A deblocking Filter with Two Separate Modes in Block-Based Video Coding

Authors: S.D. Kim, H.M. Kim, and J.B. Ra
Publisher: IEEE Transactions on Circuit and Systems for Video Technology, 1999

Key laboratory of Intelligent perception and image understanding
content

1. The origin of problem
2. The solution of problem
3. Proposed deblocking filtering
The origin of problem

Blocking artifacts

Procedure of coding
1. Divide the image into 8*8 pixel blocks
2. Perform DCT transforms in each block, respectively. In the transforming, measurement error (or offset) will be produced in each block. Unfortunately, the errors in each block are not equal.

From the procedure, we can imagine that discontinuous jump and change will appear in the Block boundary of reconstructed image. That is blocking artifacts.
The solution of problem

- nonblock-based coding
  - The lapped orthogonal transform
  - Embedded zero-tree wavelet coding

However, it is not a good method when we consider performance, complexity, compatibility, market requirements.

So, we need an image post-processing technology, which does not change the coding process, and can keep the code rate.

Therefore, as a post-processing method, deblocking filtering is regarded as important due to improvement of visual quality in low bit-rate video coding.
Proposed deblocking filter

In order to find an efficient deblocking filter, we investigate smoothing features in a video sequence in terms of the human visual system (HVS).

**Based on three major observations**

1. The HVS is more sensitive to blocking artifacts in flat regions than in complex regions.
2. Smoothing operations tend to introduce more undesirable blur in complex regions than in flat regions.
3. Because of motion compensation, blocking artifacts are propagated, and the propagated artifacts are more visible in flat regions than in complex regions.

**Hence, we need a deblocking filtering with two separate modes.**

The filtering modes for flat regions will be called a **smooth region mode**; the filtering modes for complex regions will be called a **default mode**.
Proposed deblocking filter

Mode Decision

- To select a proper mode between the smooth region mode and the default mode, local image characteristics in the region are to be examined. In the proposed scheme, we examine the flatness of the region by using the following measurement:

\[
F(u) = \sum_{i=0}^{8} f(v_i - v_{i+1}) \quad \text{where} \quad f(\|D\|) = \begin{cases} 
1, & \|D\| < T_1 \\
0, & \text{otherwise}
\end{cases}
\]

In formula, \(T_1\) is set to a small value so that may reflect the flatness of the local image across a block boundary. If \(F(v)\) has a big value, larger than a certain threshold \(T_2\), in that case, \(V\) is assigned to the smooth region mode, and strong smoothing is applied; otherwise, \(V\) is assigned to the default mode, and accurate and adaptive filtering is applied.

where \(u = [v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8]\)
Proposed deblocking filter

Filtering in the Smooth Region Mode

As we mentioned, The HVS is more sensitive to blocking artifacts in flat regions than in complex regions. So the filter for flat regions should provide a strong smoothing effect inside a block as well as on block boundaries.

The algorithm for smooth region mode follows:

\[
v'_n = \frac{1}{16} \sum_{k=-4}^{4} b_k \cdot p_{n+k}, \quad 1 \leq n \leq 8,
\]

where

\[
\begin{align*}
p_0 &= \begin{cases} v_0, & |v_1 - v_0| < QP \\ v_1, & \text{otherwise} \end{cases} \\
p_9 &= \begin{cases} v_9, & |v_9 - v_8| < QP \\ v_8, & \text{otherwise} \end{cases} \\
p_m &= \begin{cases} v_m, & 1 \leq m \leq 8 \\ p_9, & m > 8 \end{cases}
\]

and \( \{b_k : -4 \leq k \leq 4\} = \{1, 1, 2, 2, 4, 2, 2, 1, 1\} \)
Proposed deblocking filter

Filtering in the Smooth Region Mode

To prevent real edges in the filtering region from smoothing, however, filtering is not performed when the difference between the maximum value and the minimum value of is larger than a certain value, $2Q_P$. Here, $Q_P$ is the quantization parameter of the macroblock to which pixel belongs. This is because the offset related to blocking artifacts is usually a small value, and is highly related to quantization parameter.

