Image Enhancement for Low Bit-rate JPEG and MPEG Coding via Postprocessing

Yung-Kai Lai, Jin Li, and C.-C. Jay Kuo

Signal and Image Processing Institute
Department of EE-Systems
University of Southern California
Los Angeles, CA90089-2564

ABSTRACT

JPEG and MPEG compression standards are based on block discrete cosine transform (BDCT) and, when the bit rate becomes low, visually annoying blocking artifacts appear in decompressed images and videos. In this work, we attempt to characterize and quantify the blocking artifact, and then propose an iterative method for its removal by using block classification and space-frequency filtering. The proposed new method is better than the POCS (projection onto convex sets) method in terms of convergence rate, visual appearance and objective mean square error measure.

Keywords: image compression, video compression, blocking artifact, JPEG, MPEG, POCS, postprocessing.

1 INTRODUCTION

Block-based transform coding techniques are widely used in image and video compression. The block discrete cosine transform (BDCT) has been the most popular one, since it has a comparable energy compaction property as the Karhunen-Loeve transform (KLT), which has the optimal performance, while allowing easy implementation and fast computation. BDCT has been adopted in JPEG and MPEG compression standards and used in many applications. When the bit rate becomes very low, the BDCT-based compression methods do result in unpleasant visual degradation. The most obvious compression artifact is the the blocking effect, which appears as discontinuities along block boundaries. This artifact is primarily due to the negligence of correlations among adjacent blocks by a coarse quantization of DCT coefficients independently in these blocks. It is therefore important to analyze the formation of such an artifact, exploit the knowledge of human visual system for its quantification and study some techniques to remove it.

Many approaches have been proposed to remove the blocking artifact. These artifact removal algorithms are often implemented as a postprocessing unit, where the codec itself remains unchanged. With the same transmission/storage bandwidth, the algorithms improve both the visual quality and the mean square error (MSE). One interesting class of methods is based on the idea of iterative projection onto convex sets (POCS). They are very attractive since the visual performance of the decompressed image can be gradually improved with a moderate computational complexity. One of the pioneering POCS work was carried out by Rosenholtz and Zakhor [1], where a decoded image is projected alternatively between two convex sets: one is specified by the quantization bin and the other is resulted from the continuity assumption along block boundaries. Reeves and Eddins [2] proved that the POCS method provides the solution to a constrained minimization problem. Yang *et al.* [3] proposed two regularized approaches to reduce the blocking artifact, one of which can be viewed as a rigorous version of the POCS. Their work has been generalized to the enhancement of MPEG coded video by Yuen *et al.* [4].

A key issue in POCS is the implementation of the smoothing operation, which is usually done by convolving a blocky image with a lowpass filter. The size of the region involved in lowpass filtering is often difficult to adjust. On one hand, if we apply the lowpass filtering only to a few boundary pixels, the convergence rate of POCS is slow, and the blocking effect is still visible after many iterations. On the other hand, if we apply the lowpass filtering further into the interior region of blocks (i.e. a larger size filter), it will smooth edges and textures inside the block. The resulting image looks blurry and the corresponding MSE increases. Even though projecting the filtered image back to the original quantization bin prevents oversmoothing, the blurry appearance along edges and within textures still appears. To overcome these difficulties, two new techniques are adopted in the proposed new postprocessing algorithm. First, a classified smoothing scheme is used to discriminate smooth regions from edge and texture regions. Second, a lowpass filtering is performed in both the frequency and the spatial domains to speed up the convergence rate.

This paper is organized as follows. We give an overview of major visual artifacts of image/video compression and discuss the cause of the blocking artifact in Section 2. Then, a metric is developed to measure the extent of blocking artifact in Section 3. An iterative postprocessing method is proposed to exploit the correlation among adjacent blocks in Section 4. This postprocessing technique is applied to the JPEG and MPEG compressed image/video in Section 5. Experimental results are shown in Section 6 to demonstrate the performance of our new algorithm in terms of convergence rate, MSE, visual appearance, and blockiness measure.

