Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

FAST SIGNAL-INDUCED TRANSFORMS IN IMAGE ENHANCEMENT

Khalil Naghdali, Raghunath Ranjith, and Artyom M. Grigoryan

The University of Texas at San Antonio
Department of Electrical and Computer Engineering
One UTSA Circle, San Antonio, TX 78249-0669

ABSTRACT

In this paper, the concept of the discrete unitary transforms induced by given signals or input signals is modified and developed. The basic transformations composing such transforms are parameterized and the energy (or its partial part) of signals is transferred to one of their components in different paths. We focus on the case when all basic transformations themselves represent Givens rotations of the inputs or modified inputs. The consideration of these transformations is of much interest in the unique angular forms, which allows us to consider signals and their transforms in the very compact region of 2π. Applications of proposed angular transforms for image enhancement are described.

1. INTRODUCTION

Together with the Fourier transform, many other discrete unitary transforms are used widely in different areas of signal and image processing [1, 2]. We can mention the Hadamard, Hartley, Haar, cosine, and wavelet transforms [3]-[8]. In applications to discrete-time signals or discrete images, these transforms are determined by different complete systems of basis functions. Some of these systems have been derived or chosen based on certain properties of signals under consideration. It is very important to define a system of basic functions that carry much information of processed signals. Such an example is the system of eigenfunctions of the autocorrelation function of the random image for the Karhunen-Loeve transform which, however, does not have fast algorithms. On the other hand, the Hartley and Haar transforms have preassigned systems of functions. In works [9]-[11], a simple model of composing the discrete signal-induced heap transforms (DsiHT) has been introduced. These transforms are fast for any order and are defined by consequent heaping of the energy of a generator-signal in one or a few specified locations of the signal. The complete systems of basis functions of heap transforms are referred to as waves generated by input signals, or the waves with their specific motion in the space of functions.

In this paper, we describe modified DsiHTs, when the decision equations are parameterized in order to transfer the energy of the generator-signal to one heap in different proportions depending on the arrangement of components of the signal. Each DsiHT can be represented in the unique angular form, and each signal can be described as an inverse angular transform. Our preliminary experimental results show that the angular transforms can be used in signal and image processing, such as image enhancement.

Address all correspondence to Artyom M. Grigoryan. E-mail: amgrigoryan@utsa.edu, Tel/Fax: (210) 458-7518/5947.

2. PARAMETERIZED DSIHTS

Given a vector-generator $\mathbf{x} = (x_0, x_1, ..., x_{N-1})'$, the N-point discrete signal-induced heap transform (DsiHT) of a signal $\mathbf{z} = (z_0, z_1, ..., z_{N-1})$ is defined by $N-1$ basic transformations

$$\begin{aligned} T(\varphi_k):\ \mathbf{z} \to (z_0, ..., z_{k_1-1}, f(z_{k_1}, z_{k_2}, \varphi_k), z_{k_1+1}, ..., \\ z_{k_2-1}, g(z_{k_1}, z_{k_2}, \varphi_k), z_{k_2+1}, ..., z_{N-1}) \end{aligned} \tag{1}$$

where the pair of numbers (k_1, k_2) is uniquely defined by k, and $0 \leq k_1 < k_2 \leq (N-1)$. We consider the case, when $k_1 = 0$ and values of k_2 are equal k; in other words, the path P of processing components of the signal starts from $k_2 = 1$ and consequently continues until $N-1$. The value of z_0 is renewed on each stage of calculation. The parameter φ_k is defined from the second equation, which we call *the angular equation* of the system of decision equations

$$\left\{ \begin{array}{l} f(x_0, x_{k_2}, \varphi_k) = y_0 \\ g(x_0, x_{k_2}, \varphi_k) = a_k \end{array} \right. \tag{2}$$

for a given parameter a_k. Then, the value of y_0 is calculated and considered to be a new value of x_0 for the next stage of calculations. System of equations (2) is solving for a vector-generator $\mathbf{x} = (x_0, x_1, ..., x_{N-1})$. It is assumed that the angular equation has solutions $\varphi = r(x, y, a_k)$ for given parameters a_k.

We now recall the composition of the DsiHT, then consider examples of parameterized DsiHTs.

