# Fast Unitary Heap Transforms: Theory and Application in Cryptography

**Artyom Grigoryan and Khalil Naghdali**

EE Dept. UTSA, San Antonio

*Presented by Khalil Naghdali*

To the SPIE Defense, Security, and Sensing 2009

# *OUTLINE*

- ***Introduction***
- ***Heap Transforms***
  - ***Definition and basic transformation***
  - ***Composition and application***
  - ***Vector generators and examples***
  - ***Basic Functions***
  - ***Cosine wave***
- ***Angular Representation***
  - ***Speech signal***
- ***DsiHT***
  - ***Speech signal***
  - ***Speech signals by binary generators***
- ***Heap transform of images***
  - ***1-D and 2-D image processing***
  - ***Examples***
- ***Advantages***
- ***Conclusions***
- ***References***

# Introduction

- We discuss a class of discrete unitary transforms, which we call the heap transforms, or transforms which are generated by input signals.
- The complete systems of basic functions of heap transforms are referred to as waves generated by input signals, the waves with their specific motion in the space of functions.
- We stand on the linear heap transforms defined by simple rotations, or Givens rotations.
- The heap transforms are defined by signal-generators which allow for tuning the transforms when applying them for specific classes of applied signals and images.
- These generators play the role of keys without which it is not possible to reconstruct the initial signals or images.

# Heap Transforms: *Definition*

Discrete signal induced heap transforms (DsiHT) are defined by one or a few given generators (or signals) and a system of specific decision equations.

We discuss the case, when the basic transformations of the DsiHT are defined by the Givens transformations, or elementary rotations which satisfy 2 decision equations.

The composition of the N-point DsiHT by a given vector-generator $\boldsymbol{x} = (x_0, x_1, x_2, \ldots . x_{N-1})$ is performed by the sequential calculation of basic transforms

$$\boldsymbol{H} = \boldsymbol{T}_{N-1} \boldsymbol{T}_{N-2} \boldsymbol{T}_{N-3} \ \ldots \ \boldsymbol{T}_2 \, \boldsymbol{T}_1$$

# Heap Transforms: *Composition*

Each transform $T_k$ changes only two components of the input $\mathbf{z} = (z_0, z_1, z_2, \ldots, z_{N-1})$ which are processed in order $z_0, z_1, z_2, \ldots, z_{N-2}$, and then $z_{N-1}$. This is a natural path $P$, and such a path can be taken in many different ways.

Transformations $\mathrm{T}_\mathrm{k}$ are parameterized, and values of their parameters are defined by a given vector-generators $\mathbf{x} = (x_0, x_1, x_2, \ldots, x_{N-1})$.

The process of the $\mathbf{x}$-signal DsiHT of the signal $\mathbf{z}$:

$$\mathbf{x} \to \{T_k;\ k=1:(N-1)\} \to H=H_x \text{ and } \mathbf{z} \to H[\mathbf{z}].$$

# Heap Transforms: *Basic Transformations*

The basic transformations $T_k$ are parameterized by an angle $\varphi_k$ .The selection of $\{\varphi_k\}$ is initiated by the vector-generator through the so-called decision equations and a given set of constants $\boldsymbol{A} = \{a_1, a_2, \ldots, a_{N-1}\}$ in the following way.

Let $f(x,y,\varphi)$ and $g(x,y,\varphi)$ be functions of three variables. It is assumed that, for each $a$ from $\boldsymbol{A}$ and point $(x,y)$ on the plane the equation

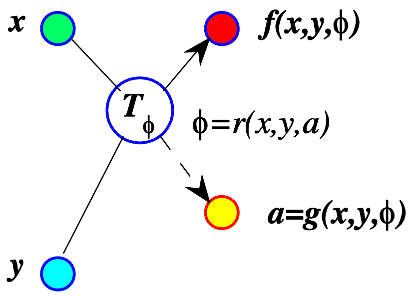
$$g(x,y,\varphi)=a$$

has a unique solution $\varphi = r(x,y,a)$.

# Heap Transforms: *Composition*

The system of equations

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



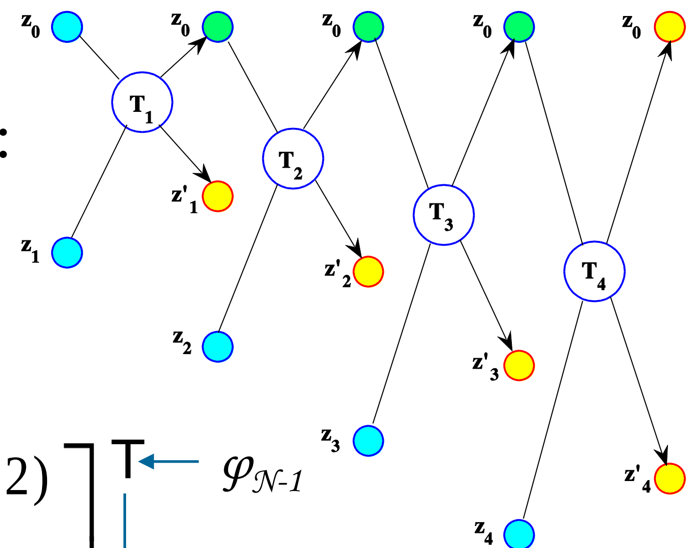

is called *the system of decision equations*.

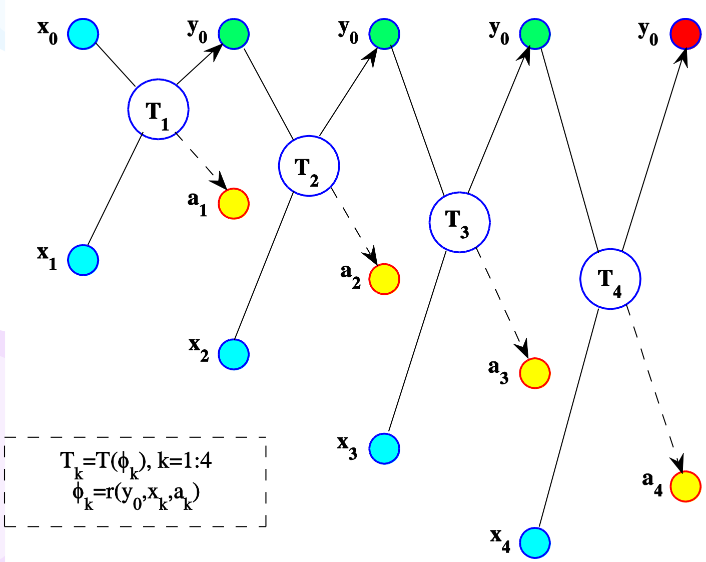
1. The value of $\varphi$ is calculated from the second equation which is called *the angular equation*.
2. The value of $y_0$ is calculated from the given input $(x,y)$ and obtained $\varphi$.

