# IMAGE RESOLUTION SCALING WITH ARBITRARY UNEQUAL RATIOS IN EACH DIRECTION IN THE DCT DOMAIN

*Kebin An, Jun Sun, member, IEEE, and Lei Zhou*

Institute of Image Communication and Information Processing
Shanghai Jiaotong University, Shanghai,China

**ABSTRACT**

To meet with the different client end devices and network bandwidth transmission requirements, it is necessary to convert the high definition resolution images and videos to standard definition resolution format. A novel approach to convert image resolution with arbitrary ratios in the DCT (Discrete Cosine Transform) domain is proposed, which exploits the relationship of a block and its subblocks with differing size. It can realize arbitrary unequal ratios in the horizontal and vertical direction, respectively. Unlike the present methods, it does not need upsampling-downsampling process. It can perform downsizing directly to the original data and confirms to the standard decoder. The proposed approach is computationally fast and memory efficient and produces visually better images with higher PSNR compared to the spatial methods.

## 1. INTRODUCTION

High definition resolution images or video frames are becoming widely used in consumer electronics. However, limited network bandwidth brings forward the demand of smaller image size for fast transmission. In order to meet with the different client end devices and network width constraints, the spatial resolution of the original image/video needs to be changed in such situations. So it is necessary to convert the high definition images into standard definition images. Both of them use the DCT-based coding standard for transmission, but they have different spatial resolutions. Spatial resolution downscaling is an important issue in HD to SD image transcoding. The simple approach is to downsize the image in the spatial domain as follows. First, fully decode the compressed image and then apply spatial filter techniques to downsize the decoded image. Finally, encode the image again. This straightforward approach is undesirable due to its significant computational overhead associated with decoding and encoding as well as the large memory requirement. As most images are stored in the compressed format, it is more desirable to perform the image downsizing directly in the compressed domain. Many approaches have been proposed to resize image in the compressed domain [1-3]. The recent approaches showed good results in both computational complexity and image quality [2-3]. Their image downsizing processes are performed in the DCT domain by the low-pass truncated approximation [2] and the subband approximation [3] of DCT coefficients, respectively, both of which showed good PSNR improvement. However, all these approaches can only realize downsizing by a factor of two, or the downsizing ratio must be integer. They lack flexibility for arbitrary image downsizing. In [4], Mehta and Desai have proposed an arbitrary image downsizing method. Their method can realize image downsizing with ratio $m/n$, but it is computationally expensive and memory inefficient due to the large size of the matrix multiplication. Park, H.W et al have proposed an image resizing method with arbitrary scales of $L/M \times L/M$ in the DCT domain using the multiplication-convolution properties of DCT [5]. Anyway, all of these methods can only process image resizing with equal ratio in the horizontal and vertical direction, respectively. Whereas, in reality, there are needs for image spatial resolution conversion with unequal downsizing ratio in both direction, such as from $1920 \times 1080$ to $720 \times 576$. In [7], Shu and Chau proposed an arbitrary image downsizing method with the horizontal downsizing ratio $R_x$ differing vertical downsizing ratio $R_y$, which partitioned the original image into many supporting areas of size $8R_x \times 8R_y$, and then downsized the supporting areas into $8 \times 8$ output blocks. From the analysis, we can see that $R_x$ and $R_y$ must satisfy some conditions to make the $8R_x$ and $8R_y$ be integers. It is the limited arbitrary factors. In [8], J. Mukherjee presented an arbitrary resizing method with factors $P/Q \times R/S$ using the spatial relationships of the DCT coefficients between a block and its subblocks developed by Jiang and Feng [6], where $P$, $Q$, $R$, and $S$ are positive integers. However, the rational factors image downsizing is carried out by upsampling an image with the integral factor $P \times R$, followed by downsampling with the factor $Q \times S$. This upsampling-downsampling process demands large computational complexity and memory requirements.

Furthermore, the output blocks did not satisfy the requirement of size $8\times 8$ in the DCT domain.

In this paper, a generalized algorithm that can convert image resolution with unequal ratios in the horizontal and vertical direction, respectively, in the DCT domain is proposed, which exploits the relationship of a block and its subblocks with differing size developed by Brynmor [9] and Shu's high frequency DCT coefficients discarding technique [7]. The presentation in [9] is the extension of that in [6]. The main difference of them is that the subblock can only be of size $N\times N$ in [6], however, the subblock can be of arbitrary size $L\times M$ in [9]. Our method can also realize the image downsizing with factors $P/Q\times R/S$. But it does not need the upsampling-downsampling process, and it performs the downsizing directly to the original data. The output blocks are also of size $8\times 8$. Compared to the method in [8], our method is computation and memory efficient, and with higher image quality objectively and visually.