Example 1 The four-point DsiHT by the vector-generator $\mathbf{x} = (x_0, x_1, x_2, x_3)'$ is defined in matrix form as follows:

$$\begin{aligned} \mathbf{y} &= \mathbf{T}\mathbf{x} = \mathbf{T}_3\mathbf{T}_2\mathbf{T}_1\mathbf{x} \\ &= \begin{bmatrix} c_3 & 0 & 0 & -s_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ s_3 & 0 & 0 & c_3 \end{bmatrix} \begin{bmatrix} c_2 & 0 & -s_2 & 0 \\ 0 & 1 & 0 & 0 \\ s_2 & 0 & c_2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \\ &\times \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix}, \end{aligned}$$

where the cosine and sine coefficients are calculated from $\mathbf{x}$ by $c_k = \cos(\varphi_k)$, $s_k = \sin(\varphi_k)$, $\varphi_k = -\tan^{-1}(x_k/y_0)$, for $k = 1, 2, 3$. Here we consider that $y_0 = x_0$, for $k = 1$, and $y_0 = y_0^{(2)} = (\mathbf{T}_1\mathbf{x})_0$ and $y_0 = y_0^{(3)} = (\mathbf{T}_2\mathbf{T}_1\mathbf{x})_0$. For instance, the vector $\mathbf{x} = (1, 2, 3, 4)'$ induces the four-point DsiHT whose matrix is composed as

$$\mathbf{T} = \begin{bmatrix} 0.6831 & 0 & 0 & 0.7303 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -0.7303 & 0 & 0 & 0.6831 \end{bmatrix} \begin{bmatrix} 0.5976 & 0 & 0.8018 & 0 \\ 0 & 1 & 0 & 0 \\ -0.8018 & 0 & 0.5976 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$
$$\times \begin{bmatrix} 0.4472 & 0.8944 & 0 & 0 \\ -0.8944 & 0.4472 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

and the output $\mathbf{y} = (5.4772, 0, 0, 0)'$ is calculated as

$$\mathbf{y} = \mathbf{T}\mathbf{x} = \begin{bmatrix} 0.1826 & 0.3651 & 0.5477 & 0.7303 \\ -0.8944 & 0.4472 & 0 & 0 \\ -0.3586 & -0.7171 & 0.5976 & 0 \\ -0.1952 & -0.3904 & 0.5855 & 0.6831 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}.$$

Angles of rotations of the vector-generator equal $\varphi_1 = -1.1071$, $\varphi_2 = -0.9303$, and $\varphi_3 = -0.8188$ (in radians). The number $5.4772^2 = 30$ is the energy of the signal $\mathbf{x}$. It is important to note that the normalized vector-generator lies in the first row of the matrix $\mathbf{T}$ of the heap-transform

$$[0.1826, 0.3651, 0.5477, 0.7303] \cdot 5.4772 = [1, 2, 3, 4].$$

In general, up to the normalized coefficient, the first basis function of the heap-transform coincides with the vector-generator. As shown in [10], there are three stages which can be separated during the process of motion and transformation of one basis function into another one, when starting from the wave-generator. In the first stage, *the statical stage,* the generator itself is lying as the basis function. The second stage, *the evolution stage,* is related to the formation of a new wave. The last stage is *the dynamical stage,* when the new established wave is moving to the end of the path. This wave is composed by two parts. The first part resembles the generator and the second part, or a splash, is a static wave increasing by amplitude. For example, Figure 1 shows the eight waves defining the matrix of the heap transform, when the generator is $\mathbf{x} = (1, -1, 1, -1, 1, -1, 1, -1)'$. One can note that the generator in the last stage is trying to restore itself, while moving the large splash to the end. The maximum reconstruction of the generator is in the last movement, when it gives the much power to the splash, whose amplitude increases linearly. The generator "induces" some field and then tries to pass through it, and that is fulfilled with the loss which occurs in the form of a high but narrow splash moving to the end of the path.

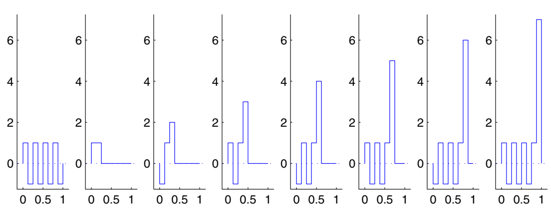

Fig. 1. Non-normalized basis functions of the 8-point DsiHT

The DsiHT is described completely in angular representation, or transform $\mathbf{x} \to \mathcal{A}[\mathbf{x}] = (||\mathbf{x}||, \varphi_1, \varphi_2, \ldots, \varphi_{N-1})$. As an example, we consider the DsiHT of a sine signal with two noise-pikes

$$z(t) = x(t) + n(t) = \sin(\pi t/L) + n(t),$$

where $L = 32$, and the noise-pikes are sine half-waves of lengths $L_1 = 2$ and $L_2 = 8$. Figure 2 shows the original sampled 501-point signal $\mathbf{x}$ in part a, along with the noisy signal $\mathbf{z}$ in b, the 500-point angular DsiHT (generated by $\mathbf{x}$) in c, and the 500-point angular DsiHT (generated by $\mathbf{z}$) in d. One can see the location of the noise and its length in the angular transforms. This is a time-length, or time-frequency analysis of these signals.

