Lossless Image Compression With IDP Decomposition

R. Kountchev[1], Vl. Todorov[2], R. Kountcheva[3]

Abstract - **A recursive algorithm for lossless compression of halftone and colour images, in correspondence with the decomposition method INVERSE DIFFERENCE PYRAMID (IDP), based on two-dimensional orthogonal transforms, is described in this paper. Special investigation was performed on the results obtained with highest quality or lossless coding applied for different kinds of images: natural, graphics, medical, etc. The results are compared with those for JPEG and JPEG2000.**

Keywords - **lossless image compression, inverse difference pyramid decomposition.**

I. INTRODUCTION

In the paper are presented the specific features of the Inverse Difference Pyramid decomposition [1,2], used for lossless still image compression. In the section II are presented the basic principles of the method. In the section III are given the results, obtained with the IDP lossless compression for medical ultrasound images, natural grayscale and colour images and graphics, and are pointed some of its main advantages, compared with the widely used standards JPEG and JPEG2000 [3,4]. In the section IV are presented the most efficient trends for future method development.

II. BASIC PRINCIPLES OF THE IDP DECOMPOSITION

The Inverse Difference Pyramid (IDP) decomposition is presented in brief, as follows. The basic approach is that the matrix [X] of the original image is divided in sub-images with size $2^n\times 2^n$ and each is processed with a two-dimensional (2D) orthogonal transform using only a limited number of spectrum coefficients. The values of the coefficients, calculated in result of the transform, constitute the first pyramid level. Using the values of these coefficients, the sub-image is restored with the corresponding inverse orthogonal transform and subtracted pixel by pixel from the original one. The difference sub-image with elements $e_p(i,k)$ in the level p of the IDP is defined as:

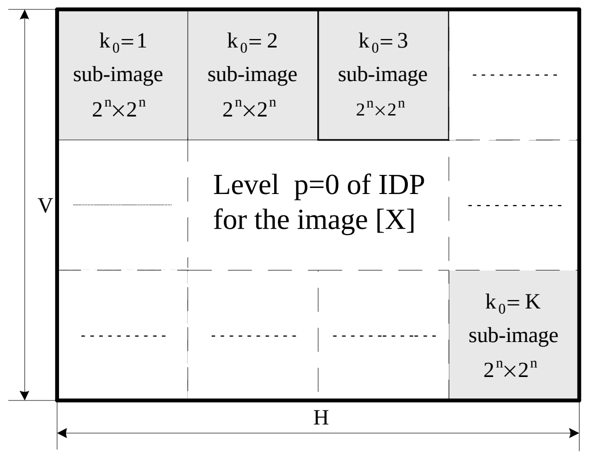
$$e_p(i,k)=\begin{cases} x(i,k)-\tilde{x}(i,k) & \text{for} \quad p=0; \\ e_{p-1}(i,k)-\tilde{e}_{p-1}(i,k) & \text{for} \ \ p=1,2,..,P, \end{cases} \qquad (1)$$

where x(i,k) is the pixel (i,k) in a sub-image with size $2^n\times 2^n$ of the input image [X] (Fig.1a); $\tilde{x}(i,k)$ and $\tilde{e}_{p-1}(i,k)$ are the pixels of the recovered input and difference sub-images in the level p of the IDP, obtained with the inverse orthogonal transform using the selected spectrum coefficients only. The obtained difference sub-image is divided in 4 sub-images with size $2^{n-1}\times 2^{n-1}$. Each sub-image is processed with the 2D orthogonal transform again; the values of the transform coefficients build the second pyramid level. Then the image is restored again and the second difference image is calculated. The process continues in a similar way with the next pyramid levels. In this way all pyramid levels, consisting of coefficients values only, are calculated. The processing of the image is shown in Fig. 1.

