Histogram-Based Image Enhancement in Quasi-Spatial Domain for Compressed Image

Chung-Ming Kuo*, Nai-Chung Yang, Jeng-Yan Wu, and Sheng-Chang Chen

Department of Information Engineering, I-Shou University
Tahsu, 840, Kaohsiung, Taiwan R.O.C
*corresponding author, e-mail: kuocm@isu.edu.tw

Abstract. Images with JPEG format using discrete cosine transform (DCT) is widely used in various network applications. Image enhancement in the compressed domain has benefits including low computational complexity and low storage space. However, the compression is achieved by block-based transform; therefore, it is restricted by local block and hard to enhance image globally. In this paper, we will present a new quasi-spatial representation to solve the limitation in compressed domain. The main contributions of proposed method are that it developed a fast quasi-spatial transform and then effectively integrates the histogram-based method to address the drawbacks of image enhancement in compressed domain such as blocking artifacts and the adjustment of enhancement factors. To compare with existing methods, the proposed method achieves not only excellent visual quality but also eliminates blocking artifacts.

Keywords: JPEG, DCT, Image enhancement, Compressed domain, Histogram-based

1. Introduction. Image enhancement is generally used to improve the visual quality of images [19, 9, 10, 5, 14, 21, 3, 7, 20, 11, 4, 13, 6, 15, 12, 17, 1]. Most developed methods enhance images in spatial domain. However, image compression such as JPEG [22] that is the most popular standard has been widely adopted in various network applications. Therefore, to develop image enhancement method in compression domain is very important. Enhancement in the compressed domain is with not only low computational complexity but also low storage space [19, 3]. It is very desirable for mobile devices such as smart phone and tablet PC. Except these advantages, enhancement in compressed domain can achieve performance improvement for specific frequency details, which cannot achieve in spatial domain methods. Therefore, enhancement in DCT domain has the potential to enhance image more precisely than that of the spatial domain approaches.

Recently, several methods focused on enhancement in the compressed domain have been proposed [19, 10, 3, 7]. For example, one early used algorithm is the contrast measure-based approach [19], which enhances the image by manipulating the DCT coefficients. Typically, the enhancement algorithms improve overall performance, but introduce block artifacts. In [10], Lee developed a content-based enhancement algorithm based on Retinex theory; the algorithm enhances the details both in the dark and the bright areas. In [9], several algorithms had been investigated for image enhancement based on alpha-rooting theory. The methods could enhance the visualization performance and depress the image noise. Another approach is to enhance the high-frequency characteristics of the image by sharpening the DCT coefficients [13, 12]. Due to the advantages of enhancement in compressed domain, many researchers still try to find new directions in recent years such
as [6, 15, 12, 17, 1, 18], which include applications [15, 12] and new developments [17, 1] and extension to wavelet domain [18].

In general, the enhancement algorithms mentioned above achieve low computational cost; however, there are several major problems with these techniques. Firstly, due to the blocks compressed independently, the blocking artifacts are inevitable. Although, many methods [10, 7] were proposed to solve this problem, the processing for complex or strong contrast are not well. Secondly, it will produce color-distortion problem. It is worth mentioning that both spatial domain and compressed domain suffer from this problem. In general, to consider large region of illumination in image can achieve reasonable results for color constancy. Different from our previous work [7], in this paper we will propose a new spatial like method to address this problem. We can conclude that image enhancement in compressed domain have intrinsic limitation. If we can extract the spatial information in compress domain, then the benefit of enhancement in compress domain and addressing blocking artifacts and color distortion will achieve simultaneously.

Practically, we can select larger DCT blocks to suppress the block artifacts; however, the complicated frequency distribution makes difficult control factor adjustment. Thus, it is impossible to achieve satisfactory enhancement result. In spatial domain, Although the problem does not exist, it will lose the advantages of compressed domain enhancement as in [7]. Therefore, we aim to propose a new approach that has the advantages of spatial and compressed domain algorithms.

In summary, there will be three main contributions in this paper.

1. An efficient technique to transform the DCT blocks into quasi-spatial domain is developed. The spatial domain enhancement methods can be easily applied.
2. The blocking artifacts do not exist.
3. It doesn’t need to consider the adjustment of enhancement control factor and thus achieves more powerful enhancement.

This paper is organized as follows. In Section 2, we will briefly review the related works of compressed domain approaches. Detail descriptions of the proposed method and experiment are in Section 3 and Section 4, respectively. And our conclusion follows in the Section 5.

2. Problem Analysis. In this section, we will briefly review necessary foundations for explaining the proposed method. First, the algorithm of JPEG will be briefly discussed, and then the local contrast measure [22] will be explained.