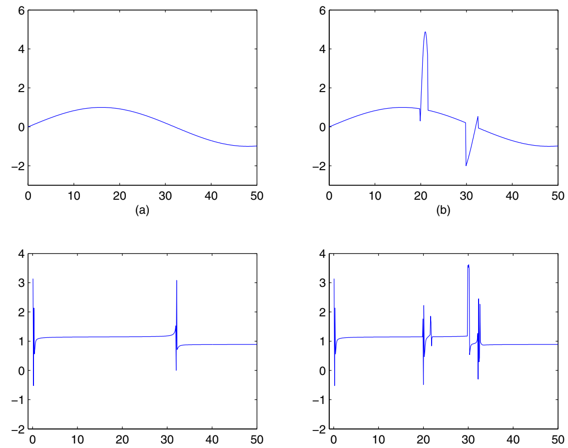

Fig. 2. (a) Sine wave of half-period L, (b) noisy signal, and the angular representations of (c) the sine wave and (d) noisy signal.

Let α be a positive number, and let A be a set of numbers a_k which are defined by

$$a_k = g(x_{k-1}, x_k, \varphi_k) = \alpha^{-1} f(x_{k-1}, x_k, \varphi_k), \tag{3}$$

where $k = 1 : (N-1)$. For instance, we can consider α to be an integer number $n \geq 1$. To describe the N-point transform corresponding to such a parameterized case, we analyze the basic two-point transformation, $T_\alpha(\varphi)$, in matrix form

$$\begin{bmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} y_0 \\ \alpha^{-1} y_0 \end{bmatrix}. \tag{4}$$

The angle φ and the component y_0 are calculated respectively by

$$\tan(\varphi) = \frac{x_1 + \alpha x_0}{x_0 - \alpha x_1}, \qquad y_0 = \pm \frac{\alpha}{\sqrt{1+\alpha^2}} \sqrt{x_0^2 + x_1^2}. \tag{5}$$

Thus, $y_0^2 < x_0^2 + x_1^2$ and less energy is transferred to the first component but more to the second component, if $\alpha < 1$, and vice versa, if $\alpha > 1$. Figure 3 shows the graph of the basic transformation along with the example when $\alpha = 1/2$ and the generator $(x_0, x_1) = (4, 3)$. The transform of (x_0, x_1) equals $(\sqrt{5}, 2\sqrt{5}) = (2.236, 4.472)$, and the angle $\varphi = \tan^{-1}(2) = 63.4349°$. This transform can be applied now to any vector $\mathbf{z} = (z_0, z_1)'$.

We now construct an N-point transform which is induced by a vector $\mathbf{x} = (x_0, x_1, \ldots, x_{N-1})'$. To find values of angles φ_k, $k = 1 : (N-1)$, the decision equations are used. Let C be the square of the normalized coefficient in (5), i.e. $C = C(\alpha) = \alpha^2/(1+\alpha^2)$. The first pair of components $\bar{x}_0 = (x_0, x_1)'$ is rotated to the vector $(y_0^{(1)}, \alpha^{-1} y_0^{(1)})'$. Then, on the next kth stage of calculations, where $k \geq 1$, similar rotations are accomplished over the vector

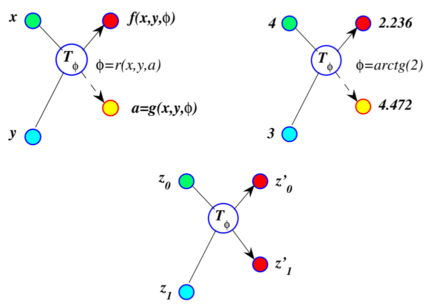

Fig. 3. Graphs of the 2-point transform T_φ which is defined by the input $(x, y) = (4, 3)$ and $\alpha = 0.5$ and, then, applied to (z_0, z_1).

$\bar{x}_{k-1} = (y_0^{(k-1)}, x_k)'$, where $y_0^{(k-1)}$ denotes a new value of the first component on the $(k-1)$th stage, and $y_0^{(0)} = x_0$. On this stage of composition, the basic transform is calculated by

$$\left[\begin{array}{c} y_0^{(k)} \\ \alpha^{-1} y_0^{(k)} \end{array}\right] = \left[\begin{array}{cc} \cos\phi_k & -\sin\phi_k \\ \sin\phi_k & \cos\phi_k \end{array}\right]\left[\begin{array}{c} y_0^{(k-1)} \\ x_k \end{array}\right], \qquad (6)$$

where

$$\tan(\phi_k) = \frac{x_k + \alpha y_0^{(k-1)}}{y_0^{(k-1)} - \alpha x_k}. \qquad (7)$$