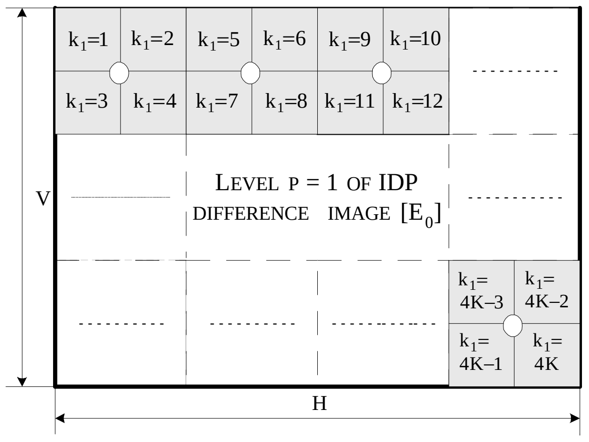
Roumen Kountchev is with the Faculty of Telecommunications, Technical University of Sofia, Boul. Kl. Ohridsky, 8, Sofia 1000, Bulgaria. E-mail: rkountch@tu-sofia.bg

Vladimir Todorov is with T&K Engineering, Mladost 3, P.O.Box 12, Sofia 1712, Bulgaria. E-mail: vtodorov@yahoo.com

Roumiana Kountcheva is with T&K Engineering, Mladost 3, P.O.Box 12, Sofia 1712, Bulgaria. E-mail: rkountcheva@yahoo.com


$k_0=1$ sub-image $2^n\times 2^n$
$k_0=2$ sub-image $2^n\times 2^n$
$k_0=3$ sub-image $2^n\times 2^n$
Level p=0 of IDP for the image [X]
$k_0=K$ sub-image $2^n\times 2^n$
V
H

a. The original image [X], divided in K sub-images ($2^n\times 2^n$) for pyramid level p=0.


$k_1=1$ $k_1=2$ $k_1=5$ $k_1=6$ $k_1=9$ $k_1=10$
$k_1=3$ $k_1=4$ $k_1=7$ $k_1=8$ $k_1=11$ $k_1=12$
LEVEL P = 1 OF IDP DIFFERENCE IMAGE $[E_0]$
$k_1=4K-3$ $k_1=4K-2$ $k_1=4K-1$ $k_1=4K$
V
H

b. Each sub-image of difference image $[E_0]$ the level p=0, is divided in 4 sub-images ($2^{n-1}\times 2^{n-1}$) in the pyramid level p=1

Fig.1. The IDP levels p=0,1 for the image with HxV pixels.

Usually the processing does not require all the pyramid levels to be used, because the necessary image quality is obtained earlier, in one of the lower levels. Such pyramid is called “truncated”. The only requirement for the case of lossless image coding is that in the last pyramid level all the possible coefficients should be used.

The approximation models of the input or difference image in the level p could be represented with the relations:

$$\tilde{x}(i,k)/\tilde{e}_{p-1}(i,k) = IT[y_p(u,v)], \qquad (2)$$

$$y_p(u,v) = T[x(i,k)/e_{p-1}(i,k)]$$

where T[•] is the operator for the “truncated” direct two-dimensional linear transform applied on the input block with size $2^n \times 2^n$, or on the difference sub-image with size $2^{n-p} \times 2^{n-p}$ from the pyramid level p=1,2,..,P (Fig. 2b); IT[•] is the operator for the inverse linear transform of the spectrum coefficients, $y_p(u,v)$ from the level p of the “truncated” transform $2^{n-p} \times 2^{n-p}$, obtained in result of the transformation of every ¼ part of the difference sub-image, $e_{p-1}(i,k)$.

Specific for IDP is that the set of coefficients of the orthogonal transform, chosen for every pyramid level, can be different. The coefficients obtained in result of the orthogonal transform from all pyramid levels are sorted in accordance with their frequency, and scanned sequentially. The obtained one-dimensional massif of spectrum coefficients for the s-th frequency band of the two-dimensional linear transform of the input or of the difference image for the level p of the IDP is represented with:

$$y_p(s) = y_p[u=\varphi(s), v=\psi(s)], \qquad (3)$$

where $u=\varphi(s)$ and $v=\psi(s)$ are the functions, which define the transformation for the two-dimensional massif of spectrum coefficients in the s-th frequency band for the level p. In order to achieve higher compression ratio the data is processed, applying adaptive entropy and run-length coding, performed in two steps: adaptive coding of the lengths of the series of equal symbols (run-length encoding, RLE), and adaptive coding with modified Huffman code (HE). The compressed and decompressed data of the one-dimensional massif of spectrum coefficients for the s-th spectrum band in the level p is presented as follows:

$$\alpha_p(s) = RLE/HE[y_p(s)]; \quad y_p(s) = RLD/HD[\alpha_p(s)] \qquad (4)$$

where RLE/HE[•] and RLD/HD[•] are operators for entropy coding and decoding with Run-Length and Huffman coding.

