Deblocking And Denoising Filter To Remove Blocking Artifacts From A Noisy Image

Meera Thapar Khanna  
Sr. Lecturer, CSE Dept. LLRIET, Moga

Jagroop Singh Sidhu  
AP, ECE Dept. DAVIET, Jalandhar

Abstract-A major drawback of block discrete cosine transform-based compressed image is the appearance of visible discontinuities along block boundaries in low bit-rate coding, which are commonly referred as blocking artifacts. Another fundamental problem in the field of image processing is image denoising. There are several ways through which noise can be introduced into an image, depending on how the image is created. In this paper, a post-processing method based on three separate filtering modes is proposed for removing these discontinuities. Then a median filter is applied to remove the noise from the image. Then evaluate the performance of the proposed method using performance metric- Peak Signal-to-Noise Ratio. Experiments shows that the proposed method gives better performance in terms of PSNR and excellent results compared with other approaches.

Key Words-Blocking effect, Denoising, Deblocking filter, DCT, PSNR.

1. Introduction

Digital images play an important role both in daily life applications such as satellite television, magnetic resonance imaging, computer tomography as well as in areas of research and technology such as geographical information systems and astronomy. Data sets collected by image sensors are generally contaminated by noise [26]. Thus, denoising is often a necessary and the first step to be taken before the images are analyzed. It is necessary to apply an efficient denoising technique to compensate for such data corruption [26]. Image denoising is an important image processing task, both as a process itself, and as a component in other processes. Many ways to denoise an image or a set of data exists. The main properties of a good image denoising model are that it will remove noise while preserving edges.

On the other hand, Blocking artifact is one of the most annoying defects in DCT-based (Discrete Cosine Transform) image compression standards (e.g. JPEG, MPEG) [1, 2], especially when image quality is compromised for low bit-rates. This phenomenon is characterized by visually noticeable changes in pixel values along block boundaries. The degradation is a result of a coarse quantization of the DCT coefficients of each image block without taking the inter-block correlations into account. It degrades the visual quality of the reconstructed images, it is desirable to be able to monitor and control the visibility of blocking effects in DCT-coded images.

In B-DCT the quantization noise is highly correlated with the characteristics of the original signals, so that different areas of the coded image suffer from distinctly different impairments [3]. In particular, the artifacts create two kinds of visual distortions (i) the artifacts create two kinds of visual distortions (i) blurring of sharp edges and changes in the texture patterns, and (ii) formation of false edges at interblock boundaries. The first kind of distortion is generally due to near elimination or improper truncation of the high- and mid-frequency DCT coefficients and is efficiently reduced by the proposed AC distribution-based restoration. The other kind is due to severe reductions in the low-frequency DCT coefficients (especially in the DC coefficient) and is tackled with the proposed adaptive spatial filtering. Hence, the two stages are acting complementarily for the removal of blocking artifacts.

There are many techniques for the distribution-based estimation of the DCT coefficients. A new method is described in this paper that exploits the activity across a region boundary to identify high and low-detail regions of the decoded image. The filtering process is then based on the region classification.

2. Background

Many approaches have been proposed in the literature aiming to remove the noise and blocking artifacts in the B-DCT image coding technique. At the encoding end, different transform schemes have suggested, such as the interleaved block transforms [4], the combined transform [5], and so on. However, none of these transform schemes conform to the existing image/video-coding standards such as JPEG or MPEG, it is difficult or impossible to integrate them with the existing standards.

The other strategy is via post-processing techniques at the decoder side. It appears to be the most practical solution. It does not require changes to existing standards, and with the rapid increase of
available computing power more sophisticated methods can be implemented. The blocking-effect is a major obstacle for using B-DCT to achieve very low bit-rate compression.

Various post-processing techniques have been suggested for the reduction of blocking artifacts, but they often introduce excessive blurring, ringing and in many cases they produce poor deblocking results at certain areas of the image. Moreover, they fail to handle a wide range of bit rates.