# Heap Transforms: *Composition and Application*

$$\begin{bmatrix} x_0 \\ x_1 \end{bmatrix} \xrightarrow{T \leftarrow \varphi_1} \begin{bmatrix} y_0^{(1)} \\ a_1 \end{bmatrix}, \quad \begin{bmatrix} y_0^{(1)} \\ a_1 \end{bmatrix} \xrightarrow{T \leftarrow \varphi_2} \begin{bmatrix} y_0^{(1)} \\ x_2 \end{bmatrix} \quad \cdots \quad \begin{bmatrix} y_0^{(N-2)} \\ x_{N-1} \end{bmatrix} \xrightarrow{T \leftarrow \varphi_{N-1}} \begin{bmatrix} y_0^{(N-1)} \\ a_{N-1} \end{bmatrix}$$

$z \rightarrow H(z):$




Generator is processed first and during this process all angles $\varphi_k$ are calculated

# Heap Transform: *x-generated DHT*

The transform of the vector ***x***:

$$H: x \to H(x) = (y_0^{(N-1)}, a_1, a_2, \ldots, a_{N-1}).$$

*H* is the *N*-point discrete ***x***-signal induced heap transformation (DsiHT), and the vector ***x*** is the generator of this transformation.

The components of ***z*** can be processed sequentially together with components of the vector-generator at the same time the angles $\varphi_k$ are calculated:

$$z \to H(z) = (z_0^{(N-1)}, z_1^{(1)}, z_2^{(1)}, \ldots, z_{N-1}^{(1)}).$$

$$z_0 \to z_0^{(1)} \to z_0^{(2)} \to \ldots \to z_0^{(N-1)}.$$

$$z_K^{(1)} = g(z_0^{(k-1)}, z_k, \varphi_k), \qquad (z^{(0)} = z_{(0)}).$$

# Example 1: *Elementary Rotations*

Given a real number $a$ consider the following functions defined on the unbounded set of points $\{(x,y);\ x^2+y^2\geq a^2\}$

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

It is a rotation of the point $(x,y)$ to the horizontal $Y=a$,

$$T_\varphi\colon (x,y)\rightarrow(y_0,a)=(x\cos\varphi - y\sin\varphi, a).$$

$$T_\varphi : \begin{bmatrix} x \\ y \end{bmatrix} \rightarrow \begin{bmatrix} y_0 \\ a \end{bmatrix} = \begin{bmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix},$$

$$\varphi = \arccos\left(\frac{a}{\sqrt{x^2+y^2}}\right) - \arctan\left(\frac{x}{y}\right), \qquad \left(\varphi = \arccos\left(\frac{a}{x}\right) \text{if } y=0\right).$$

# Heap Transform: *Example N=5*

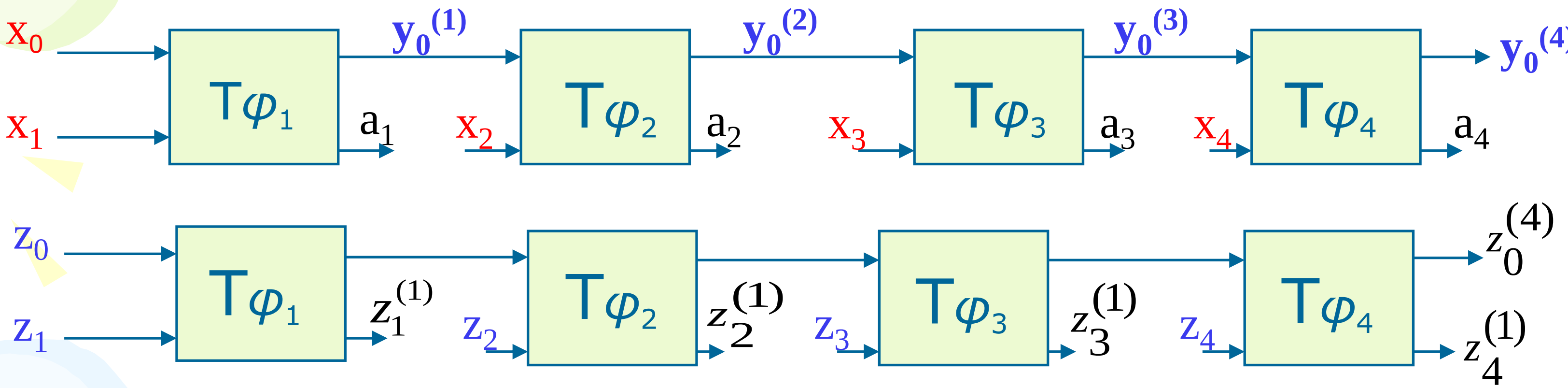



**Fig 1.** Signal-flow network of the five-point DsiHT of a vector **z**.

***Case A={0,0,…,0}:***

$$\tan(\varphi_k) = -\frac{x_k}{y_0^{(k-1)}}, \quad k = 1(N-1), \qquad y_0^{(0)} = x_0,$$

$$y_0^{(k)} = (T_{\varphi_k} T_{\varphi_{k-1}} \cdots T_{\varphi_1} x)_0 = \pm\sqrt{x_0^2 + x_1^2 + \ldots + x_k^2}.$$

As a result, the whole energy of the input signal is collected consequently, and then transferred to the first component (heap)

$$y_0 = y_0^{(N-1)} = \pm\sqrt{x_0^2 + x_1^2 + \ldots + x_{N-1}^2} = \pm\|x\|. \qquad y_0^{(N-1)} = sign(x_0)\|x\|.$$

# Heap Transform: *Example N=5*

Let $H$ be the DsiHT generated by vector $x=(1,2,3,2,1)'$.

The generator is transformed into the scaled unit vector

$$H(x)=\|x\|e_1=(\|x\|,0,0,\ldots,0)'=(\sqrt{19},0,0,\ldots,0)'$$

$$\mathrm{H}=\begin{bmatrix} 0.2294 & 0.4588 & 0.6882 & 0.4588 & 0.2294 \\ -0.8944 & 0.4472 & 0 & 0 & 0 \\ -0.3586 & -0.7171 & 0.5976 & 0 & 0 \\ -0.1260 & -0.2520 & -0.3780 & 0.8819 & 0 \\ -0.0541 & -0.1081 & -0.1622 & -0.1081 & 0.9733 \end{bmatrix},$$

and the angels of rotations are

$\varphi_1$ =- 63.4349°, $\varphi_2$ =-53.3008°,

$\varphi_3$ =-28.1255°, $\varphi_4$ =-13.2627°.