2 CHARACTERIZATION OF COMPRESSION ARTIFACTS

2.1 OVERVIEW

Even though the commonly used pixel-by-pixel error measures such as MSE is simple to calculate, they often do not correlate well with human perception. Since compressed images and videos have to be viewed by human being in practice, it is important to understand the sources of different image and video compression artifacts and define proper measures for their quantification. Several commonly encountered compression artifacts are reviewed in this section.

A. Blocking Effect

The blocking effect is observed for all coding techniques which involve block partitioning and appears to be the major visual defect of coded images. The DCT-based compression (JPEG), vector quantization (VQ), block truncation coding (BTC) and fractal-based compression methods all share the same coding artifact. It is also one of the major visual degradations of the MPEG standard. The blocking effect is due to surface discontinuities along block boundaries. The artifact is observable as vertical and horizontal false edges periodically appearing in the image, especially in the smooth region.

B. Overall Smoothness

The overall smoothness may be the most common picture degradation. It is one of the major criticisms of the conventional TV standards (NTSC, PAL or SECAM), which lead towards the rapid development of digital TV. However, it still occurs for all digital coding techniques at a low bit rate. It appears in different forms such as the edge smoothness due to the loss of high frequency components, the texture and color blur due to the loss of resolution and the temporal blur of moving objects due to poor motion compensation. Although the segmentation based coding techniques [8] claim to preserve the major edge components in the image, they often smooth out smaller edge components.

C. Ringing Effect

The ringing effect is another notorious visual distortion which is observable as periodic pseudo-edges around original shape edges for the DCT-compressed and subband/wavelet compressed image/video. It is also visible on the textured region of the compressed images, appearing as a different texture pattern. The ringing effect is resulted from improper truncation of high frequency components, also known as the Gibbs effect.

D. Texture Deviation

Another type of distortion is called texture deviation, which is caused by loss of fidelity in mid-frequency components, and appears as granular noise or the dirty window effect. Human eyes are less sensitive to texture deviation in textured areas with transform-based coding. However, in model- [9] or segmentation-based coding, texture deviation is often represented as an oversmoothing of texture patterns, which turns out to be visually annoying.

E. Geometrical Deformation

In model-based coding, objects in an image (e.g. the human face) are often obtained via image synthesis. The most disturbing effect of this compression approach is geometrical deformation, namely, the synthesis procedure may change the shape and position of some crucial features and lead to perceptual inconsistency.

2.2 CAUSE OF BLOCKING ARTIFACT

For the rest of this work, we will focus on the blocking artifact, since it is the most visible degradation in JPEG and MPEG compression standards. The blockiness in JPEG is due to the negligence of the interblock correlation, where an independent quantization of DCT coefficients in neighboring blocks is used. Even though the local variations represented by the high frequency components are not much correlated in a natural image, the global smoothness across multiple blocks represented by the low frequency components are strongly correlated. In the JPEG quantization procedure, the encoder divides the DCT coefficient by using the corresponding entry in the quantization table and encode the level indexes and the decoder de-quantize the coefficient by setting the value to the midpoint of the quantization bin. The scheme is optimal in the minimum MSE (MMSE) sense with respect to a uniform DCT coefficient distribution. Without considering the correlation between low frequency components across blocks, the quantization/dequantization step may reduce the correlation among adjacent blocks by putting low frequency DCT coefficients in different bins. For example, the two low frequency coefficients in neighboring blocks can be very close to each other in the original image, but quantized to two different quantization bins so that the continuity across the boundary is broken. The human visual system is very sensitive to the inconsistency among adjacent blocks in a smooth region caused by mismatched low frequency components, and the large differences among boundary pixels. As the bit rate is lowered, the quantization bins grow wider and the blocking effect becomes more severe. Most work devoted to DCT blocking artifact removal does exploit the above knowledge of blocking artifact formation.

3 QUANTIFICATION OF BLOCKING ARTIFACT

The traditional objective image quality measure such as PSNR is not sufficient for the quantification of visual quality of an image with the blocking effect. Since PSNR is a pixel-by-pixel distortion metric, it does not take into account the visual blockiness seen along block boundaries. Research has been done to measure the blockiness between the original and decoded images [5]. In this section, we propose a very simple evaluation metric to measure the blockiness without reference to the original image. Such a measure can be used by the decoder to identify which region in the decoded image is blocky. Also, since the blocking effect can be considered along the horizontal and vertical directions separately, we will focus our discussion on the 1D case.

