IMAGE AND AUDIO SIGNAL FILTRATION WITH DISCRETE HEAP TRANSFORMS

Artyom M. Grigoryan and Mehdi Hajinoroozi

Department of Electrical and Computer Engineering,
The University of Texas at San Antonio, Texas USA

ABSTRACT

Filtration and enhancement of signals and images by the discrete signal-induced heap transform (DsiHT) is described in this paper. The basic functions of the DsiHT are orthogonal waves that are originated from the signal generating the transform. These waves with their specific motion describe a process of elementary rotations or Givens transformations of the processed signal. Unlike the discrete Fourier transform which performs rotations of all data of the signalon each stage of calculation, the DsiHT sequentially rotates only two components of the data and accumulates a heap in one of the components with the maximum energy. Because of the nature of the heap transform, if the signal under process is mixed with a wave which is similar to the signal-generator then this additive component is eliminated or vanished after applying the heap transformation. This property can effectively be used for noise removal, noise detection, and image enhancement.

KEYWORDS

Heap Transform, Fourier Transform, Filtration, Image Enhancement

1.INTRODUCTION

Filtration of discrete signals and images in certain frequency bands with the discrete Fourier transform is extensively used in signal and image processing systems [1]-[3]. In order to process discrete signals with the discrete Fourier transform (DFT), first the DFT of the signal is calculated, afterwards some components of the DFT are removed or modified, and finally the inverse of the DFT is calculated. For instance, in the alpha-rooting method which is one of the most popular transform-based techniquesof image enhancement, the two-dimension (2-D) DFT is modified in one or a few bright visible zones in the frequency domain[4]-[9]. The filtration of colour images in the frequency domain by the quaternion 2-D DFT is described in [10].Many other transforms such as elliptic Fourier transforms [11]-[13], Fibonacci Fourier transform [14], discrete cosine transform [1,16,17], and wavelet transforms also are used for filtering signal and images[18]-[20].The 1-D discrete Fourier transform is composed of a complex system of rotations of the input signal around different circles with following summing of all rotated points [21].The consequent and counter clockwise rotations of the obtained data are referred to as the inverse DFT.The 2-D DFT represents a more complicated system of rotations since the transform can be split by 1-D DFTs in tensor representation [22]-[26].

In this paper we analyze another system of rotations, which does not use rotation of all data on each stage of calculations. Unlike the N-point DFT when rotations are performed only by $(N-1)$

angles uniformly distributed in the interval [0,2π), the rotations in a new system are different, they are accomplished by angles defined by the given signal (or signals) [27]-[31]. These signals are called the generators of the system, or the transformation, and selection of these signals is determined by the nature of the problem under the consideration. The signal generators play an important role in processing signals and images. The discrete unitary transformation which is defined by such models of rotation is called the discrete signal-induced heap transformation (DsiHT). The complete system of basic functions of the heap transformis referred to as a system of waves generated by input signals, the waves with their specific motion in the space of functions[27]. By the unique set of angles of rotations the energy of the processed components of the signal-generator is moving to one heap located in one of the components of the transform. In the case when a similar system of rotations is applied over a signal which is mixed with the component the same as the signal-generator, this component will be eliminated at all points except the one point where the energy of the component is transformed. For instance, when the heap transform is generated by a sinusoidal wave, the transform removes the component of the signal which has the frequency equal to the sine wave. Consequently the DsiHT can be used as a linear filter, and the preliminary experimental results show that the filtration by the heap transform is comparative with the Fourier transform method. The heap transform is fast due to fast algorithms for any length of the signals and with less operations of multiplication when compared with the DFT. It is notable that the calculation of the inverse of the heap transform is not needed, because the heap transform transfers the signal into another signal and during this transformation the signal is filtered and the desired frequency componentsare removed.

The rest of the paper is organized in the following way. In Section 2, the definition of the heap transform is described. Properties of the heap transform are explained in Section 3. In Sections 4and 5,the applications of the heap transform in image and audio processing are described and examples are given.

2. Discrete Heap Transforms

In this section, the definition and examples of the DsiHT are given. The composition of the N-point discrete heap transformation, T, is based on the special selection of a set of parameters $\varphi_1, \dots, \varphi_m$, or angles, where $m \geq (N-1)$. The transformation T is considered to be separable, which means that there exist such transformations $T_{\varphi_1}, T_{\varphi_2}, \dots, T_{\varphi_m}$ that

$$T = T_{\varphi_1,\dots,\varphi_m} = T_{\varphi_m} \cdots T_{\varphi_2} T_{\varphi_1}. \qquad (1)$$

We consider the simple case when each transformation T_{φ_k}, $k \in \{1,2,\dots,m\}$, changes only two components of the vector $\mathbf{z} = (z_1, \dots, z_{N-1})$. Thus, T_{φ_k} is represented as

$$T_{\varphi_k} : \mathbf{z} \rightarrow (z_1, \dots, z_{k_1-1}, \underline{f_{k_1}(\mathbf{z}, \varphi_k)}, z_{k_1+1}, \dots, z_{k_2-1}, \underline{f_{k_2}(\mathbf{z}, \varphi_k)}, z_{k_2+1}, \dots, z_{N-1}) \qquad (2)$$

where the pair of numbers (k_1, k_2) is uniquely defined by k and $1 \quad k_1 < k_2 \quad m$. The set of all these pairs (k_1, k_2) defines the path of the transformation. For simplicity of calculations, we assume that all first functions $f_{k_1}(\mathbf{z}, \varphi_k)$ in (2) are equal to a function $f(\mathbf{z}, \varphi_k)$ and all second functions $f_{k_2}(\mathbf{z}, \varphi_k)$ equal to a function $g(\mathbf{z}, \varphi_k)$. The N-point transformation $T = T_{\varphi_1,\dots,\varphi_m}$ is defined by the basic transformations

$$T_{k_1,k_2}(\varphi_k) : \left(z_{k_1}, z_{k_2}\right) \rightarrow \left(f(z_{k_1}, z_{k_2}, \varphi_k), g(z_{k_1}, z_{k_2}, \varphi_k)\right). \qquad (3)$$

2

The selection of parameters $\varphi_k, k = 1: m$, is based on specified vector-generators $\boldsymbol{x}$, the number of which is defined through the so-called decision equations, to achieve a uniqueness of parameters and desired properties of the transformation T [27]. Here, we consider the case of two decision equations with one vector-generator.

Let $f(x, y, \varphi)$ and $g(x, y, \varphi)$ be functions of three variables; φ is referred to as the rotation parameter such as an angle, and x and y as the coordinates of a point (x, y) on the plane. The function $g(x, y, \varphi)$ is parameterized and it is assumed that for a specified set of numbers α, the equation $g(x, y, \varphi) = \alpha$ has a unique solution with respect to φ for each point (x, y) on the plane or its chosen subset.