# Heap Transform: *Example N=5*

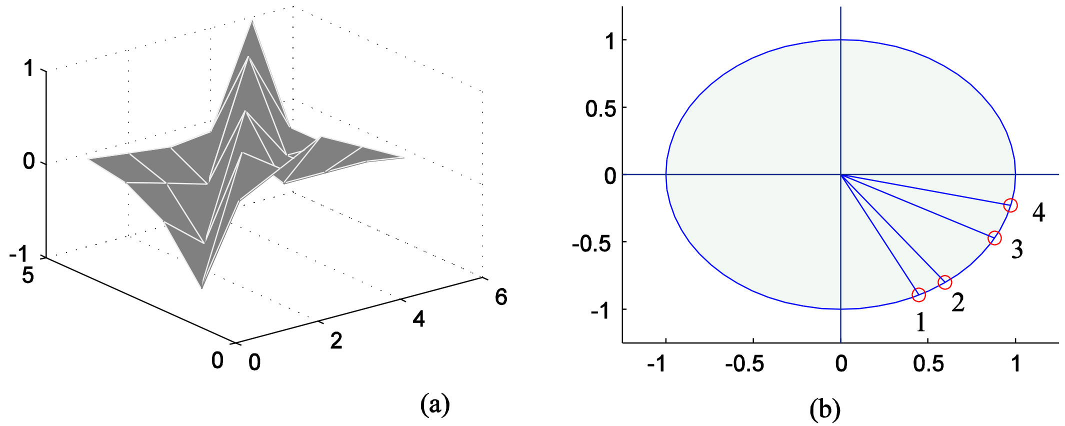
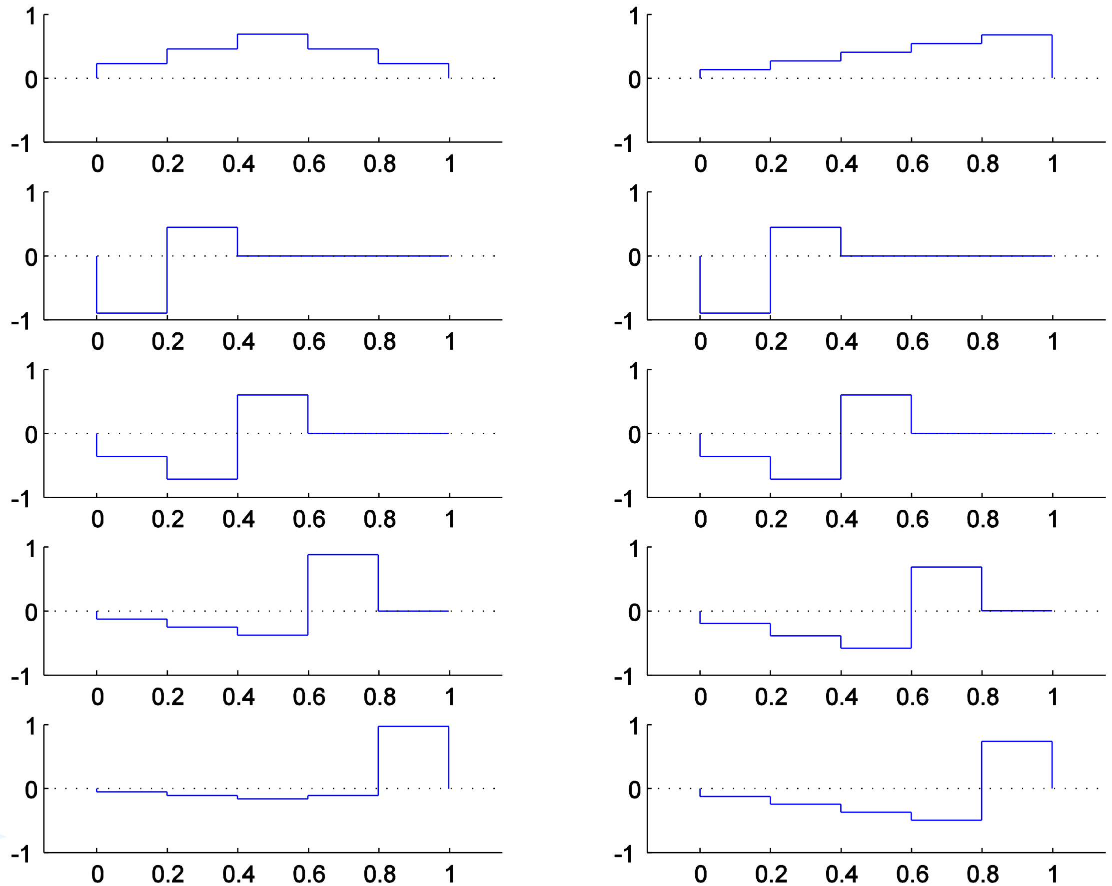



Fig. 2. Mesh and angles of the five-point DsiHT.

$$\mathrm{H} = \begin{bmatrix} 0.2294 & 0 & 0 & 0 & 0 \\ 0 & 0.4472 & 0 & 0 & 0 \\ 0 & 0 & 0.3586 & 0 & 0 \\ 0 & 0 & 0 & 0.1260 & 0 \\ 0 & 0 & 0 & 0 & 0.0541 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 & 2 & 1 \\ -2 & 1 & 0 & 0 & 0 \\ -1 & -2 & \frac{5}{3} & 0 & 0 \\ -1 & -2 & -3 & 7 & 0 \\ -1 & -2 & -3 & -2 & 18 \end{bmatrix}.$$