## 2. PROPOSED RESOLUTION SCALING ALGORITHM

The relationship between the DCT coefficients of a block and those of its subblocks facilitates image processing in the transform domain [9]. Let $x$ be a 2D signal with the size of $M\times N$, the 2D DCT transformation coefficients can be expressed as:

$$X = TxU^T \tag{1}$$

$$x = (T^{-1})X(U^{-1})^T \tag{2}$$

where, $T$ and $U$ are DCT transformation matrix.

For an image, if subblock transforms are taken in two different ways, defined by $T_1, U_1$ and $T_2, U_2$, the relation between the two sets of DCT coefficients of $X_1$ and $X_2$ can be obtained as in [9]. The detailed description can be referred to [9].

### 2.1 Decomposite-ImageDownscaling-Decomposite(DIDD)

In this section, we take the image resolution conversion from $1920\times 1080$ to $720\times 576$ as an example to illustrate our proposed arbitrary image downsizing technique.

For the rational image downsizing, the horizontal downsizing ratio $R_x$ can differ from the vertical downsizing ratio $R_y$ [7]. For the image spatial resolution conversion from $1920\times 1080$ to $720\times 576$, $R_x$ is 8:3, $R_y$ is 15:8. One 8×8 output block in the downsized image can come from a supporting area in the original image. Using the presentation in [9], The original image can be partitioned into many supporting areas. In a natural image, most of the signal energy is concentrated in the lower frequency part in the DCT domain. A reasonable downsizing scheme as proposed is to retain only the lower frequency coefficients and discard the high frequency components of a block [7]. So, for the resolution conversion, our approach can realize the downsizing by first obtaining the supporting areas from the original image, and then downsizing and composing them into 8×8 blocks of output image.

For the convenience of computation, let the image horizontal downsizing ratio be $D_x$ = 64:24, vertical ratio be $D_y$ = 45:24, and $L_x = 24\times D_x$, $L_y = 24\times D_y$. So, for the conversion of image resolution from 1920 × 1080 to $720\times 576$, we need first transform the image composed of blocks of size $8\times 8$ into the one composed of supporting areas of $64\times 45$, and then down-sampled the each supporting areas into the size of $24\times 24$. Finally, in order to make the downsized image output compliant to the image compression standard, we need to further partition the down-sampled image composed of blocks of size $24\times 24$ into the one of blocks $8\times 8$. In this way, we can convert $X_1$ to $X_2$ region by region.

Let $m_x/n_x$ =8:1, $m_y/n_y$ =45:8. Let $y$ be a matrix with the size of $m_x\times m_y$, each element of which is an 8×8 matrix. $Y_1$ and $Y_2$ are its two different DCT transform representations, their DCT transform matrixes can be expressed with $T_1, U_1$ and $T_2, U_2$ as follows:

$$T_1 = \begin{bmatrix} T^{(8)} & O^{(8)} & \cdots & O^{(8)} \\ O^{(8)} & T^{(8)} & \cdots & O^{(8)} \\ \vdots & \vdots & \ddots & \vdots \\ O^{(8)} & O^{(8)} & \cdots & T^{(8} \end{bmatrix}_{m_x\times m_x}$$

$$U_1 = \begin{bmatrix} T^{(8)} & O^{(8)} & \cdots & O^{(8)} \\ O^{(8)} & T^{(8)} & \cdots & O^{(8)} \\ \vdots & \vdots & \ddots & \vdots \\ O^{(8)} & O^{(8)} & \cdots & T^{(8} \end{bmatrix}_{m_y\times m_y}$$

$$T_2 = \left[T^{(64)}\right]_{n_x\times n_y}$$

$$U_2 = \begin{bmatrix} T^{(45)} & O^{(45)} & \cdots & O^{(45)} \\ O^{(45)} & T^{(45)} & \cdots & O^{(45)} \\ \vdots & \vdots & \ddots & \vdots \\ O^{(45)} & O^{(45)} & \cdots & T^{(45)} \end{bmatrix}_{n_y\times n_y} \tag{3}$$

where, $T^{(P)}$ is a $P$-by-$P$ DCT transform matrix, and $O^{(P)}$ is a $P$-by-$P$ zero matrix with all the elements are zero. Thus, we can obtain $Y_2$ from $Y_1$ using the relationship described as in [9], Using this way, image region composed of blocks of size $8\times 8$ can be transformed into the one composed of blocks of $64\times 45$.