The new metric is developed by considering the human visual response to the blocking effect. Not all discontinuities across block boundaries appear the same. The blocking effect tends to be more visible in smooth regions, where consistent discontinuities across block boundaries line up to form horizontal or vertical edges and these edges can be easily detected by human eyes. Some researchers [6], [7] characterized the blocking effect as the slope discontinuity across the block boundaries. In contrast, the blocky effect is less visible in the edge or texture region since there exist a lot of discontinuities already. We use the variance of a local region to characterize the background. The variance of a smooth block is very small while the variance of a textured block is large.

Motivated by the above discussion, our blockiness metric is described below. We denote the pixels in the two adjacent 1-D blocks (consisting of $2N$ pixel width per block) by

$$(p_1, ..., p_N, p_{N+1}, ..., p_{2N}) \text{ and } (p_{2N+1}, ..., p_{3N}, p_{3N+1}, ..., p_{4N}).$$

Note that $N = 4$ in the JPEG compression standard. As shown in Fig. 1, we use P_L and P_R to represent the sets of half-block pixels to the left and right of the block boundary (denoted by the vertical line), respectively. That is,

$$P_L \triangleq \{p_{N+1}, ..., p_{2N}\}, \text{ and } P_R \triangleq \{p_{2N+1}, ..., p_{3N}\}.$$

The difference of the block boundary can be calculated by:

$$d = p_{2N+1} - p_{2N}. \tag{1}$$

We use l_L and l_R to denote the first-order regression lines of points in P_L and P_R with slopes m_L and m_R, respectively.

Furthermore, we would like to separate the smooth (low variance) regions from the textured (high variance) regions. A block edge belongs to the set S of smooth regions, if

$$\text{Var}[P_L] \le T_s \text{ or } \text{Var}[P_R] \le T_s,$$

where $\text{Var}[\cdot]$ is the variance of the points with respect to the regression line and T_s is a certain threshold. If there is a smooth edge across the block boundary, there should be a natural transition in the slopes. That is, by viewing d defined in 1 as a slope over the 1-pixel width, the value of d should lie between the two neighboring slopes m_L and m_R. Thus, the individual boundary discontinuity D in homogeneous regions is defined as the mismatch of slopes

$$D^2 = \left| d - \frac{1}{2}(m_L + m_R) \right|^2$$

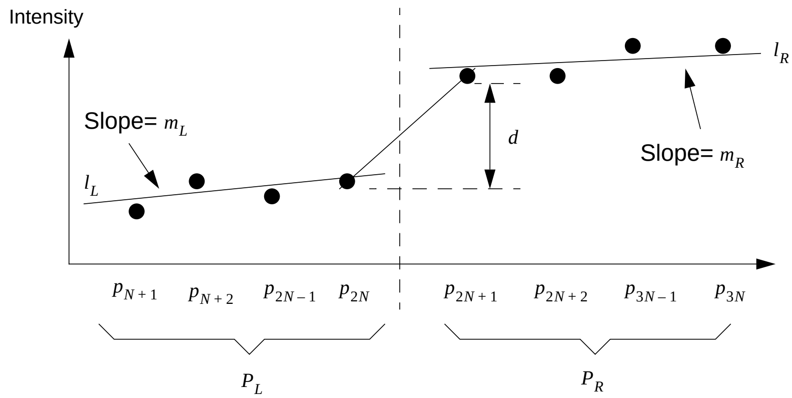


Figure 1: Blockiness measurement parameters

Finally, our new blocking artifact metric is defined as the squared sum of of individual boundary discontinuity in the smooth region set $\mathcal{S}$:

$$B = \sum_{P_L, P_R \subset \mathcal{S}} D^2. \tag{2}$$

Even though this new metric provides a new way to examine the compression artifact, the result is still preliminary. We are still actively working on this topic, and would like to report better results in the near future. For the rest of the paper, we still adopt the traditional quality measure such as MSE and PSNR for our discussion on the postprocessing technique.