The first component $y_0 = y_0^{(k-1)}$ is renewed as

$$y_0^{(k)} = \pm\sqrt{C x_k^2 + C^2 x_{k-1}^2 + \cdots + C^k x_1^2 + C^k x_0^2},$$

and the kth component equals $\alpha^{-1} y_0^{(k)}$, $k = 1 : (N-1)$. The N-point discrete transform defined by the vector $\mathbf{x}$ such as

$$T : \mathbf{x} \to \left(y_0, \alpha^{-1} y_0^{(1)}, \alpha^{-1} y_0^{(2)}, \ldots, \alpha^{-1} y_0^{(N-1)}, \alpha^{-1} y_0\right)$$

is called the *modified* DsiHT (MDsiHT). On the final stage of calculations when $k = N-1$, we obtain the following transformation of energy of the generator to the first component

$$y_0^2 = \left(y_0^{(N-1)}\right)^2 = C^{N-1} x_0^2 + \sum_{k=1}^{N-1} C^{N-k} x_k^2. \qquad (8)$$

The energy of the first component x_0 is transformed with the factor of C^{N-1}; and for remaining components x_k, their energies are transformed with factors of C^{N-k}, $k \geq 1$. Since $C < 1$, the smaller the number of the component, the smaller the increment of energy is transferred to the first component. The term "energy" of the vector is referred to as the norm to the square, i.e. $E^2(\mathbf{x}) = x_0^2 + x_1^2 + \ldots + x_{N-1}^2$, or x_k^2 when only one component x_k is mentioned. In the $\alpha = 1$ case, $C = 1/2$ and the energy is distributed equally between the outputs of basic transforms. In the limit $\alpha = \infty$ case, $C = 1$ and the energy of the signal is transferred fully to the first component of the transform; this is the case of the DsiHT. The above example is a deviation from the original method of construction of the DsiHT, when the second component of the basic transform is always considered to be zero.

Example 2 We consider the case when the energy of components of the signal is collected along a specified path and transferred with the same factor that does not depend on the distance of components from the origin point. Given a number $\lambda > 0$, we define the parameterized two-point basic transformation, $T_\lambda(\phi)$, by

$$\left[\begin{array}{cc} \cos\phi & -\lambda\sin\phi \\ \sin\phi & \lambda\cos\phi \end{array}\right]\left[\begin{array}{c} x_0 \\ x_1 \end{array}\right] = \left[\begin{array}{c} y_0 \\ 0 \end{array}\right].$$

The values of ϕ and y_0 are defined respectively by $\tan(\phi) = -\lambda x_1/x_0$ and $y_0 = \sqrt{x_0^2 + \lambda^2 x_1^2}$. The determinant of the transform matrix equals λ. The transformation $T_\lambda(\phi)$ is reduced to the original $\lambda = 1$ case, by changing x_1 by λx_1.

The N-point MDsiHT based on such basic transformations transfers to the first sample the energy which equals

$$y_0^2 = x_0^2 + \lambda^2(x_1^2 + x_2^2 + \ldots + x_{N-1}^2) = (1-\lambda^2)x_0^2 + \lambda^2 E_N^2.$$

The transferred energy is increasing when $\lambda > 1$. When λ varies in the interval $(0, 1]$, the amount of energy transformed to the first component y_0 varies from x_0^2 to $E^2(\mathbf{x})$.

Example 3 We now consider the transform which is associated with the case when energies of components of the signal are transferred with factors that increase as the distance of components from the original increases. A parameterized two-point basic transformation, $T_\lambda(\phi)$, with $\lambda \in (0, 1)$, is defined as

$$\left[\begin{array}{cc} \lambda\cos\phi & -\sin\phi \\ \lambda\sin\phi & \cos\phi \end{array}\right]\left[\begin{array}{c} x_0 \\ x_1 \end{array}\right] = \left[\begin{array}{c} y_0 \\ 0 \end{array}\right]$$

where $\tan(\phi) = -x_1/(\lambda x_0)$ and $y_0^2 = \lambda^2 x_0^2 + x_1^2$. The determinant of the transform matrix equals λ and the transform is reduced to the original $\lambda = 1$ case, when changing x_0 by λx_0.