The MPEG4 standard [6] offers a deblocking algorithm, which operates in two modes: dc offset mode for low activity blocks and default mode. Block activity is determined according to the amount of changes in the pixels near the block boundaries. All modes apply a one-dimensional (1-D) filter in a separable way. The default mode filter uses the DCT coefficients of the pixels being processed and the dc offset mode uses a Gaussian filter. However, this is not exactly a "pure" post processing method since every quantization factor from each macro block has to be fed into the algorithm.

Another class of postprocessors using iterative image recovery methods based on the theory of projections onto convex sets (POCS) is proposed in [7, 8, and 9]. In the POCS based method, closed convex constraint sets are first defined that represents all of the available data on the original uncoded image. Then alternating projections onto these convex Combined Frequency and Spatial Domain Algorithm for the removal of Blocking Artifacts 603 sets are iteratively computed to recover the original image from the coded image. POCS are effective in eliminating blocking artifacts but less practical for real-time applications, since the iterative procedure increases the computational complexity.

Luo and Bovik [10] proposed a DCT-domain method for blind measurement of blocking artifacts, by modeling the artifacts as 2-step functions in shifted blocks. In [11], Zeng proposed a simple DCT-domain method for blocking effect reduction; applying a zero masking to the DCT coefficients of some shifted image blocks. However, the loss of edge information caused by the zero masking schemes can be noticed. Luo and Ward [12] and Singh et al. [13] gave a new approach which preserved the edge information. These methods are based on reducing the blocking artifacts in the smooth regions of the image. The correlation between the intensity values of the boundary pixels of two neighboring blocks in the DCT domain is used to distinguish between smooth and non-smooth regions.

Most post-processing methods of removing deblocking artifacts result in the filtration of images in the spatial domain [14]-[19] or the transformed domains [20] and wavelet domain [21]-[23]. Often some constraints reflecting the properties of unprocessed images are imposed on the filtering result. Since the filters commonly have some lowpass properties, this deblocking procedure is actually a smoothing operation. The major challenge is how to effectively smooth out the blocking artifacts without blurring image details.

3. Model of Image Degradation And Restoration Process

As figure 1 shows, the degradation process [2] is modeled as a degradation function that, together with an additive noise term, operates on an input image \( f(x,y) \) to produce a degraded image \( g(x,y) \). Given \( g(x,y) \), some knowledge about the degradation function \( H \), and some knowledge about the additive noise term \( \eta(x, y) \), the objective of restoration is to obtain an estimate \( f'(x, y) \) of the original image. The estimate should be as close as possible to the original input image and in general, the more we know about \( H \) and \( \eta \), the closer \( f'(x,y) \) will be to \( f(x,y) \).

\[
g(x,y) = h(x,y) * f(x,y) + \eta(x,y) \quad (1)
\]

where \( h(x, y) \) is the spatial representation of the degraded function. The symbol "*" indicates spatial convolution. Because convolution in spatial domain is equal to multiplication in frequency domain, so equation (1) can be written in an equivalent frequency domain representation.

\[
G(u,v) = H(u,v) F(u,v) + N(u,v) \quad (2)
\]

where the terms in capital are the Fourier Transforms of the corresponding terms in equation (1).

In this analysis, the only degradation present in an image is noise, so equations (1) and (2) become

\[
g(x,y) = f(x,y) + \eta(x,y)
\]
and
\[ G(u, v) = F(u, v) + N(u, v). \]

Spatial noise may be considered as random variables which are characterized by a probability density function (PDF). Some common noise types are Gaussian, Rayleigh, Erlang, Exponential, Uniform and Impulse noise.

4. Model of Blocking Artifacts

From the earlier discussion, it has been cleared that the blocking effect normally occurs at the 8×8 block boundary, and is visualized as a false edge when the compression ratio is high [24]. It can be efficiently suppressed by a smoothing procedure. According to the characteristics of a local region to the blocking area, the blocking effect becomes more visible. In designing a deblocking filter, observations of the reconstructed image are useful in formulating the appropriate characteristics of a filtering procedure.

Let \( b_1 \) and \( b_2 \) be the neighboring blocks that are horizontally adjacent to each other. Figure 2 shows intermediate block \( b \) that includes the right part of \( b_1 \) and left part of \( b_2 \).