As most of the signal energy is concentrated in the lower frequency part in the DCT domain. To realize the image downsizing, we need to discard the high frequency component and extract only the low frequency part of a supporting area of $64\times45$ to downsize it into the size of $24\times24$ in the DCT domain as follows:

$$\hat{B}=\begin{bmatrix}I_{24} & 0\end{bmatrix}_{24\times24D_x}\cdot B\cdot\sqrt{\frac{1}{R_xR_y}}\cdot\begin{bmatrix}I_{24}\\0\end{bmatrix}_{24D_y\times24}$$

$$=P_1\cdot B\cdot\sqrt{\frac{1}{R_xR_y}}\cdot P_2 \qquad (4)$$

The block $\hat{B}$ is an output DCT block of size $24\times24$, $B$ is the supporting area with the size of $L_x\times L_y$ in $Y_2$. Then, downsizing is realized from $(n_x\times n_y)\times(L_x\times L_y)$ blocks to $(n_x\times n_y)\times(24\times24)$ blocks in the DCT domain as follows:

$$C=H_1\cdot Y_2\cdot\sqrt{\frac{1}{R_xR_y}}\cdot H_2$$

$$=H_1\cdot R_{12}\cdot Y_1\cdot\sqrt{\frac{1}{R_xR_y}}\cdot S_{12}^T\cdot H_2$$

$$=H_L\cdot Y_1\cdot\sqrt{\frac{1}{R_xR_y}}\cdot H_R \qquad (5)$$

where, $H_L=H_1\cdot R_{12}$, and $H_R=S_{12}^T\cdot H_2$. The obtain of $R_{12}$ and $S_{12}^T$ can be referred to [9]. $H_1$ is a diagonal matrix with $n_x$ diagonal elements, each of the diagonal elements is $P_1$, and $H_2$ is also a diagonal matrix with $n_y$ diagonal elements, each of the diagonal elements is $P_2$. The scale factor can be merged into $H_L$ or $H_R$. Finally, using the method in [5], we can partition the each block of size $24\times24$ into $(3\times3)$ blocks of size $(8\times8)$ in the DCT domain. As the DCT transform is block independent, the image $X_1$ can be divided into many non-overlapping regions composed of $Y_1$, each region can be processed independently as discussed above.

From the above analysis, we can see that image resolution conversion from $1920\times1080$ to $720\times576$ in the DCT domain can be realized by subdividing the image into non-overlapping subband regions, and each subband region is processed independently.

As we can see from DIDD methods, both pre-matrix $H_L$ and post-matrix $H_R$ are independent of the input blocks so that they can be precomputed and stored in the memory. With the look up table-based implementation method, no delays are imposed while processing real-time image process. Compared to the conventional method that processes in the spatial domain, the DCT and inverse DCT are avoided, and hence, the computation can be saved.

So for the arbitrary image downsizing, we can use the method described as above in the DCT domain.