The N-point MDsiHT based on such transformations transfers to the first component the energy which is equal to

$$y_0^2 = x_{N-1}^2 + \lambda^2 x_{N-2}^2 + \lambda^4 x_{N-3}^2 + \cdots + \lambda^{2(N-1)} x_0^2.$$

The energy of each component x_k is added to y_0 after being multiplied by factor of $\lambda^{2(N-k-1)}$, $k \geq 0$, which tends to zero for large k. If $\lambda < 1$, the energies of the first components go out; the longer the signal, the faster they go out, and $y_0^2 \leq E_N^2(\mathbf{x})$. As an example, Figure 4 shows the graph of calculation of the 5-point MDsiHT of a vector $\mathbf{z} = (z_0, z_1, z_2, z_3, z_4)'$.

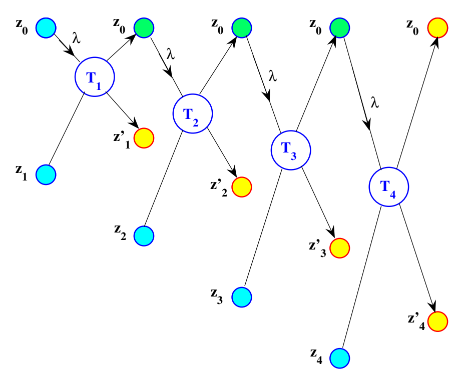

Fig. 4. Signal-flow graph of calculation of the five-point MDsiHT of a vector $\mathbf{z}$.

To smooth the second components of basic transformations $T(\varphi_k)$ in the decomposition of N-point discrete heap transform, we consider the following angular equations

$$a_k = g(x_{k-1}, x_k, \varphi_k) = \frac{x_{k-1} + x_k}{\lambda_k}, \quad k = 1 : (N-1), \quad (9)$$

in the case when all $\lambda_k = 2$. The two-point basic transformation, $T(\varphi)$, is defined in matrix form as follows:

$$\begin{bmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} y_0 \\ \frac{1}{2}(x+y) \end{bmatrix}. \quad (10)$$

The angle φ is calculated by

$$\varphi = \tan^{-1}\left(\frac{y}{x}\right) + \arccos\left(\frac{x+y}{2\sqrt{x^2+y^2}}\right), \quad (11)$$

and $y_0 = x\cos\varphi - y\sin\varphi$. If we assume the conservation of energy, then $y_0^2 = x^2 + y^2 - (x+y)^2/4 > 0$.

3. EXAMPLES AND APPLICATIONS OF THE DSIHT

In this section, we describe a few examples of the heap transform applied to 1-D signals and 2-D images. In all examples, the heap transform T is composed by basic transformations T_φ defined as

$$T_\varphi : \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} \to \begin{bmatrix} y_0 \\ 0 \end{bmatrix} = \begin{bmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix}.$$

If $\mathbf{x} = (x_0, x_1, ..., x_{N-1})$ is the generator of the transform T, then

$$(T[\mathbf{x}])_k = ||\mathbf{x}||\delta_k, \quad k = 0 : (N-1),$$

where δ_k is the delta symbol defined as $\delta_0 = 1$ and $\delta_k = 0$ if $k \neq 0$. The heap transform is described in the angular form as

$$\mathbf{x} \to \mathcal{A}[\mathbf{x}] = (||\mathbf{x}||, \mathcal{A}_{N-1}[\mathbf{x}]) = (||\mathbf{x}||, \{\varphi_1, \varphi_2, ..., \varphi_{N-1}\})$$

where φ_k are the angles of rotations of basic transformations $T_k = T_{\varphi_k}$ composing T. The first component is the energy of the signal and stands separately. It may take large values, especially for long length signals. All other components are angles and lie in the interval $(-\pi, \pi)$, or $(0, 2\pi)$. The heap transformation in angular representation is thus a unique transformation of the vector space R^N of N-dimensional signals into the subspace $R \times [-\pi, \pi]^{N-1}$.

As an example, Figure 5 shows a signal $\mathbf{x}$ of length 256 in part a, along with the angular transform in the form $\mathcal{A}_{255}[\mathbf{x}] + \pi$ in b. The signal is row number 64 of the clock-and-moon image (which is shown in Figure 6(a)). We now consider application of the heap transform generated by this signal to the next row of the image. Row-signal number 65 is shown in c, and the heap transform on this signal in d. The heap transform shows the interval, where the second signal differs greatly because of the noise.

Certain operations over the angular transforms can be developed and used in practice in order to obtain some useful properties of signals processed through their angular transforms. It is desired thus to study the angular transform in detail and understand how to manipulate the rotation angles φ_k of basic transformations composing the heap transformation.

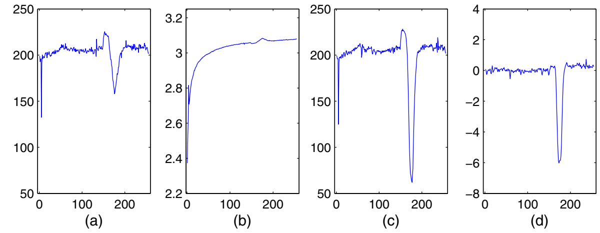

Fig. 5. (a) Signal-generator, (b) the angular transform of the signal, (c) another signal, and (d) the heap transform of the signal.