The system of equations

$$\begin{cases} f(x, y, \varphi) = y_0 \\ g(x, y, \varphi) = \alpha \end{cases} \tag{4}$$

is called the system of decision equations [21,27]. First, the value of φ is calculated as a function $r(x, y, \alpha)$ from the second equation which is called the angular equation. Then, the value of y_0 is calculated as $y_0 = f(x, y, r(x, y, \alpha))$. It is also assumed that the two-point transformation is unitary.

$$T_\varphi : (z_0, z_1) \rightarrow (z_0', z_1') = \big(f(z_0, z_1, \varphi), g(z_0, z_1, \varphi) \big) \tag{5}$$

Example 1: Given a real number α, we consider the following functions that describe the elementary rotation:

$$\begin{cases} f(x, y, \varphi) = x \cos \varphi - y \sin \varphi, \\ g(x, y, \varphi) = x \sin \varphi + y \cos \varphi. \end{cases} \tag{6}$$

The basic transformation is defined as the rotation of the point (x, y) to the horizontal $Y = \alpha$,

$$T_\varphi : (x, y) \rightarrow (x \cos \varphi - y \sin \varphi, \alpha), \tag{7}$$

where the rotation angle φ is calculated by

$$\varphi = \arccos \left(\frac{\alpha}{\sqrt{x^2 + y^2}} \right) - \arctan \left(\frac{x}{y} \right). \tag{8}$$

The input signal $\boldsymbol{z}$ is processed in the same order, or path P, as the vector-generator $\boldsymbol{x}$ when composing the heap transform. When the path is ordinary, the signal is processed as shown in Figure 1 for the five-point signal $\boldsymbol{x}$. The first pair to be processed is (x_0, x_1), the next is (y_0, x_2), then (y_0, x_3) with a new value of y_0, and so on. The first component of the vector is renewed and

participates in the calculation of all $(N–1)$ basic transformations $T_k = T_{\varphi_k}, k = 1 : (N - 1)$. Therefore we can write that $y_0 = x_0^{(k)}$ on the stage k. The transform of the vector-generator $\mathbf{x}$ is

$$T(\mathbf{x}) = (y_0, \alpha_1, \alpha_2, \dots, \alpha_{N-1}), \qquad \left(y_0 = x_0^{(N-1)} \right). \tag{9}$$

Now we consider the case when all parameters $\alpha_k = 0$, i.e., the whole energy of the vector $\boldsymbol{x}$ is collected in one heap and then transferred to the first component. In other words, we consider the Givens rotations of vectors, or points (y_0, x_k), on the horizontal $Y = 0$. Figure 2 shows the network of the transform of the signal $\mathbf{z} = (z_0, z_1, z_2, \dots, z_{N-1})'$. The parameters (angles) of the transformation are generated by the signal-generator $\boldsymbol{x}$. In the 1st level and the kth stage of the flow-graph, the angle φ_k is calculated by the inputs $(x_0^{(k-1)}, x_k)$ where $k \in \{1, N - 1\}$ and $x_0^{(0)} = x_0$. This angle is used in the transform $T_k = T_{\varphi_k}$ to define the next component $x_0^{(k)}$ and to transform the input signal $\mathbf{z}$ in the 2nd level. The full graph represents a coordinated network of transform of the vector $\mathbf{z}$, under the action on $\boldsymbol{x}$.
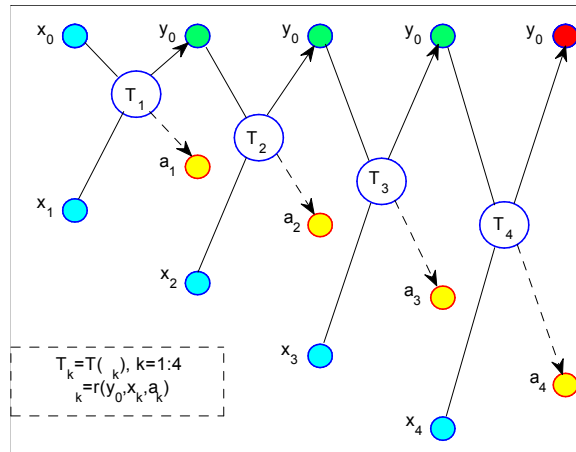

Fig.1. Signal-flow graph of determination of the five-pointtransformation by a vector
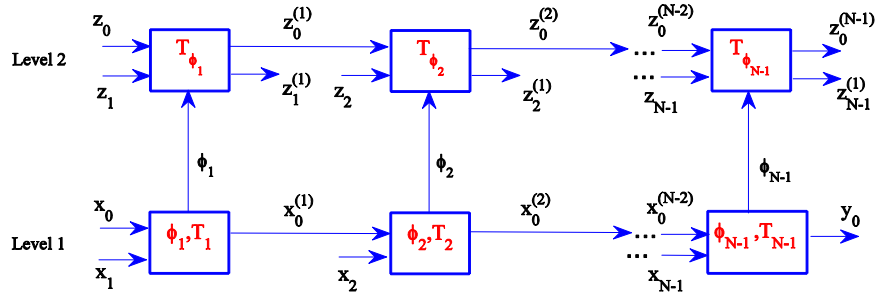$$\mathbf{x} = (\mathbf{x_0}, \mathbf{x_1}, \mathbf{x_2}, \mathbf{x_3}, \mathbf{x_4})'.$$

Fig. 2. Network of the N-point x-induced DsiHT of the signal z.

Example 2: In the $N = 6$ case when the generator $x = (1, -1, 1, -1, 1, -1)'$, the matrix $\boldsymbol{T}$ of the heap transformation generated by this vector can be written as $\boldsymbol{T} = \boldsymbol{DM}$, with the integer matrix

$$M = \begin{bmatrix} 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ -1 & 1 & 2 & 0 & 0 & 0 \\ 1 & -1 & 1 & 3 & 0 & 0 \\ -1 & 1 & -1 & 1 & 4 & 0 \\ 1 & -1 & 1 & -1 & 1 & 5 \end{bmatrix}.$$

The diagonal matrix $D = \text{diag}\{0.4082, 0.7071, 0.8165, 0.8660, 0.8944, 0.9129\}$; the angles φ_k are $\{0.7854, -0.6155, 0.5236, -0.4636, 0.4205\}$ in radians.

The six basic waves composing the matrix T of the six-point heap transformation are shown in Fig. 3. There are three stages which can be separated during the process of motion and transformation of one basic function into another one, when starting from the wave-generator. In part (a), the first stage, the static stage, the generator itself is lying as the basis function. The second stage from (b) through (e), the evolution stage, is related to the formation of a new wave.

For example, the wave $m_3 = [-1,1,2,0,...,0]$ in (c) can be described as the sum of the wave $m_2 = [1,1,0,0,...,0]$ in (b) with its shift $\hat{m}_2 = [0,1,1,0,...,0]$ as $m_3 = 2\hat{m}_2 - m_2$. The last stage (f) 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 is a splash with high amplitude.
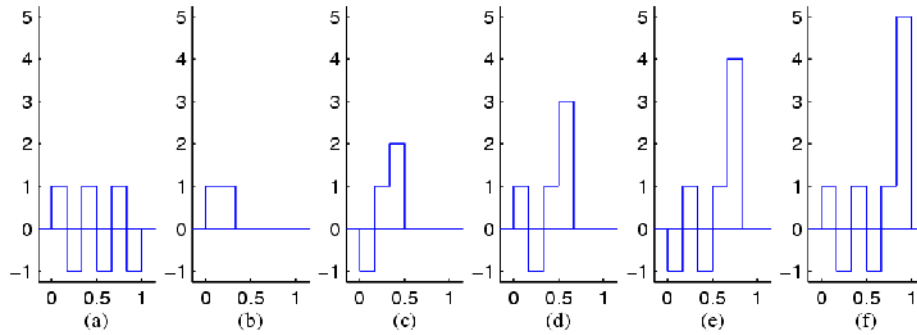

Fig. 3. The basic functions of the six-point DsiHT.

The inverse DsiHT is described by the similar process when in the static stage the last basic function of the DsiHT is used, as shown in Fig. 4.
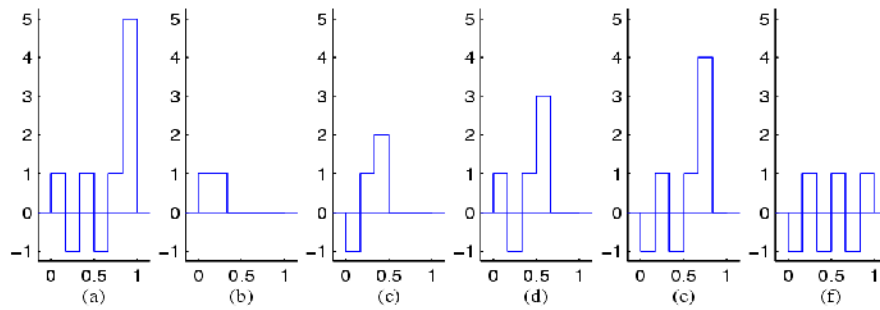

Fig. 4. The basic functions of the six-point inverse DsiHT.

