A Polar-Fourier-Wavelet Transform for Effective CBIR

Dimo T. Dimov

Institute of Information Technologies, Bulgarian Academy of Sciences,
Acad.G.Bonchev Str., Block 29-A, 1113 Sofia, Bulgaria
dtdim@iinf.bas.bg

Abstract. Problems of effective rapid and reliable CBIR (Content Based Image Retrieval) are discussed in the paper, generally concerning the so called "primary" CBIR, i.e. the image query by example into a large database of images of hallmark type. A novel Polar-Fourier-Wavelet transform (PFWT) is proposed herein, allowing both high invariance towards accidental input image rotation and low correlation among the transform coefficients, the latter expected significantly to reduce the retrieval errors. Beside the theoretical aspects of the PFWT, result analysis is reported on tests over three IDBs from the real practice.

Keywords: Content Based Image Retrieval (CBIR), Image Databases (IDB), Polar mapping of images, Fourier transform, Wavelet transform, Polar-Fourier-Wavelet-Transform (PFWT).

1 Introduction

The state of the art in CBIR (Content Based Image Retrieval) is often referred to as "early" because it is based mainly on the extraction of the image primary features that are used for similar image retrieval from a database of images (IDB) [5], [9]. Thus the early CBIR is expected to overcome the current "semantic gap" on the path from low-level features towards the (user preferred) integrated semantic concepts over them [9], i.e. to become a natural base for further development to the "logical" and/or "abstract" CBIR [5].

There are many CBIR problems, the solution to which requires a rapid and reliable basic access method into an IDB. A classical task in this sense is the image query-by-example into an IDB of hallmarks [3], [5]. A promising approach to this task's solution is the generation of adequate image descriptors to be used as keys for fast index search into the IDB [3], [4], [10]. The approach is well applicable to the great amount of large-scale IDBs already created by conventional DB management tools [3], [4], [5], [9].

The paper defines a novel Polar-Fourier-Wavelet transform (PFWT), which not only provides emphasis to the essential image content and to the desirable high invariance towards accidental image rotation at input, but also ensures a lower correlation among transform coefficients than what the known transforms do. The

latter contributes significantly to the noise tolerance of CBIR. Beside the theoretical aspects of PFWT, the paper also analyzes the PFWT experimental results as of the CBIR practice of the Bulgarian Patent Office (PORB), with their large IDB of professional and/or service marks.

2 Problem Description in Brief

The needed processing of a given image $g(x,y)$ to obtain a CBIR descriptor D can generally be described by the following 3 steps [3], [4], [10]:

- Transform $g(x,y)$ to underline the image significant features: T: $g(x,y) \rightarrow g_T(f)$, where $g_T(f)$ can be considered as a field of potentials for the emphasized feature set f, or even a function over the emphasized feature vector space f.
- Scan $g_T(f)$ into a string and cut off this string to a definite length L to obtain a descriptor $D=(d_0, d_1,\ldots d_{L-1})$ of the initial image.
- Use D as a key for (fast) access into the given IDB. It can be considered that the features' order of D corresponds to a given (user's) interpretation of their significance "$\succ$" for the initial image description, i.e. $(d_0) \succ (d_1) \succ \ldots \succ (d_{L-1})$.

Assigning of the "classical" 2DFT (2-Dimensional Fourier Transform) to T results into a not bad solution to the problem of fast CBIR access to images of given IDB. Unfortunately, this solution is highly dependent on the accidental/random rotation of the input query image and often causes (unacceptable) errors of CBIR, [4], [10].

A possible approach addressing the rotational (and scale) invariance lies in the preliminary evaluation of the accidental rotation (and/or scale) at the input and its (their) back compensation. But this usually leads to a heavy and unreliable software performance.

The more apt and widely spread approach to the above issue is to subject the image to preliminary LogPMT (Logarithmic Polar Mapping Transform), by which the rotation and scale are converted into translations, [8].

Additionally, the logarithmic part of LogPMT can be safely omitted since it is ineffective when applied to non-isotropic scale, i.e. different along both axes ($0x$) and ($0y$). Thus, in practice, the simpler PMT is commonly used, and the compensation of accidental scale is usually left to the above mentioned input preprocessing.

In a similar fashion, the combination of PMT and 2DFT can lead to the so-called "Generic Fourier Descriptors for Object-Based Image Retrieval" of [10]. We will name this "classical" combination as P2DFT and will consider the hereafter proposed PFWT (Polar-Fourier-Wavelet-Transform) as an improving modification [3], [4] to the former [10].