4 NEW POSTPROCESSING TECHNIQUE FOR BLOCKING ARTIFACT REMOVAL

We illustrate our algorithm with a blockdiagram as shown in Fig. 2. The function of each block is detailed below.

We would like to discuss first the convergence behavior of the POCS approach proposed by Rosenholtz and Zakhor in [1] (denoted by POCS-RZ). The MSE is lowered after the first iteration, but goes up later and converges to a higher MSE than the raw decompressed image. On one hand, this could be justified by arguing that the MSE is not a good quality measure. On the other hand, one can find that edges and textures were improperly oversmoothed after a careful examination. The lowpass filtering in the beginning does reduce quantization noise around the block boundaries and lowers the MSE, but successive filtering blurs the texture and edge regions and reduce the fidelity. In other words, the MSE gained by filtering in the smooth regions is offset by the MSE loss in edge/texture oversmoothing. In POCS-RZ, a quantization bin constraint was introduced to bring the modified DCT coefficient back to its original quantization interval with an objective to prevent the frequency components from excessively amplified or attenuated, However, as the bit rate becomes lower, the quantization bin grows larger so that there is a larger room for each DCT coefficient to settle. Then, the quantization bin constraint is not sufficient to overcome the effect of oversmoothness. Additional constraints should be incorporated to improve the performance.

In our new scheme, the spatial filtering is performed on the entire image only at the first iteration. To avoid unnecessary edge/texture oversmoothing, classification is used to separate smooth blocks from others after the first iteration. For a block of size $M \times M$, we adopt the following simple criterion for

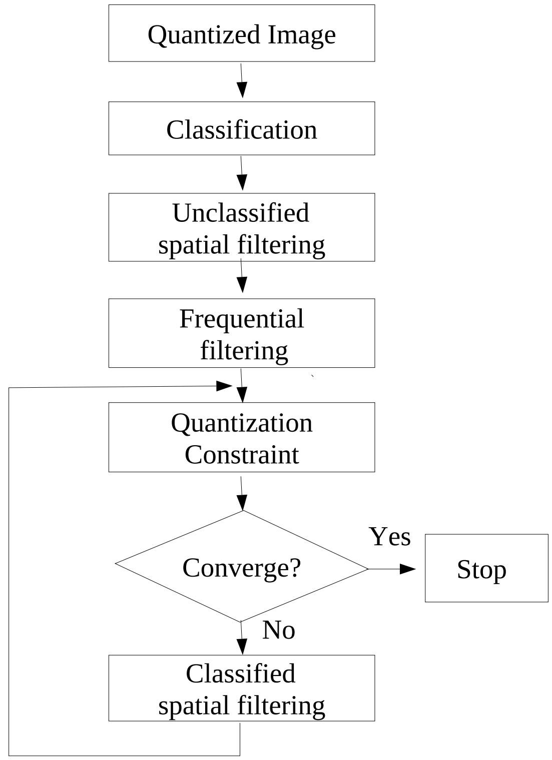


Figure 2: Block diagram of the proposed algorithm.

classification:

$$\sum_{i=k}^{M^2-1} C_{Q,i} \leq T_f, \tag{3}$$

where $C_{Q,i}$ are the quantized DCT coefficients of the block, k is a certain integer between 0 and M^2-1 and T_f is a threshold value. The parameters k and T_f can be adjusted in the experiment. The left-hand side of (3) gives the sum of quantized high frequency (from kth to (M^2-1)th) coefficients within a block. If it is smalled than threshold T_f, we classify the block as a smooth block, otherwise an edge/texture block. Since the second iteration, we apply lowpass filtering only to smooth blocks. By doing so, we are able to preserve the sharpness of the edges and keep MSE decreasing after the first iteration.

