Abstract. Mirroring an image (horizontally or vertically) and rotation by 90°, 180° and 270° in the spatial domain has been implemented by changing the signs of the 2D-DCT coefficients of the original image appropriately. This approach leads to an efficient compressed domain-based image mirroring and rotation. This technique is now extended to 2D-DST, i.e., 2D-DST has properties similar to 2D-DCT and can be applied to image mirroring (horizontally or vertically) and also rotation by 90°, 180° and 270°. We illustrate these methods in video editing application.

1 Introduction

Image or video coding standards like JPEG, MPEG-1,2,4 and latest H.264 utilize DCT or integer DCT\(^1\) to compress image or video data. We can compress images to save storage space or transmit and then edit in the compressed-domain. Compressed-domain image manipulation has been introduced by Smith and Rowe\(^2\). By using the linearity of the DCT, video dissolve and caption insert were processed in the compressed domain. Chang and Messerschmitt\(^3\) introduced a method to reconstruct DCT block using neighboring
four DCT blocks in the compressed domain based on the fact that the DCT is distributive to matrix multiplication. Merhav and Bhaskaran\textsuperscript{4} introduced fast DCT-domain bilinear interpolation and motion compensation. Shen, Sethi and Bhaskaran\textsuperscript{5} introduced logotype insertion in the compressed video. They showed that DCT-domain convolution which corresponds to multiplication in the spatial domain is more efficient because of the orthogonality of the DCT. As an extension of these studies, a discrete cosine transform (DCT)-domain image flipping scheme has been introduced.\textsuperscript{6} We present a similar discrete sine transform (DST)-domain\textsuperscript{7,8} image mirroring and rotation scheme in this paper.

2 Proposed Algorithm

By changing the signs of the DST (discrete sine transform) coefficients, we can change the order of input sequence elements into the reverse order in the time domain. In other words, let \( \{X(1), X(2), \ldots, X(N-1)\} \) be the DST coefficients of a sequence, \( \{x(1), x(2), \ldots, x(N-1)\} \). Change the signs of the DST coefficients as follows, \( \{X(1), -X(2), X(3), -X(4), \ldots, -x(N-1)\} \). The inverse DST of the latter is \( \{y(n)\}_{n=1}^{N-1} = \{x(N-1), x(N-2), \ldots, x(2), x(1)\} \) which has elements in reverse order.

This can be proved as follows. For a sequence, \( x(n) \) for \( n = 1, 2, \ldots, N-1 \), the DST Type I and its inverse\textsuperscript{7} are defined as follows:

\[
X(m) = \sqrt{\frac{2}{N}} \sum_{n=1}^{N-1} x(n) \sin \left( \frac{mn\pi}{N} \right) \quad m = 1, 2, \ldots, N-1 \tag{1}
\]

\[
x(n) = \sqrt{\frac{2}{N}} \sum_{m=1}^{N-1} X(m) \sin \left( \frac{mn\pi}{N} \right) \quad n = 1, 2, \ldots, N-1. \tag{2}
\]
From Eq. (2) we can recover the sequence \( \{x(n)\}_{n=1}^{N-1} \) as follows:

\[
x(1) = \sqrt{\frac{2}{N}} \left[ X(1) \sin \frac{\pi}{N} + X(2) \sin \frac{2\pi}{N} + X(3) \sin \frac{3\pi}{N} + \cdots \right] 
\]

\[
x(2) = \sqrt{\frac{2}{N}} \left[ X(1) \sin \frac{2\pi}{N} + X(2) \sin \frac{4\pi}{N} + X(3) \sin \frac{6\pi}{N} + \cdots \right] 
\]

\[
\vdots
\]

\[
x(N-1) = \sqrt{\frac{2}{N}} \left[ X(1) \sin \frac{N-1}{N} \pi + X(2) \sin \frac{N-1}{N} 2\pi \\
+ X(3) \sin \frac{N-1}{N} 3\pi + X(4) \sin \frac{N-1}{N} 4\pi + \cdots \right] 
\]

\[
= \sqrt{\frac{2}{N}} \left[ X(1) \sin \frac{\pi}{N} - X(2) \sin \frac{2\pi}{N} + X(3) \sin \frac{3\pi}{N} \\
- X(4) \sin \frac{4\pi}{N} + \cdots \right]
\]