The recovered sub-image x(i,k) is calculated recursively from the components of all the (P+1) IDP levels:

$$x(i,k) = \tilde{x}(i,k) + \sum_{r=1}^{P} \tilde{e}_{r-1}(i,k). \qquad (5)$$

In the start of the arranged data sequence is inserted a special header, which contains information about the number of the used pyramid levels, the retained “meaning” coefficients, the kind of the selected orthogonal transform for every level (usually DCT or WHT), the kind of the lossless coding, etc. The process of the image decompression is performed in reverse order. The block diagram of the described IDP algorithm is shown in Fig. 2.

III. Results and Evaluation

The experiments were performed for the following basic image kinds: graphics, text, medical ultrasound images, natural grayscale images, fingerprints, and colour images. In the investigation were used hundreds of images, but in TABLE 1 are presented only a part of the results, which show the specific features of the method behavior best. The used orthogonal transform was Walsh-Hadamard (WHT). The most typical images, used as examples and shown in Fig.3 were as follows: *Graphics*: Circles, Squares, Crosses; *Text images*: Text1, Text2, Text5; *Medical images*: Coronal, Axial, Cells; *Natural grayscale images*: Lena, Peppers, Souliers; *Uniform Color images*: Yellow, Blue, Green, White.

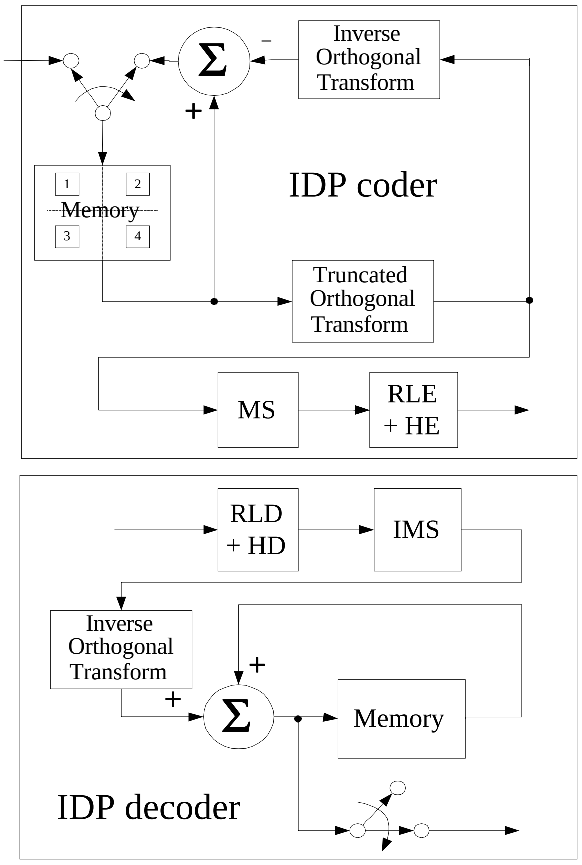
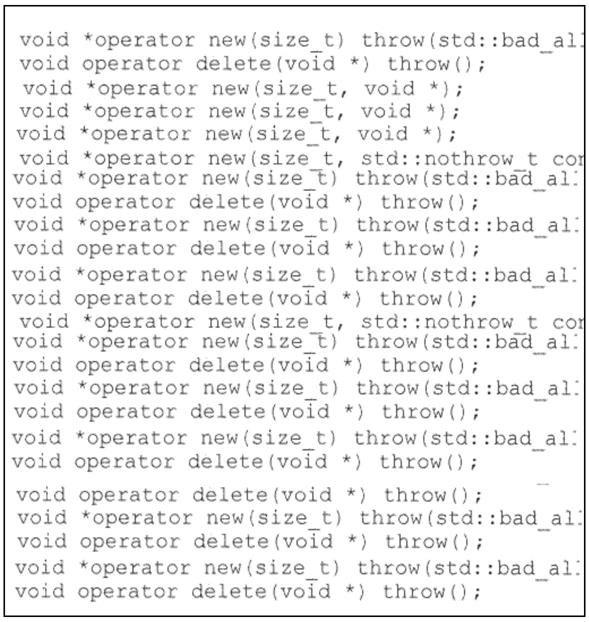

Σ
Inverse Orthogonal Transform
–
+
1
2
Memory
3
4
IDP coder
Truncated Orthogonal Transform
MS
RLE + HE
RLD + HD
IMS
Inverse Orthogonal Transform
+
+
Σ
Memory
IDP decoder

Fig. 2. Block diagram of the IDP algorithm: MS and IMS are the blocks for direct and inverse processing of the two-dimensional data massif consisting of coefficients’ values; M is the memory, and (RLD+HD) and (RLE+HE) are the blocks for decoding and coding with RLE and Huffman code.