3.1. Heap transforms and 2-D images

The concept of the angular transform can be used and generalized for 2-D signals and images. We first consider the case when the 1-D heap transforms are generated by each row of an image and then represented in their angular forms. We call such a set of 1-D heap transforms the *2D-row heap transform* and the set of corresponding angular forms the *2D-row angular transform*. Similarly the concepts of the *2D-column heap* and *angular* transforms are defined. As an example, Figure 6 shows the clock-moon image (256×256) in part a, along with the mesh of the 2D-row angular transform in b, the energy curve of all rows in c, and the mesh of the image in d. The picture in b resembles the light field of the

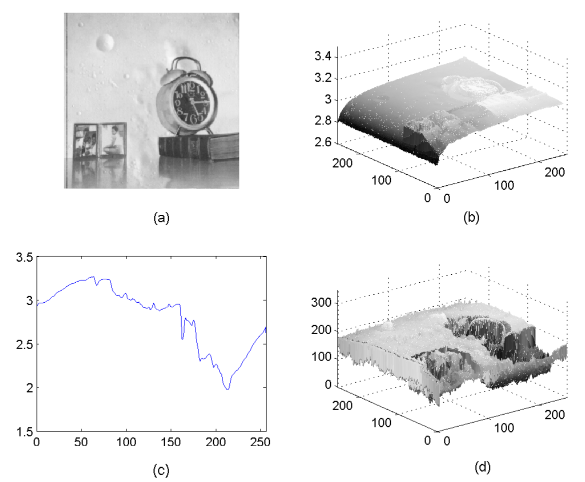

Fig. 6. (a) The image, (b) the mesh of the 2D-row angular transform of the image, (c) the energy curve of all row-signals (which has been normalized by 256), and (d) the mesh of the image.

angular transform with a 3-D copy of the original image on the surface of that field. Many details of the image can be observed on such a surface. The picture of the ordinary mesh in d does not provide us with such a compressed information of the image.

There are many ways in the 2-D case to define paths P for heap transforms. The transforms can be performed block-wise with different starting points and along specified directions and paths to complete 1-D heap transforms within blocks. These transforms can also be generated by selected rows or columns and applied to other ones, and so on. We consider the case when the heap transforms are sequentially generated by rows and applied to the next ones. As an example, Figure 7 shows the image (256×256) in part a, and the curves of energies of all 128 rows $\mathbf{x}_k$ with odd numbers $k = 1, 3, ..., 255$ in b, the image of angular transforms $\mathcal{A}_{255}[\mathbf{x}_k]$ generated by these row-signals in c, and the heap transforms $T_{\mathbf{x}_k}[\mathbf{x}_{k+1}]$ applied to rows with even numbers $k+1$ in d. The 2D-row angular transform in c is of size 255×128; and together with the energy curve of length 128 in b, it allows for reconstructing the original image in all odd rows. The image in d allows for reconstructing the remaining rows of the original image.

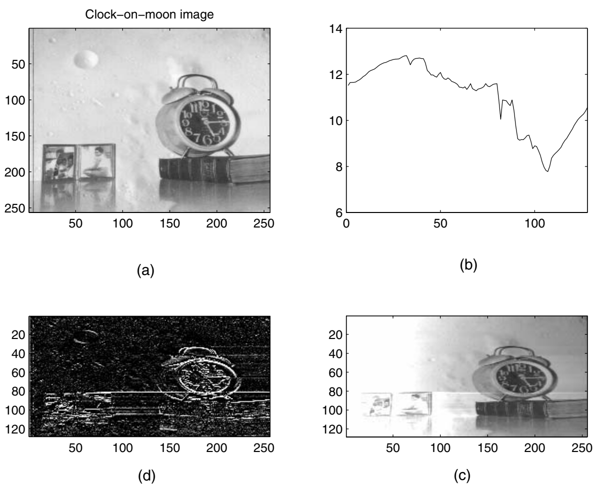

Fig. 7. (a) The image, (b) energy curve of all even row-signals, (c) image of the angular transforms generated by odd-row signals, and (d) the image of the 1-D DsiHTs applied on even-row signals.