Let \( Y(m) \) for \( m = 1, 2, \ldots, N-1 \) be the DST coefficients of \( y(n) \) for \( n = 1, 2, \ldots, N-1 \), and be related to \( X(m) \) for \( m = 1, 2, \ldots, N-1 \) as follows:

\[
Y(2k-1) = X(2k-1)
\]

\[
Y(2k) = -X(2k)
\]

for \( k = 1, 2, \ldots, (N-1)/2 \). The sequence \( y(n) \) for \( n = 1, 2, \ldots, N-1 \) is the mirror sequence of \( \{x(n)\}_{n=1}^{N-1} \), i.e.,

\[
y(1) = \sqrt{\frac{2}{N}} \left[ X(1) \sin \frac{\pi}{N} - X(2) \sin \frac{2\pi}{N} + X(3) \sin \frac{3\pi}{N} \\
- X(4) \sin \frac{4\pi}{N} + \cdots \right]
\]

\[
= x(N-1)
\]

\[
\vdots
\]
\[ y(N-2) = \sqrt{\frac{2}{N}} \left[ X(1) \sin \frac{N-2}{N} \pi - X(2) \sin \frac{N-2}{N} 2\pi \\
+ X(3) \sin \frac{N-2}{N} 3\pi - X(4) \sin \frac{N-2}{N} 4\pi + \ldots \right] = x(2) \]

\[ y(N-1) = \sqrt{\frac{2}{N}} \left[ X(1) \sin \frac{N-1}{N} \pi - X(2) \sin \frac{N-1}{N} 2\pi \\
+ X(3) \sin \frac{N-1}{N} 3\pi - X(4) \sin \frac{N-1}{N} 4\pi + \ldots \right] = x(1). \]

\[ \{y(n)\}_{n=1}^{N-1} = \{x(N-1), x(N-2), \ldots, x(2), x(1)\} \text{ is } \{x(n)\}_{n=1}^{N-1} \text{ in reverse order.} \]

3 Extension to the Two-D Case

Mirroring and rotation of an image is illustrated using a (4 × 4) block. Given

\[
\begin{bmatrix}
x_{11} & x_{12} & x_{13} & x_{14} \\
x_{21} & x_{22} & x_{23} & x_{24} \\
x_{31} & x_{32} & x_{33} & x_{34} \\
x_{41} & x_{42} & x_{43} & x_{44}
\end{bmatrix}
\]

Horizontally mirrored sequence of \([x]\) is

\[
\begin{bmatrix}
x_{14} & x_{13} & x_{12} & x_{11} \\
x_{24} & x_{23} & x_{22} & x_{21} \\
x_{34} & x_{33} & x_{32} & x_{31} \\
x_{44} & x_{43} & x_{42} & x_{41}
\end{bmatrix}
\]

Vertically mirrored sequence of \([x]\) is

\[
\begin{bmatrix}
x_{41} & x_{42} & x_{43} & x_{44} \\
x_{31} & x_{32} & x_{33} & x_{34} \\
x_{21} & x_{22} & x_{23} & x_{24} \\
x_{11} & x_{12} & x_{13} & x_{14}
\end{bmatrix}
\]
\[ x \] rotated by 90° is
\[
[x_{90°}] = \begin{pmatrix}
    x_{41} & x_{31} & x_{21} & x_{11} \\
    x_{42} & x_{32} & x_{22} & x_{12} \\
    x_{43} & x_{33} & x_{23} & x_{13} \\
    x_{44} & x_{34} & x_{24} & x_{14}
\end{pmatrix}.
\]

\[ x \] rotated by 180° is
\[
[x_{180°}] = \begin{pmatrix}
    x_{44} & x_{34} & x_{24} & x_{14} \\
    x_{34} & x_{24} & x_{14} & x_{13} \\
    x_{24} & x_{14} & x_{13} & x_{12} \\
    x_{14} & x_{13} & x_{12} & x_{11}
\end{pmatrix}.
\]

\[ x \] rotated by 270° is
\[
[x_{270°}] = \begin{pmatrix}
    x_{14} & x_{24} & x_{34} & x_{44} \\
    x_{13} & x_{23} & x_{33} & x_{43} \\
    x_{12} & x_{22} & x_{32} & x_{42} \\
    x_{11} & x_{21} & x_{31} & x_{41}
\end{pmatrix}.
\]

Then \([x_{90°}] = h\{x\}^{T}\), where \(h\{\}\) denotes horizontal flipping an input matrix and is defined by
\[
[x_{h}] = h\{x\} = [x \, J]
\]
where
\[
[J] = \begin{pmatrix}
    O & 1 \\
    1 & 0 \\
    \vdots & \ddots \\
    1 & O
\end{pmatrix}
\]
is the opposite diagonal unit matrix. Here the superscript \(T\) represents transpose.