3 The Proposed Image Transform PFWT

The proposed PFWT can be presented by the following algorithm of 4 steps:

3.1 The PFWT algorithm

PFWT step 1. Polar mapping of the given image $g(x,y)$, P: $g(x,y) \rightarrow g_P(\rho,\theta)$,

$$\begin{aligned} &(\rho + j\theta, g_P) = P(\rho \exp(j\theta), g) \\ &\rho = \sqrt{x^2 + y^2},\ \theta = \text{arctg}(y/x),\ g_P = g(x,y),\ j^2 = -1. \end{aligned} \quad (\mathbf{1})$$

I.e., P represents a conformal transform that stretches the Cartesian plane ($0xy$), which is also of polar coordinates (ρ,θ), to a new Cartesian plane ($0\rho\theta$), [8], see Fig. 2, the second row of images. The digital implementation of P naturally introduces noise; we will call it additive "regular" noise in 4.1.

PFWT step 2. 1-D discrete complex Fourier transform along the θ axis: F: $g_P(\rho,\theta) \rightarrow g_{PF}(\rho,\omega_\theta)$,

$$g_{PF}(\rho,\omega_\theta) = \sum_\theta g_P(\rho,\theta) \exp(-j\theta\omega_\theta),\ \ j^2 = -1, \quad (\mathbf{2})$$

where ω_θ is the frequency variable, corresponding to θ, and the result g_{PF} is a complex function of (ρ,ω_θ). The software implementation of F most often uses the so-called Fast Fourier Transform, [2].

PFWT step 3. 1-D complex wavelet transform along the ρ axis: W: $g_{PF}(\rho,\theta) \rightarrow g_{PFW}(\tau_\rho,\omega_\theta)$:

$$\begin{aligned} &W: (f_0,f_1,...f_{N-1}) \rightarrow (c_0,c_1,...c_{N-1}),\ \ N = 2^k\ \ k > 0 \\ &f_i = f(\rho) = \text{Re}(f(\rho)) + j\,\text{Im}(f(\rho)),\ \ \rho = i\rho_{\max}/N,\ \ i = 0,1,...(N-1),\ \ j^2 = -1 \\ &c_\tau = c(\tau) = (f{*}\Phi)(\tau) = \sum_\rho f(\rho)\Phi(\tau - \rho) = (\text{Re}(f){*}\Phi)(\tau) + j(\text{Im}(f){*}\Phi)(\tau) \\ &g_{PFW}(\tau_\rho,\omega_\theta) = c(\tau_\rho) = W(\text{Re}(g_{PF}(\rho,\omega_\theta)) + j\,\text{Im}(g_{PF}(\rho,\omega_\theta)),\ \ \tau_\rho = \tau \\ &\qquad\qquad \tau = 0,1,...(N-1). \end{aligned} \quad (\mathbf{3})$$

where $(f_0,f_1,\ldots f_{N-1})$ is the discrete representation of $g_{PF}(\rho,\theta)$, $\rho_{\max}$ determines the interval $(0,\rho_{\max})$ for $g_{PF}(\rho,\omega_\theta)$ and $g_P(\rho,\theta)$, while $(c_0,c_1,\ldots c_{N-1})$ is the series of the respective wavelet coefficients of $g_{PF}(\rho,\omega_\theta)$, i.e. the leaves in the full binary tree of a detailed decomposition at a resolution J, $J>0$, in accordance with the discrete algorithm of Mallat [6]. In our case, the band filter Φ defined in $(-\Delta\tau/2,+\Delta\tau/2)$, $\Delta\tau=\rho_{\max}2^{-J}$, corresponds to the chosen wavelet for W, and to the respective J for its scaling, [6]. Φ is introduced in formulation (3) to stress out the commutativity between W and the preceding F from (2). This occurs important for the software performance, and is visible from the representation of W as a convolution (.)*(.) of $g_{PF}(\rho,\omega_\theta)$ or $g_P(\rho,\theta)$ with a filter Φ.

PFWT step 4. Finally, perform a module transform M: $g_{PFW}(\rho,\theta) \rightarrow |\, g_{PFW}(\tau_\rho,\omega_\theta)\,|$. As established, this step is necessary for assuring translation invariance; the same is valid for P2DFT, [3], [4], [8], [10].

