American Journal of Applied Sciences 10 (7): 664-668, 2013
ISSN: 1546-9239

doi:10.3844/ajassp.2013.664.668 Published Online 10 (7) 2013 (http://www.thescipub.com/ajas.toc)

A Non-Iterative Method for Factorization of Positive Matrix in Discrete Wavelet Transform Based Image Compression

Nallathai Perumalsamy and Nithyanandam Natarajan

Department of Electronics and Communication Engineering,
B.S. Abdur Rahman University, Chennai, India

Received 2013-05-21, Revised 2013-06-17; Accepted 2013-07-01

ABSTRACT

A non-iterative method of factorizing a 4×4 positive matrix, with the application to image compression is explained using an example. The procedure is applied to all the 4096 number of 4×4 pixel sub-blocks of a 256×256 image for compression. The proposed compression technique can be applied to the Discrete Wavelet Transform (DWT) coefficients of the test image. The 16 Pixel Intensity Values (PIV) or their DWT coefficients of a 4×4 pixels sub-block of the image can be represented by the outer product of a 4×1 column matrix and a 1×4 row matrix, with Least Mean Square Error (LMSE) criterion. Hence, instead of transmitting the 16 PIVs or their DWT coefficients, the values of the 4 elements of the column matrix and the 4 elements of the row matrix alone are transmitted resulting in a maximum compression ratio of 2 (16/4+4). The receiver can recreate the 4×4 pixels sub-block or their DWT coefficients, by calculating the outer products of 4 values of column matrix with 4 values of row matrix. In case of DWT coefficients inverse DWT is applied to recreate the pixels. This principle is extended to all the sub-blocks of the 256×256 image to compress and later reconstruct the image.

Keywords: DWT, Lossy Image Compression, Positive Matrix Factorization, PSNR

1. INTRODUCTION

Data compression for fast transmission with minimum error is desirable to save data transmission time and data storage requirements, two of the important parameters of any data processing system. The above requirements are more significant in image data processing. A number of image compression methods based on outer product expansion and tensor decomposition have been proposed in the past (O'Leary and Peleg, 1983; Tucker, 1996; Kolda and Bader, 2009; Welling and Weber, 2001; Cichocki *et al*., 2011; Karami *et al*., 2012).

Let us consider a 256×256 monochrome image, which is normally divided into blocks of 8×8 pixels for processing. The gray level intensities of the pixels will range from the minimum of black to the maximum of white. Assuming 8 bits are used to represent the gray levels, we have 256 levels. In this study, for the convenience of explanation we consider a sub-block of 4×4 pixels with 16 gray scale intensities numbered from 1 to 16. Thus the 16 PIVs of the sub image are in the range from 1 to 16. Normally either the 16 PIVs of each and every sub-image of an image or their Harr wavelet based DWT coefficients are to be transmitted as such for lossless transmission. In either case we shall refer them as 16 Numerical Values (NV). If the 16 NVs of a sub-image are represented as a 4×4 matrix, it will be a positive

Corresponding Author: Nallathai Perumalsamy, Department of Electronics and Communication Engineering, B.S. Abdur Rahman University, Chennai, India

matrix, in which all the 16 elements will have positive values. Using a method of factorization, explained in 2.1, the 4×4 matrix can be represented as the outer product of one 4×1 column matrix and one 1×4 row matrix. In most cases a factorization may not be exact and hence a best match approximation based on Least Mean Square Error (LMSE) criterion is adopted. After factorization the 4 elements of the column matrix along with the 4 elements of the row matrix alone are transmitted. At the receiving end the best match of 16NVs are estimated as the outer products of 4 elements of the column matrix and the 4 elements of the row matrix. This will result in a maximum compression ratio of 2 (16/4+4). Because of the approximation in factorization, the compression is lossy with LMSE.

2. MATERIALS AND METHODS

This section deals about the matrix factorization method for positive matrix of size 4×4 with numerical example and generalized matrix factorization.