2.1. Preliminaries. In JPEG algorithm, source image I(i, j) is partitioned into $8 \times 8$ non-overlapping blocks. Each block is transformed into an $8 \times 8$ coefficient matrix by DCT. The mathematical definition of each coefficient of a block is defined as

$$d_{u,v} = \frac{C_u C_v}{4} \sum_{i=0}^{7} \sum_{j=0}^{7} I_{i,j} \cos \left( \frac{(2i+1)u\pi}{16} \right) \cos \left( \frac{(2j+1)v\pi}{16} \right)$$

(1)

And the inverse DCT (IDCT) outputs is

$$I_{i,j} = \frac{C_u C_v}{4} \sum_{u=0}^{7} \sum_{v=0}^{7} d_{u,v} \cos \left( \frac{(2i+1)u\pi}{16} \right) \cos \left( \frac{(2j+1)v\pi}{16} \right)$$

(2)

where $C_w$ is defined as

$$C_w = \begin{cases} \frac{1}{\sqrt{2}}, & \text{if } w=0 \\ 1, & \text{otherwise} \end{cases}$$

(3)
As shown in Fig. 1, the output array contains $8\times8$ DCT coefficients in which the frequency distribution from low to high is arranged as a zigzag form. According to the frequency distribution, the $n$th frequency band of the original block is defined as $n = u + v$. For example, $n$ is 3 for the 3rd frequency band in Fig. 1. Typically, the coefficients with low frequencies are localized in the upper left corner, represent significant energy in images. As the band number increases, the DCT coefficients carry lesser energy that could be neglected.

\[
D = \begin{bmatrix}
d_{0,0} & d_{0,1} & d_{0,2} & d_{0,3} & d_{0,4} & d_{0,5} & d_{0,6} & d_{0,7} \\
d_{1,0} & d_{1,1} & d_{1,2} & d_{1,3} & d_{1,4} & d_{1,5} & d_{1,6} & d_{1,7} \\
d_{2,0} & d_{2,1} & d_{2,2} & d_{2,3} & d_{2,4} & d_{2,5} & d_{2,6} & d_{2,7} \\
d_{3,0} & d_{3,1} & d_{3,2} & d_{3,3} & d_{3,4} & d_{3,5} & d_{3,6} & d_{3,7} \\
d_{4,0} & d_{4,1} & d_{4,2} & d_{4,3} & d_{4,4} & d_{4,5} & d_{4,6} & d_{4,7} \\
d_{5,0} & d_{5,1} & d_{5,2} & d_{5,3} & d_{5,4} & d_{5,5} & d_{5,6} & d_{5,7} \\
d_{6,0} & d_{6,1} & d_{6,2} & d_{6,3} & d_{6,4} & d_{6,5} & d_{6,6} & d_{6,7} \\
d_{7,0} & d_{7,1} & d_{7,2} & d_{7,3} & d_{7,4} & d_{7,5} & d_{7,6} & d_{7,7} \\
\end{bmatrix}
\]

\[E_5\]

\[\text{Figure 1. } 8\times8 \text{ DCT coefficient matrix}\]

### 2.2. Measuring Contrast

The local contrast measure $c_n$ in [19] is defined as the ratio of high and low-frequency content in the DCT matrix to reflect the human visual detection

\[c_n = \frac{E_n}{\sum_{t=0}^{n-1} E_t}, \quad (4)\]

where $E_n$ is the total frequency content of the $n$th band ($n \geq 1$); $E_t$ is the average amplitude over a specific spectral band $t$:

\[E_t = \frac{\sum_{u+v=t} |d_{u,v}|}{K} K = \begin{cases} t + 1, & t < 8 \\ 14 - t + 1, & t \geq 8. \end{cases} \quad (5)\]

In [19], the image enhancement is achieved in the compressed domain. The contrast can be enhanced uniformly by multiplying an enhancement factor $\lambda$:

\[\overline{c_n} = \lambda c_n \Rightarrow \overline{c_n} = \frac{\overline{E_n}}{\sum_{t=0}^{n-1} \overline{E_t}} = \lambda c_n = \frac{E_n}{\sum_{t=0}^{n-1} E_t}. \quad (6)\]

Eq. (4) can be represented as,

\[\overline{E_n} = \lambda H_n E_n, \quad (7)\]

where $H_n$ is given by

\[H_n = \frac{\sum_{t=0}^{n-1} \overline{E_t}}{\sum_{t=0}^{n-1} E_t}. \quad (8)\]

In eq.(8), $\overline{E_n}$ is the enhanced frequency band, and $\sum_{t=0}^{n-1} \overline{E_t}$ is the total enhanced frequency content less than that of nth band. Finally, from Eq. (7), the enhanced coefficients can then be derived as $\overline{d_n} = \lambda d_n H_n$. 
The $\lambda$ is an image enhancement control factor which can be determined by the user. The image with $\lambda = 1$ reproduces the original image, and the image with $\lambda > 1$ enhances images. When $\lambda < 1$, the image will be softened.

