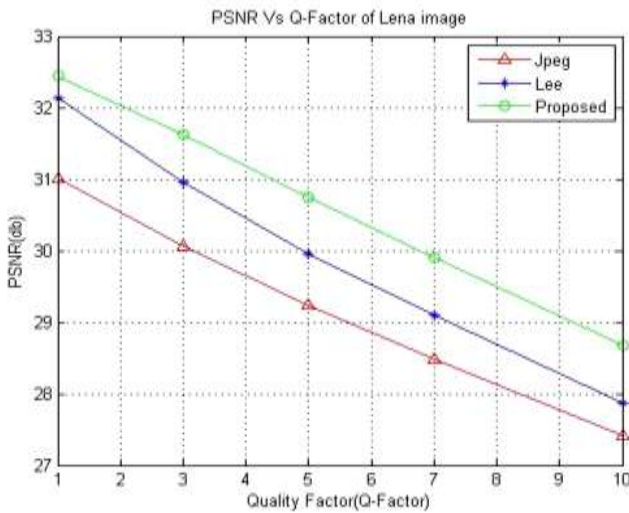


Figure 5: Relationship between PSNR and Quality Factor of Lena Image

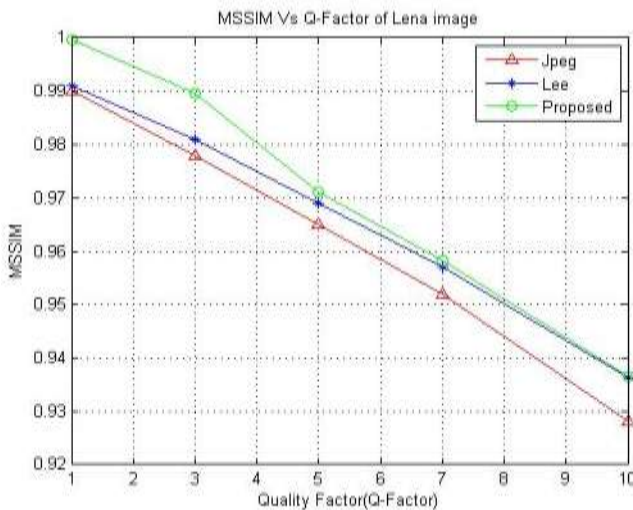


Figure 6: Relationships between MSSIM and Quality Factor of Lena Image

The figure 6 shows the relationship between MSSIM and Quality Factor of Lena (512X512) Image using JPEG compression Method, Lee Method and Proposed Algorithm. It is very clear from the graph that there is improvement in MSSIM value of Lena image with the use of proposed method over the JPEG method and Lee Method. This increase represents improvement in the objective quality of the image.