3.1.1. *Image enhancement*

The presentation of the image in an angular form can be used for image enhancement. Our preliminary results show that many details can be observed better through angular forms defined by different paths P. Such paths can be defined to obtain desired properties of processed images. As an example, we consider the girl image (256×256) shown in Fig. 8 in part a. This image is presented in b in the form of the 2D-row angular transform (2D-R AT), i.e. 1-D angular transforms are generated by rows of the image with the natural path P and the direction from left to right for each. Similar image, when the 1-D angular transforms of rows are defined from right to left, is given in c. The angles of rotations for the 1-D heap transforms change greatly in the beginning of their paths; this is why we can notice zones with very bright first and last columns in the image in b and c, respectively. The average of these two images is given in d. Many details of the girl image can be observed in the processed images. The 2D-column heap transform of the image, when the path P of each 1-D heap transform is defined from the top to bottom, is shown in e. The average of three obtained angular transforms is given in f. It is also possible to process the image by 1-D heap transforms which are defined by paths along diagonals and other directions as well.

Figure 9 shows results of the similar processing of the tree image (256×256) by rows. It is interesting to notice in c that

Fig. 8. (a) Girl image, the 2D-R ATs with the (b) left-to-right and (c) right-to-left directions, (d) average of images in b and c, (e) 2D-R ATs with the top-to-bottom direction, and (f) average of images in b, c, and e.

a part of the ground in shadow in front of the big stone becomes more visible. This example shows that, by performing the angular transformations along specified directions, we are able to enhance images along the directions. We can also apply the concepts of the

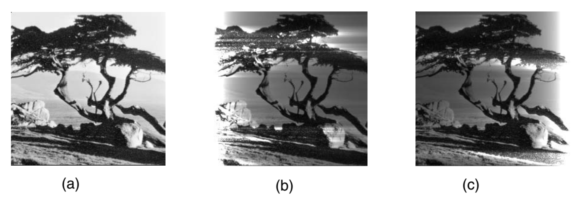

Fig. 9. (a) Tree image, the 2D-R ATs with the (b) right-to-left and (c) left-to-right directions.

2D-column and 2d-row heap transforms for processing the image block-wise, for instance by blocks 8×8 or 16×16.

3.2. Inverse angular transform

In image enhancement, other methods based on the heap and angular transforms can be developed. We dwell upon the following concept of the inverse angular representation. Every 1-D signal $\mathbf{x} = \{x_0, x_1, ..., x_{N-1}\}$ can be considered in the angular form of the heap transform (3) generated by $\mathbf{x}$. At the same time, this signal itself can be considered as the angular transform, or representation of an $(N+1)$-point signal, which we denote by $\mathbf{y} = \{y_0, y_1, ..., y_N\}$,

$$\mathbf{y} \to \mathcal{A}[\mathbf{y}] = \{||\mathbf{y}||, \mathcal{A}_N[\mathbf{y}]\} = \{||\mathbf{y}||, \mathbf{x}\}. \tag{12}$$

In other words, we assume that $x_0 = \phi_1, x_1 = \phi_2, ..., x_{N-1} = \phi_N$, where ϕ_k, $k = 1 : N$, are rotation angles of the heap transform generated by signal $\mathbf{y}$. Thus, we define the inverse angular transformation by $\mathbf{x} \to \mathcal{A}_N^{-1}[\mathbf{x}]$. In order to avoid ambiguity of the inverse transform, we assume that $||\mathbf{y}|| = 1$. In other words, the input signal is considered to be equal $\{1, \mathbf{x}\}$ and the signal $\mathbf{x}$ normalized before applying the inverse transform,

$$\mathbf{x} \to \left\{1, \frac{x_0}{||x||}, \frac{x_1}{||x||}, \ldots, \frac{x_{N-1}}{||x||}\right\}.$$

As an example, Figure 10 shows the 256-point sine signal $\mathbf{x}$ of period 128 in part a, along with the angular transform $\mathcal{A}_{255}[\mathbf{x}]$ in b, the inverse angular transform $\mathbf{y} = \mathcal{A}_{256}^{-1}[\mathbf{x}]$ of this signal in c, and the second inverse angular transform $\mathcal{A}_{256}^{-1}[\mathbf{y}]$ in d. The similar graphs for the sine signal of period 32 are given in (e)-(h). One can note the damped oscillations of the sine waves in b and f. In contrast, the inverse angular transforms of these sine waves change the sign and oscillate with increasing amplitudes. The sine waves $\mathbf{x}$ considered in a and e as the functions of angles are represented in c and g as sine-type increasing functions $\mathbf{y}$ of time. Consequent rotation of the signals $\mathbf{y}$ by the angles of the corresponding sine waves $\mathbf{x}$ result in the unit impulse. This is a time-to-time transformation $\mathbf{x} \to \mathbf{y}$. We can continue and calculate next inverse angular transforms over signals $\mathbf{y}$; however, these transforms will be changed very little, as shown in d and h.