3.2 Extra notes to PFWT

As defined, PFWT is obviously valid for half-tone (gray) images, i.e. similarly to P2DFT. Color images can also be processed, after a preliminary conversion to gray.

At the time being, the *db2* wavelet of Daubechies has been chosen for the WT part of PFWT, [6]. It is smooth enough as well as simple enough for quantitative considerations, comparatively to *db1* that is equivalent to Haar wavelet and often preferred in image processing, [7].

IDB key generation. The IDB keys used to IDB access can be considered as composite keys, i.e. consisting of byte items $(d_0) \succ (d_1) \succ \ldots \succ (d_{L-1})$, L – the length of the key. Each key can simply be represented as a number of a 256-based number system. A given input image key D is obtained by scanning the resulting PFWT or P2DFT map of the image.

Considering the PFWT coefficients to be more significant the closer they are to the map beginning $(0,\tau_\rho,\omega_\theta)$, the scan of PFWT map is ordered by the (Euclidian) distance on the map item's positions. The similar is concerned for P2DFT, see Fig.1.

ρ\ω θ	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10
0	–	–	–	–	–	–	–	–	–	0	2	6	12	20	30	42	56	72	90	110
1	109	89	71	55	41	29	19	11	5	1	4	10	18	28	40	54	70	88	108	–
2	–	107	87	69	53	39	27	17	9	3	8	16	26	38	52	68	86	106	–	–
3	–	–	105	85	67	51	37	25	15	7	14	24	36	50	66	84	104	126	–	–
4	–	–	125	103	83	65	49	35	23	13	22	34	48	64	82	102	124	–	–	–
5	–	–	–	123	101	81	63	47	33	21	32	46	62	80	100	122	–	–	–	–
6	–	–	–	–	121	99	79	61	45	31	44	60	78	98	120	–	–	–	–	–
7	–	–	–	–	–	119	97	77	59	43	58	76	96	118	–	–	–	–	–	–
8	–	–	–	–	–	–	117	95	75	57	74	94	116	–	–	–	–	–	–	–
9	–	–	–	–	–	–	–	115	93	73	92	114	–	–	–	–	–	–	–	–
10	–	–	–	–	–	–	–	–	113	91	112	–	–	–	–	–	–	–	–	–
11	–	–	–	–	–	–	–	–	–	111	–	–	–	–	–	–	–	–	–	–

Fig. 1. The scan ordering applied to both PFWT and P2DFT

Normalization: The process of IDB key generation needs a normalization of the key items (d_i), i=1,2,…L-1 in the range [0÷255] determined by the bytes representation.

- Zhang and Lu [10] propose the following formulae:

$$D = (d_0, d_1, \ldots d_{mn-1}) = \left(\frac{\text{P2DFT}(0,0)}{area}, \frac{\text{P2DFT}(0,1)}{\text{P2DFT}(0,0)}, \frac{\text{P2DFT}(0,2)}{\text{P2DFT}(0,0)}, \ldots \frac{\text{P2DFT}(m,n)}{\text{P2DFT}(0,0)} \right),$$

where *area* means the spectrum area (or some part of it), and the indexing is 2D.

- The above normalization is almost adopted hereinafter with modifications:

$$D = (d_0, d_1, \ldots d_{L-1}) = \left(b, \left\lfloor \frac{\text{PFWT}(1)}{\text{PFWT}(0)/b} \right\rfloor_b, \left\lfloor \frac{\text{PFWT}(2)}{\text{PFWT}(0)/b} \right\rfloor_b, \ldots \left\lfloor \frac{\text{PFWT}(L-1)}{\text{PFWT}(0)/b} \right\rfloor_b \right), \quad (4)$$

where L is the key length. The modifications reflect the chosen way of map scanning as well as the restrictions $\lfloor \cdot \rfloor_b$ of the greatest key items to b-1, b =256.

PFWT basic properties. The two considered transforms can shortly be presented as compositions as follows:

$$\begin{aligned} \text{PFWT} &= P \circ W_\rho \circ F_\theta \circ M = P \circ F_\theta \circ W_\rho \circ M \ , \\ \text{P2DFT} &= P \circ F_\rho \circ F_\theta \circ M = P \circ F_\theta \circ F_\rho \circ M \ , \end{aligned} \tag{5}$$

where the transform components P, F, W and M have the sense of 3.1.