Our second improvement over POCS is in the speed up of the convergence rate. In conventional POCS, lowpass filtering is only applied to the boundary pixels and the amount of energy that redistributes among various frequency bands is limited. The convergence rate is relatively slow (about 20 iterations were needed for POCS-RZ [1]), which could be an problem for real-time applications. As discussed in Section 2.2, since the DC and low AC frequency coefficients are strongly correlated across blocks in smooth regions, we can interpolate these coefficients from adjacent blocks for a quick and better estimate confined to the original quantization bin. We refer the smoothing operation in the frequency domain as **frequency filtering**. In terms of mathematics, the frequency filtering can be written as

$$C_j^{(i+1)}(x,y) = \sum_{(k,l)\subset\mathcal{N}} C_j^{(i)}(x+k, y+l) h_f(k,l), \quad j = 0, 1, \ldots, n-1 \tag{4}$$

where $C_j^{(i)}(x,y)$ denotes the jth low-frequency coefficient in block (x,y) at the i-th iteration, n is the number of low-frequency coefficients on which we perform frequency filtering, and $\mathcal{N}$ denotes a neighborhood region.

According to the experience gaind from extensive experiments, we choose to perform frequency filtering only in smooth regions and at the first iteration as shown in Fig. 2. Although the operation is performed once, this scheme readjusts the values of low frequency components directly in the frequency domain so that the convergence rate is significantly accelerated. For texture/edge regions which do not have little across-block correlation, successive spatial filtering propagates low frequency components outward at a slower pace and affect gradually neighboring blocks.

5 APPLICATION OF THE PROPOSED METHOD TO MPEG VIDEO CODING

In this section, we discuss briefly the application of our new scheme to MPEG-I decoded video. The blocking artifact in MPEG-I is more complicated and, therefore, difficult to analyze. Most results are very preliminary.

In addition to the intraframe-coded I frames, MPEG also has predictive-coded P/B frames which are encoded by using motion compensation. Although all coding in MPEG are still block-based, block boundary discontinuities in P- and B- frames can shift due to motion compensation, and as a result the blocking artifact is not restricted to a fixed grid structure as in JPEG. As the depth of repeated motion compensation references increases, discontinuities appear virtually anywhere in the blocks. Thus, for frames near the end of Group Of Pictures (GOP), we see a distortion which is more like the granular noise than the vertical or horizontal edges observed in JPEG.

Compression of motion compensated residuals (or the displacement frame difference) smoothes some of the blockiness in the predicted frame. However, they are also scalar-quantized. So far, the net effect is that the blocking artifact removal method does not work well for P and B frames. Since the POCS method only applies the filtering operation to block boundaries, it is not sufficient to remove the distortion propagated due to motion. The major difficult of our proposed method is classification. Since only residuals are quantized, the coefficients given by Equation (3) are not immediately available. One approach is to add the residual back to the predicted block and recalculate the DCT coefficients. However, this approach increases the computational complexity.

In the experiment for MPEG decoded video postprocessing, we treat the intra-coded I blocks exactly the same as in the JPEG case. For predict coded P/B blocks, we consider both the motion compensated prediction and the residual. We assume the encoder to be an ordinary MPEG-I without postsmoothing in the encoder loop. In our MPEG-I decoder with postprocessing, the motion compensated prediction is based upon frames already postprocessed (i.e. smoothed). Even though we do not have the exact residuals for the B- and P-blocks, we exploit the fact that edge/textures block are very unlikely to use a smooth blocks as their references and vice versa in motion compensation. Therefore, we can perform lowpass filtering on the smooth block since the other smooth blocks refer to it will also be smoothed. We keep the edge/texture block as is since they may be referred by later edge/texture blocks. Since the referenced frame is also classified in the artifact removal process, no additional computational cost is needed.