2.3. Combined-blocks representation. The algorithms of image enhancement in compressed domain usually introduce blocking artifacts since the image is partitioned into small blocks and then each block is enhanced independently. A direct way to solve this problem is to modify the enhancement factor adaptively [3, 20, 11]. However, it results in compromising quality of the enhancement. In addition, it also increases the burden of tuning the enhancement factors. In our earlier work [7], we suggested that if larger blocks are used for enhancement, the block artifacts can be improved because large blocks contain more global image information. We combined $8 \times 8$ coding blocks into large blocks, e.g., $8n \times 8n$ block, in DCT domain. In general, this approach reduces the block artifacts effectively.

![Figure 2](image_url). Four $8 \times 8$ DCT blocks construct a $16 \times 16$ extended DCT matrix

2.4. The Procedure of Combining DCT Blocks. By using Haar transform [8, 23], we can combine the four neighboring blocks, i.e., $A$, $B$, $C$ and $D$ into one large block $G$ as shown in Fig. 2.

First, the DCT coefficients are processed in zigzag order, and we group the four coefficients in the same place (e.g., $A$, $B$, $C$ and $D$) into a large sample as shown in Fig. 2. We define an extended DCT matrix $G$, and construct the elements of a $16 \times 16$ coefficient array using Haar transform from the four $8 \times 8$ DCT blocks.

$$
\begin{align*}
    d^G_{i,j} &= \frac{1}{4}(d^A_{i,j} + d^B_{i,j} + d^C_{i,j} + d^D_{i,j}) \\
    d^G_{i,j+1} &= \frac{1}{4}(d^A_{i,j} - d^B_{i,j} + d^C_{i,j} - d^D_{i,j}) \\
    d^G_{i+1,j} &= \frac{1}{4}(d^A_{i,j} + d^B_{i,j} - d^C_{i,j} - d^D_{i,j}) \\
    d^G_{i+1,j+1} &= \frac{1}{4}(d^A_{i,j} - d^B_{i,j} - d^C_{i,j} + d^D_{i,j})
\end{align*}
$$

For example, the four coefficients can be separated in four bands as

$$
\begin{bmatrix}
    d^G_{0,0} & d^G_{0,1} \\
    d^G_{1,0} & d^G_{1,1}
\end{bmatrix}
= \begin{bmatrix}
    \frac{d^A_{0,0} + d^B_{0,0} + d^C_{0,0} + d^D_{0,0}}{4} & \frac{d^A_{0,0} - d^B_{0,0} + d^C_{0,0} - d^D_{0,0}}{4} \\
    \frac{d^A_{0,0} - d^B_{0,0} - d^C_{0,0} - d^D_{0,0}}{4} & \frac{d^A_{0,0} - d^B_{0,0} - d^C_{0,0} + d^D_{0,0}}{4}
\end{bmatrix}.
$$
The relation of coefficients between four 8×8 DCT coding blocks and a 16×16 extended DCT block can be expressed by matrix form

\[
X = H \Lambda H^T,
\]

where

\[
X = \begin{bmatrix}
    d_{2i,2j}^G & d_{2i,2j+1}^G \\
    d_{2i+1,2j}^G & d_{2i+1,2j+1}^G
\end{bmatrix}
\]

is the coefficient matrix in 16×16 extended DCT block, \( \Lambda = \begin{bmatrix}
    d_{A} & d_{B} \\
    d_{C} & d_{D}
\end{bmatrix} \) is the coefficient matrix in 8×8 DCT coding blocks, and \( H = \begin{bmatrix}
    1 & 2 & 1 & 2 \\
    2 & 1 & 2 & 1
\end{bmatrix} \) is the transform matrix, and 0 ≤ \( i \leq 7 \), 0 ≤ \( j \leq 7 \).

The inverse equation for transforming a 16×16 extended DCT block back to four 8×8 DCT coding blocks is given by

\[
\Lambda = 4H^TXH
\]

The coefficients of the four 8×8 DCT blocks can be transformed into a 16×16 extended DCT block by Eq. (11). As shown in Figure 3, the lower frequency coefficients locate in the upper-left corner of the DCT-like matrix \( G \).