Reversibility of the basic part of PFWT:

$$(\text{PFW})^{-1} = (F_\theta)^{-1} \circ (W_\rho)^{-1} \circ (P)^{-1} = (W_\rho)^{-1} \circ (F_\theta)^{-1} \circ (P)^{-1} . \tag{6}$$

Linearity of the basic part of PFWT:

$$(\text{PFW})(\alpha g(x, y) + \beta h(x, y)) = \alpha(\text{PFW})(g(x, y)) + \beta(\text{PFW})(h(x, y)) . \tag{7}$$

Inner commutativity of the basic part of PFWT:

$$(\text{PFW})(g(x, y)) = W_\rho F_\theta (f(\theta, \rho)) = F_\theta W_\rho (f(\theta, \rho)) , \quad f(\theta, \rho) = P(g(x, y)) . \tag{8}$$

Rotational invariance of PFWT (including *M,* the 4-th step of PFWT). The polar mapping P of the input image $g(x,y)$, by a centre – the center of image "gravity", transforms the accidental input rotation into translation of $g_P(\rho,\theta) = P(g(x,y))$. The final operation M ignores the phase thus ignoring the random input rotation.

Expected advantages of PFWT towards P2DFT. There is no primary necessity of using second 1D Fourier transform, namely F_ρ, as it is in P2DFT. In PFWT it is replaced by a 1D wavelet transform. The WT-coefficients are statistically non-correlated, as well as the Fourier coefficients are [1], [6], but WT allows a full independence between the object space and the frequency space representations, while FT is limited in this respect [1]. Thus, it is expected that the cumulative effect of using WT instead of second FT will bring to improvement of the new PFWT.

Processing resources necessary. Processing speed of PFWT can be simply estimated as similar to this of P2DFT, using the following estimations by parts: $O(P) \sim (x_m y_m \theta_m \rho_m)^{1/2}$, $O(F_\theta) \sim O(W_\theta) \sim \theta_m \log(\theta_m)$, and $O(F_\rho) \sim O(W_\rho) \sim \rho_m \log(\rho_m)$, where x_m, y_m are the normalized input image dimensions, in our case $x_m = y_m = 128$, and respectively $\theta_m = 512$, $\rho_m = 128$. Some speed improvements can also be expected considering the choice of L, $L \le x_m y_m/2$, in our case $L = 126$.

4 Experiments

The experimental research of the proposed PFWT was conducted in the frames of the EFIRS (Effective and Fast Image Retrieval System), developed by IIT-BAS, [3], [4].

EFIRS is a C/C++ written Windows-XP application operating on an IBM compatible PC: Pentium 2.8 GHz, MM 1.0 GB, HDD 160 GB.

The test is implemented as a separate module of EFIRS, called SLT (Simple Locate Test), and is described in brief below.

The tested IDBs are from the PORB practice – IDB_1 of 4834 mark images, IDB_2 of 30001 mark images, and IDB_3 of 58338 mark images. The three IDBs include each other as follows: $IDB_1 \subset IDB_2 \subset IDB_3$, which contributes to the result analysis.

4.1 Test algorithm

The SLT algorithm consists of 2 general parts – a basic loop and a final computation. The definition of the so-called *Z*-distance is essential for the SLT description.

Z-distance between two images. We define the *Z*-distance by the number of positions to the end of the IDB keys, cf. descriptors in 2. Thus, the *Z*-distance between two images under comparison, in our case – the query image at the input and the retrieved one from IDB, is measured as the maximal tail difference length between the respective image keys. The maximal tail difference length denotes the remainder from the key after position l, $l<L$, as to which the CBIR method considers both images similar enough, i.e.:

$$Z(K_1, K_2) = L - \max_{0 \le l < L} \{ l : | k_{1,i} - k_{2,i} | \le \varepsilon, i = 0,1,...l \} , \qquad (9)$$

where L is the length of the IDB keys, K_1 and K_2, $K=(k_0, k_1,\ldots,k_{L-1})$ are the keys of compared images, and the positive constant ε reflects the EFIRS's built-in degree of noise tolerance.

The SLT basic loop. For each image from the test IDB do the following 3 steps:

- Rotate the image at α, a given angle, obeying no image cut (see also Fig.2).
- Use the above rotated (and scaled) image as a query content to search for its similarities into the same IDB. The number k of the similar images from the IDB search is considered to be given a priori, $0<k<N$, where $N=|IDB|$ is the IDB volume (in number of images).
- List the retrieved images into ascending order of their *Z*-distances to the query content (descending order of similarities). Visualize the results graphically and protocol them.