6 EXPERIMENTAL RESULTS

The proposed method is first applied to JPEG compressed images, where the Lena image is chosen to be the test image. The image is of size 512×512 with 256 gray scales per pixel. It is divided into blocks of size 8×8 for JPEG compression and the quantization table used is shown in Table 1. For the classification criterion in (3), we choose parameters

$$k = 1 \text{ and } T_f = 1.$$

The 1-D spatial filter used was

$$h_s(-1) = h_s(1) = 1/6, h_s(0) = 2/3.$$

In (4), the number of frequency filtering coefficients n is set to 3, and the neighborhood $\mathcal{N}$ is the set of all eight blocks surrounding the center block. The frequency filter used was

$$\begin{aligned} h_f(-1,-1) &= h_f(-1,1) = h_f(1,-1) \\ &= h_f(1,1) = 0.0625 \\ h_f(-1,0) &= h_f(1,0) = h_f(0,-1) \\ &= h_f(0,1) = 0.1250 \\ h_f(1,1) &= 0.2500 \end{aligned}$$

80	55	50	80	120	200	255	255
60	60	70	95	130	255	255	255
70	65	80	120	200	255	255	255
70	85	110	145	255	255	255	255
90	110	185	255	255	255	255	255
120	175	255	255	255	255	255	255
245	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255

Table 1: Quantization table

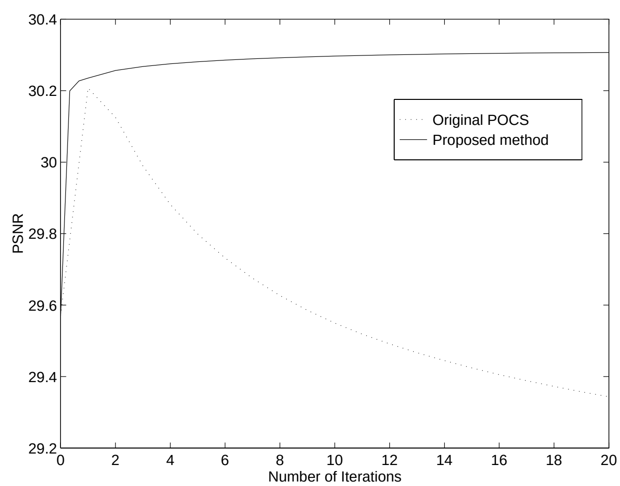
A portion (256×256) of the decoded image with a bit rate equal to 0.25bpp is shown in Fig. 3 (a). The results obtained by using POCS-RZ [1] and our new algorithm after 4 iterations are shown in Fig. 3 (b) and (c), respectively. Both algorithms use the same spatial filter for fair comparison. Our algorithm has a better visual appearance in smooth regions due to frequency filtering. The sharpness of the edges and textures is also better preserved in our algorithm. A dramatic improvement can be observed on the shoulder and right-hand-side background, where the blocking artifacts are most prominent and annoying. The reconstructed PSNR versus the number of iterations is plotted in Fig. 3 (d). Our algorithm converges faster and has a higher PSNR than POCS-RZ. Although only applied once in the smooth region, the frequency filtering brings the image quickly to the final converged state. Note also that POCS-RZ actually has a reconstructed PSNR below the original, which partly reflects the fact that the edges and textures are oversmoothed and the restored image is brought away from the original image in these regions. This observation can be confirmed by examining the restored image visually.

We also applied the new scheme to the MPEG-decoded SUSIE sequence. The post-processing parameters are the same as before. The PSNR values for I and P, B frames after 4 iterations are shown in Fig. 4. We see from the figure that the new method is able to increase the PSNR values for the I-frame and the first several B- and P-frames. The PSNR values drop when reaching the end of the group of pictures (GOP). Our blocking artifact removal algorithm is still not able to handle motion compensated predicted sequences properly since the boundary discontinuities inside blocks may shift. However, the visual appearance of the sequence still improves as shown in Fig. 5 (a) and (b), where the 7th frame of SUSIE before and after postprocessing are compared. Although the post-processed frame has a lower PSNR than the one without postprocessing, it is still more pleasant visually.

(a)

(b)

(c)



(d)

Figure 3: (a) Decompressed Lena image (0.25bpp), (b) restored Lena image with POCS (4 iterations), (c) restored Lena image with the proposed scheme (4 iterations), and (d) convergence history (PSNR versus the number of iteration).