3. PROPERTY OF THE DSIHT

The DsiHT generated by the vector-generator **x** operates over the generator and the input signal **z** as

$$T(\mathbf{x}) = (\|\mathbf{x}\|, 0, 0, \dots, 0)',$$
$$T(\mathbf{z}) = (z_0^{(N-1)}, z_1^{(1)}, z_2^{(1)}, \dots, z_{N-1}^{(1)})'. \tag{10}$$

It follows that the original vector-generator is almost dissolved, or eliminated, in its transform. Since the DsiHT is the linear transformation, it is clear that if the input signal **z** contains an additive component similar to the generator, this component will be eliminated but its energy will be preserved in the first point. This property of the heap transform can be illustrated on examples. Figure 5 shows the original discrete-time signal **z** in part a, which has been obtained by sampling the function

$$z(t) = 3\sin(t) - 2\cos(8t), \qquad t \in [0, 2\pi],$$

at 512 equidistant time-points of the interval $[0, 2\pi]$. We consider two heap transforms generated by the waves of frequencies $\omega_1 = 8$ and $\omega_2 = 1$ (in rad/s) which are the carrier frequencies of the signal $z(t)$. Two vector-generator$\mathbf{x}_1$ and $\mathbf{x}_2$ of length 512 each, that correspond to the sampled waves $\cos(8t)$ and $\sin(t)$, are shown in b. The 512-point discrete heap transform of the signal z, which was generated by the discrete-time signal $\mathbf{x}_1$, is shown in c, and the 512-point heap transform of the signal **z**, which was generated by $\mathbf{x}_2$, is shown in d. One can notice that the transform of the signal by generator $\mathbf{x}_1$ deletes (or filters) the contribution of the cosine wave of frequency $\omega_1 = 8$. A wave of the high frequency presents with a small magnitude in the beginning of the transform, but the remaining part is similar to the wave $\sin(t)$. The application of the transform generated by $\mathbf{x}_2$ illuminates the frequency $\omega_2 = 1$ and results in a wave which is similar to the wave $\cos(8t)$.
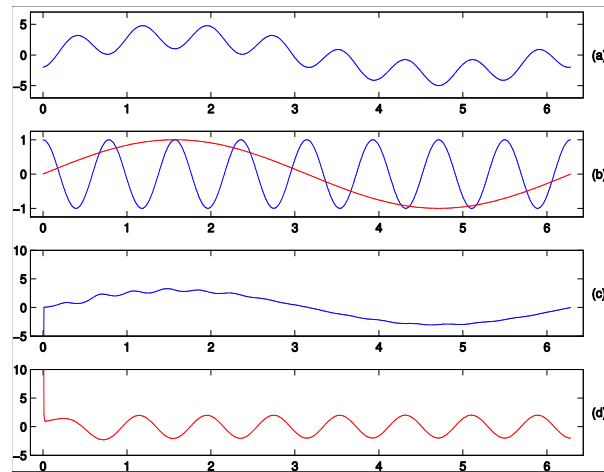

Fig. 5. (a) Original 512-point discrete signal, (b) generators$x_1(t) = cos(8t)$ and$x_2(t) = sin(t)$, and the 512-point (c)x_1-induced and (d) x_2-induced heap transforms of the signal.

For comparison, the traditional method of the Fourier transform is shown Figure 6. The original signal is shown in part a, along with the magnitude of the DFT in the small interval of frequencies

[−0.25,0.25] in b, and the result of filtration of two low frequencies in c. This method requires computation of the direct and inverse DFTs. We note that the fast N-point DFT uses $\mu(N) = N/2(\log_2 N - 3) + 2$ complex operations of multiplication [21], or half of this number when the signal is real. Therefore, the number of complex multiplications for the direct and inverse DFTs approximately equals $\mu_F(N) = 1.5\,\mu(N) = 3N/4(\log_2 N - 3)$ or $3\mu_F(N)$ real operations of multiplication. Here, we assume that one complex multiplication uses three (not four) real operations of multiplication. The DsiHT uses only $\mu_H(N) = 5(N-1)$ real multiplications, and therefore the ratio of these numbers is estimated as $r(N) = 3\mu_F(N)/\mu_H(N) \approx 9/20(\log_2 N - 3)$. In the considered $N = 512$ case, the corresponding saving in operations of multiplication is estimated as $r(512) \approx 2.7$.
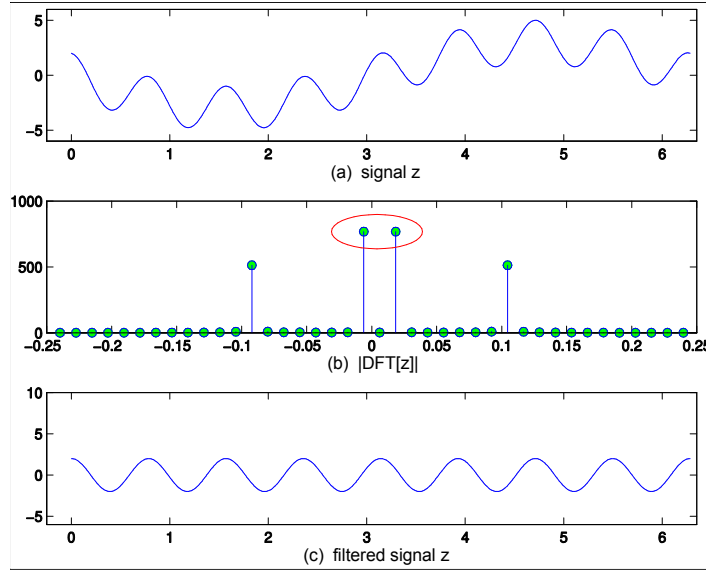

Fig. 6. (a) The original 512-point discrete signal, (b) the magnitudeof the Fourier transform of the signal, and (c)the signal afterfiltering the low frequencies.

It is important to note that the basic transformations $T_k = T_{\varphi_k}$, $k = 1 : (N-1)$, that compose the N-point DsiHT can be performed without calculation of angles and trigonometric functions. Analytical formulas can be derived for calculation of coefficients of the DsiHT [27,28]. For that, we introduce the following notations which represent respectively the partial cross-correlation of z with the vector generator x and energy of x:

$$E_k(\mathbf{z}, \mathbf{x}) = z_0 x_0 + z_1 x_1 + \cdots + z_{k-1} x_{k-1},$$
$$E_k^2(\mathbf{x}) = E_k(\mathbf{x}, \mathbf{x}) = x_0^2 + x_1^2 + \cdots + x_{k-1}^2, \tag{11}$$

where $k = 1 : (N-1)$. The components of the DsiHT on the$k$th iteration can be expressed by thecorrelation data as

$$z_k^{(1)} = \frac{E_k(\mathbf{x}, \mathbf{x}) z_k - E_k(\mathbf{z}, \mathbf{x}) x_k}{E_{k+1}(\mathbf{x}) E_k(\mathbf{x})}. \tag{12}$$

On the final stage, the value of the first component is definedby

$$z_0^{(N-1)} = E_N(\mathbf{z},\mathbf{x})/E_N(\mathbf{x}) \tag{13}$$

which is the correlation coefficient of the input signal $\mathbf{z}$ with the normalized generator $\mathbf{x}$. For a given generator, all values of $E_k(\mathbf{x},\mathbf{x})$ and $E_{k+1}(\mathbf{x})E_k(\mathbf{x})$, $k = 0 : (N-1)$, can be calculated in advance. One can consider that $E_N(\mathbf{x},\mathbf{x}) = 1$. This calculation requires N real multiplications for $E_k(\mathbf{z},\mathbf{x})$ and $3(N-1)$ multiplications and divisions for $z_k^{(1)}$, $k = 1 : (N-1)$. The last value $z_0^{(N-1)} = E_N(\mathbf{z},\mathbf{x})$ and the total number of real multiplications equals $\mu_H(N) = 4N - 3$. This estimation allows for saving more operations of multiplication when comparing with the method of the DFT.