The SLT final computation. After exhausting the images of the test IDB, generalize and protocol the results.

The generalization terms will be considered hereinafter in respect to *only the image retrieved at the first position* in the list of similarities:

- Warning (W): reports about an inessential error that the retrieved image does not coincide with the query one, but the *Z*-distance between them is "zero". I.e., the tested access method is too generalized to discriminate both images. *W* will stand for the number of W-errors. Either *W* or its percentage towards the volume *N* of the entire IDB will be worth for this analysis.

- Rough error (R): indicates an essential error – the retrieved image does not coincide with the query one, and the *Z*-distance is positive, i.e. the tested access method is not precise enough to cluster (to associate) both images. *R* will stand for the number of R-errors.
- Total of errors (*W*+*R*): as well as the percentage towards *N*=|IDB|.
- Averaged *Z*-distance on the R-cases only.
- Averaged *Z*-distance on hits only: i.e. on successful cases, without any errors, when the retrieved (at the first position from the list) image coincides with the initial query content (at step 1 of the SLT loop), no matter how big their *Z*-distance is.
- Averadged access time per image

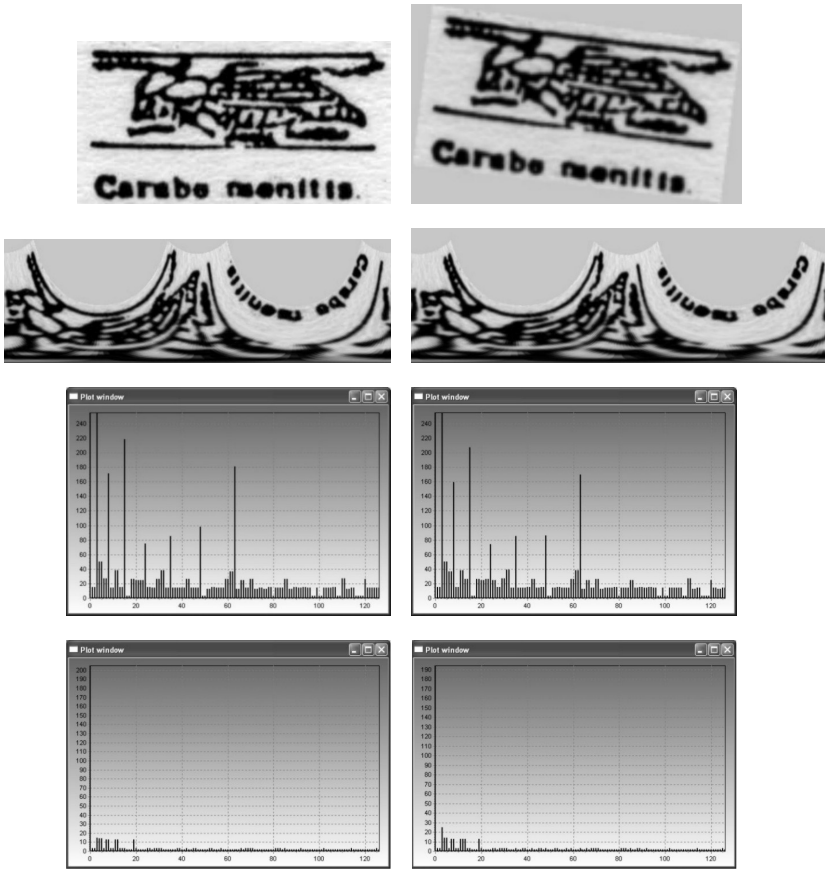

Fig. 2. Two images processed by both methods of comparison, PFWT and P2DFT. Illustrations by rows: (1) the input images, (2) their polar mappings, (3) their PFWT keys to an IDB, and (4) their P2DFT keys. The left vertical of illustrations stands for an original mark image while the right one – for its rotation (at α=7°).

Regular noise introduced by the test. We will call *regular noise* the noise introduced in the input image by the digital performance of the SLT preliminary rotation. Besides, we will also consider as regular the noise caused by the SLT scaling as well as the extra background added in order to avoid any cutting out of the resulted image, see Fig.2, the upper row, on the right. This type of noise can also be attributed to the performance inaccuracies of the polar mapping, etc. In view of the linearity of the used transforms, we can treat the regular noise additively.

