Influence of execution of orthogonal block transform types and results of comparison

A G Shoberg, K A Shoberg

Pacific National University, 136, Tikhookeanskaya St., Khabarovsk, 680035, Russia

E-mail: Shoberg@rambler.ru

Abstract. This article considers variants of orthogonal block transforms execution. Block sampling and recording variants are described when performing a transform. Two variants of the block transforms execution with rotation invariance elements are proposed. All proposed expressions are simulated and the results are compared. The approach to the frequency transforms can be used in any dimension. It works for any basic function set (symmetrical, anti-symmetrical, without symmetry). This approach can be used in the various fields related to the data processing, data transfer, etc.

1. Introduction

Most fixed images and video frames are presented in a frequency domain on digital media at the present time. Modern multimedia applications, including video servers, image databases, video conferencing in most cases are compressed. Effective processing algorithms for search and visualization of a large amount of data are required. Data manipulation functions include scaling, shifting, rotation, deformation, etc. Such actions are usually performed in a spatial domain. The algorithms are well developed and there is application software. A number of works suggest such operations of implementation in a frequency domain. The paper [1] proposes direct data processing in the frequency domain to obtain equivalent changes in a spatial domain based on DCT. An image mirroring was produced with blocks redistribution and rotation was performed in a matrix multiplication. This method can be used when editing images and video, where image and video data are available in the JPEG format. The authors [2] propose an approach to images resizing with a selected scaling factor in the DCT domain. If an image is set by blocks 8x8, then its changed image is also obtained in 8x8 coefficients of DCT blocks. It is argued that the proposed approach is far from a computational point of view and creates visually clear images with a good PSNR values. A filtration in the frequency domain is considered in [3]. A concept of transform domain filtering is defined in [4], and a pipelining structure is proposed as a means to implement it. It refers to filtering applied directly to transform domain data. The papers [5, 6] are concerned with the indexing and retrieval of images based on features extracted directly from the JPEG DCT domain. The authors examine some possible ways of manipulating DCT coefficients by the standard image analysis approaches to describe an image shape, texture, and color. Researchers [7] have developed compressed-domain computer vision algorithms. They propose a compressed-domain corner detection for a DCT-based compressed image. It partially decodes the image to obtain DCT data and then split the data into the high precision data. The edge map is then estimated by gradient patterns and coefficients to detect corners. A number of other articles [8,9,10] are devoted to the data processing in the frequency domain.
2. Orthogonal transform and its block form

Orthogonal transform is a linear transformation of a Euclidean vector space that preserves the lengths and the scalar vectors product. On an orthonormal basis, the orthogonal transformation corresponds to an orthogonal matrix.

An orthogonal transform executes the whole source data set (frequency characteristic usage) or set of standard blocks (for example, effectively storage).

If there is any transform, then there is the basis function set \( \Phi(v,u,m,n) \). For a block with a size \( 2KN \times 2KN \) of the direct transform it can be written as:

\[
X(v,u) = C(v)C(u) \sum_{m=0}^{2KN-1} \sum_{n=0}^{2KN-1} x(m,n)\Phi(v,u,m,n),
\]

where \( x(m,n) \) – source data set;
\( m,n \) – indexes on the internal block;
\( v,u \) – indexes in the frequency domain;
\( C(v),C(u) \) – coefficients depending of the orthogonal functions set;
\( X(v,u) \) – transform result.

Inverse transform is:

\[
x(m,n) = \sum_{m=0}^{2KN-1} \sum_{n=0}^{2KN-1} C(v)C(u)X(m,n)\Phi(v,u,m,n).
\]

Equations (1) and (2) correspond to the standard no block transform. The test image is shown in figure 1, a. Its representation (1) in the DCT frequency domain \( X \) is placed in figure 1, b.

![Figure 1. The test image and its representations in the frequency domain.](image)

There are \( 2K \) blocks with \( N \times N \) size in the second case.

