A Novel Deblocking Algorithm Using Edge Flow-Directed Filter and Curvelet Transform

ZhiMing Zhang, JeongHoon Park, YongJe Kim
Digital Media R&D Center, Samsung Electronics

Abstract

In this paper, a new post-processing approach based on “edge flow” and curvelet transform is proposed for the suppression blocking artifacts in block discrete cosine transform (BDCT) compressed images. Firstly, by exploiting the edge flow correlations, edge information in the compressed images is extracted and protected, while blocky noise in the smooth background regions is smoothed out by edge flow-directed filter in the wavelet domain. Then the curvelet transform coefficients in different subbands are filtered with adaptive thresholds that are obtained according to the edge flow boundary map. The advantage of the new method is that it retains sharp features in the images and compared with other wavelet-based methods, it is capable of achieving the higher peak signal-to-noise ratio (PSNR) improvement and giving visually very pleasing images as well.

1. Introduction

In the past, various post-processing techniques have been proposed for removing the blocking artifacts to improve the visual quality of BDCT compressed images in the decoder without increasing the bit rate. Some efficient deblocking methods using wavelet representation have been proposed [3][4][5]. But these methods may not be suitable for images containing a large portion of texture, such as the “Barbara” image. Ref. [10] proposed an adaptive method for determining the soft threshold and different threshold values and strategies were used for wavelet coefficients at different high frequency subbands, but had less PSNR improvements compared to projection onto convex sets (POCS)-based method [7][9]. While the Gaussian assumption in [1] is not valid for the quantization noise, it is known that such noise is structured around block boundaries. So the deblocking problem simply boils down to smoothing out discontinuity across block boundaries only in smooth background regions but protecting edges that might occur at block boundaries [5]. Given the edge and texture information, which are provided by the following edge flow method, curvelet representations are well suited for deblocking of BDCT compressed images.

2. Curvelet and edge flow

The basic idea of curvelet transform was developed to get such a kind of expansions that only a few coefficients are represented either for the smooth parts or the edge parts, thus the balance between parsimony and accuracy will be much more favorable and a lower mean squared error (MSE) results [1]. Although such kind of algorithm cannot be directly used in the deblocking process because of the different noise distribution, it can be circumvented with wavelet-based denoising and filters to protect the edges information around block boundaries [4][5]. While these methods only use a simple fixed coefficients one dimensional (1-D) filter in the horizontal direction, we propose a curvelet-based scheme with a texture adaptive 2-D filter where its weights are controlled by the edge flow [2] directions, so that edge pixels will not be mixed and pixels in the smoothing area will get more weights from the nearby zones with similar texture.
using intensity edge flow method in [2]. To simplify the computation of the edge flow, the angle of edge flow vector $\vec{e}(x,y)$ is quantized into steps of $\pi/4$. Further, based on $\vec{e}(x,y)$, the edge flow boundary map $E(x,y)$ can be obtained, where $E(x,y) = 1$ means $f(x,y)$ is an edge pixel, and $E(x,y)=0$ means $f(x,y)$ belongs to a background region.

### 3.2. Dyadic wavelet transform

The $J$-level overcomplete 2-D wavelet representation of $f(x,y)$, $W^j_x f$ and $W^j_y f$, is obtained the same as in [5]. Here, with the edge flow boundary map $E(x,y)$ at hand, we define $C = \{(x,y) | E(x,y)=1\}$ and $N_c$ as the number of non-zero elements in $E(x,y)$. Then the threshold $T^j$ for scale $j$ is defined as:

$$T^j = \left[ \sum_{x,y \in C} \left( |W^j_x f|^2 + |W^j_y f|^2 \right) \right]^{1/(N_c \times N \times N)} \quad (1)$$

### 3.3. Edge flow-directed filter

Taking a $J$-scale discrete dyadic wavelet transform of a BDCT compressed image, the blockiness in $f(x,y)$ has a strong showing in the first scale highpass wavelet images $W^j_x f$ and $W^j_y f$. And following filtering rules are used:

1) If the central point of the $3 \times 3$ filter kernel is on the edge flow boundary, i.e. corresponding $E(x,y) = 1$, no 2-D filtering operation is performed. The processed pixel represents an edge pixel that means the existence of the statistically different regions and so any further smoothing will blur the corresponding edges.

2) If any edge point of the edge flow boundary is not included in the $3 \times 3$ filter window, following average filtering is performed. Suppose the edge flow matrix covered by filter kernel is $\vec{e}(x,y)$, $0 \leq x, y \leq 2$ and $\vec{e}(1,1)$ is the center point of the matrix. Then the actual weights in the filter kernel is calculated as:

$$Kernel(x,y) = K1(x,y) \cdot \left| \vec{e}(x,y) \right| \cdot \left| \vec{e}(1,1) \right| \cdot \left( 1 + \cos \left( angle(\vec{e}(x,y)) - angle(\vec{e}(1,1)) \right) \right) \quad (2)$$

where $\text{angle}(\vec{e}(x,y))$ is the angle of vector $\vec{e}(x,y)$ and $\left| \right|$ means module.