# **Heap Transform:** *Basic functions (N=5)*



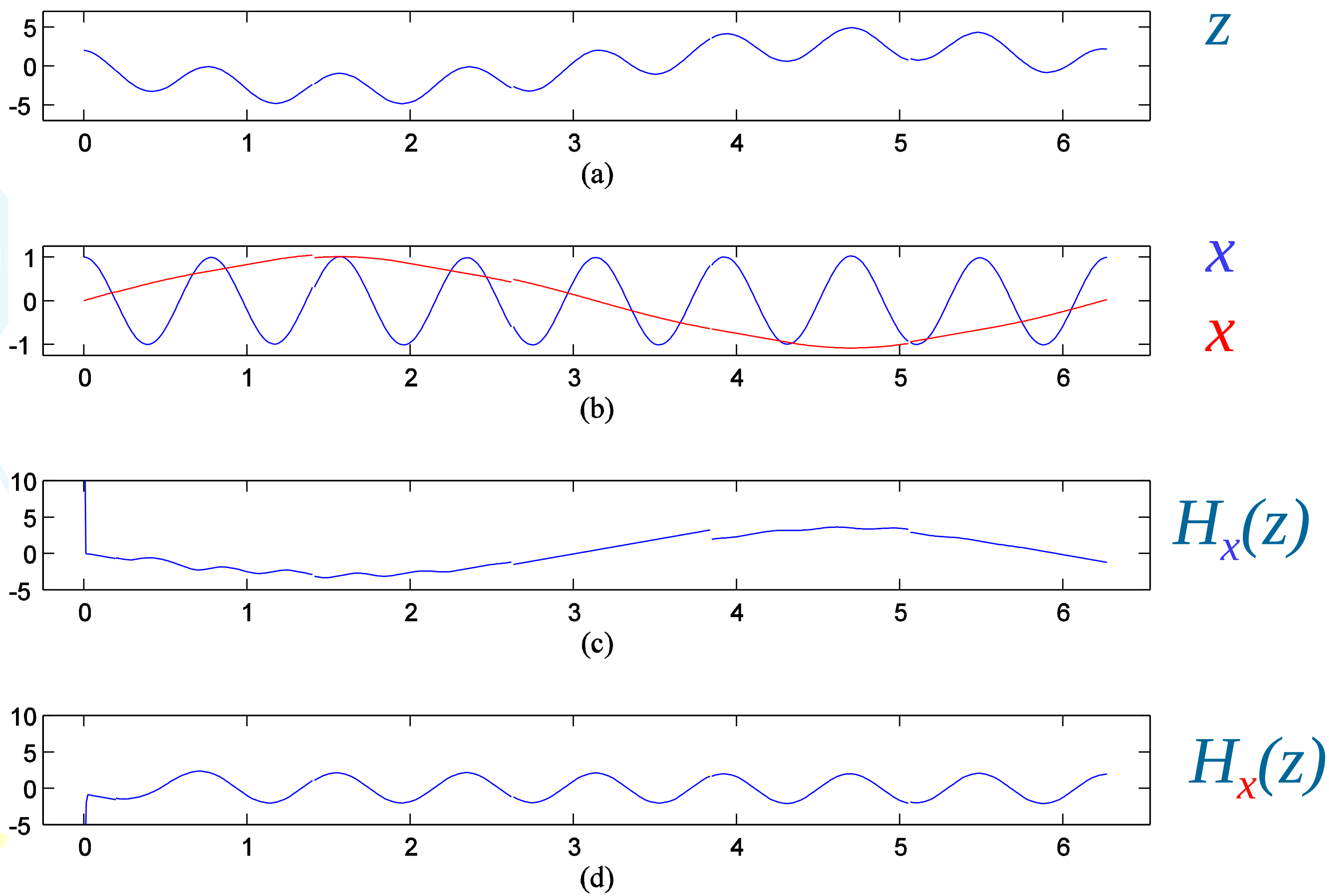

(a) (b)

Fig 3: Basic functions of the DsiHTs generated by (a) the vector *(1,2,3,2,1)'* and (b) the vector *(1,2,3,4,5)'*.

# **Heap Transforms:** *Cosine wave z(n)*

Consider a the discrete-time signal of length 512 sampled from

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

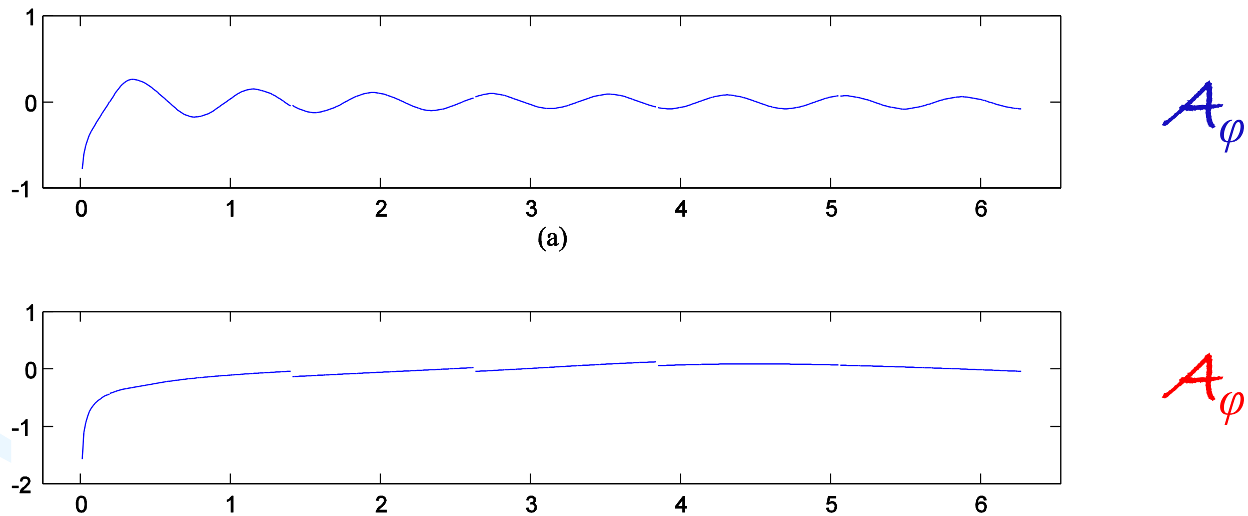


# Angular Representation: *DsiHT*

Given a generator $\boldsymbol{x}$, the DsiHT is described uniquely by its angular representation

$$\mathbf{x} \rightarrow \mathcal{A}_{\mathbf{x}} = \{||\mathbf{x}||, \mathcal{A}_{\varphi}\} = \{||\mathbf{x}||, \varphi_1, \varphi_2, \ldots, \varphi_{N-1}\}.$$

$\mathbf{A}_{\varphi}$ is referred to as an angular signal.




(a) (b)

The generator is reconstructed by its angular representation as

$$\mathbf{x} = \mathcal{H}^{-1}(||\mathbf{x}||, 0, 0, \ldots, 0)' = ||\mathbf{x}|| \left(\mathcal{H}^{-1}(1, 0, 0, \ldots, 0)'\right).$$

# Heap Transform: *Speech Signal*

The generator is a part of the speech signal. We will apply the transform generated by this part to the whole signal.