2.1. Numerical Example for Positive Matrix Factorization of 4×4 Matrix

Step1: 4×4 sub-block

Either the intensity values of the 16 pixels of the 4×4 sub image or their DWT coefficients, commonly called as Numerical Values (NVs), forming a 4×4 matrix are represented in a 4×4 grid as shown in **Fig. 1**. The sum of the NVs in each row and column are shown separately. The highest sum among these eight sum values is identified as 43 corresponding to the second column:

Step 2: The sixteen NVs of the sub image are to be factorized as the outer products of 4 values of row factors (x_1,x_2,x_3,x_4) and 4 values of column factors (y_1,y_2,y_3,y_4). Normally, after factorization the NVs of 2^{nd} column which has the maximum column sum is likely to contribute maximum error. To avoid this error, the NVs of column 2 are taken as the row factors (x_1,x_2,x_3,x_4) and y_2 is taken as 1, resulting in zero error from the 4 NVs of this column

Step 3: The unknown column factors y_1,y_3 and y_4 are estimated in such a way that the estimated product NVs of corresponding columns 1,3 and 4 result in minimum error in their respective columns. For example, considering column 1, the estimated squared error will be $[(5-12y_1)^2 + (8-2y_1)^2 + (7-14y_1)^2 + (4-15y_1)^2]$. Differentiating this squared error with respect to y_1 and equating the differentiated expression to zero we get the value of y_1 for minimum squared error in column 1. Thus the value of y_1 is calculated to be 0.4112 giving a minimum of the squared error. Likewise the value of y_3 and y_4 are calculated to be 0.4059 and 0.8189 respectively to have minimum squared error in column 3 and column 4:

$x_1 = 12$; $x_2 = 2$; $x_3 = 14$; $x_4 = 15$

$Y_1 = (5\times12)+ (8\times2)+(7\times14)+ (4\times15)/(12^2 + 2^2+14^2 +15^2) = 234/569 = 0.4112$

$Y_2 = (12\times12)+ (2\times2)+(14\times14)+ (15\times 15)/(12^2 + 2^2+14^2 +15^2)= 569/569 = 1$

$Y_3 = (7\times12)+ (9\times2)+(6X14)+ (3\times 15)/(12^2 + 2^2+14^2 +15^2) = 231/569 = 0.4059$

$Y_4 = (11\times12)+ (1\times2)+(13\times14)+ (10\times 15)/(12^2 + 2^2+14^2 +15^2) = 466/569 = 0.8189$

The row factor $x_1(12)$, $x_2(2)$, $x_3(14)$ and $x_4(15)$, along with the column factors of y_1 (0.4112), $y_2(1)$, $y_3(0.4059)$ and $y_4(0.8189)$ are transmitted to the receiver **Fig. 2**. The receiver will calculate the 16 outer products of the 4 number of row factors and 4 number of column factors. These 16 values are the reconstructed pixels or their DWT coefficients at the receiving end. Inverse DWT is applied to the DWT coefficients to obtain the pixel intensity values:

Step 4: Reconstructed 4×4 sub-block with original NVs shown in brackets **Fig. 3**

Step 5: RMSE Calculation: Making use of the 16 numbers of original pixel intensities at the transmitting end and the 16 values of intensity values are reconstructed at the receiving end, the RMSE is calculated:

$$RMSE = \{[(5- 4.9344)^2 + (12-12)^2 + (7-4.8708)^2 +(11-9.8268)^2 + (8-0.8224)^2 + (2-2)^2 + (9-0.8118)^2 + (1-1.6378)^2 + (7-5.7568)^2 + (14-14)^2 + (6-5.6826)^2 + (13-11.4646)^2 + (4-6.1680)^2 + (15-15)^2 + (3-6.0885)^2 + (10-12.2835)^2] / 16 \}^{1/2} = 3.0448$$

In 2.2, we describe the procedure to factorize a 4×4 general matrix.

Column factor

Row factor	Y_1	Y_2	Y_3	Y_4	Row sum
X_1	5	12	7	11	35
X_2	8	2	9	1	20
X_3	7	14	6	13	40
X_4	4	15	3	10	32
Column sum	24	43	25	35	

Fig. 1. 16 NVs of a sub-block of 4×4 pixels shown in a grid form

	Y_1	1	Y_3	Y_4
12	5	12	7	11
2	8	2	9	1
14	7	14	6	13
15	4	15	3	10

Fig. 2. Choosing the row factor values

Column factor

Row factor	0.4112	1	0.4059	0.8189
12	4.9344 (5)	12 (12)	4.8708 (7)	9.8268 (11)
2	0.8224 (8)	2 (2)	0.8118 (9)	1.6378 (1)
14	5.7568 (7)	14 (14)	5.6826 (6)	11.4646 (13)
15	6.1680 (4)	15 (15)	6.0885 (3)	12.2835 (10)