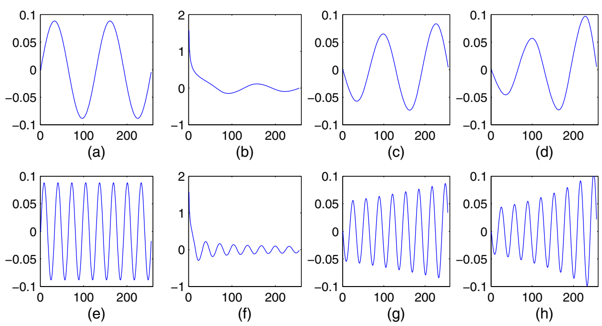

Fig. 10. (a) The sine signal of period 128, (b) angular transform of the signal, (c) inverse angular transform of the signal, and (d) the second inverse angular transform. (e) The sine signal of period 32, (f) angular transform of the signal, (g) inverse angular transform of the signal, and (h) the second inverse angular transform.

To achieve image enhancement by inverse angular transforms, we can manipulate, for instance, values of the inverse angular transforms along rows or columns and then perform the reconstruction of row- or column-signals. For instance, for the image shown in Fig. 11(a), the following simple operations have been used. The inverse angular transforms $\mathbf{y} = \{y_n\}$ of normalized row-signals have been amplified slightly by the linear function $(1 + \alpha n)$, where $\alpha = 0.005$ and $n = 0 : 255$. Then, the angular transforms applied to modified inverse angular transforms $\{(1 + \alpha n)y_n\}$ have been performed and multiplied by the corresponding normalized coefficients to compose a new image. In parts b and c, the results of such operations are given for the cases when the paths P of all transforms are defined in directions from left to right and right to left, respectively. The result of the similar image which is processed by column-signals when the paths of the angular transforms are defined from the top to bottom is shown in d. In above examples, the 1-D DsiHTs defined with angular equations $g(x, y, \varphi) = a = 0$ have been used; however, other angular equations can also be considered and applied for image enhancement.

Fig. 11. (a) Girl image, (b) 2D-row left-to-right, (c) 2D-row right-to-left, and (d) 2D-column top-to-bottom inverse angular transforms of the image.

4. CONCLUSION

Examples of composing parameterized discrete unitary transforms generated by signals have been described. The composition of such transforms is based on solving decision equations which determine the angles of rotations of basic transformations which are then used to transform the input signals. Parameters were introduced in order to collect and transfer the energy of signals in different ways to one of the components of the generator-signals. The application of the proposed transforms in the angular forms for image enhancement has been described.

5. REFERENCES

[1] N. Ahmed and K.R. Rao, *Orthogonal transforms for digital signal processing*. Berlin, Germany: Springer-Verlag, 1975.

[2] L.R. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing,* Prentice-Hall, Englewood Cliffs, 1975.

[3] A.M. Grigoryan and S.S. Agaian, *Multidimensional Discrete Unitary Transforms: Representation, Partitioning and Algorithms,* Marcel Dekker Inc., New York, 2003.

[4] D. Sundararajan and M.O. Ahmad, "Fast computation of the discrete Walsh and Hadamard transforms," *IEEE Trans. on Image processing,* vol. 7, no. 6, pp. 898-904, June 1998.

[5] S. Mitra, O. Shentov, and M. Petraglia, "A method for fast approximate computation of discrete cosine transforms," *Proc. of the IEEE ICASSP'90,* Albuquerque, NM, pp. 2025-2028.

[6] A.M. Grigoryan, "An algorithm for calculation of the discrete cosine transform by paired transform," *IEEE Trans. on Signal Processing,* vol. 53, no. 1, pp. 265-273, Jan. 2005.

[7] S.G. Mallat, *A Wavelet Tour of Signal Processing,* San Diego: Academic Press, second edition, 1998.

[8] J.C. Goswami and A.K. Chan, *Fundamentals of Wavelets: Theory, Algorithms, and Applications,* New York: Wiley, 1999.

[9] A.M. Grigoryan and M.M. Grigoryan, "Nonlinear approach of construction of fast unitary transforms," *Proc. of the 40th Annual Conference on Information Sciences and Systems (CISS 2006),* pp. 1073-1078, March 22-24, 2006, Princeton.

[10] A.M. Grigoryan and M.M. Grigoryan, "Discrete unitary transforms generated by moving waves," *Proc. of the International Conference: Wavelets XII, SPIE: Optics+Photonics 2007,* San Diego, CA, August 27-29, 2007.

[11] A.M. Grigoryan and M. M. Grigoryan, *Brief Notes in Advanced DSP: Fourier Analysis with MATLAB,* CRC Press Taylor& Fransic Group, New York, 2009.