The so-defined regular noise is a constructive specificity of the EFIRS system and can be interpreted as a deviation of the SLT rotation towards the virtual/ideal case of infinitely high resolution. At this stage, the level δ of the introduced regular noise can be manipulated mainly by choosing the preciseness level of the SLT rotation module.

4.2 Generalized experimental results

Tables 1, 2 and 3 generalize the experimental results for the three tested IDBs, each one by rotation angle α=7°. The table column names correspond to the description in 4.1, while the table rows – to both methods of comparison, P2DFT and PFWT.

Table 1. Both approaches P2DFT and PFWT on the first IDB (~5ooo marks).

4834 marks	Warn-ings (*W*)	Warn -ings (%)	Rough errors (*R*)	Rough errors (%)	Total *W+R*	Total (%)	averaged *Z*-distance on R-cases	averaged *Z*-distance on hits (no err.s)
P2DFT	906	*18.7*	48	*1.0*	954	19.7	107.6	0.322
PFWT	682	*14.1*	100	*2.1*	782	16.2	102.7	0.628

Table 2. Both approaches P2DFT and PFWT on the second IDB (~30ooo marks).

30001 marks	Warn-ings (*W*)	Warn -ings (%)	Rough errors (*R*)	Rough errors (%)	Total *W+R*	Total (%)	averaged *Z*-distance on R-cases	averaged *Z*-distance on hits (no err.s)
P2DFT	8008	*26.7*	344	*1.2*	8352	27.8	103.1	0.230
PFWT	6574	*21.9*	698	*2.3*	7272	24.2	96.3	0.518

Table 3. Both approaches P2DFT and PFWT on the third IDB (~60ooo marks).

58338 marks	Warn-ings (*W*)	Warn -ings (%)	Rough errors (*R*)	Rough errors (%)	Total *W+R*	Total (%)	averaged *Z*-distance on R-cases	averaged *Z*-distance on hits (no err.s)
P2DFT	16188	*27.7*	494	*0.8*	16682	28.6	100.9	0.175
PFWT	13626	*23.4*	1066	*2.3*	14692	25.2	93.4	0.335

As can be seen from Table 4, the averaged times per CBIR access for both methods are commensurable to 0.6÷1.8 s. The similar times in the single search regime "locate", will increase up to 2÷7 s, mostly depending on the IDB volume *N*. More formally

$$t_{\text{SLT}} = t_{\text{init}} + N.t_{\text{access}}\ , \quad t_{\text{locate}} = t_{\text{init}} + t_{\text{access}}\ , \qquad (\mathbf{10})$$

where t_{SLT} is the entire processing time for the SLT experiment for a given IDB of volume N, t_{init} is the preprocessing time (for the EFIRS initialization with the given IDB), and t_{access} is the average access time.

Table 4. Processing speed evaluated for both methods, P2DFT and PFWT, on the three IDBs.

-	Entire processing time [min]			Pre-processing time [s]			Average access time [s/rec]		
IDB volume	4834	30001	58338	4834	30001	58338	4834	30001	58338
P2DFT	43.4	584.6	1916.6	0.8	7.9	12.3	0.54	1.17	1.97
PFWT	51.8	619.3	1799.9	2.0	7.5	12.6	0.64	1.24	1.85

4.3 Experimental result analysis

Zhang & Lu [10] propose an averaged evaluation of hits of searching in their IDBs. In this work, another evaluation is proposed stressing on the retrieval at the first position only. The obtained results are not directly comparable with [10] because of the different test IDBs. That's why the P2DFT that is available in EFIRS [3], [4] and considered similar enough to [10] is chosen for comparison with the proposed PFWT.

Ascertainments. The generalized experimental results (Tables 1, 2, and 3) show that:

- The proposed PFWT wins against the P2DFT approach towards the error rate of warning type, in the order of 4.3 ÷ 4.8 % towards N.
- In the opposite, towards the essential R-errors' rate, PFWT loses against P2DFT, in the order of 1.1 ÷ 1.5 % towards N.