Fig. 3. Reconstructed NVs along with original values for comparison

2.2. The General Procedure for Factorization with Minimum RMSE

Step 1: Let us consider a 4×4 sub-block of an image with NVs I_{11}, I_{12}, ….. I_{44} as shown in the **Fig. 4** below. The sum S_{R1}, S_{R2}, S_{R3} and S_{R4} of the 4 rows of the NVs of the sub image and the sum S_{C1}, S_{C2}, S_{C3} and S_{C4} of the NVS of the 4 columns of the sub image are calculated and indicated as shown in the Figure. The 16 NVs are to be factorized as the outer products of 4 row factors (x_1, x_2, x_3, x_4) and 4 column factors (y_1, y_2, y_3, y_4). Therefore, ideally $I_{11} = x_1y_1$; $I_{12} = x_1y_2$ …… $I_{44} = x_4y_4$. However, in practice the factorization may not be exact and hence the factorization should be optimized resulting in minimum error. The minimum error is estimated as the Least Mean Squared error.

Step 2: Let us assume that the estimated row factors (x_1, x_2, x_3, x_4) and the estimated column factors (y_1, y_2, y_3, y_4) result in $x_1y_1 = \bar{I}_{11}$, $x_1y_2 = \bar{I}_{12}$ . . . . $x_4y_4 = \bar{I}_{44}$. Hence sum of the squared errors of the 16 NVs will be equal to $[(I_{11} - \bar{I}_{11})^2 + (I_{12} - \bar{I}_{12})^2 + ... + (I_{44} - \bar{I}_{44})^2]$. Considering the logic that the larger values of NVs are likely to contribute large errors, we identify the largest value among the S_{R1}, S_{R2}, S_{R3}, S_{R4}, S_{C1}, S_{C2}, S_{C3} and S_{C4}. For example let S_{C2} be the largest sum. To have no error in this column 2, it is taken that $x_1 = I_{12}$, $x_2 = I_{22}$, $x_3 = I_{32}$, $x_4 = I_{42}$ and $y_2 = 1$. The remaining column factors y_1, y_3 and y_4 are estimated in such a way as to give minimum sum of squared errors in their respective columns 1, 3 and 4.

Step 3: We shall now illustrate the method of estimating y_1 and the same procedure will be adopted to estimate y_3 and y_4 column 1:

$$\text{error}^2 = E = [(I_{11} - \bar{I}_{11})^2 + (I_{21} - \bar{I}_{21})^2 + (I_{31} - \bar{I}_{31})^2 + (I_{41} - \bar{I}_{41})^2]$$
$$= [(I_{11} - x_1y_1)^2 + (I_{21} - x_2y_1)^2 + (I_{31} - x_3y_1)^2 + (I_{41} - x_4y_1)^2]$$
$$= [(I_{11} - I_{12}y_1)^2 + (I_{21} - I_{22}y_1)^2 + (I_{31} - I_{32}y_1)^2 + (I_{41} - I_{42}y_1)^2]$$

Differentiating E with respect to y_1 and equating it to zero we get:

$$\frac{dE}{dy1} 2[(I_{11} - I_{12}y_1)(- I_{12}) + (I_{21} - I_{22}y_1)(- I_{22}) + (I_{31} - I_{32}y_1)(- I_{32}) + (I_{41} - I_{42}y_1)(- I_{42})] = 0$$

Solving this equation we get:

$$Y_1 = [I_{11}\, I_{12} + I_{21}\, I_{22} + I_{31}\, I_{32} + I_{41}\, I_{42}]/[(I_{12})^2 + (I_{22})^2 + (I_{32})^2 + (I_{42})^2]$$

Similarly:

$Y_3 = [I_{13}\ I_{12}+I_{23}\ I_{22}+I_{33}\ I_{32}+I_{43}\ I_{42}]/[(I_{12})^2+(I_{22})^2+(I_{32})^2 + (I_{42})^2]Y_4 = [I_{14}\ I_{12}+I_{24}\ I_{22}+I_{34}\ I_{32}+I_{44}\ I_{42}]/[(I_{12})^2+(I_{22})^2+(I_{32})^2 + (I_{42})^2]$

For image compression instead of transmitting the 16 NVs of I_{11}, I_{12}….. I_{44}, the 4 values of row factors (x_1, x_2, x_3, x_4) and the 4 values of column factors (y_1, y_2, y_3, y_4) are transmitted, using which the receiver is ble to estimate the 16 NVs $\bar{I}_{11}$, $\bar{I}_{12}$ . . . $\bar{I}_{44}$. Of these 16 values the four values corresponding to column 2 will be with no error since $\bar{I}_{12} = I_{12}$; $\bar{I}_{22} = I_{22}$; $\bar{I}_{32} = I_{32}$; $\bar{I}_{42} = I_{42}$. The remaining 12 NVs alone will contribute to the error. The estimation of error is possible only at the transmitting end, which has the 16 original NVs $I_{11,}$ $I_{12,}$ ….. I_{44} and the 16 estimated NVs $\bar{I}_{11}$, $\bar{I}_{12}$ . . . $\bar{I}_{44}$ which are the outer products of row factors (x_1, x_2, x_3, x_4) and the column factors (y_1, y_2, y_3, y_4).