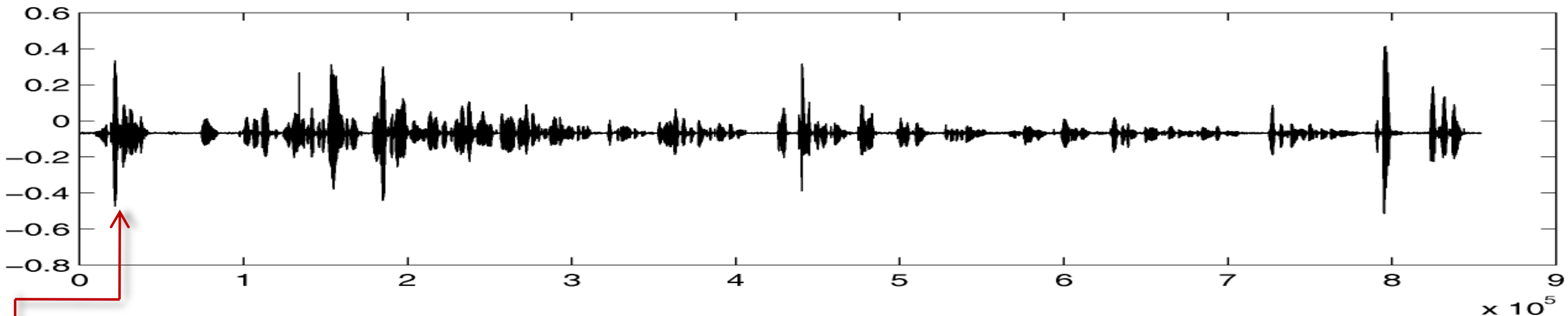
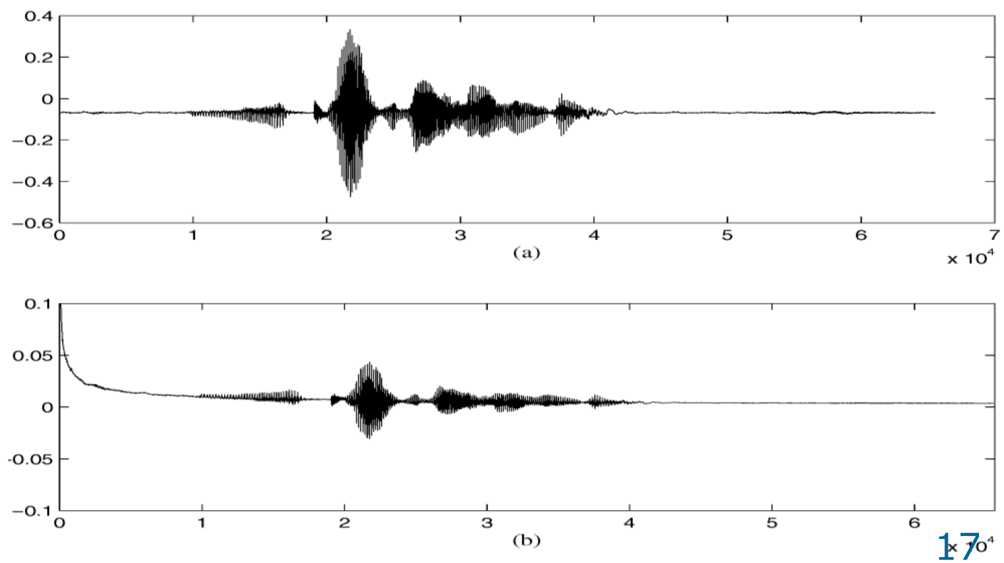



Fig 4: A speech signal of 38.7360 sec ($f_s$=22050Hz).

***x***-signal:2.9722 sec ($2^{16}$ pts).

Angular representation of ***x***

Fig 5:



# Angular Representation: *Speech Signal*

- It is very important to note, that we can hear the angular signal $\boldsymbol{A}_\varphi$; it reproduces all sounds and words as the original signal. The human ear is able thus to reproduce the speech signal in its angular representation; it needs only to be magnified.
- In the above example, the angular signal $\boldsymbol{A}_\varphi$ has been magnified by factor of *32*. Thus, together with the time and frequency representation of the signal, we obtain a new, angular representation.
- Our preliminary experimental results show that the angular transformation has a root property.

$$\mathbf{x} \to \mathbf{u} = \mathcal{A}_{\mathbf{x}} = \{||\mathbf{x}||, \mathcal{A}_\varphi\} \to \mathcal{A}_{\mathbf{u}} = \{||\mathbf{u}||, \mathcal{A}_\phi\}$$

# Heap Transform by parts: *Speech Signal*

- When applying sequentially the DsiHT which is generated by a specific part to the whole speech signal, we observe the heaps of the transforms at the beginning of each part.

- It can be heard clearly: *Within each part the heap transform reproduces the signal-generator.*

- DsiHT includes much information of the signal-generator into the output.

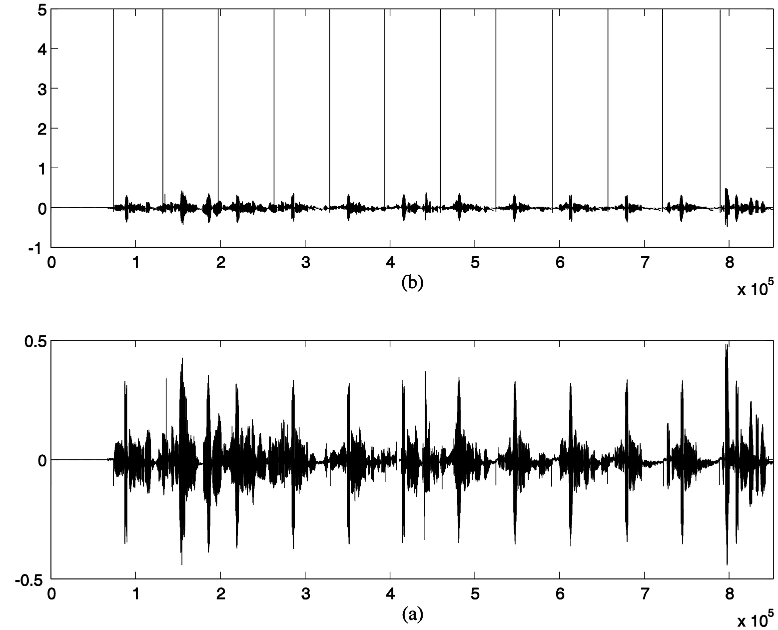



Fig 6: DsiHTs of the parts.