These ascertainments are also confirmed by a comparison on the "averaged Z-distance":

- The averaged Z-distance in rough error cases differs in about 2 times between the proposed PFWT and the "classical" P2DFT as tested by the three IDBs. More exactly it differs in about 19÷33% (for PFWT) and 14÷20% (for P2DFT) towards the chosen length L of IDB keys, L=126.
- The averaged Z-distance on hits, i.e. on non error cases, is small enough towards the chosen L, L=126, with both of the approaches and for all tested IDBs. Here, PFWT loses towards P2DFT, generally because of the greater "density" of PFWT-key-values around the query image than the corresponding "density" of P2DFT; and the latter case should be considered occasional situations.

Interpretations. For given IDB and fixed ε – the level of the built-in noise tolerance of EFIRS, see (9), the idea of testing with given access method consists in the following:

- At high enough level δ of the introduced regular noise (see 4.1), it is natural to find a multitude of essential errors by the SLT experiment.
- When decreasing δ, we may expect that the number R of essential errors will diminish, and that at some δ_{opt} , $\delta_{opt}>0$, R will reach "zero". The non-occurrence of this will be an indication for software performance errors either in the access method performance or in the test experiment (!)

- Becides, if the IDB consists of clusters of very close or even equal images, then the number W of warnings is going to remain high, even in the case of $\delta < \delta_{opt}$. To improve the distinguishability between the two methods of comparison, PFWT and P2DFT, a larger key length L can be chosen, $L \leq x_m y_m/2$, where x_m, y_m are the normalized input image dimensions, in our case $x_m = y_m = 128$, i.e. $L \leq 8192$.

5 Conclusion

The introduction of WT, instead of FT, along the polar distances in the images after PMT, determines a much lower correlation among the transformed image coefficients by PFWT compared to P2DFT. Thus, the eventual additive input noise is better localizing using PFWT, instead of P2DFT. Meanwhile, the noise influences much more coefficients at P2DFT than at PFWT, which explains the larger number of warnings in the P2DFT case. The relatively bigger number of essential errors by the proposed PFWT can be explained by the impreciseness of the software implementation of the experiments at the time being.

By these reasons we still consider the proposed PFWT more effective that the "classical" P2DFT approach for CBIR performances.

Acknowledgements. This work is partially supported by following grants: Grant # B-G-17/2005 and Grant # VU-MI-204/2006 of the Bulgarian Ministry of Education & Science, and Grant # 010088/2007 of BAS. A special gratitude to PORB (the Patent Office of R Bulgaria) for placing at our disposal a test IDB of about 60 000 images.

References

1. Benedetto, J., Frazier, M.: Introduction, In *Wavelets: Mathematics and Applications,* J. Benedetto and M. Frazier, Eds. Boca Raton, FL: CRC Press, 1-20 (1994)
2. Chu, E., George, A.: Inside the FFT Black Box, CRC Press, New York (2000)
3. Dimov, D.: Rapid and Reliable Content Based Image Retrieval. Proceedings of NATO ASI, Multisensor Data and Information Processing for Rapid and Robust Situation and Threat Assessment, 16-27 May 2005, Albena-Bulgaria, IOS Press, 384-395 (2007)
4. Dimov, D.T.: Harmonic and Wavelet Transform Combination for Fast CBIR, Int. Conf. on PDE Methods in Applied Mathem. and Image Proc., Sep. 7-10, 2004, Sunny Beach Bulgaria, http://www.math.bas.bg/~kounchev/2004_conference/HomePage2004.html/
5. Eakins, J.P.: Towards intelligent image retrieval, Pattern Rec. J., **35** 3-14 (2002)
6. Mallat, S.G.: A Wavelet Tour of Signal Proc., Acad. Press, San Diego (1998)
7. Misiti, M., Misiti, Y., Oppenheim, G., Poggi J.-M.: Wavelet Toolbox: Computation, Visualization, in Programming User's Guide, Ver.1, MATLAB® v.5.2, ..\MatLab\Help\PDF_doc\wavelet\wavelet_ug.pdf.
8. Reiss, T. H.: Recognizing Planar Objects Using Invariant Image Features, Springer-Verlag, Berlin (1993)
9. Smeulders, A.W.M., Worring, M., Santini, S., Gupta, A., Jain, R.: CBIR at the End of the Early Years, IEEE Trans. on PAMI, **22**, 12 1349-1380 (2000)
10. Zhang, D., Lu, G.: Enhanced Generic Fourier Descriptors for Object-Based Image Retrieval. Proceed. IEEE ICASSP'02 (1520-6149) 4 3668-3671 (2002)