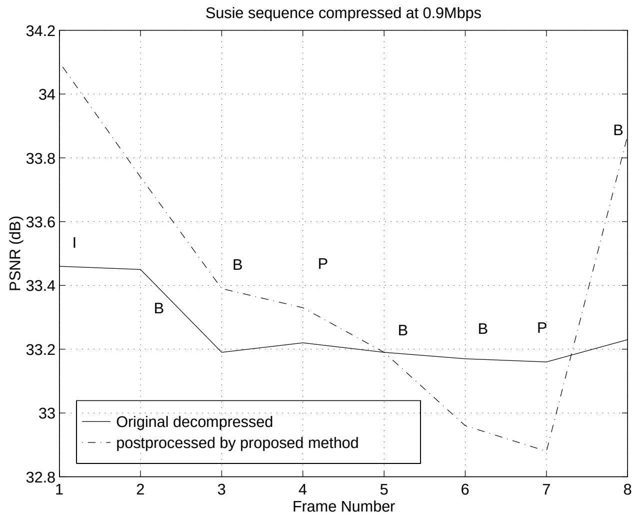


Figure 4: The PSNR of the luminance for the reconstruction of the SUSIE sequence (coded at 0.9Mbps).

(a)

(b)

Figure 5: (a) The original decompressed frame no. 7 and (b) the restored frame no. 7 by the proposed method.

7 CONCLUSION

In this paper, we examined several compression artifacts commonly found in digital compression techniques and then focused on the understanding of the blocking artifact appearing in JPEG and MPEG compression standards. Then, we proposed an iterative postprocessing approach for blocking artifact removal. A block classification scheme is developed to prevent oversmoothing in the edge and texture regions. We also proposed a frequency-domain filtering which exploits the correlation of low frequency components among adjacent blocks. It was demonstrated in experiments that our approach was very successful in improving the JPEG decoded image to achieve a better visual appearance, faster convergence rate and higher PSNR value. Some preliminary work on MPEG artifact removal was also presented and more research effort along this direction remains to be done.

8 ACKNOWLEDGMENTS

This work was supported in part by Intel and in part by the National Science Foundation Presidential Faculty Fellow (PFF) Award ASC-9350309.

9 REFERENCES

[1] R. Rosenholtz and A. Zakhor, "Iterative procedures for reduction of blocking effects in transform image coding," *IEEE Trans. Circuits and Syst. for Video Tech.*, Vol. 2, pp. 91–95, Mar. 1992.

[2] S. J. Reeves and S. L. Eddins, "Comments on 'Iterative procedures for reduction of blocking effects in transform image coding,'" *IEEE Trans. Circuits and Syst. for Video Tech.*, Vol. 3, pp. 439–440, Dec. 1993.

[3] Y. Yang, N. P. Galatsanos, and A. K. Katsaggelos, "Regularized reconstruction to reduce Blocking artifacts of block discrete cosine transform compressed images," *IEEE Trans. Circuits and Syst. for Video Tech.*, Vol. 3, pp. 421–432, Dec. 1993.

[4] M. Yuen, H. R. Wu, and K. R. Rao, "Performance evaluation of POCS loop filtering in generic MC/DPCM/DCT video coding" in *Visual Communications and Image Processing,* SPIE, vol. 2501, pp. 65–75, 1995.

[5] S. A. Karunasekera and N. G. Kingbury, "A distortion measure for blocking artifacts in images based on human visual sensitivity," *IEEE Trans. Image Processing*, Vol. 4, pp. 713–724, June 1995.

[6] S. Minami and A. Zakhor, "An optimization approach for removing blocking effects in transform coding," *IEEE Trans. Circuits and Syst. for Video Tech.*, Vol. 5, pp. 74–82, Apr. 1995.

[7] H. Paek, J.-W. Park, and S.-U. Lee, "Non-iterative post-processing technique for transform coded image sequence," *ICIP*, 1995.

[8] M. Kunt, A. Ikonmopoulos, and M. Kocher, "Recent results in high-compression image coding," *IEEE Trans. on Circuits and Systems*, No. 11, pp. 1306–1336, Nov. 1987.

[9] H. Li, A. Lundmark, and R. Forchheimer, "Image sequence coding at very low bitrates: a review," *IEEE Trans. on Image Processing*, No. 5, pp. 589–609, Sep. 1994.