Fig.3a. Circles (256x256, 8bpp) b. Squares (256x256, 8 bpp)



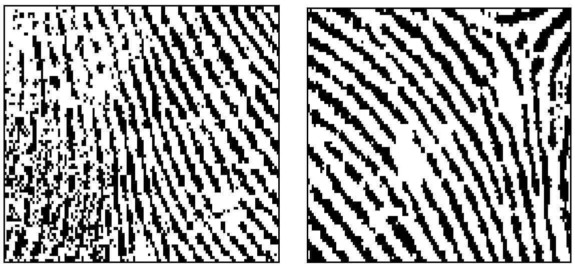
```
void *operator new(size_t) throw(std::bad_al
void operator delete(void *) throw();
void *operator new(size_t, void *);
void *operator new(size_t, void *);
void *operator new(size_t, void *);
void *operator new(size_t, std::nothrow_t co
void *operator new(size_t) throw(std::bad_al
void operator delete(void *) throw();
void *operator new(size_t) throw(std::bad_al
void operator delete(void *) throw();
void *operator new(size_t) throw(std::bad_al
void operator delete(void *) throw();
void *operator new(size_t, std::nothrow_t co
void *operator new(size_t) throw(std::bad_al
void operator delete(void *) throw();
void *operator new(size_t) throw(std::bad_al
void operator delete(void *) throw();
void *operator new(size_t) throw(std::bad_al
void operator delete(void *) throw();

void operator delete(void *) throw();
void *operator new(size_t) throw(std::bad_al
void operator delete(void *) throw();
void *operator new(size_t) throw(std::bad_al
void operator delete(void *) throw();
```

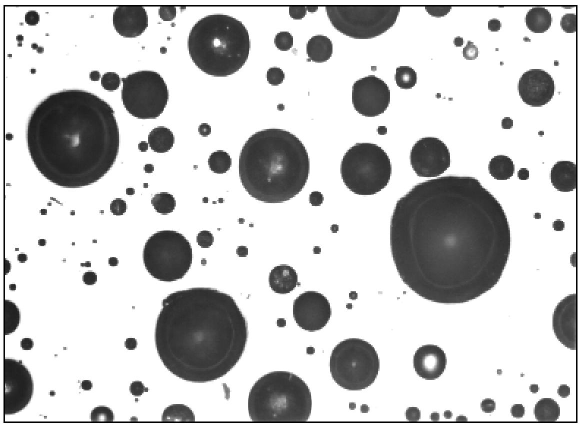
c. Text 2 (512x512, 8bpp)

d. Peppers ((512x512, 8bpp)

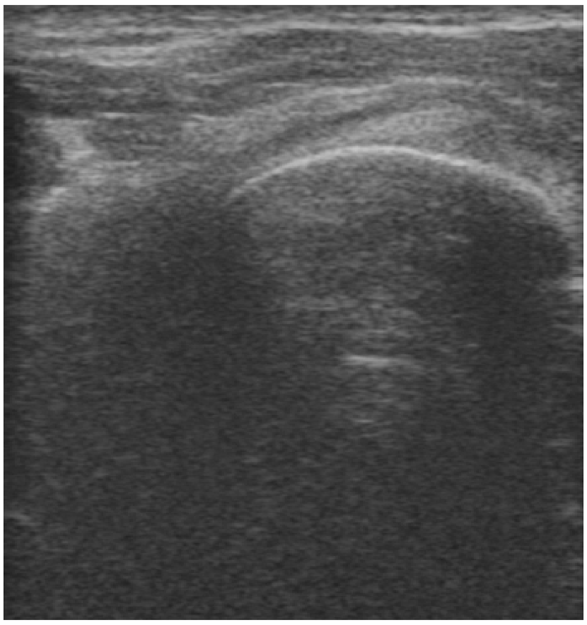
e. Souliers (551x369, 8 bpp)



f. Fingerprint 2 and Fingerprint 3 (128x88, 8 bpp each)



g. Cells (390x277, 8 bpp)



h. Coronal (350x432, 24bpp)

Fig.3, a - h. Example test images.