5. Proposed Measurement System

In this analysis, a noise pattern like salt & pepper noise is superimposed upon the image and then the proposed method for deblocking and a denoising filter is applied to the image for reconstruction. If any blocking artifact is introduced between \( b_1 \) and \( b_2 \), the pixel values in \( b \) will be abruptly changed. By modeling the abrupt change in \( b \), the blocking artifacts can be measured. The proposed filter attempts to remove blockiness and noise from a degraded image.

5.1. An overview of proposed deblocking algorithm

The proposed algorithm is based on three separate modes, smooth, intermediate, and complex because a single smooth filter may not effectively reduce blocking effects with respect to both visual quality and fidelity. It appropriately classifies the local characteristics of images according to the above requirements. The extent of the blocking artifacts clarifies the type of filtering appropriate for each region. In designing a Debloking filter, observations of the reconstructed image are useful in formulating the appropriate characteristics of a filtering procedure. Based on the observation, strong filtering is applied to the flat area of block boundary, whereas weak filtering is to be applied to preserve the details in areas of high spatial or temporal activity [25].

![Figure 3. Position of filtered pixels](image)

An intermediate mode is designed to solve the problem of a too simplistic decision, and either excessive blurring or inadequate removal of the blocking effect. Flatness can be classified using a threshold and a simple subtraction from local pixel values. Then, according to the local characteristics, a Debloking filter may have two different effects on the region to be filtered.

5.2. Mode Detection

The key idea behind smoothing the blocked image is to reduce the extent of blockiness without blurring the image. It is independent of the blocking position, implying that the existence of the offset between the two flat regions is of interest instead of the discontinuity around the block boundary. It is measured by analyzing the pixels between two adjacent blocks [25]. Smothing is done by considering three neighboring pixels on either side of pixels containing the block edge. As depicted in fig.
4, x is a 1-D array of pixel values across a block boundary edge.

\[
I = \sum_{j=1}^{5} \phi(x_j - x_{j+1})
\]

(3)

Where

\[
\phi(\Delta) = \begin{cases} 
0, & \Delta \leq S \\
1, & \text{otherwise}
\end{cases}
\]

(4)

The threshold S is set to a low value so that the function \( \phi(\cdot) \) will return zero to represent an insignificant difference between neighboring pixels.

After the five values are determined, then their sum can be seen as a suitable measure of activity. The low value of I indicate a smooth region, whereas a high value indicates a region with edge detail. The detection is performed on the basis of the value of I. The value of I is compared to two thresholds, T1 and T2, to determine the appropriate filtering mode. That is, the pixel being deblocked depends upon the detection criteria. When I<T1 apply smooth mode filtering and when I>T2 then the use of complex mode filtering is specified. When I is between T1 and T2, intermediate filtering is used to improve the PSNR and visual quality. S is set to a minimal value so that I may reflect the flatness of the local image across a block boundary.

5.3. Deblocking smooth regions

In this mode, the block boundary is between two adjacent smooth 8x8 blocks. The region of pixels near the block boundary, at which the deblocking filter has updated the pixel values, must be accurately determined. First, the offset is determined from the difference between two pixels across the block boundary. It affects the strength of the blocking effect. So filtering based on offset eliminates the blocking effects in smooth regions. A total of six pixels are updated across the boundary. To prevent real edges filtering is not performed when the offset is larger than a certain value 2Q. Here Q is the quality parameter. The value of these updated pixels must be adjusted again within the grayscale range, from 0-255. Figure 5 shows how offset is calculated between two adjacent regions.

5.4. Deblocking complex regions

If the region is of high activity then strong filtering is not appropriate because it over-blurs the true edges of the image. In this region, filtering is applied only to two pixels from both the boundary regions. So it requires simple control mechanism and reduces the computational overhead.

5.5. Deblocking intermediate regions

A 3x3 low pass filter is presented as an intermediate mode filtering. The filtering for an intermediate region balances the strong filtering in a smooth region with the weak filtering in a complex region. It also reduces the computational overhead because only shifting is applied as compared to the division in other filters.