A transform in block form based on (1) is written as:

\[
Xr(v,u,l,k) = C(v)C(u)\sum_{m=0}^{N-1} \sum_{n=0}^{N-1} x(m,n,l,k)\Phi(v,u,m,n),
\]

where \( l=0,\ldots,K \), \( k=0,\ldots,K \) – block numbers in vertical and horizontal;
\( x(m,n,l,k) \) – source data set for block transform in the traditional form with fixed \( l \) and \( k \);
\( Xr(v,u,l,k) \) – result of block transform in the traditional form with fixed \( l \) and \( k \).

Block selection (3) from input two dimensional dataset \( x(m,n,l,k) \) is:

\[
x(m,n,l,k) = \begin{cases} 
    x(IN+m,kN+n) \\
    x(IN+m,(k+1)N+n) \\
    x((I+1)N+m,kN+n) \\
    x((I+1)N+m,(k+1)N+n)
\end{cases}
\]
The record will be done in a block

\[
X_{r}(v,u,l,k) = \begin{cases} 
X_{r}(lN + v, kN + u) \\
X_{r}(lN + v, (k + 1)N + u) \\
X_{r}((l + 1)N + v, kN + u) \\
X_{r}((l + 1)N + v, (k + 1)N + u)
\end{cases}
\]  

(5)

Restoration is performed in accordance with the expression:

\[
x(m,n,l,k) = \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} C(v)C(u)X_{r}(v,u,l,k) \Phi(v,u,m,n),
\]  

(6)

where result block is (4) and source block – (5).

Expression (3) of simulation result \( X_{r} \) with \( 2 \times 2 \) blocks is shown in figure 1, c. The test image has rotation invariant elements and its frequency representations do not have it. The frequency domain representations are inverted (here and below).

3. **Two block transform types**

2 types of changed transforms based on (3) are offered. Proposed block transform type 1 for 4 blocks is:

\[
x(m,n,l,k) = \begin{cases} 
x((l + 1)N - (m + 1), (k + 1)N - (n + 1)) \\
x((l + 1)N - (m + 1), (k + 1)N + n) \\
x((l + 1)N + m, (k + 1)N - (n + 1)) \\
x((l + 1)N + m, (k + 1)N + n)
\end{cases}
\]  

(7)

A placement will be made by similar groups (\( X_{r} \) is changed on \( X_{s1} \)):

\[
X_{s1}(v,u,l,k) = \begin{cases} 
X_{s1}((l + 1)N - (v + 1), (k + 1)N - (u + 1)) \\
X_{s1}((l + 1)N - (v + 1), (k + 1)N + u) \\
X_{s1}((l + 1)N + v, (k + 1)N - (u + 1)) \\
X_{s1}((l + 1)N + v, (k + 1)N + u)
\end{cases}
\]  

(8)

Frequency representation \( X_{s1} \) based on (3) with (7) and (8) in the DCT frequency domain is placed in figure 2, a.

**Figure 2.** Representations in the frequency domain on the proposed form.

The same approach is used for proposed direct \( X_{s2} \) and inverse block transform of type 2. Expression (3) is used for 4 block groups in accordance with:
\[
x(m,n,l,k) = \begin{cases}
  x(IN + m,kN + n) \\
x(IN + m,(k + 2)N - n) \\
x((l + 2)N - m,kN + n) \\
x((l + 2)N - m,(k + 2)N - n)
\end{cases}, \quad (9)
\]

The record will be made as follows:

\[
X_{22}(v,u,l,k) = \begin{cases}
  XS_2(IN + v,kN + u) \\
XS_2(IN + v,(k + 2)N - u) \\
XS_2((l + 2)N - v,kN + u) \\
XS_2((l + 2)N - v,(k + 2)N - u)
\end{cases}. \quad (10)
\]

Frequency representation \(X_{S1}\) based on (3) with (9) and (10) in the DCT frequency domain is placed in figure 2, b. A restoration will be performed according to expression (6) with the same substitutions.