We now consider another example. Figure 7 shows the N-point discrete signal $y(t)$ in the interval $[0, 2\pi]$ in part a, when $N = 512$. The noise signal is composed by three frequencies $\omega_1 = 2$, $\omega_1 = 9$, and $\omega_3 = 32$ as

$$n(t) = \cos(\omega_1 t) + 3\sin(\omega_2 t) + 5\sin(\omega_3 t)$$

with 512 sampled points $t = t(k) \in [0, 2\pi]$, $k = 0 : 511$. The noisy signal $z(t) = y(t) + n(t)$ is shown in b. The mean-square-root error of this degradation is 0.1847. The magnitude of the discrete Fourier transform of the noise is shown in c. The DFT of the noise has large values $(0.2091 - 1.5275i)10^3$ and $(0.2091 + 1.5275i)10^3$ at the frequency-points $33 = \omega_3 + 1$ and $481 = (N - \omega_3) + 1$ when counted from 1 to N. The result of filtration of the high frequency of the noise in the frequency domain is shown in d. For that, two components of the 512-point DFT of $z(t)$ at frequency-points 33 and 481 were zeroed, and then, the inverse DFT was calculated. The mean-square-root error of the filtration equals 0.1062.
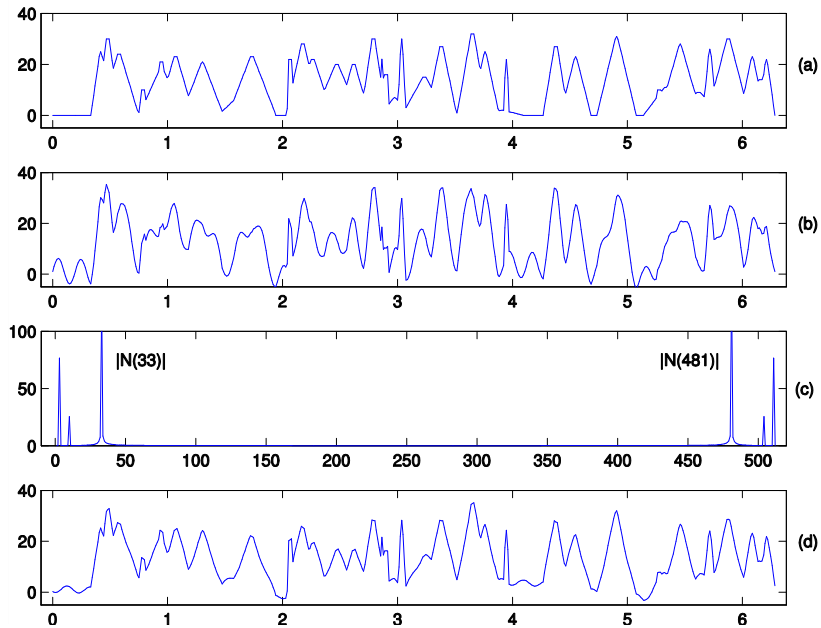

Fig. 7.(a) The original signal, (b) the noisy signal, (c) the magnitude of DFT of the noise, and (d) the filtered signal.

The results of processing the same signal by the heap transformare shown in Figure 8. In part a, the heap transform of the noisy signal $z(t)$ is shown when the vector-generator $\boldsymbol{x}$ is the high frequency wave $\sin(\omega_3 t)$, along with the filtered signal by the DFT in b, for comparison. The large values of the DsiHT at the first points were zeroed. The mean-square-root error of the filtration by the heap transform equals 0.1068. This error is a little greater than the error of the DFT and the results of both transforms are very close. It is difficult to notice the difference between these two different applications.We consider these two results of filtration in more detail. Let us denote by $y_F(t)$ and $y_T(t)$ the results of the filtration by the DFT and DsiHT, respectively. Figure 9 shows the difference signal $d_F(t) = y_F(t) - y(t)$ in part a, along with the difference signal $d_T(t) = y_T(t) - y(t)$ in b. These two difference signals are shown together in c.
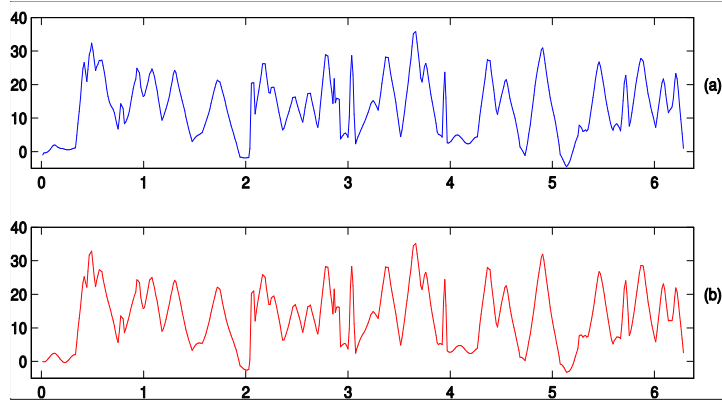

Fig. 8. (a) The DsiHT generated by $\sin(\omega_3 t)$ and (b) the method of filtration by the DFT.

A big error of the filtration by the DsiHT is in the beginning, namely in the interval (0.5,1), then it approximates the signal $y(t)$ with the smaller error 0.1125 than the DFT does, 0.1135.
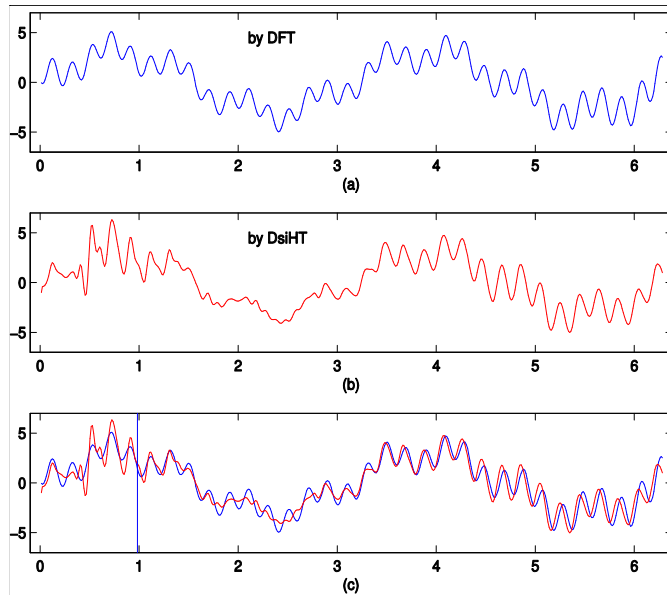

Fig. 9.Signals (a) $d_F(t)$, (b) $d_T(t)$, and (c) $d_F(t)$ and $d_T(t)$.