Applying this low pass filter to the pixels on the either side of the block boundary, C and D, reduces the blocking effect with minimal loss of image content. This filter is adaptive in two ways. First, only the pixels near the boundary are selected in the filtering window and their gray value is within a specified range around the gray value of the pixel to be filtered. Secondly, the threshold Th (0-30) is adjusted according to the JPEG quality parameter Q (1-100) of the block [25].
6. Results and Conclusion

PSNR - Peak Signal to Noise Ratio is used to measure the blocking artifacts in the image. PSNR is basically a logarithmic scale of the mean squared difference between two sets of values (pixel values, in this case). It is used as a general measure of image quality, but it does not specifically measure blocking artifacts. In observed literature, PSNR is used as a source-dependant artifact measure, requiring the original, uncompressed image to compare with. PSNR is defined as:

\[
PSNR = 20 \log_{10} \left( \frac{255}{RMSE} \right)
\]  

(5)

It is easily seen that this blockiness measure is not actually aware of the artifacts it is measuring - it is simply a gauge of how different the corresponding (that is, the same position) pixel values are between two images.

So PSNR is an acceptable measure, and hence the primary measure used to compare the proposed method. However, two images with completely different levels of perceived blockiness may have almost identical PSNR values.

In this experiment, the proposed algorithm depends on some predefined parameters. For the mode to be selected, the activity between block boundaries must be measured. The threshold S is set to 2 to correspond to the smooth region appropriate for intensive filtering. After the activity across block boundary is measured, two thresholds, T1 and T2 determine the appropriate deblocking mode. T1 is set to 2 and T2 is set to 3.

In order to evaluate the performance of proposed technique two 512x512 images Lena and Pentagon are coded at different bit rates using the JPEG standard and compared with ward [12]. Figure 7 show the results of two test images which are Lena and Pentagon respectively. It shows the result after applying proposed method and ward method to the compressed images after adding noise to it. It shows that the filtering of ward [12] tends to blur the image at the expense of removing the artifacts. In contrast the proposed scheme adapts to various image features successfully and removes noticeable blocking artifacts without degrading the details. In this case a median filter is applied after the deblocking algorithm. Median filter is the non-linear filter which changes the image intensity mean values. So results show that it doesn’t leave any blurring in the image.
Table 1
The RMSE values in comparison with Ref. [12] and proposed algorithm at different bit rate after adding noise Salt & Pepper

<table>
<thead>
<tr>
<th>Image</th>
<th>Bit rate</th>
<th>JPEG</th>
<th>WITH NOISE</th>
<th>Proposed</th>
<th>Ref. [12]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lena</td>
<td>0.12</td>
<td>11.8817</td>
<td>23.1469</td>
<td>10.6921</td>
<td>10.6050</td>
</tr>
<tr>
<td></td>
<td>0.14</td>
<td>10.7404</td>
<td>22.4571</td>
<td>9.6044</td>
<td>9.6969</td>
</tr>
<tr>
<td></td>
<td>0.16</td>
<td>9.4419</td>
<td>21.9722</td>
<td>8.3872</td>
<td>8.7687</td>
</tr>
<tr>
<td></td>
<td>0.20</td>
<td>8.1087</td>
<td>21.2059</td>
<td>7.2449</td>
<td>7.8666</td>
</tr>
<tr>
<td></td>
<td>0.25</td>
<td>6.6075</td>
<td>20.9905</td>
<td>6.0240</td>
<td>7.0519</td>
</tr>
<tr>
<td></td>
<td>0.35</td>
<td>4.9438</td>
<td>20.7163</td>
<td>4.8149</td>
<td>6.2162</td>
</tr>
<tr>
<td>Pentagon</td>
<td>0.10</td>
<td>16.4047</td>
<td>24.6198</td>
<td>15.2590</td>
<td>15.4112</td>
</tr>
<tr>
<td></td>
<td>0.12</td>
<td>15.3523</td>
<td>23.8999</td>
<td>14.3213</td>
<td>14.6160</td>
</tr>
<tr>
<td></td>
<td>0.15</td>
<td>14.2047</td>
<td>23.1347</td>
<td>13.2604</td>
<td>13.8503</td>
</tr>
<tr>
<td></td>
<td>0.20</td>
<td>12.9983</td>
<td>22.2374</td>
<td>12.1621</td>
<td>13.0299</td>
</tr>
<tr>
<td></td>
<td>0.28</td>
<td>11.6079</td>
<td>21.7422</td>
<td>11.0383</td>
<td>13.0299</td>
</tr>
<tr>
<td></td>
<td>0.45</td>
<td>9.8722</td>
<td>20.9328</td>
<td>9.8290</td>
<td>11.3007</td>
</tr>
</tbody>
</table>