## 3. EXPERIMENTS AND DISCUSSION

In this section, we present the experimental results for our proposed image resolution conversion algorithm. The original image size is $1920\times1080$ [10] and is downsized into the size of $720\times576$. In the experiments, to evaluate the performance of the proposed method, downsized pictures are upsized to the original image size and compared to the original image. We use the method proposed by Mukherjee [8] as a reference in the DCT domain. For the Mukherjee's method, the image conversion from $1920\times1080$ to $720\times576$ is realized by upsampling the image with the integral factor $8\times15$, followed by downsampling with the factor $3\times8$. This process is computational inefficient and needs large memory requirement. It is inefficient for real time applications. Mukherjee only gave the subjective images and did not present the objective quality assessment in [8]. To evaluate the performance of the Mukherjee's method, we compare it with the spatial bicubic interpolation method. The picture downsized by the spatial domain bicubic interpolation is upsized by bicubic interpolation (BI-BI) and by DCT-domain zero padding method (BI-ZPD). We use one frame from the CIF sequence of "tempetel" to evaluate the performance of BI-BI, BI-ZDP and the Mukherjee's method. The CIF image is downsized into the size of $256\times192$. The reason we use the CIF sequence is to make the computation more efficient. From the result, we find that the PSNR is 30.79dB using BI-BI method, the PSNR is 31.45dB using BI-DZP method, and the PSNR is 30.96dB using Mukherjee's method, which is inferior to the BI-DZP. So, in this section, we only compare our method with BI-DZP to get visual and objective quality evaluation and do not make Mukherjee's method as a comparison to avoid the large computational complexity. Due to space limit, we only give one image for presentation. Fig.1 is the original "Sunlight" image. Fig.2 (a) is the downsized image by Bicubic method, Fig.2 (b) is the downsized image by DIDD method. Fig. 3 shows the picture by the BI-DZP method, the PSNR is 38.73 dB. For our proposed method, the downsized picture is upsampled to its original size by DCT-domain zero padding method. Fig. 4 shows the picture downsized by DIDD and upsized to its original size by DCT-domain zero padding and the PSNR is 42.19 dB. Comparisons show that our proposed DCT domain resolution conversion method can well preserve the most important information of the picture and present a better visual quality. Less information is lost in the downsizing process. Our method can preserve most detail information, while the spatial method produces the blurring in the edges. Table I gives PSNR comparison for different images resolution conversion of four test images. It is obvious that the our schemes outperform the spatial domain bicubic interpolation method and produce PSNR improvement.

When compared the computational complexity of our method with the reference method, The BI-DZP is the combination of spatial and DCT domain processing, so we just compared our method with Mukherjee's DCT domain method. It does not need upsampling-downsampling process, and is memory efficient. From the simulations, our method can speed up approximate 65%. Therefore, our proposed method outperforms the state-of-the-art method visually and objectively and easier for hardware implementation.

TABLE I
PSNR COMPARISONS FOR DIFFERENT IMAGE DOWNSIZING

| | BI-ZPD (dB) | DIDD (dB) |
|---|---|---|
| Sunlight | 38.73 | 42.19 |
| Eye | 40.86 | 42.41 |
| Outdoor | 37.83 | 41.19 |
| Fish | 33.60 | 35.37 |

**Fig. 1. Original Sunlight image.**

(a) (b)

**Fig. 2. (a) Downsized image of 720 × 576 by Bicubic method; (b) Downsized image of 720 × 576 by DIDD method.**

**Fig. 3. Image downsized by Bicubic method and upsized to its original size by DCT domain zero padding.**

**Fig. 4. Image downsized by DIDD method and upsized to its original size by DCT domain zero padding.**

## 4. REFERENCES


[1] N. Merhav and V. Bhaskaran, "Fast algorithms for DCT-domain image down-sampling and for inverse motion compensation," IEEE Trans. Circuits Syst. Video Technol., pp. 468–476, vol. 7, June 1997.

[2] R. Dugad and N. Ahuja, "A fast scheme for image size change in the compressed domain," IEEE Trans. Circuits Syst. Video Technol., pp. 461–474, vol. 11, April 2001.

[3] J. Mukherjee and S. K. Mitra, "Image resizing in the compressed domain using subband DCT," IEEE Trans. Circuits Syst. Video Technol., pp. 620–627, vol. 12, July 2002.

[4] D. Mehta and U. B. Desai, "A primer to video transcoding: Image transcoding," in Proc. 8th Nat. Conf. Communications, pp. 528–532, Jan. 2002.

[5] Park, H.W., Park, Y.S., and Oh, S.K., "L/M-image folding in block DCT domain using symmetric convolution," IEEE Trans. Image Processing, pp. 1016–1034, vol. 12, September 2003.

[6] J. Jiang and G. Feng, "The spatial relationship of DCT coefficients between a block and its subblocks," IEEE Trans. Signal Processing, pp. 1160–1169, vol. 50, May 2002.

[7] Haiyan Shu and Lap-Pui Chau, "An efficient arbitrary downsizing algorithm for video transcoding," IEEE Trans. On Circuits and Systems for Video Technology, pp. 887-891, vol. 14, No. 6, June 2004.

[8] J. Mukherjee and S.K. Mitra, "Arbitrary resizing of images in DCT space," IEE Proc.-Vis. Image Signal Processing, pp.155-164, vol. 152, No. 2, April 2005.

[9] Brynmor J.Davis and S. Hamid Nawab, "The relationship of transform coefficients for differing transforms and/or differing subblock sizes," IEEE Trans. Signal Processing, pp.1458-1461, vol. 52, May 2004.

[10] http://www.siliconimaging.com/hdtv-images.htm.