Although the approximation is not exactly identical to the real DCT expression of the 16×16 matrix, it allows us to approximate the coarse distribution of the source image so that the resulting output achieves desired enhancement. Moreover, this procedure can be easily extended to larger blocks until original image size is reached. On the other hand, after enhancement, we can also easily recover the original coding block used in JPEG by Eq. (12). Therefore, it is very efficient and effective to achieve enhancement for any desired scale.

**Figure 3.** Extended DCT output block

### 3. Histogram-Based Image Enhancement in Quasi-Spatial Domain

In practice, the larger DCT blocks will effectively suppress the block artifacts; however, it will cause very complicated frequency distribution and therefore it is impossible to use only one control factor to achieve satisfactory enhancement result. On the contrary, there is no such problem in spatial domain. Although the enhancement algorithms in pixel domain can solve this problem, it will lose the advantages of compressed domain enhancement as in [7]. Therefore, we aim to propose a new approach that has the advantages of spatial and compressed domain algorithms.
3.1. Fast Quasi-Spatial Transform. In last Section, it has been shown that the DCT coefficients can be determined from different sized DCT blocks by Haar transform. However, each transform step allows us to combine four blocks of size \( n \times n \) blocks into a \( n \cdot 2^1 \times n \cdot 2^1 \) one, or to decompose a \( n \times n \) block into four \( n \cdot 2^{-1} \times n \cdot 2^{-1} \) blocks. We can repeat the process until reaching the desired block size. For example, we can divide an \( 8 \times 8 \) block into four \( 4 \times 4 \) blocks. Each \( 4 \times 4 \) block can be further divided into four \( 2 \times 2 \) blocks. Finally, each \( 2 \times 2 \) block can be divided into four \( 1 \times 1 \) blocks. Note that the block size is with \( 1 \times 1 \), then the DCT coefficients are very similar to the image in spatial domain, therefore it can be considered as an approximation of the image in pixel domain. In our work, we will call such approximation as quasi-spatial domain. According to the quasi-spatial domain, it becomes possible to apply any conventional algorithms developed in spatial-domain.

In Eq. (9), the combined blocks is using Haar transform, i.e., combining four \( n \times n \) blocks to size \( n \cdot 2^1 \times n \cdot 2^1 \) block. On the contrary, the inverse Haar transform is to decompose a \( n \times n \) block to four \( n \cdot 2^{-1} \times n \cdot 2^{-1} \) blocks. The equation is as follow.

\[
\begin{align*}
    d_{i,j}^A &= d_{i,j}^G + d_{i,j+1}^G + d_{i+1,j}^G + d_{i+1,j+1}^G, \\
    d_{i,j}^B &= d_{i,j}^G - d_{i,j+1}^G + d_{i+1,j}^G - d_{i+1,j+1}^G, \\
    d_{i,j}^C &= d_{i,j}^G + d_{i,j+1}^G - d_{i+1,j}^G - d_{i+1,j+1}^G, \\
    d_{i,j}^D &= d_{i,j}^G - d_{i,j+1}^G - d_{i+1,j}^G + d_{i+1,j+1}^G.
\end{align*}
\]  

(13)

For various block size, the Haar transform procedures should be repeated until the desired size is achieved. In order to transform the DCT block, i.e., \( 8 \times 8 \), to spatial-like domain, i.e., \( 1 \times 1 \), the necessary procedure should be repeated 3 times, and so it is very time consuming. For improving computational efficiency, in the following, we will propose a fast Haar transform, which can be performed in one step.

At first, we assume that the \( 8 \times 8 \) DCT block as an extended block \( G \) as in Fig. 2. Then we partition the \( 8 \times 8 \) block \( G \) into four \( 4 \times 4 \) blocks \( A, B, C \) and \( D \) as shown in Fig. 4 (a) and (b), respectively. We let

\[
\begin{align*}
    &d_{i,j}^G \in A, \quad \text{if } 0 \leq i < 4 \text{ and } 0 \leq j < 4 \\
    &d_{i,j}^G \in B, \quad \text{if } 4 \leq i < 8 \text{ and } 0 \leq j < 4 \\
    &d_{i,j}^G \in C, \quad \text{if } 0 \leq i < 4 \text{ and } 4 \leq j < 8 \\
    &d_{i,j}^G \in D, \quad \text{if } 4 \leq i < 8 \text{ and } 4 \leq j < 8
\end{align*}
\]  

(14)