Table 2
The PSNR values in comparison with Ref. [12] and proposed algorithm at different bit rate after adding noise Salt & Pepper

<table>
<thead>
<tr>
<th>Image</th>
<th>Bit rate</th>
<th>JPEG</th>
<th>WITH NOISE</th>
<th>Proposed</th>
<th>Ref. [12]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lena</td>
<td>0.12</td>
<td>26.6333</td>
<td>20.8410</td>
<td>27.5496</td>
<td>27.6206</td>
</tr>
<tr>
<td></td>
<td>0.14</td>
<td>27.5104</td>
<td>21.1037</td>
<td>28.4814</td>
<td>28.3981</td>
</tr>
<tr>
<td></td>
<td>0.16</td>
<td>28.6297</td>
<td>21.2933</td>
<td>29.6585</td>
<td>29.2721</td>
</tr>
<tr>
<td></td>
<td>0.20</td>
<td>29.9518</td>
<td>21.6017</td>
<td>30.9302</td>
<td>30.2040</td>
</tr>
<tr>
<td></td>
<td>0.25</td>
<td>31.7300</td>
<td>21.6904</td>
<td>32.5331</td>
<td>31.1647</td>
</tr>
<tr>
<td></td>
<td>0.35</td>
<td>34.2496</td>
<td>21.8799</td>
<td>34.4467</td>
<td>32.2603</td>
</tr>
<tr>
<td></td>
<td>0.10</td>
<td>23.8314</td>
<td>20.3051</td>
<td>24.4603</td>
<td>24.3741</td>
</tr>
<tr>
<td></td>
<td>0.12</td>
<td>24.4073</td>
<td>20.5629</td>
<td>25.0111</td>
<td>24.8343</td>
</tr>
<tr>
<td></td>
<td>0.15</td>
<td>25.0822</td>
<td>20.8455</td>
<td>25.6797</td>
<td>25.3016</td>
</tr>
<tr>
<td></td>
<td>0.20</td>
<td>25.8531</td>
<td>21.1891</td>
<td>26.4306</td>
<td>25.8320</td>
</tr>
<tr>
<td></td>
<td>0.28</td>
<td>26.8358</td>
<td>21.3847</td>
<td>27.2727</td>
<td>25.8320</td>
</tr>
<tr>
<td></td>
<td>0.45</td>
<td>28.2425</td>
<td>21.7142</td>
<td>28.2858</td>
<td>27.0687</td>
</tr>
</tbody>
</table>

Figure 8 is the graphical representation of bits per pixel versus PSNR and RMSE respectively in JPEG, proposed, and Ward [12] images. The result of the proposed algorithm demonstrates not only alleviating the blocking artifacts but also removing the edges near the real edges. Resulting bit rates and RMSE and PSNR values are listed in Table 1 and Table 2 respectively. At the same bit rates, all of the Deblocking filter images PSNR values are absolutely higher than these of images compressed by JPEG.

7. Conclusion

This paper proposed a post-processing algorithm for reducing blocking artifact in transforming coded images. The proposed algorithm is based on the 1-D filtering of block boundaries. Results show that using three filtering modes effective deblocking is achieved. The proposed technique provides satisfying image quality across a wide variety of images. To
demonstrate the performance of the proposed algorithm PSNR has been used. It is found that there is a significant improvement in the perceptual quality of the JPEG compressed imagers after removal of blocking artifact by the proposed method. Due to its low computational cost, the technique can be integrated into real-time image/video applications as a method for online quality monitoring and control.

ACKNOWLEDGMENT

Meera Thapar Khanna sincerely thanks Mr. Jagroop Singh Sidhu, Assistant Professor D.A.V. Institute of Engineering and Technology, Jalandhar for his guidance and support.

REFERENCES