9

Figure 10 shows in part d the result of the heap transform of the signal in c, which is the sum of the signal in a, and the short sine wave in b in the interval [3.0740,3.8117].
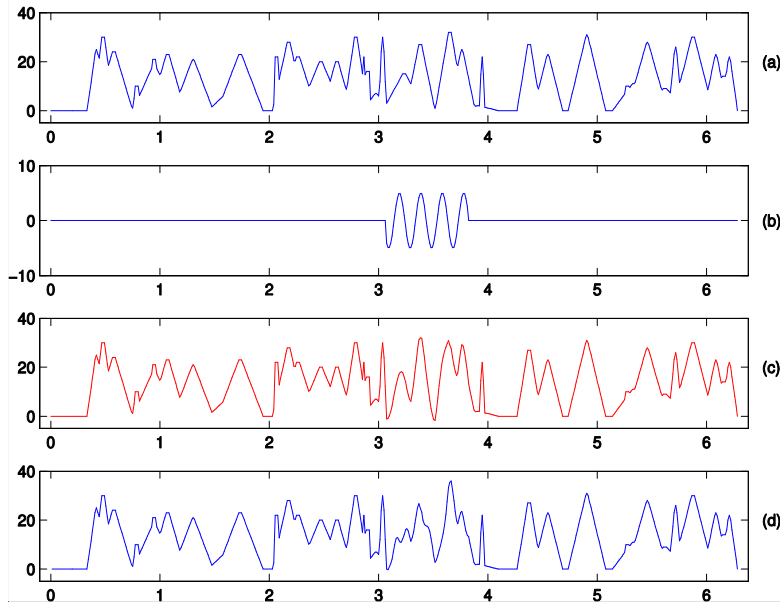

Fig. 10. (a) The 512-point discrete signal $y(t)$, (b) short sine wave $x(t)$, (c) degraded signal $z(t) = y(t) + x(t)$, and (c) heap transform of $z(t)$ when the generators is $x(t)$.

Now we assume that a signal **z** is composed by sine waves and an impulse noise. As mentioned above, if a wave **x** of a frequency ω does not lie in the signal **z** then the heap transform of the signal, $T_x[z]$, which is generated by the wave **x**, will run over the field of this wave without change. And opposite, if the signal carries frequency ω then the component of the signal will be dissolved partially in the field generated by wave of this frequency. This property can be used for noise detection.
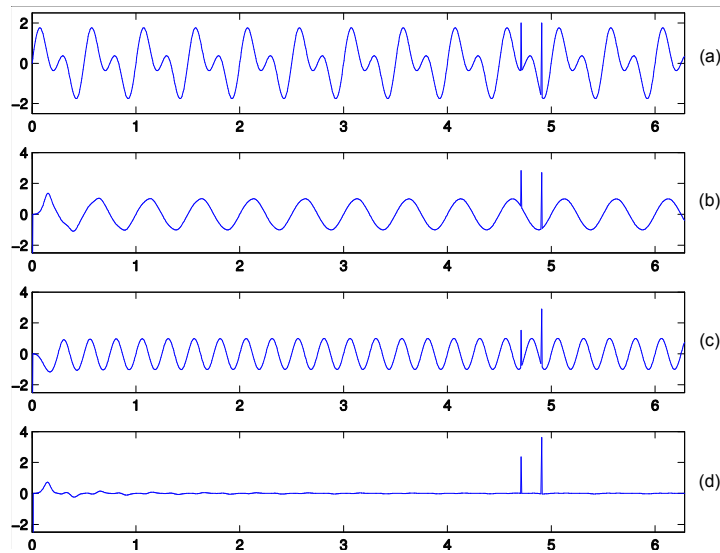

Fig. 11. (a) The input signal DsiHTs of the signal when the generator is (b) $\sin(8\pi t)$, (c) $\sin(4\pi t)$, and (d) the DsiHT of the signal in (b) when the generator is $\sin(4\pi t)$.

As an example, Figure 11 shows the 8192-point discrete-time signal

$$z(t) = \text{sins}(4\pi t) + \sin(8\pi t) + 2[\delta(t - t_1) + \delta(t - t_2)]$$

where t∈[0, 2π].The locations of two impulses are at the points $t_1 = 4.7124$ and $t_2 = 4.8597$. In parts b and c, the heap transforms of this signal are shown when the generators are $x_1(t) = \sin(8\pi t)$ and $x_2(t) = \sin(4\pi t)$, respectively. One can see that the component of the signal with the frequency 4π is removed from the signal in b, and the component of the signal with the frequency 8π is removed from the signal in c. In both cases, the impulse noise is well preserved. In part d, the heap transform generated by $x_1(t)$ is applied to the filtered signal of b which removes the frequency 4π from the signal, and it shows well the noise signal.

4. Filtration of 2-D Images

In this section, some examples which show the application of the DsiHT for filtration of noisy images and noise detection are described. Images are two dimensional signals and are considered as sets of rows or/and sets of columns and each individual row and /or column is processed by the 1-D signals being the generator of the heap transform. There are many other ways to process the images.For instance, with tensor representation [21]-[26], the image is the sum of the direction images and each of these images can be processed by the same or different DsiHTs. It means that the image can be processed separately along different directions. It is notable that selection of the path for the dish has an important role when the heap transform is used in image/signal processing. Here, the simple row-wise and column-wise methods of image processing are considered. In fact, each row or column of the image is processed by the 1-D heap transform generated by the signal which carries the frequencies that we would like to remove from the image. In noise detection, the original rows/columns of the image may play the role of signal generators for the heap transformations.

The following pseudo code of row/column wise heap transform of the image of size $N \times N$ is provided.

```
Pseudo code of row/column wise heap transform
for row(or column)=1:N
    z = noisy_input_signal // each row/column
    x = periodic_noise
    for k=1: N-1
        corr(x,z,k) // cross-correlation
        corr(x,x,k) // autocorrelation
        Z11(k) =- (corr(z,x)x(k)-corr(x,x)z(k))
        Z1(k) = Z11(k) / √corr(x,x,k+1)corr(x,x,k)
    end
    Z0(N-1) = Corr(x,z,N)/ √corr(x,x,N)
    new_row = Z
end
```

We consider different images processed by DsiHTs which are generated by the sinusoidal waves with different frequencies. As an example, Figure 12 shows the tree image of size 256×256 in part a, which is processed row-wisely by the 1-D heap transform generated by the sampled waves $x_1(t) = \cos(4t)$ and $x_2(t) = \cos(16t)$ where t∈[−π,π], which are shown together in part b. The

tree image processed by the 1-D heap transform generated by the signal $x_1(t)$ of low frequency $\omega_1 = 4$ is shown in c. The image processed by the 1-D heap transforms generated by the signal $x_2(t)$ of four times higher frequency $\omega_2 = 4\omega_1 = 16$ is shown in part d. The image is now without the high frequency ω_2.
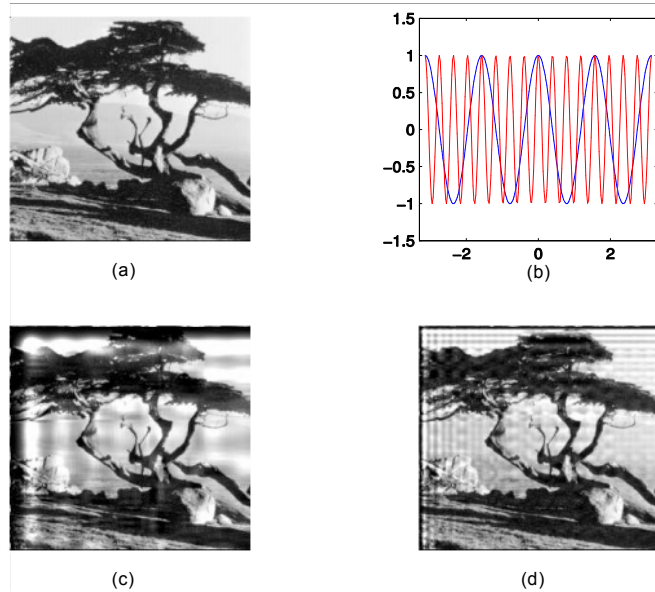

Fig. 12. (a) The image, (b) two cosine waves of frequencies 4 and 16, and the image processed by the DsiHTs generated by (c) the 1st and (d) the 2nd cosine waves.