Image can be represented as nonoverlapping blocks of size \((N-1) \times (N-1)\) (for example \(N - 1 = 8\)). Each block can be represented as a matrix \([x] = \{x(n_1, n_2)\}_{n_1, n_2}^{N-1}\). The DST is defined in matrix form as \([S] = \{s(m_1, n_1)\}_{m_1, n_1}^{N-1}\) where
\[ s(m_i, n_j) = \sqrt{2/9} \sin(n_j m_i \pi /9) . \] (13)

Then we can compute the two-dimensional DST coefficients matrix \([X]\) of an input image block or matrix, \([x]\) as follows:

\[ [X] = [S][x][S]^T . \] (14)

Let the 2D-DST of horizontally mirrored sequence \([x_h]\) be \([X_h]\) and the 2D-DST of vertically mirrored sequence \([x_v]\) be \([X_v]\). Then we can compute \([X_h]\) and \([X_v]\) as follows:

\[ [X_h] = [X][M] \] (15)

\[ [X_v] = [M][X] \] (16)

where the matrix \([M]\) is defined as

\[
[M] = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -1
\end{bmatrix} . \] (17)

Thus we can select a group of consecutive 2D-DST blocks of an image and flip horizontally according to the following steps:

1. Apply the \((8 \times 8)\) 2D-DST to the nonoverlapping blocks of size \((8 \times 8)\) of the image.
2. Set the size of a rectangular block to be horizontally flipped. Horizontal and vertical sizes should be multiples of eight according to the DST size.
3. Compute DST-domain image flipping for each 2D-DST block by using Eq. (15).
4. Rotate horizontally DST blocks within the rectangular block. The most left DST block goes to the most right, and vice versa.

In general this is applicable to \([(N - 1) \times (N - 1)]\) 2D-DST where \(N = 5, 9, 17, \ldots\) We can also rotate an image block by 90° in the 2D-DST domain by using the proposed scheme.

Let us denote the 2D-DST of the 90°, 180° and 270° rotated blocks in the spatial domain \([x_{90°}], [x_{180°}] \) and \([x_{270°}]\) as \([X_{90°}], [X_{180°}] \) and \([X_{270°}]\) respectively. Then we can compute 2D-DST of rotated blocks in the spatial domain as follows:

\[
[X_{90°}] = [S]^{T} [J] [S] = [S]^{T} [S] [M] \\
= [X]^{T} [M] = [X_{r}]^{T} .
\]

(18)

Let the symbol \(\otimes\) represent element-by-element multiplication.

\[
[X_{180°}] = [M] [X] [M] = [W] \otimes [X] \tag{19}
\]

where the matrix \([W]\) is defined as

\[
[W] = \begin{pmatrix}
1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\
-1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\
1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\
-1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\
1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\
-1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\
1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\
-1 & 1 & -1 & 1 & -1 & 1 & -1 & 1
\end{pmatrix} .
\]

(20)

Similarly,

\[
[X_{270°}] = [S]^{T} [J]^{T} [S] = [M] [S]^{T} [S] \\
= [M] [X]^{T} = [X_{r}]^{T} .
\]

(21)
We can also show Eqs. (19) and (21) by using \[ x \leftrightarrow [X] \], where the symbol \( \leftrightarrow \) indicates forward and inverse 2D-DST pair.

4 Simulations

Using Eqs. (15)-(21), the 2D-DST domain manipulation is utilized to obtain image mirroring and rotation by 90\( ^\circ \), 180\( ^\circ \), and 270\( ^\circ \) in spatial domain. Note that image mirroring and rotation by 90\( ^\circ \), 180\( ^\circ \), and 270\( ^\circ \) in spatial domain is accomplished by changing the signs of the 2D-DST coefficients of the original image appropriately. See Eqs. (15)-(21). These operations are implemented on images Lena (Fig. 1) and Boat (Fig. 2).

5 Complexity Comparison

Image or video coding standards like JPEG, MPEG-1,2,4 and H.264 utilize DCT and integer DCT to compress image or video data. We can compress images to save storage space or transmit and then edit in the compressed-domain. In that case, the DCT-domain image mirroring scheme\(^6\) is very efficient compared to the existing method, which does the IDCT and flipping operation and then the DCT. The technique developed\(^6\) does not need any multiplications and only needs to change signs of the DCT coefficients. Similarly by changing the signs of the 2D-DST coefficients, mirroring and rotation of the images are accomplished.
6 Conclusions

Image or video coding systems can utilize DST to compress image or video data. By using the proposed scheme, we can flip images horizontally and/or vertically in the spatial domain by appropriately changing the signs of the 2D-DST coefficients. The proposed scheme is very efficient compared to the existing method, which does the 2D-IDST, flipping operation in the spatial domain and then the 2D-DST, since the DST/IDST need multiplications. We also showed efficient compressed-domain scheme for rotation of images by 90°, 180°, and 270° in the spatial domain.

Acknowledgments

Do Nyeon Kim acknowledges the support by the Korea Ministry of Information and Communication.

References


Fig. 1 Mirroring or rotation of portion of Lena image in the spatial domain by manipulating the 2D-DST coefficients of the original block: (a) horizontal mirroring, (b) vertical mirroring, (c) rotation by 90°, (d) rotation by 180°, and (e) rotation by 270°.
Fig. 2 Mirroring or rotation of portion of Boat image in the spatial domain by manipulating the 2D-DST coefficients of the original block: (a) translation, (b) horizontal mirroring of (a), (c) vertical mirroring of (a), (d) translation, (e) rotation of (d) by 90°, (f) rotation of (d) by 180°, and (g) rotation of (d) by 270°.