We consider the block \( A \) as DC block \( ADC \), and the \( A \) should be with the property of DC value. Therefore, the coefficients in block \( ADC \) should adjust the value to the DC level and can be reassigned as

\[
d_{i,j}^{ADC} = \begin{cases} 
    d_{0,0}^A, & \text{if } i = j = 0 \\
    d_{0,0}^A + d_{i,j}^A, & 0 < i < 4, \ 0 < j < 4
\end{cases}
\]  

(15)

Then, the four \( 4 \times 4 \) blocks \( ADC, B, C \) and \( D \) are transformed to \( A', B', C' \) and \( D' \) by following equation,

\[
\begin{align*}
    d_{i,j}^{A'} &= d_{i,j}^{ADC} + d_{i,j}^B + d_{i,j}^C + d_{i,j}^D, \\
    d_{i,j}^{B'} &= d_{i,j}^{ADC} - d_{i,j}^B + d_{i,j}^C - d_{i,j}^D, \\
    d_{i,j}^{C'} &= d_{i,j}^{ADC} - d_{i,j}^B - d_{i,j}^C + d_{i,j}^D, \\
    d_{i,j}^{D'} &= d_{i,j}^{ADC} - d_{i,j}^B - d_{i,j}^C - d_{i,j}^D.
\end{align*}
\]  

(16)

where the \( A'=ADC \).
The four blocks is viewed as a 8×8 block H, which is an approximation of the original image in spatial domain. For convenience, it is called as quasi-spatial domain.

![Image](image in spatial domain. For convenience, it is called as quasi-spatial domain.

**Figure 4.** Illustration of fast quasi-spatial transform as in Eq. (14) to (16)

The computational complexity can be significantly reduced by this approach. Complexity comparison of the IDCT and proposed fast Haar transform are listed in Table I.

**Table 1.** Performance comparison of the different 2-D IDCT architectures and proposed Haar transform [8, 23]

<table>
<thead>
<tr>
<th>Algorithmic Approach</th>
<th>Complexity of Multiply</th>
<th>Complexity of Additions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct 2-D IDCT algorithm [13]</td>
<td>$\frac{N^2 \log_2 N}{2}$</td>
<td>$\frac{1}{2}(5N^2 \log_2 N - 2N) + 2$</td>
</tr>
<tr>
<td>Haar transform</td>
<td>$N^2 \sum_{n=0}^{N/2} \left( \frac{1}{2n!} \right) &lt; 2N^2$</td>
<td>$N^2$</td>
</tr>
<tr>
<td>Proposed Haar transform</td>
<td>0</td>
<td>$\frac{3N^2}{2}$</td>
</tr>
</tbody>
</table>

In the following, we use an 8×8 DCT coefficient block as example,

$$G = \begin{bmatrix}
54.91 & 1.43 & -0.38 & -0.15 & -1.03 & -0.05 & -0.17 & 0.07 \\
1.50 & 0.15 & -0.87 & 1.36 & 0.07 & 0.45 & 0.26 & -0.01 \\
-1.56 & 1.56 & 1.94 & -0.09 & 1.00 & 1.22 & -0.34 & -0.06 \\
-1.73 & 0.05 & -1.33 & 0.33 & -0.38 & -0.54 & 0.08 & -0.05 \\
-0.24 & -0.07 & -0.38 & 0.09 & 0.08 & -0.14 & 0.24 & 0.01 \\
-0.90 & -0.53 & 1.21 & 0.36 & 0.36 & 0.36 & -0.20 & 0.05 \\
-0.46 & 0.30 & -0.40 & -0.02 & -0.13 & 0.36 & 0.34 & -0.09 \\
-0.19 & -0.00 & -0.09 & -0.09 & -0.03 & -0.06 & -0.14 & -0.20 & 0.00
\end{bmatrix} \quad (17)$$

Using Eq. (14) to (16), the value of quasi-spatial transform is as

$$H = \begin{bmatrix}
53.71 & 56.07 & 54.98 & 55.05 & 55.61 & 56.46 & 54.85 & 55.25 \\
55.96 & 55.07 & 57.06 & 53.95 & 55.07 & 53.99 & 56.92 & 53.86 \\
56.88 & 55.24 & 56.45 & 54.61 & 55.14 & 52.05 & 56.44 & 54.95 \\
56.01 & 54.17 & 53.37 & 55.15 & 56.91 & 55.53 & 53.60 & 55.27 \\
54.03 & 56.50 & 55.26 & 55.22 & 56.28 & 56.33 & 56.10 & 54.72 \\
57.03 & 55.96 & 55.03 & 53.11 & 57.61 & 55.22 & 54.09 & 53.26 \\
58.07 & 53.89 & 56.56 & 54.87 & 55.80 & 52.17 & 57.94 & 54.81 \\
56.52 & 54.47 & 53.96 & 55.21 & 54.15 & 55.26 & 53.38 & 55.33
\end{bmatrix}. \quad (18)$$

As an example, Lena in spatial and quasi-spatial domain are shown in Fig. 5. Although the image gray level is not exactly same, the range of gray value distribution is very similar. Therefore, applying spatial-based methods to quasi-spatial image is reasonable.
3.2. Quasi-Spatial Domain Enhancement. In image enhancement, spatial-domain based algorithms are preferable for spatial domain images and usually improve the global visual perception. Since the proposed fast quasi-spatial transform is very computationally efficient, the benefit is that it allows for enhancing images by using spatial-domain based algorithms in compressed domain with limited computational effort.