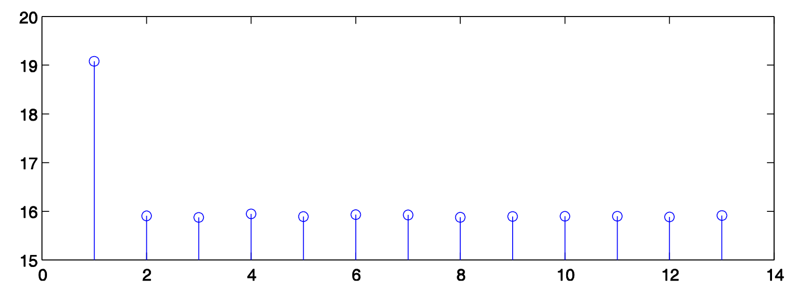
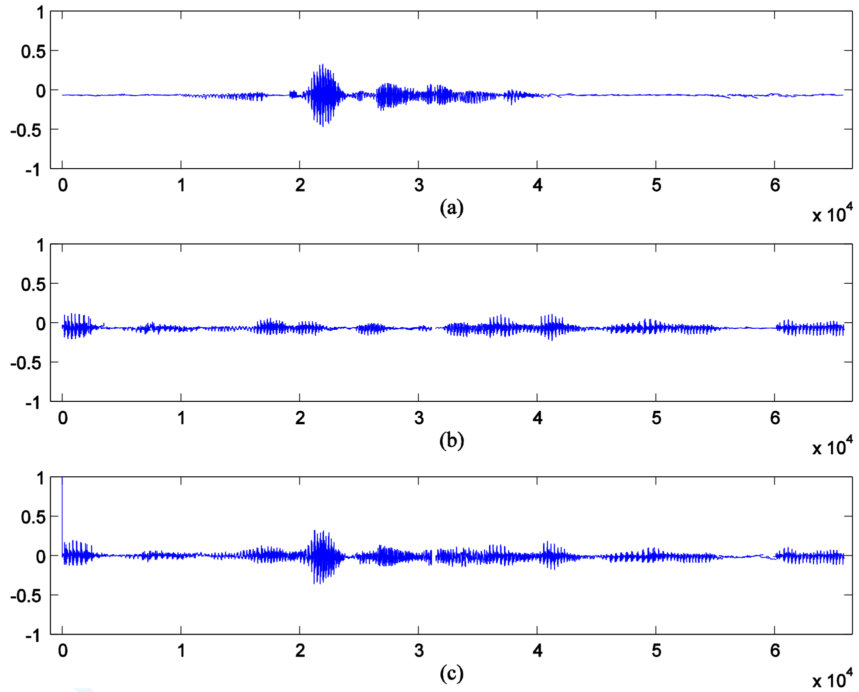



Fig 7: 13 DsiHT Amplitudes

# DsiHT: *Speech Signals*

When processing one signal by the DsiHT generated by a given signal, the information of the generator is imposed in the processed signal.

Fig. 8:




generator ***x***

signal ***z***

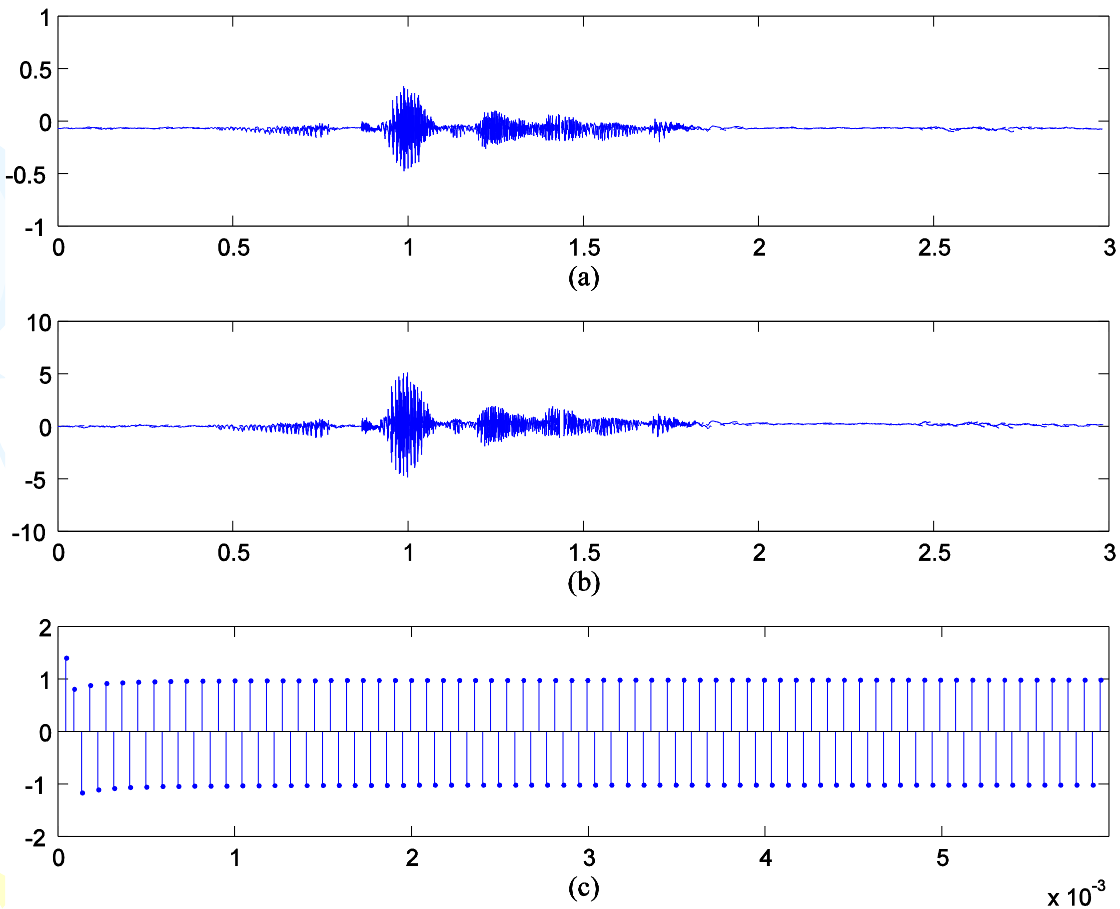
transform ***$H_x[z]$***

The transform contains the information about both signals, and during the reproduction of this transform as a speech signal, in many cases, both signals can be recognized.

# DsiHT: *Speech Signals by binary generators*

Depending on the vector-generator, a signal processed by the DsiHT can be enhanced or made imperceptible:

Fig. 9:




Input signal ***z***

Transform $\boldsymbol{H_{(1,1,\dots,1,1)}[z]}$

Transform $\boldsymbol{H_{(1,-1,\dots,-1,1)}[z]}$

# 1-D or 2-D DsiHT: *Image processing*

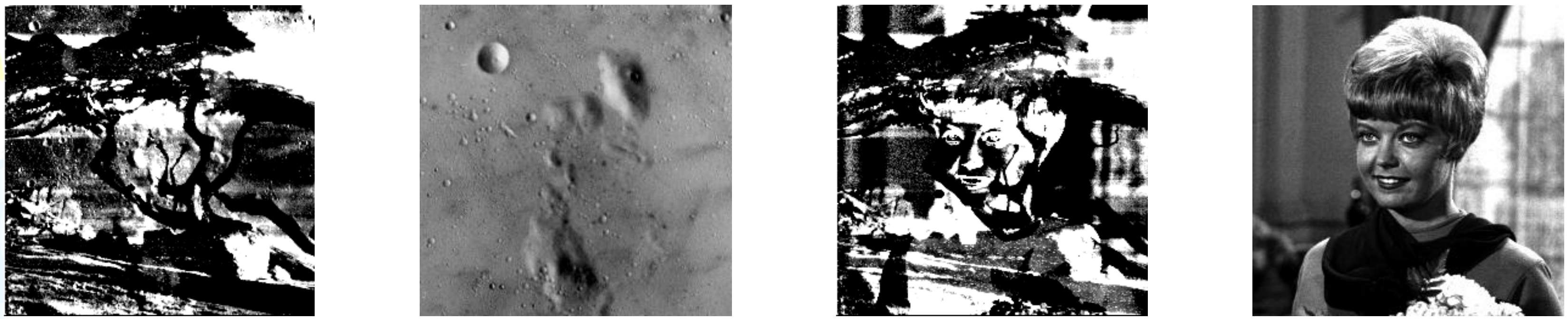
The selection of generators plays important role when applying the heap transform in image processing.

1. We can consider the simple row-wise method of image processing, when each row of the image is processed by the 1-D DsiHT generated by the same row or a different generator.

2. We can use the row-and-column method to define the 2-D DsiHT of the image with different generators by rows and columns.

# 1-D DsiHT by rows: *Example with the tree*

The tree image in (a) is processed row-wise by DsiHTs generated by rows of the moon image (b), and the tree



(a) (b) (c) (d)

image (c) processed row-wise by DsiHTs generated by rows of the girl image (d).

One can notice the changes in the tree image; the moon image cannot be seen in the image, but some details of the girl image can be observed.

# 1-D DsiHT by rows: *Example with the tree*

- Importance of the generators
- Key-generator is presented in the transform together with the image.

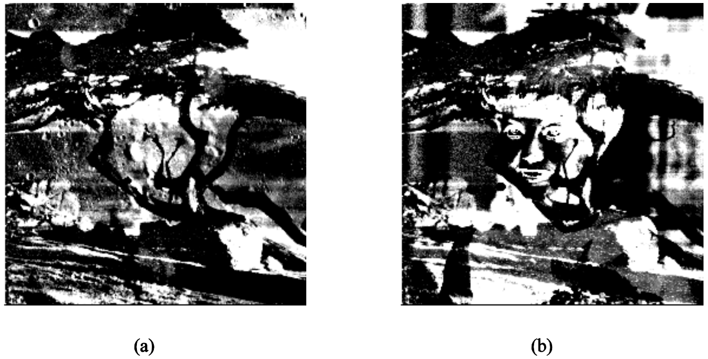



Fig. 11: DsiHTs of the tree

- The generator can be extracted from the transform and processed image. In other words, in general, if a signal ***z*** has been processed by the DsiHT generated by a signal ***x***, then there is the unique transformation restoring ***x***

$$\mathcal{R} : \{H_{\mathbf{x}}[\mathbf{z}], \mathbf{z}\} \rightarrow \mathbf{x}.$$

Thus ***z*** is the key to obtain ***x***.

# 1-D DsiHT by rows: *Example 3 with the tree*

- We can select generators to make the information we want invisible.

*Example:* We consider the tree image and the sine-wave generators of length 256, which are sampled from the signals *cos(2t)* and *cos(8t)* in the interval *[-π,+π]*.


cos(2t)-row


cos(2t)-column


cos(8t)-row


cos(8t)-column

Fig. 12: 2-D DsiHT of the tree image by discrete sine wave generators cos(2t) and cos(8t)

# 1-D DsiHT by rows: *Example 4 with the tree*

• We can select generators to make the information we want invisible.

*Example:* We consider the tree image generator and the sine-wave image 256x256. The sine wave is sampled from the signal $x(t)=\cos(32t)$ in the interval $[-\pi,+\pi]$.

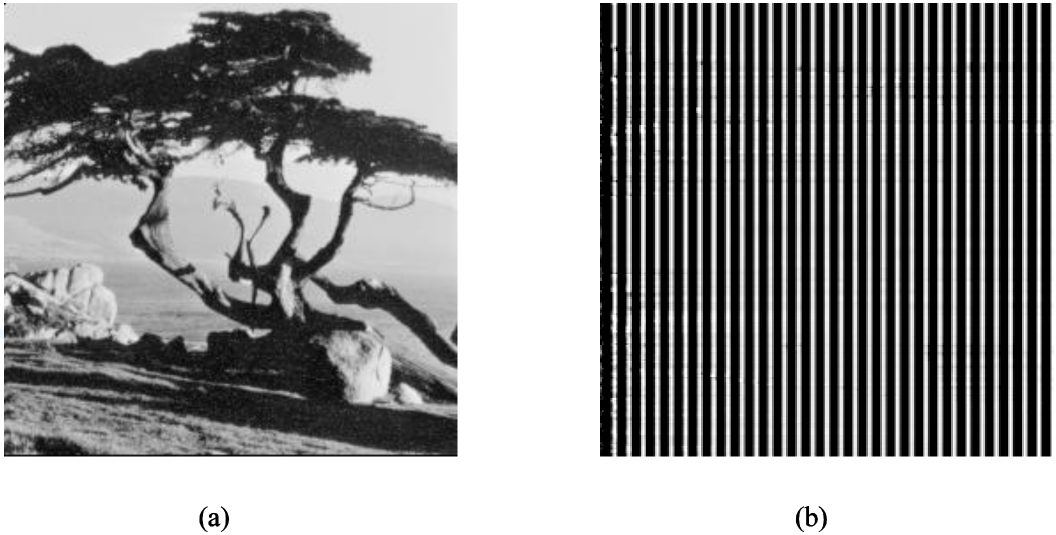



Fig 12. (a) The tree image and (b) the DsiHT of cos(32t).

# 2-D DsiHT row-column: *Example 5 with the tree*

(a) (b) (c)

Fig. 13: Tree image (a) and the 2-D DsiHT of the image performed by the row-column method.
(b): rows (columns) by 1-DsiHT by rows (columns)
(c): rows (columns) by 1-DsiHT by columns (rows)

# DsiHT: *Example with the tree and speech*

Consider the first fragment (of length $2^{16}$=256x256) of the speech signal and the DsiHTs of this signal, when 256 generators represent the rows of the tree image. Each DsiHT was applied to one part of the speech signal.