The compression ratio is 2, since 8 values of factors are transmitted instead of 16 NVs.

Column factor

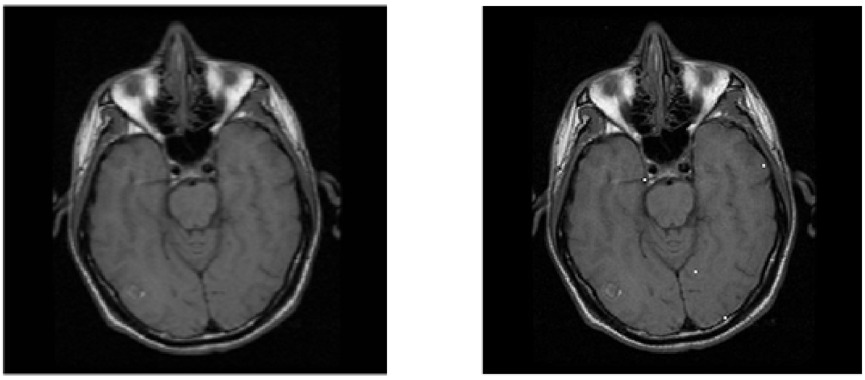
Row factor		Y_1	Y_2	Y_3	Y_4		Row factor
	X_1	I_{11}	I_{12}	I_{13}	I_{14}	S_{R1}	
	X_2	I_{21}	I_{22}	I_{23}	I_{24}	S_{R2}	
	X_3	I_{31}	I_{32}	I_{33}	I_{34}	S_{R3}	
	X_4	I_{41}	I_{42}	I_{43}	I_{44}	S_{R4}	
		S_{C1}	S_{C2}	S_{C3}	S_{C4}		

Column factor

Fig. 4. Sub-block of 4x4 pixels with general NVs

(a) (b)

Fig. 5. (a). Original image (b). Reconstructed image



(a) (b)

Fig. 6. (a). Original image (b). Reconstructed image

(a) (b)

Fig. 7. (a). Original image (b). Reconstructed image

Table 1. Processing parameters

	Pixel intensity based compression		DWT of the pixel intensity based compression	
Name of the image	PSNR (dB)	Time (sec)	PSNR (dB)	Time (sec)
Cameraman (256×256)	10.4066	0.6432	45.3712	0.9752
Barbara (512×512)	17.7321	2.2556	56.8010	3.4945
MRI (512×512)	23.4940	1.9883	71.4597	3.1107

3. RESULTS AND DISCUSSION

The proposed method of image compression based on factorization, as explained in the previous sections, is applied to a set of images, using MATLAB. The results are shown below in **Fig. 5-7**. The comparative analysis of the PSNR and the processing time values are listed in **Table 1**.

4. CONCLUSION

Based on the reconstructed images it is observed that the DWT based compression is better in terms of increased PSNR. The approximation in the factorization process results in noisy patches in the reproduced image which can be minimized by suitable filters.

5. REFERENCES

Cichocki, A., S. Cruses and S. Amari, 2011. Generalized alpha-beta divergences and their application to robust nonnegative matrix factorization. Entropy, 13: 134-170. DOI: 10.3390/e13010134

Karami, A., M. Yazdi and G. Mercier, 2012. Compression of hyperspectral images using discerete wavelet transform and tucker decomposition. IEEE J. Selected Topics App. Earth Observ. Remote Sens., 5: 444-450. DOI: 10.1109/JSTARS.2012.2189200

Kolda, T.G. and B.W. Bader, 2009. Tensor decompositions and applications. SIAM Rev., 51: 455-500. DOI: 10.1137/07070111X

O'Leary, D.P. and S. Peleg, 1983. Digital image compression by outer product expansion. IEEE Trans. Commun., 31: 441-444. DOI: 10.1109/TCOM.1983.1095823

Tucker, L.R., 1996. Some mathematical notes on three-mode factor analysis. Psychometrika, 31: 279-311. DOI: 10.1007/BF02289464

Welling, M. and M. Weber, 2001. Positive tensor factorization. Patt. Recogn. Lett., 22: 1255-1261. DOI: 10.1016/S0167-8655(01)00070-8