\[ \text{where} \]
\[ \max = \text{MAX} (u) \]
\[ \min = \text{MIN} (u) \]
\[ u = [v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8] \]
Proposed deblocking filter

Filtering in the Default Mode

- In this mode, we smooth the two block boundary pixels $v_4$ and $v_5$ only, and use the four-point DCT as a frequency analysis tool to get the feature information of the pixel array. If a four point pixel array $S_1$ is located across the block boundary, four-point DCT basis vectors of $S_1$ have symmetric and anti-symmetric properties around the center of four points, or the block boundary. We define $a_{0,1}$, $a_{1,1}$, $a_{2,1}$, $a_{3,1}$ and as the four-point DCT coefficients of $S_1$. Then it is noticed that the high-frequency anti-symmetric component $a_{3,1}$ is a major factor affecting the blocking artifact. This means that the proper adjustment of $a_{3,1}$ is directly related to the reduction of block discontinuity in the spatial domain. Thus, in this mode, the magnitude of is reduced by using a scaling factor whose value is between 0 and 1.

If we describe a four-point DCT/IDCT of in matrix form, we have

$$
egin{pmatrix}
  k_0 & k_0 & k_0 \\
  k_3 & -k_3 & -k_3 \\
  -k_2 & -k_2 & k_2 \\
  -k_1 & k_1 & -k_3
\end{pmatrix}
= \begin{pmatrix}
  a_{0,1} \\
  a_{1,1} \\
  a_{2,1} \\
  a_{3,1}
\end{pmatrix}
$$

where

$$
k_0 = 0.5 \quad k_1 = \frac{1}{\sqrt{2}} \cos \frac{\rho}{8} = \frac{c_1}{c_3} = 0.6523
$$

$$
k_2 = \frac{1}{\sqrt{2}} \cos \frac{2\rho}{8} = 0.5 \quad k_3 = \frac{1}{\sqrt{2}} \cos \frac{3\rho}{8} = \frac{c_1}{c_4} = 0.2706
$$
Proposed deblocking filter

Filtering in the Default Mode

- New values of $v_4$ and $v_5$ due to the change of $a_{3,1}$ can be easily obtained without performing a full four-point IDCT. If $a_{3,1}$ increases by $\theta$, we can find that $v'_4 = v_4 - k_1 \cdot \theta$ and $v'_5 = v_5 + k_1 \cdot \theta$. We should also note that the scaling of $a_{3,1}$ can be easily achieved without a dividing operation because the scaled value $a_{3,1}$ is given as $\text{SIGN}(a_{3,1}) \times \text{MIN}(|a_{3,0}|, |a_{3,1}|, |a_{3,2}|)$.

The algorithm for Default mode follows:

$$v'_4 = v_4 - d,$$
$$v'_5 = v_5 + d,$$

where

$$d = \text{CLIP}(\frac{c_2}{c_3} \cdot \text{MIN}(a_{3,1}, 0), \frac{v_4 - v_5}{2}),$$

$$a'_{3,1} = \text{CLIP}(\frac{1}{c_3} \cdot \text{MIN}|a_{3,0}|, |a_{3,1}|, |a_{3,2}|, |a_{3,1}|, 0),$$

and $\text{CLIP}(x, p, q)$ clips $x$ to a value between $p$ and $q$. 

Horizontal block boundary $S_1$
Proposed deblocking filter

Experiment Design

1. Use C or MATLAB program for the implementation of deblocking filter

2. Use proposed deblocking filter for video test sequences of low bit-rate DCT-based compression

3. Observe the results to find whether proposed deblocking filter is effective or not.

4. Compare the proposed algorithm with other existing postprocessing methods in observation for subjective image quality and in PSNR for objective image quality.
Proposed deblocking filter

Future Schedule

Read the relevant papers and materials to understand the background knowledge and think over methods to solve the problem.

(about 30~40 days)

Programming and testing, analyse the results and make a comparation

(about 40 days)

End work. (in 7 days)
超越在于创新