This investigation was done only for lossless compression. The compression with JPEG was performed with MS Photo Editor, and for JPEG2000 was used LuraTech Algovision. With MS Photo Editor is impossible to perform lossless image compression, but the results show that for best quality compressions the image quality for JPEG was worse than that obtained with IDP.

The results, presented in TABLE 1, illustrate the following specific features and advantages of the method:

- The IDP method offers much higher lossless compression for graphic images. All text images were compressed with unchanged quality and the compression ratio was higher than that, obtained with JPEG and JPEG2000.
- Very good results were obtained for uniform color examples (images Yellow, Blue, Green-1024x1024 pixels). In these cases, the compression ratio for lossless IDP was extremely high.
- The comparison with JPEG standard shows better results for IDP lossless compression. In some cases, the highest quality with JPEG lossless compression resulted even in enlarging of the image file (pictures Text1, Fingerpr2 and Fingpr3).
- The IDP method has lower computational complexity than JPEG2000.
- For natural images (peppers, Lena, etc.) the results, obtained with IDP, are worse than these with JPEG and JPEG2000.

IV. FUTURE METHOD DEVELOPMENT

In case, when the IDP method is used for surveillance applications, the basic processing should be made more flexible, setting regions of interest. For this purpose, the sub-images of the picture should be processed with different masks, in order to select the most suitable set of coefficients, and to increase the compression efficiency.

- Special stress will be put on the lossless compression and data base management for fingerprints and graphics.
- The compression method permits the insertion of digital watermark in every pyramid level. This makes the application area of the method very wide, including multimedia contents protection and authorized distribution check, performed together with the compression.

ACKNOWLEDGEMENT

The authors thank the National Fund for Scientific Research of the Bulgarian Ministry of Education and Science for the financial support. (Contract No. I 1304/2003).

REFERENCES

[1] R. Kountchev, V. Haese-Coat, J. Ronsin. Inverse Pyramidal Decomposition with multiple DCT. Signal Processing: Image Communication, Vol. 17, January 2002, pp. 201-218.

[2] R. Kountchev. Image Compression with Recursive IDP Algorithm. ICEST 2003, 16-18 Oct. Sofia, Bulgaria, pp. 273-276.

[3] G. Wallace. The JPEG Still Image Compression Standard. IEEE Trans. Consumer Electronics, Vol. 38, No.1, Feb. 1992.

[4] M. Rabbani, R. Joshi. An Overview of the JPEG 2000 Still Image Compression Standard. Signal Processing: Image Communication, Vol.17, Jan.2002, pp.3-48.

TABLE 1

Image	IDP lossless	IDP PSNR[dB]	JPEG HQ Compr.	JPEG HQ PSNR[dB]	JPEG2000 compr.	JPEG2000 PSNR[dB]
Coronal	5.11	Infinity	5.59	58.48	6.62	Infinity
Axial1	5.35	Inf.	5.73	58.43	6.67	Inf.
Axial2	5.89	Inf.	5.70	18.52	6.56	Inf.
Text5	12.87	Inf.	2.50	65.96	4.31	Inf.
Text1	13.46	Inf.	0.85	60.83	1.89	Inf.
Text2	7.54	Inf.	1.19	62.79	2.25	Inf.
Squares	468.11	Inf.	55.02	Inf.	43.92	Inf.
Crosses	42.17	Inf.	3.13	67.11	7.45	Inf.
Circles	46.38	Inf.	4.85	68.15	8.57	Inf.
Cells	2.97	Inf.	1.95	62.43	2.74	Inf.
Souliers	1.74	Inf.	2.29	58.58	2.53	Inf.
Fingerp3	6.15	Inf.	0.57	63.32	12.09	Inf.
Fingerpr2	4.97	Inf.	0.55	63.02	9.24	Inf.
Peppers	1.36	Inf.	1.59	58.51	1.80	Inf.
Lena	1,21	Inf.	1,47	58,45	1.69	Inf.
Yellow	27594.00	Inf.	61.62	Inf.	3972.00	Inf.
Blue	27594.00	Inf.	61.62	Inf.	3972.00	Inf.
Green	27594.00	Inf.	61.62	Inf.	3972.00	Inf.

TABLE 1. Results from the compression with IDP, JPEG and JPEG 2000 (LuraTech)