The discrete heap transform is a powerful tool for image processing and can be useful in cleaning and repairing the images degraded with various kinds of noises. As a matter of fact the heap transform can be designed to be a notch filter and it is able to remove unwanted signals and noises from damaged images. Periodic noise is one of the noises that can affect the images which are facing periodic phenomena [32]. Figure 13 shows the Barbara image in part a, along with the image contaminated with the sinusoidal $sin(64t)$ noise along the columns in part b, and the filtered Barbara image in c. The filtration has been accomplished by the heap transform. In fact, each column of the noisy Barbara image is considered as the input signal **z** of the heap transform and vector-generator of the heap transform $\boldsymbol{x}$ equals $sin(64t)$.
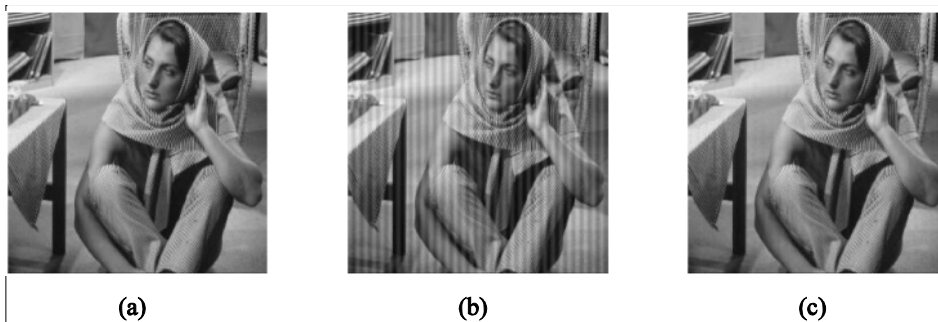

Fig. 13. (a) The Barbara image, (b) the image corrupted with $sin(64t)$ along the columns, and (c) the image filtered by the $sin(64t)$-induced heap transform.

Figure 14 shows another example with the clock image in part a. In part b, the image is corrupted with the sine wave of a the high frequency, $\sin(64t)$, along the rows. With consideration of the wave $\sin(64t)$ as a vector generator x and each row of the noisy tree image as input signal z of the heap transform, the image is filtered and the periodic noise is removed from the tree image. The filtered image with the sine-induced DsiHT is depicted in c.
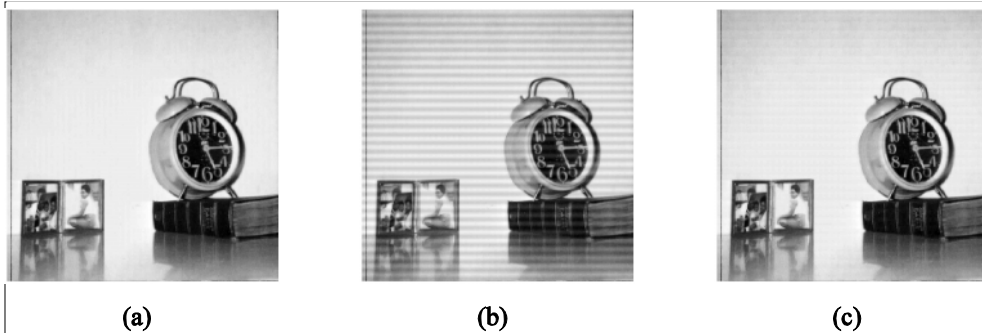

(a)   (b)   (c)

Fig. 14. (a) The clock image, (b) the image corrupted with $sin(64t)$ along the rows, and (c) the image filtered by the $sin(64t)$-induced heap transform.

In Figure 15 the Cameraman image is shown in part a, and the image contaminated with the $sin(64t)$ wave along the rows is depicted in part b. With application of the heap transform induced by the vector x which equals $sin(64t)$, and with consideration of each row of the corrupted image as input signal z, the image is filtered and the noise is removed. The filtered image is shown in part c.The similar filtered image is shown in part d where the imagecorrupted with the low frequency wave $sin(8t)$ along the rows is processed by the $sin(8t)$-induced heap transform.
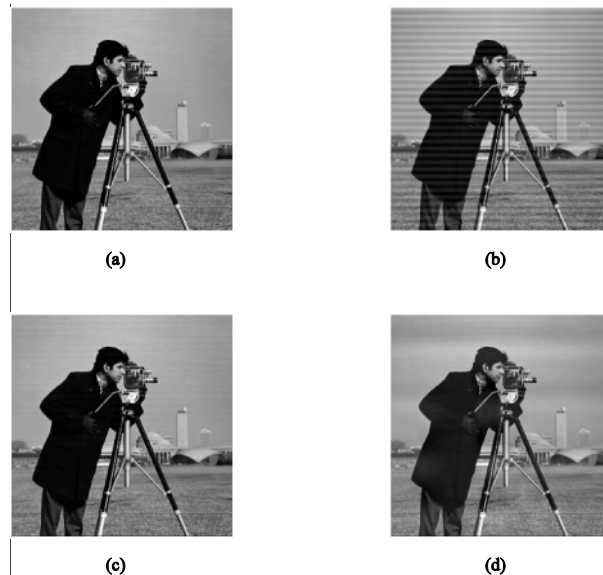

(a)   (b)

(c)   (d)

Fig. 15. (a) The Cameraman image, (b) the image corrupted with sin(64t) along the rows, (c) the filtered image by the sin(64t)-induced heap transform, and (d) the filtered image by the sin(8t)-induced heap transform over the image corrupted with sin(8t) along rows.

In both cases, the results with restored Cameraman images show that the heap transform removes the periodic noise from the corrupted images; the heap transform serves here as a notch filter. As mentioned in the previous section, the noise can be detected in a signal with the application of the heap transform.

Figure 16 shows the Cameraman image in part a, along with the image corrupted with a Gaussian noise with the distribution $N(0,0.1)$ in b. Each row of the original image is considered as a vector-generator x and each row of the noisy image as the input signal z. After application of the heap transform over all rows of the noisy image, the original image has been removed and only the noise has been remained. Part c shows the detected noise.
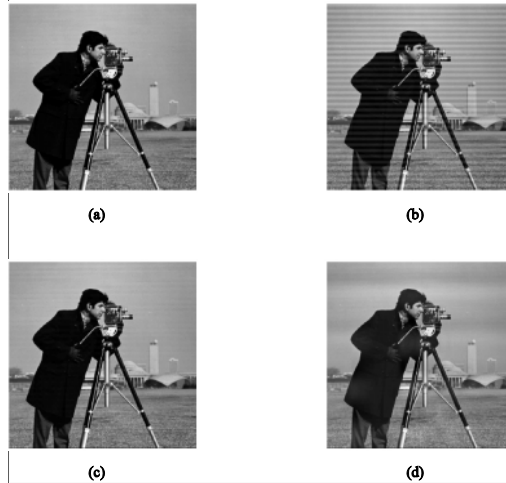

Fig. 15. (a) The Cameraman image, (b) the image corrupted with sin(64t) along the rows, (c) filtered image by the sin(64t)-induced heap transform, and (d) filtered image by the sin(8t)-induced heap transform over the image corrupted with sin(8t) along the rows.
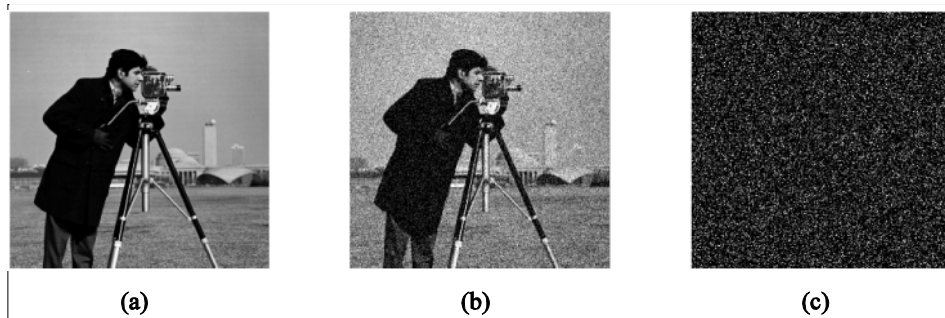

Fig. 16. (a) The Cameraman image, (b) the image corrupted with a Gaussian noise, and (c) the detected Gaussian noise from the corrupted image with the help of the DsiHT (the noise image was scaled).