In this work, a novel histogram equalization (HE) algorithm is proposed for both simplicity and performance. Suppose that the histogram of an input image \( I(x,y) \) is with \( L \) bins and denoted by \( H_I = \{ h(0), h(1), \cdots h(i), \cdots h(L-1) \} \), where \( L \) is the largest gray level, say \( L = 255 \) for 8 bits gray level image; \( h(i) \) is the normalized histogram and can be expressed as

\[
h(i) = \frac{\text{#grey level } i}{M \times N}, M \times N = \text{image size} \tag{19}\]

We then separate the input image \( I \) into two sub-images \( I_L \) and \( I_H \) based on the mean gray level \( I_m \) [6],

\[
I_m = \sum_{i=0}^{L-1} i \times h(i) \tag{20}
\]

where the \( I_L \) and \( I_H \) are defined as \( I_L = \{ I(x,y) | I \leq I_m \} \), and \( I_H = \{ I(x,y) | I > I_m \} \)

We refer to the sub-images as the lighter and darker parts, respectively.

Histogram equalization (HE) enhances images by uniformly distributing the image histogram which can be viewed as probability of intensity values. As a result, it will flatten and stretch the dynamic range of the image and then improve overall contrast of the image. HE will change the gray value globally and significantly, therefore it may result in the fact that some of the uniform regions of the output image become saturated with very bright or very dark intensities. Therefore, many techniques have been proposed to address this problem [5, 14]. In our work, we will propose a new approach that includes the advantages of histogram partition-based method and histogram clipping-based method to further improve performance.

At first, we need to determine a suitable gray value to partitioning an image. The average values of the brighter and darker sub-image, denoted as \( T_L \) and \( T_U \), can be determined by

\[
T_L = \frac{1}{X_m + 1} \sum_{i=0}^{X_m} h_L(i) \tag{21}
\]

and

\[
T_U = \frac{1}{L - 1 - X_m} \sum_{i=X_m+1}^{L-1} h_U(i). \tag{22}
\]
The weighted average value that considers the ratio of the number of pixels in the sub-images with total pixels can be defined as

$$T_{Lu} = T_L \times \frac{n_L}{n} + T_U \times \frac{n_U}{n},$$

(23)

where $n$ is the total number of pixels of the input image $I$; $n_L$ and $n_U$ are the number of pixels of the two sub-images $I_L$ and $I_H$, respectively. However, for images with extremely low or high light conditions, the parameter should be further refined as

$$T_{Lu} = \begin{cases} 
T_{LU} \times 12, & \text{if } I_m \geq 40 \\
T_{LU} \times 1, & \text{if } I_m \geq 40 \text{ and } I_m < 0 \\
T_{LU} \times 0.08, & \text{if } I_m \geq 130 
\end{cases}$$

(24)

Then the histogram of the output image of the proposed HE can be viewed as a level shifting and expressed as

$$h_{out}(i) = h(i) \times 0.75 + T_{LU}' \times 0.25.$$  

(25)

There are two advantages that result in this modification. First, the distribution of histogram is smoother than that of original histogram. The over-saturated and under-saturated problem can then be effectively addressed. Second, the color shift can be significantly reduced due to the brightness preserving.

![Example image](image.png)

**Figure 6.** Illustration of histogram modification

4. **Experimental Results.** In order to verify the proposed method, we collect a testing image dataset which include eighty 512×512 and forty 256×256 images. Considering fair and convincing comparison, as shown in Fig. 7, we select twelve different images for demonstration. These images include four grey images, four darker color-images and four...
lighter color-images. Nevertheless, the results and conclusions for demonstration images are also applicable to the images in testing dataset. In the following, the quantitative measures will be applied to whole testing image dataset for validation. In general, the enhancement usually applied to the intensity channel for color images. In our work, the RGB image is transformed to YCbCr space, which is selected in JPEG and MPEG, the algorithm applies to Y channel and then uses the enhanced Y to replace original Y. In some extreme cases, the color may distort and color correction would be considered. However, that is not the main issue in this paper.