Similarly the results of performance indices for other images are shown below

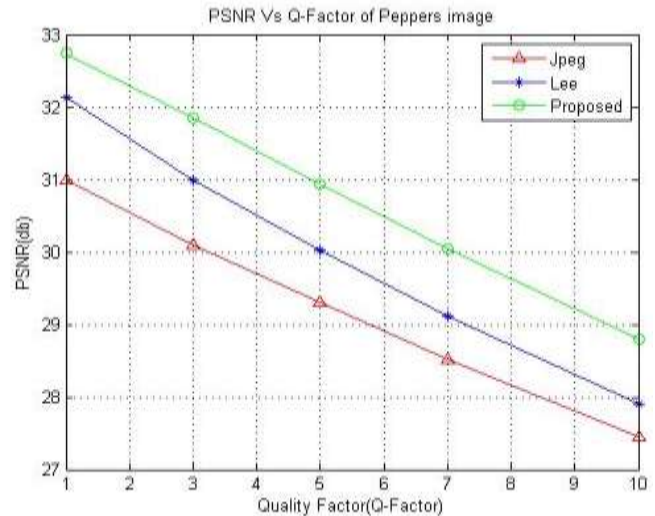


Figure7: Relationship between PSNR and Quality Factor of Peppers Image

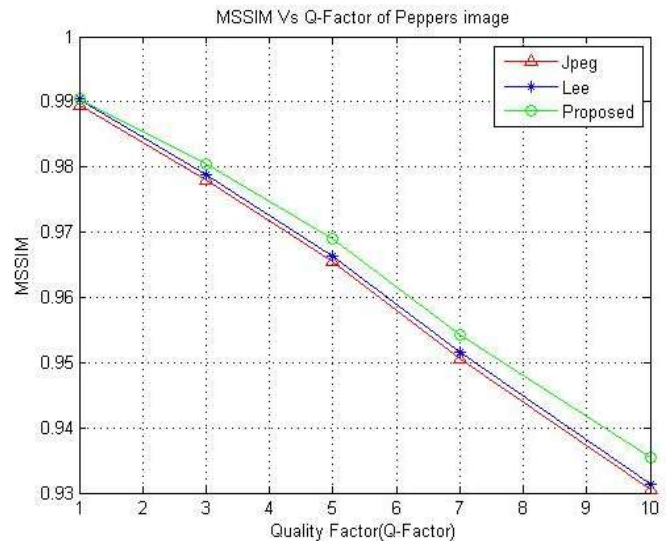


Figure8: Relationships between MSSIM and Quality Factor of Peppers Image

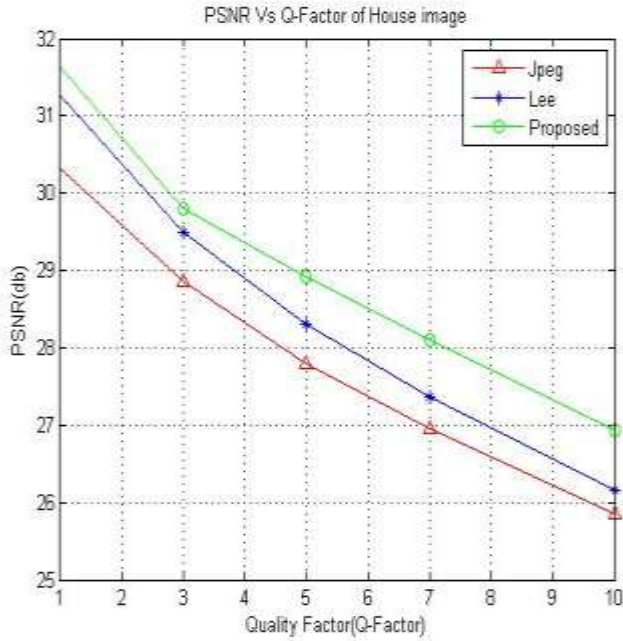


Figure 9: Relationship between PSNR and Quality Factor of House Image

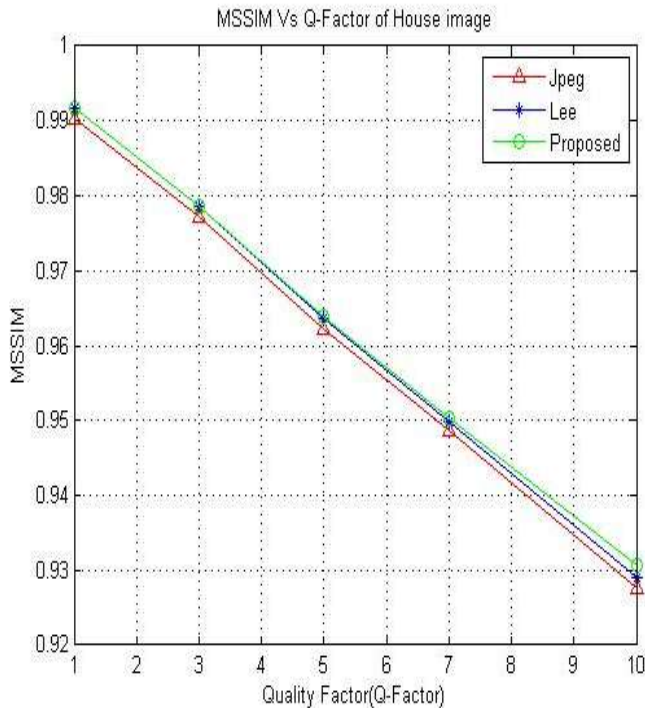


Figure 10: Relationships between MSSIM and Quality Factor of House Image

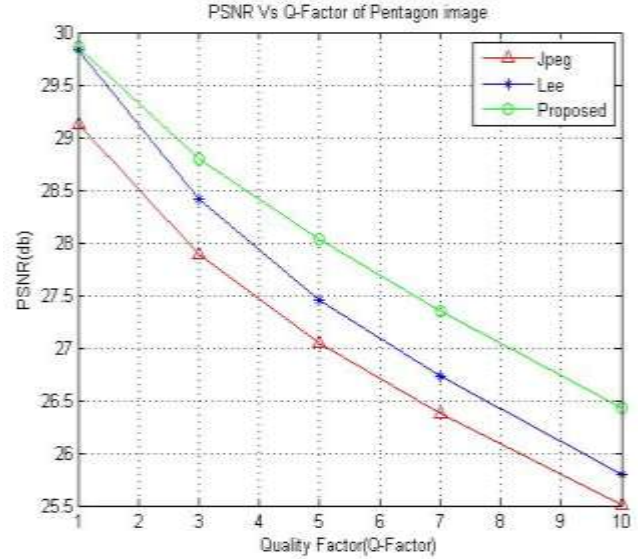


Figure 11: Relationship between PSNR and Quality Factor of Pentagon Image

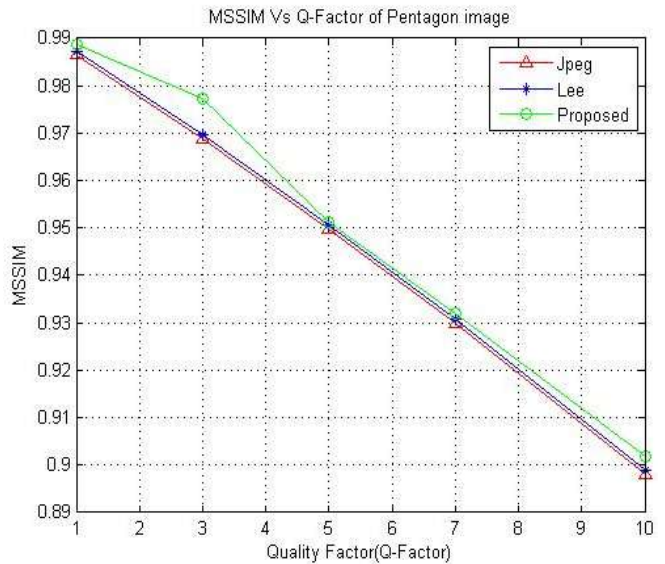


Figure 12: Relationships between MSSIM and Quality Factor of Pentagon Image

4. CONCLUSION

In this work, a new blocking-effect-reducing algorithm was proposed to improve the quality of the JPEG decompressed image. To reduce the blocking effects without degradation of the image details, the proposed algorithm uses signal adaptive filtering as well as a corner outlier detection and replacement scheme. The 2-D median filter is used in filtering of the monotone area. A noticeable corner outlier is smoothed by a corner outlier detection and replacement scheme. A new blocking artifact reduction algorithm, which is also effective at very low bit rate, is

proposed. Our algorithm takes advantage of the fact that the original pixel levels in the same block are of good continuity and we use this property and the correlation between the neighboring blocks to reduce the discontinuity of the pixels across the boundaries where blockiness appear. Our algorithm can highly preserve the high frequency components while smoothing out the boundary discontinuity. Simulation results show that the proposed algorithm significantly reduces the blocking artifacts in both the objective and the subjective measures. It can effectively remove blocking artifacts even at very low bit rates. The objective performance of the proposed algorithm was measured by the PSNR and MSSIM of the post-processed image. The measured PSNR of the proposed post-processing method showed an increase for various test images. However, the PSNR does not fully reflect the enhancement of the image quality in the psycho visual viewpoint. From a subjective view a visual improvement of the image quality, new indices MSSIM was used in the post-processed image from the proposed algorithm. In conclusion, the proposed post-processing algorithm effectively reduces the blocking effects, preserves actual edges, and enhances JPEG images. The amount of computation it requires is also acceptable compared to iterative blocking artifact removal techniques