In another example, the Lena image corrupted with the $sin(64t)$ periodic noise along the 45° direction is used. Considering the suitable vector-generator x and input signal z, the result is the filtered and noise free image, or detected image noise. Figure 17 shows the original image in part a, the noisy image in b, the filtered image in c, and the detected noise in d.
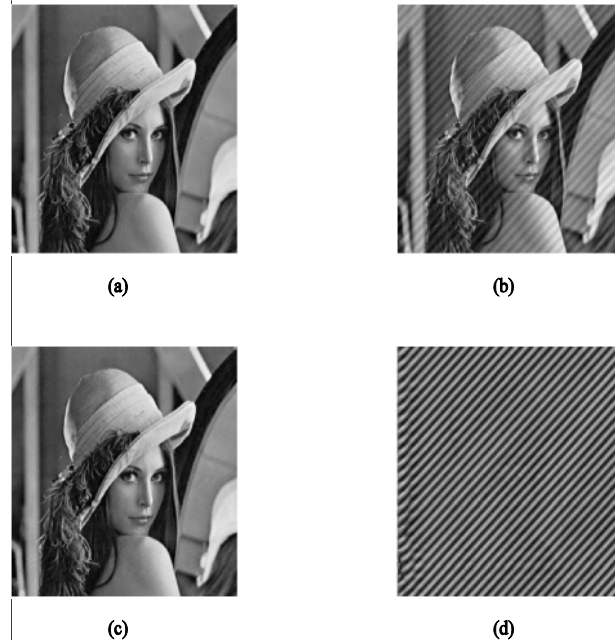
Fig. 17. (a) Lena image, (b) the image corrupted with $\sin(64t)$ along the 45° direction, (c) the filtered image by the $\sin(64t)$-induced heap transform, and (d) the detected periodic noise.

5. Filtration of 1-D audio signals

Similar to images the audio signals can be processed with the help of the discrete heap transformations. Audio signals are 1-D and the application of heap transformation on audio signals to remove an unwanted or noise signal mixed with the original signal is done through the consideration of an additive signal as the signal generator x of the heap transform and a noisy signal as the input signal z.

Heap transformation successfully and fast removes a periodic additive noise from the original audio signal and as a result a pure audio signal is ready for post processing. The experimental results confirm the precise and fast function of the heap transformation for filtration of the audio signals. The first example is depicted in Figure 18. In part a, only 256 samples of the original audio signal are shown. In part b the audio signal is mixed with the $\sin(32t)$ signal. In part c the filtered signal with heap transformation is given. The result shows a perfect filtration of the additive noise signal.

Another example is depicted in Figure 19. In part a, only 256 samples of the original audio signal are shown. In part b the audio signal is mixed with the $\sin(64t)$ signal. In part c the filtered signal with heap transformation is shown. A perfect filtration of the additive noise signal is obtained.
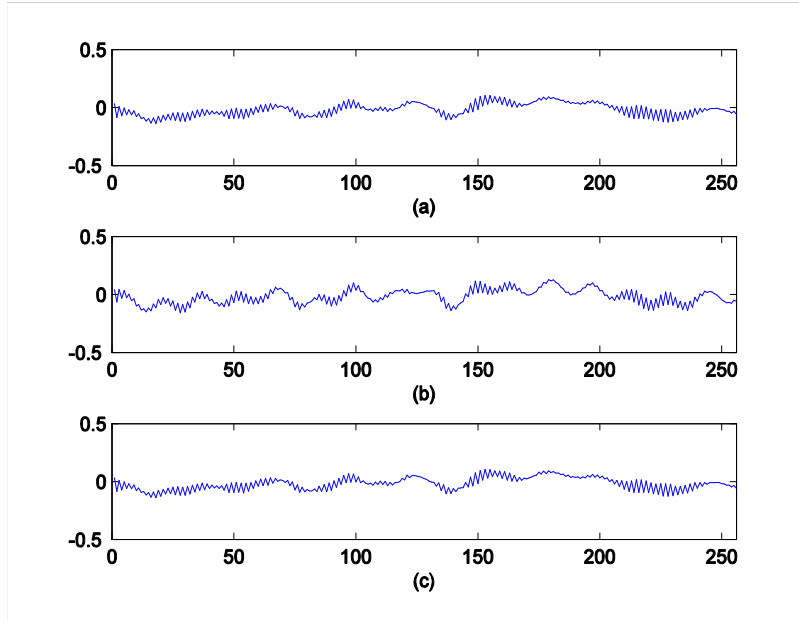
Fig. 18. (a) The 256 samples of the original audio signal, (b) the audio signal mixed with the wave $\sin(32t)$, and (c) the signal after filtering with the heap transformation.
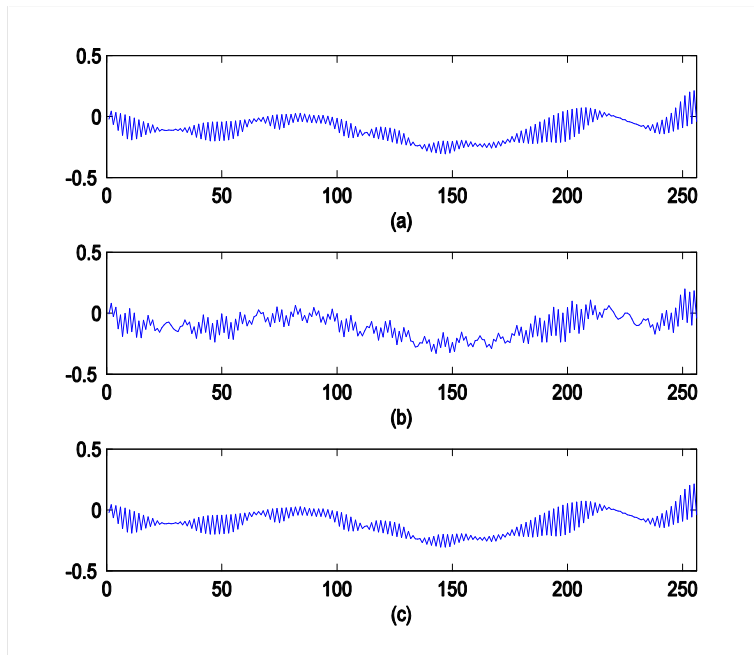
Fig. 19. (a) The 256 samples of the original audio signal, (b) the audio signal mixed with thewavesin$(64t)$, and (c) the filtered signal by the heap transformation.

6. CONCLUSIONS

The heap transformations represent a class of discrete unitary signal-induced transformations which are defined by systems of moving functions. The movement of the basic functions which is accomplished with rotation and angular representation is defined for the desired signal. The

16

transforms are fast, because of a simple form of decomposition of their matrices, and they can be applied to signals of any length. The heap transform can be effectively used for filtering signals and images. If the input signal is mixed with an additive component which is similar to the generator, this component is eliminated in the transform of the signal. When such component is the wave of a given frequency, this wave is eliminated in the heap transform. The method of heap transform is simple and requires less operations than the method of the Fourier transform. The heap transformation transfers the signal into another signal and it may filter a certain frequency during this transformation. Therefore, the calculation of the inverse heap transform is not required when filtering the signal. The preliminary experimental results show that the filtration with the discrete heap transform is an effective way for filtering the contaminated and damaged images and audio signals and also for detecting noises in images and audio signals. The discrete heap transform can also be usedeffectively in processing of color images.