![Example images](image-url)

**Figure 7.** The test images selected in this paper

For performance evaluation, we widely compare the state-of-the-art methods developed in recent years for both spatial domain [5, 14] and compressed domain [21, 3, 7]. We evaluate the performance by both subjective and objective measures, i.e., visual quality and quantitative indexes. Also, the average computational time required for each algorithm are also included in Table 2. The selective indexes are as follows.

1. The histogram flatness [16] that is used to indicate the contrast of images; a smaller number of flatness means that the image has better contrast.
2. WBQM [13], which is also called as Wang-Bovic-Quality Metric, assures perceptual quality of an image. Values of WBQM will lie in [-1, 1], and enhanced image with value close to 1 ensures good visual perception. In our work, the enhancement is applied to Y channel only, thus we provide Y-WBQM.
3. The third measure is used to indicate an average contrast in an image. It is defined by [1]

\[
EME = \frac{1}{n_1 \times n_2} \sum_{l=1}^{n_2} \sum_{k=1}^{n_1} 20 \log \frac{I_{\text{max}}(k, l)}{I_{\text{min}}(k, l)},
\]  

(26)
Histogram-Based Image Enhancement in Quasi-Spatial Domain

Table 2. The quantitative comparisons with various approaches

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Histogram Flatness</td>
<td>256×256</td>
<td>348.50</td>
<td>119.81</td>
<td>275.01</td>
<td>430.13</td>
<td>492.78</td>
<td>405.27</td>
<td>494.57</td>
<td>162.46</td>
</tr>
<tr>
<td></td>
<td>512×512</td>
<td>931.31</td>
<td>602.18</td>
<td>777.14</td>
<td>811.90</td>
<td>838.54</td>
<td>836.36</td>
<td>873.49</td>
<td>541.77</td>
</tr>
<tr>
<td></td>
<td>Fig. 7</td>
<td>863.80</td>
<td>558.55</td>
<td>657.17</td>
<td>748.86</td>
<td>704.57</td>
<td>688.37</td>
<td>768.77</td>
<td>386.52</td>
</tr>
<tr>
<td>Y-WBMQ</td>
<td>256×256</td>
<td>* 0.35</td>
<td>0.42</td>
<td>0.70</td>
<td>0.77</td>
<td>0.77</td>
<td>-0.03</td>
<td>0.82</td>
<td>0.89</td>
</tr>
<tr>
<td></td>
<td>512×512</td>
<td>* 0.34</td>
<td>0.58</td>
<td>0.74</td>
<td>0.80</td>
<td>1.80</td>
<td>0.26</td>
<td>0.81</td>
<td>0.68</td>
</tr>
<tr>
<td></td>
<td>Fig. 7</td>
<td>* 0.24</td>
<td>0.20</td>
<td>0.51</td>
<td>0.66</td>
<td>0.67</td>
<td>0.64</td>
<td>0.69</td>
<td>0.66</td>
</tr>
<tr>
<td>EME</td>
<td>256×256</td>
<td>36.20</td>
<td>42.08</td>
<td>39.37</td>
<td>37.38</td>
<td>41.17</td>
<td>41.21</td>
<td>33.33</td>
<td>41.29</td>
</tr>
<tr>
<td></td>
<td>512×512</td>
<td>15.01</td>
<td>22.96</td>
<td>16.83</td>
<td>17.92</td>
<td>20.91</td>
<td>21.16</td>
<td>16.94</td>
<td>22.19</td>
</tr>
<tr>
<td></td>
<td>Fig. 7</td>
<td>14.87</td>
<td>27.28</td>
<td>22.28</td>
<td>21.43</td>
<td>25.63</td>
<td>25.96</td>
<td>21.13</td>
<td>26.25</td>
</tr>
<tr>
<td>Execution Time (sec)</td>
<td>256×256</td>
<td>* 0.49</td>
<td>0.53</td>
<td>0.53</td>
<td>0.54</td>
<td>0.62</td>
<td>0.57</td>
<td>0.83</td>
<td>0.73</td>
</tr>
<tr>
<td></td>
<td>512×512</td>
<td>* 0.49</td>
<td>0.53</td>
<td>0.53</td>
<td>0.57</td>
<td>0.66</td>
<td>0.65</td>
<td>0.80</td>
<td>0.73</td>
</tr>
<tr>
<td></td>
<td>Fig. 7</td>
<td>* 0.49</td>
<td>0.59</td>
<td>0.59</td>
<td>0.58</td>
<td>0.63</td>
<td>0.63</td>
<td>0.80</td>
<td>0.74</td>
</tr>
</tbody>
</table>

where $I_{max}$, $I_{min}$ are maximum and minimum intensity value for block $I(k,l)$, respectively. Higher value of EME denotes a higher contrast and information clarity in the image.