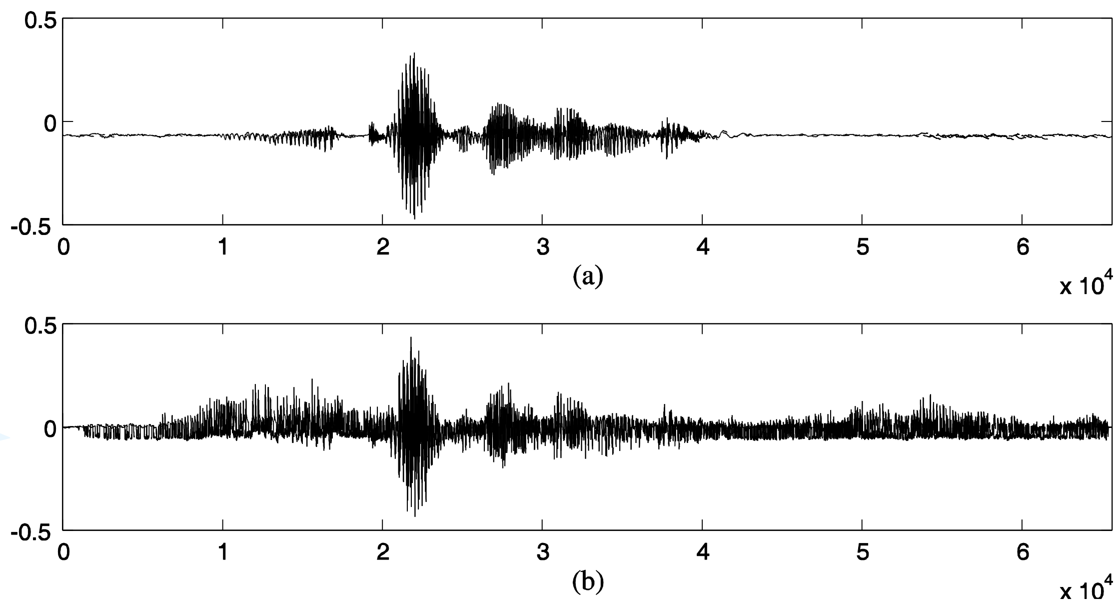



Fig. 14: Transform (b) of the speech (a) by rows of the tree image.

# DsiHT: *Example 6 with the tree and speech*

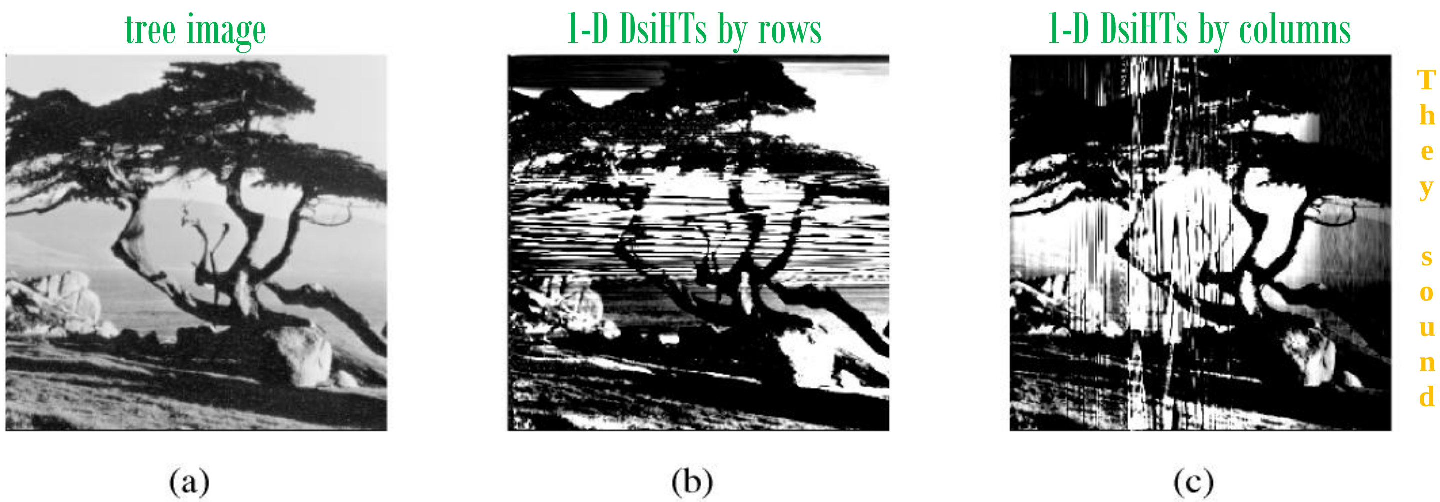



Figure 15: The original tree image (a) and the images processed row-wise (b) and column-wise (c) by the 256 DsiHTs which are generated sequentially by parts of the speech signal. (Each part is of length 256.)

# DsiHT: *Advantages*

- The transforms are linear and fast, because of a simple form of decomposition of their matrices, and they can be applied for signals of any length, as well as images of any size.
- Matrix of the heap transformation is triangle from the $2^{nd}$ row, and the 1st rows represent the generators itself.
- Fast algorithms of calculation of the direct and inverse heap transforms are unique and do not depend on the length of the processed signals.
- DsiHT provide the angular representation of signals and images.
- A few applications of the heap transforms are demonstrated for transformation and reconstruction of 1-D signals and 2-D images

# Conclusion: *DsiHT*

- Heap transformations represent a subclass of discrete unitary signal-induced transformations which are generated by input signals. More general cases with a few generators and decision equations can be also considered for the DsiHT.
- Complete set of the heap transformation represents variable waves which describe a motion in the space of signals. (These waves are not simple sliding windows as in the wavelet theory)
- The vector-generators and paths of the DsiHT are the keys of the transformation.
- We believe that DsiHT can be used in signal and image processing, image encryption, cryptography, and other areas.

# References: *DsiHT*


1. A.M. Grigoryan and M.M. Grigoryan, "Nonlinear approach of construction of fast unitary transforms," in Proc. of the 40th Annual Conference on Information Sciences and Systems (CISS 2006), Princeton University, pp. 1073-1078, March 22-24, 2006, Princeton.
2. A.M. Grigoryan and M.M. Grigoryan, "Discrete unitary transforms generated by moving waves," in Proc. of the International Conference: Wavelets XII, SPIE: Optics + Photonics 2007, vol. 6701, 27-29 August, 2007, San Diego, CA.
3. A.M. Grigoryan and M.M. Grigoryan, "New discrete unitary Haar-type heap transforms," in Proc. of the International Conference: Wavelets XII, SPIE: Optics + Photonics 2007, San Diego, CA, 27-29 August, 2007.
4. A.M. Grigoryan and M.M. Grigoryan, *Brief Notes in Advanced DSP: Fourier Analysis with MATLAB*, CRC Press Taylor and Francis Group, Feb. 2009.


5. *This presentation in pdf format is available in the Dr. Grigoryan web page:*
*http://engineering.utsa.edu/~grigoryan/posters.html*