3) If any edge pixels are in the $3 \times 3$ filter window, except on the central point, the weighted average filtering is performed by using the same equation as (2) but kernel matrix $K1$ is replaced as $K2$:

$$K1 = \begin{bmatrix} 0.5 & 1 & 0.5 \\ 1 & 2 & 1 \\ 0.5 & 1 & 0.5 \end{bmatrix}, \quad K2 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Just as noted the weights of filter kernel are changed according to the edge flow direction of the edge pixel in the filter window to protect image detail from corruption and to maintain details continuity.
3.4. Curvelet thresholding

For filtered wavelet samples \(\hat{w}_{Jf}\) and \(\hat{w}_{Jf}\) and un-filtered wavelet images \(w_{Jf}\) and \(w_{Jf}\) (\(2 \leq j \leq J\)), just as [1], such \(N \times N\) wavelet images are further decomposed into smoothly overlapping blocks of sidelength \(B\) pixels in such a way that overlap between two vertically adjacent blocks is a rectangular array of size \(B\) by \(B/2\). For an \(N \times N\) image, we count \(2N/B\) such blocks in each direction and digital ridgelet transform are done on each block. Then we use the following hard-thresholding rule for estimating the unknown curvelet coefficients. Let \(d\) be the noisy curvelet coefficients, the prediction will be:

\[
\hat{d} = d \quad \text{if } |d| \geq T
\]

\[
\hat{d} = 0 \quad \text{if } |d| < T
\]

The performance of our deblocking algorithm depends on the threshold \(T\) in (3)(4), which should be set as the variance of BDCT quantization noise. Unfortunately the correct value of \(T\) is not known a priori. We estimate it by following experimental equation:

\[
T = \lambda d^* - \lambda d^*T
\]

where \(\lambda^*_d\) is related to quantization matrix and \(\lambda^*_d\) is the coefficient which is mainly determined by the size of blocking ridgelet transform.

3.5. Reconstruction

The denoised image can be reconstructed by cascading the following operation from scale \(J\) up to the original scale and normalized within the range \([0-255]\).

4. Experimental results

Extensive deblocking experiments using images with different characteristics have been performed. For illustration, we show the deblocking results for the \(512 \times 512\) images “Lena” and “Barbara”. Three JPEG quantization tables in [4] are used to compress images. “Lena” contains mainly smooth region while “Barbara” contains regular texture pattern on her garment. Set maximum scale \(J = 3\). \((\lambda^*_d, \lambda^*_d, \lambda^*_d)\) is \((0.5,0.5,0.25)\) for quantization table \(Q1\), \((0.5,0.5,0.5)\) for \(Q2\) and \((1.0,0.75,0.75)\) for \(Q3\). \((\lambda^*_d, \lambda^*_d)\) is \((256,32,32)\) for all the quantization tables. And block size for curvelet transform is \(32 \times 32\) for scale 1 and \(64 \times 64\) for scale 2 and 3. Figure 3 shows, respectively, the original “Lena” and “Barbara”, BDCT compressed “Lena” and “Barbara” (using \(Q2\)) with blocking artifacts, and the deblocked “Lena” and “Barbara” using the proposed algorithm. The BDCT compressed “Lena” and “Barbara” in Figure 3 has PSNR of 30.091 dB and 25.591 dB, respectively, whereas the deblocked image has a PSNR of 31.240 dB and 26.541 dB.

We compare the performance of our algorithm with Xiong’s wavelet deblocking (WDx) algorithm [5], Alan’s overcomplete wavelet deblocking (WDa) algorithm [4], Hsung’s wavelet singularity detection (WSD) algorithm [6], MPEG-4 VM postfiltering (MPEG4) algorithm [7], Pack’s POCS (POCSp) algorithm [9] and Yang’s spatially adaptive POCS (POCSy) algorithm [8]. The PSNR results for the different algorithms are tabulated in Table 1. It can be seen that our algorithm outperforms other algorithms in PSNR. While the algorithm in [4] got good results in “Lena”, it is unable to perform well for the “Barbara” image that contains a large portion of texture.

Table 1. Deblocking results for the images

<table>
<thead>
<tr>
<th>Test Image</th>
<th>Lena</th>
<th>Barbara</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quant. Q1</td>
<td>30.702</td>
<td>28.679</td>
</tr>
<tr>
<td>Q2</td>
<td>30.091</td>
<td>29.095</td>
</tr>
<tr>
<td>Q3</td>
<td>27.382</td>
<td>26.679</td>
</tr>
<tr>
<td>WDa</td>
<td>31.602</td>
<td>29.679</td>
</tr>
<tr>
<td>WDx</td>
<td>31.620</td>
<td>29.591</td>
</tr>
<tr>
<td>WSD</td>
<td>31.299</td>
<td>29.095</td>
</tr>
<tr>
<td>MPEG4</td>
<td>31.211</td>
<td>29.095</td>
</tr>
<tr>
<td>POCSp</td>
<td>31.629</td>
<td>28.679</td>
</tr>
<tr>
<td>POCSy</td>
<td>31.314</td>
<td>29.095</td>
</tr>
<tr>
<td>Ours</td>
<td>31.785</td>
<td>29.095</td>
</tr>
</tbody>
</table>

5. Conclusion

In this paper, we proposed an edge flow-directed filter and curvelet-based deblocking algorithm for blocking artifacts suppression. With the edge flow-directed filter, the proposed algorithm can smooth out blocking artifacts while preserving edges and textural information. And the coefficients in curvelet domain are efficiently filtered with adaptive threshold that is obtained in relation to the edge flow boundary map. Comparative study has shown that our algorithm can retain the sharp features in the image after denoising the block artifacts and has better post-processing gains.

6. References


Figure 3. Original, BDCT compressed, and deblocked image of “Lena” (a, b, c) and "Babara" (d, e, f)