5. REFERENCES

- [1] B. Ramamurthi, A. Gersho, Nonlinear space-variant post-processing of block coded images, *IEEE Trans. Acoust. Speech Signal Process.* ASSP-34(5): 1258–1268, 1986.
- [2] R.J. Hsieh and C.J. Kuo, Adaptive post-processor for block encoded images, *IEEE Trans. Circuit Syst. Video Technol.*, 5 (4): 298-304, August 1995.
- [3] Y.L. Lee, H.C. Kim, and H.W. Park, Blocking effect reduction of JPEG images by signal adaptive filtering, *IEEE Trans. on Image Processing*, 7 (2): 229-234, February 1998.
- [4] H. W. Park and Y. L. Lee, A post-processing method for reducing quantization effects in low bit-rate moving picture coding, *IEEE Trans. Circuit Syst. Video Technol.*, 9(1): 161-171, February 1999.
- [5] S.D. Kim, J. Yi, H.M. Kim, and J.B. Ra, A de-blocking filter with two separate modes in block based video coding, *IEEE Trans. Circuit Syst. Video Technol.*, 9 (1): 156-160, February 1999.
- [6] G.C. Ray and V.K. Srivastava, Design of 2D-multiple Notch Filter and Its Application in Reducing Blocking Artifact from DCT Coded Image, in Proc. 22nd Ann. *IEEE Int. Conf. Engineering in Medicine and Biology, Chicago 2000*: 2829-2833, July 2000.
- [7] V.K. Srivastava, Post-processing of DCT coded images, *PhD dissertation*, IIT Kanpur, India: Jan. 2001.
- [8] H. Paek, R.-C. Kim, S.-U. Lee, A DCT-based spatially adaptive post processing technique to reduce the blocking artifacts in transform coded images, *IEEE Trans. Circuits Syst. Video Technol.* 10: 36–41, February 2000.
- [9] S. Minami, A. Zakhor, An optimization approach for removing blocking effects in transform coding, *IEEE Trans. Circuits Syst. Video Technol.*, 5: 74–82, 1995.
- [10] G. Lakhani, N. Zhong, Derivation of prediction equations for blocking effect reduction, *IEEE Trans. Circuits Syst. Video Technol.*, 9: 415–418, 1999.
- [11] G.A. Triantafyllidis, D. Tzovaras, M.G. Strintzis, Blocking artifact detection and reduction in compressed data, *IEEE Trans. Circuits Syst. Video Technol.*, 12: 877–890, 2002.
- [12] B. Jeon and J. Jeong, Blocking artifacts reduction in image compression with block boundary discontinuity criterion, *IEEE Trans. Circuit Syst. Video Technol.*, 8 (3): 345-357, June 1998.
- [13] Shih-Chang Hsia, Jar-Ferr Yang, and Bin-Da Liu, Efficient Postprocessor for Blocky Effect Removal Based on Transform Characteristics, *IEEE Trans. Circuits Syst. Video Technol.*, 7 (6): 924-929, December 1997.
- [14] T. Chen, H. R. Wu, and B. Qiu, “Adaptive post filtering of transform coefficients for the reduction of blocking artifacts,” *IEEE Trans. Circuits Syst. Video Technol.*, 11(5): 594–602, May 2001.
- [15] Y. Luo, R.K. Ward, Removing the blocking artifacts of block-based DCT compressed images, *IEEE Trans. Image Process.* 12 (7): 838–842, July 2003.