REFERENCES

[1]   K. Jain, (1989) Fundamentals of Digital Image Processing, Prentice Hall, Inc. Englewood Cliffs, NJ.
[2]   S.K. Mitra, (2006) Digital Signal Processing, A Computer- Based Approach, McGraw Hill, New York.
[3]   R.C. Gonzalez and R.E. Woods, (2001) Digital Image Processing, Prentice Hall, New York.
[4]   J.H. McClellan, (1980) "Artefacts in alpha-rooting of images," Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing, pp. 449-452.
[5]   R. Kogan, S. Agaian, and K.A. Panetta, (1998) "Visualization using rational morphology and zonal magnitude reduction," Proc. IS & T/SPIE's Symposium on Electronic Imaging Science & Technology, vol. 3304, IX, pp. 153-163.
[6]   S.S. Agaian, K. Panetta, and A.M. Grigoryan, (2001) "Transform-based image enhancement algorithms," IEEE Trans. on Image Processing, vol. 10, no. 3, pp. 367-382.
[7]   A.M. Grigoryan, and S. Agaian, (2003) "Tensor form of image representation: enhancement by image-signals," IS&T/SPIEs Symposium on Electronic Imaging Science & Technology.
[8]   A.M. Grigoryan and S.S. Agaian, (2004) "Transform-based image enhancement algorithms with performance measure," Advances in Imaging and Electron Physics, Academic Press, vol. 130, pp. 165-242.
[9]   F.T. Arslan and A.M. Grigoryan, (2006) "Fast splitting alpha-rooting method of image enhancement: Tensor representation," IEEE Trans. on Image Processing, vol. 15, no. 11, pp. 3375-3384.
[10] A.M. Grigoryan and S.S. Agaian, (2014) "Alpha-rooting method of color image enhancement by discrete quaternion Fourier transform," [9019-3], SPIE proceedings, 2014 Electronic Imaging: Image Processing: Algorithms and Systems XII.
[11] S.S. Agaian, B. Silver, and K.A. Panetta, (2007) "Transform coefficient histogram-based image enhancement algorithms using contrast entropy," IEEE Trans. on Image Processing, vol. 16, no. 3, pp. 741–758.
[12] A.M. Grigoryan, (2009) "Elliptic discrete Fourier transforms of Type II," Proc. IEEE International Conference on Systems, Man, and Cybernetics (SMC 2009), pp. 954-959.
[13] A.M. Grigoryan, (2011) "Two classes of elliptic discrete Fourier transforms: Properties and examples," Journal of Mathematical Imaging and Vision (0235), vol. 39, pp. 210-229.
[14] J. Xia, K.A. Panetta, and S.S. Agaian, (2011) "Color image enhancement algorithm based on logarithmic transform coefficient histogram shifting," in IS&T/SPIE Electronic Imaging 2011: Image Processing: Algorithms and Systems IX, pp. 806316-1 - 806316-12.
[15] S. Agaian, M.-C. Chen, and C.L.P. Chen, (2008) "Noise reduction algorithms using Fibonacci Fourier transforms," IEEE System, Man and Cybernetics Conference, Singapore, pp. 1048-1052.
[16] J. Xia, K.A. Panetta, and S. Agaian, (2011) "Color image enhancement algorithm based on DCT transforms," Proc. Man and Cybernetics, SMC 2011. IEEE International Conference, pp. 1496 – 1501.
[17] K. Panetta, J. Xia, and S. Agaian, (2012) "Color image enhancement based on the discrete cosine transform coefficient histogram," Journal of Electronic Imaging, vol. 21(2), pp. 021112-1, 021112-17.

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

[19] J. Xia, K. Panetta, and S. Agaian, (2010) "Wavelet transform coefficient histogram-based image enhancement algorithms," Proc. SPIE vol. 7708, 770812, 12 pages.

[20] C.B. Smith, S.S. Agaian, (2004) "Class of Fibonacci-Daubechies-4-Haar wavelets with application to ECG de-noising," Image Processing Algorithms and Systems III, Proceedings of SPIE-IS&T Electronic Imaging, SPIE vol. 5298, pp. 25-36.

[21] A.M. Grigoryan and M.M. Grigoryan, (2009) Brief Notes in Advanced DSP: Fourier analysis with MATLAB, CRC Press Taylor and Francis Group.

[22] A.M. Grigoryan, (2001) "2-D and 1-D multi-paired transforms: Frequency-time type wavelets," IEEE Trans. on Signal Processing, vol. 49, no. 2, pp. 344-353.

[23] A. Grigoryan, S. Agaian, and E. Dougherty (1999), "Splitting of the 2-D convolution: Optimal linear filtrating," Second International Conference on Information, Communications & Signal Processing, ICICS'99, Singapore, December, pp. 7-10.

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

[25] A.M. Grigoryan and N. Du, (2010) "2-D images in frequency-time representation: Direction images and resolution map," Journal of Electronic Imaging, vol. 19, no. 3, pp. 033012-1 - 033012-14.

[26] A.M. Grigoryan and N. Du, (2011) "Principle of superposition by direction images," IEEE Trans. on Image Processing, vol. 20, no. 9, pp. 2531-2541.

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

[28] A.M. Grigoryan and M.M. Grigoryan, (2008) "Discrete signal induced unitary transforms," in book Computer and Simulation in Modern Science (Editor-in-Chief: Prof. Nikos Mastorakis), vol. 1, pp. 26-31, Mathematics and Computers in Science and Engineering, A series of Reference Books and Textbooks, WSEAS Press.

[29] A.M. Grigoryan and M.M. Grigoryan, (2007) "Discrete unitary transforms generated by moving waves," [6701-25] Proc. of the International Conference: Wavelets XII, SPIE: Optics+Photonics 2007, vol. 6701, 670125, pp. 27-29.

[30] A.M. Grigoryan and K. Naghdali, (2009) "Fast unitary heap transforms: Theory and application in cryptography," [7351-16], Proc. of International Conference Mobile Multimedia/Image Processing, Security, and Applications 2009, DSS09 SPIE Defense, Security, and Sensing.

[31] K. Naghdali, R. Raghunath, and A.M. Grigoryan, (2009) "Fast signal-induced transforms in image enhancement," Proc. of IEEE International Conference on Systems, Man, and Cybernetics, pp. 565-570.

[32] A. McAndrew, (2004) An Introduction to Digital Image Processing with MATLAB Notes for SCM2511 Image Processing 1, School of Computer Science and Mathematics Victoria University of Technology.

Authors

Artyom M. Grigoryan received the MS degrees in mathematics from Yerevan State University (YSU), Armenia, USSR, in 1978, in imaging science from Moscow Institute of Physics and Technology, USSR, in 1980, and in electrical engineering from Texas A&M University, USA, in 1999, and Ph.D. degree in mathematics and physics from YUS, in 1990. In December 2000, he joined the Department of Electrical Engineering, University of Texas at San Antonio, where he is currently an Associate Professor. He is the author of three books, five book-chapters, two patents, and many journal papers and specializing in the theory and application of fast Fourier transform, tensor and paired transforms, unitary heap transforms, image enhancement, computerized tomography, processing biomedical images, and image cryptography.

Mehdi Hajinoroozi received his BS in electrical engineering from University of Tehran and his MS in electrical engineering from Technical University of Darmstadt, Germany in 2011. Currently, he is doing his graduate studies in the University of Texas at San Antonio, USA, in the Department of Electrical and Computer Engineering. His current research interests include digital signal and image processing.