Expressions (3) – (10) can be used for arbitrary images or two-dimensional functions, but the differences are most noticeable for images with the central symmetry. As can be seen from the figures 1 and 2, there is a significant difference in frequency representations, but only for the proposed block transforms of types 1 and 2, there are the rotation invariance elements for the same angles.

4. Results comparison

The expressions (4), (5), (7) – (10) determine data order and results placement. Low-frequency and high-frequency components will be in different places. Results comparison should be made with the same components location.

The block transform in traditional form \(X_T\) with the result placement in type 1 with proposed transform \(X_{S1}\) is calculated with (4) for blocks reading and (8) for blocks writing. There is a \(X_{TS1}\) variant. To obtain \(X_{TS2}\) (similarly described), it is recommended using (4) with corresponding record based on (10). In other case, the block type 1 transform is proposed to form \(X_{S1}\) with the result placement of traditional form \(X_T\) (7). There will be saving based on (5). Let us get a \(X_{TS1}\) variant. \(X_{S2}\) is calculated with (9) and (5). In addition, there are possible types: \(X_{S1S2}\) – reading (7), writing (10) and \(X_{S2S1}\) – reading (9), writing (8). The image of \(X_{S2S1}\) is shown in figure 2, c.

The differences between frequency representations of the test image are shown in figure 3.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure3.png}
\caption{Subtractions between test image representations in frequency domain.}
\end{figure}

Result \(X_{S1} - X_{TS1}\) is shown in figure 3, a. The exact coincidence of all the values is observed only in the 4 quadrant. The \(X_{S2} - X_{TS2}\) is shown in figure 3, b. Overlap is placed in the 1 quadrant. A maximum difference is presented between the block DCT in type 1 and type 2. Here in any quadrant, there is no complete correspondence of transform values.
5. Conclusion
The variants of the orthogonal block transforms execution are offered. The expressions are proposed with the selection of some basic functions. A relative block sizes affect the transform result. The block sampling and recording variants are described under performing a transform. Two variants of the block transforms execution with the rotation invariance elements are proposed. All the proposed expressions are simulated and the results are compared. Such approach to the frequency transforms can be used for any dimension. It works for any basic function set (symmetrical, anti-symmetrical, without symmetry). The proposed approach to transform implementation will lead to the reduction of calculations number.

Acknowledgments
The reported study was funded by RFBR according to the research project № 18-37-00181.

References
[1] Shen B and Sethi I K 1995 Inner-block operations on compressed images Proc. 3-rd ACM Int. Conf. on Multimedia (San Francisco) pp 489-498
[2] Park Y S and Park H W 2004 Design and analysis of an image resizing filter in the block-DCT domain IEEE Trans. Circuits Syst. Video Technol. 14 274–279
[3] Chiptrasert B and Rao K R 1990 Discrete cosine transform filtering Signal Processing 19 233-245
[4] Lee J B and Lee B G 1992 Transform domain filtering based on pipelining structure IEEE Transactions on Signal Processing 40 2061-64
[5] Ngo C W, Pong T C and Chin R T 2001 Exploiting image indexing techniques in DCT domain Pattern Recognition 34 1841-51
[6] Shneier M and Abdel-Mottaleb M. 1996 Exploiting the JPEG compression scheme for image retrieval IEEE Trans. Pattern Anal. Mach. Intell. 18 849-853
[7] Qian Z, Wang W and Quao T 2012 Int. Workshop on Inform. and Electronics Engineering pp. 344 – 348
[8] Kim B G, Shim J I and Park D J 2003 Fast image segmentation based on multi-resolution analysis and wavelets Pattern Recognition Letters 24 2995–06
[9] Shoberg A G, Sai S V 2015 The multiresolution analysis with radial symmetry elements Future Communication, Information and Computer Science ed D. Zheng (London: Tailor & Francis) 211-212
[10] Shoberg A 2015 Representation frequency transforms by matrix direct sum. Retrieved from: http://ieeexplore.ieee.org/document/7147288/