4. Executions time evaluates the computational complexity.

For convincing, the quantitative evaluations are also applied to whole dataset. As the same in previous, we give the average value for 512×512 images and 256×256 images. The histogram flatness indicates that the histogram of the proposed approach is the closest one to ideal histogram. Y-WBQM also indicates that the enhanced images of the proposed method achieve best visual perception, and finally the proposed method is also very close to best average contrast of enhanced images. Obviously, we can easily find that the proposed method achieves excellent objective performance. For the proposed method, the computational complexity increases due to introduce extra computations. However, the increase of computation time is not significant.

From Fig. 8 to Fig. 11, the visual results are presented. We compare with state-of-the-arts methods which contain three spatial domain methods, i.e., HE and [5, 14], and four compress domain methods, i.e., [19, 21, 3, 7], respectively.

Generally, when input images have large regions with very bright or very dark intensities, HE will change the gray value significantly and thus enhanced image becomes over saturated. Fig. 9(c) to 11(c) is the enhanced image by global histogram equalization. It can be seen that over-enhancement is likely to happen when dark areas are enhanced. Although the modified histogram equalization method addressed the problem, the enhancement results sometimes unsatisfactory, see Fig. 10(e) (f), and 11(e) (f).

For compress domain methods, in order to avoid the blocking artifacts, the enhancement factor is usually using small value, therefore the overall enhancement results cannot achieve satisfactory performance, as shown in Fig. 8(d) to 11(d). Although in [7], the new approach can address the drawbacks, the multi-enhancement factors need very careful adjustment.

The proposed approach indicates that the details of the dark regions are brought out more clearly, please see Fig. 8(j) to Fig. 11(j). It is also shown that the proposed method is with the advantages of the spatial and compressed domain methods. According to quantitative evaluation, we can find that the numerical values are consistent with the visual inspection.
Figure 8. The visual quality for various approaches

Figure 9. The visual quality for various approaches

Figure 10. The visual quality for various approaches
5. **Conclusions.** In this paper, we proposed an image enhancement method in compressed domain. The main advantage of proposed method is that it integrates the concepts of histogram-based method and enhance the image details effectively.

In summary, there are three main contributions in this paper. Firstly, we have successfully presented a technique to transform the DCT blocks into quasi-spatial domain, the new method is more computational efficiency when compare to conventional inverse DCT transform. In quasi-spatial domain we can apply histogram-based method and achieves very surprising performance. Secondly, to compare with conventional compressed domain methods, the blocking artifacts are not exist, it doesn’t need to consider the strength of enhancement factor and thus achieves more powerful enhancement. Thirdly, although the multi-enhancement factors [7] method seems present visually pleasure quality, it usually need manually adjusting procedure. New approach is not necessary for such procedure. Finally, we also present a new approach that integrates the advantages of histogram partition-based method and histogram clipping-based method to achieve the better performance. We can conclude that the new method is very effective and flexible. Results also indicate that the proposed method outperforms the conventional methods.

Recently, deep learning, one of machine learning algorithms, achieves great successful in various artificial intelligence applications [2]. The application of image enhancement by using deep learning is a potential issue and will be further studied.

**Acknowledgment.** This work was supported by the Ministry of Science and Technology Granted MOST 106-2221-E-214-050 -

**REFERENCES**


Chung-Ming Kuo received the M.S. and Ph.D. degrees all in electrical engineering from Chung Cheng Institute of Technology, Taiwan, in 1988 and 1994, respectively. He is currently a Professor in the Department of Information Engineering at I-Shou University. His current research interests include image/video processing, computer vision, machine/deep learning.
Nai-Chung Yang received the M.S. and Ph.D. degrees from Chung Cheng Institute of Technology, Taiwan, in 1988 and 1997, respectively, all in defense science. He is currently an associated Professor at Department of Information Engineering, I-Shou University. His research interests include image/video retrieving, multimedia signal processing, and optimal estimation.

Jeng-Yan Wu received the B.E., and M.E. degree from the Department of Information Engineering, I-Shou University, Kaohsiung, R.O.C., in 2018, and 2020. His research interests include image processing and deep learning.

Sheng-Chang Chen received the B.E., and M.E. degree from the Department of Information Engineering, I-Shou University, Kaohsiung, R.O.C., in 2012, and 2014, respectively. His research interests include image/video processing and computer vision.