On Development of Hilbert-Huang Transform Data Processing Real-Time System with 2D Capabilities

Semion Kizhner, Karin B. Blank, Jennifer A. Sichler,
Umeshkumar D. Patel and Jacqueline Le Moigne
National Aeronautics and Space Administration
Goddard Space Flight Center
Greenbelt Road, Greenbelt MD, 20771, USA
Semion.Kizhner-1@nasa.gov

Esam El-Araby and Vinh Dang
The Catholic University of America
Washington, DC 20064, USA
ALY@cua.edu

Abstract—**Unlike other digital signal processing techniques such as the Fast Fourier Transform for one-dimensional (1D) and two-dimensional (2D) data (FFT1 and FFT2) that assume signal linearity and stationarity, the Hilbert-Huang Transform (HHT) utilizes relationships between arbitrary signal's local extrema to find the signal instantaneous spectral representation. This is done in two steps. Firstly, the Huang Empirical Mode Decomposition (EMD) is separating input signal of one variable s(t) into a finite set of narrow-band Intrinsic Mode Functions {$IMF_1(t)$, $IMF_2(t)$ … $IMF_k(t)$} that add up to the signal s(t). The IMFs comprise the signal adaptive basis that is derived from the signal, as opposed to artificial basis imposed by the FFT or other heritage frequency analysis methods. Secondly, the HHT is applying the Hilbert Transform to each $IMF_i(t)$ signal constituents to obtain the corresponding analytical signal $S_i(t)$. From the analytical signal the HHT generates the Hilbert-Huang Spectrum. Namely, a single instantaneous frequency $\omega_i(t)$ for signal $S_i(t)$ at each argument t is obtained for each of the k-Huang IMFs. This yields the Hilbert-Huang spectrum {$\omega(IMF_1(t))$, $\omega(IMF_2(t))$ … $\omega(IMF_k(t))$} at each domain argument t for s(t) that was not obtainable otherwise. The HHT and its engineering implementation – the HHT Data Processing System (HHT-DPS) for 1D was developed at the NASA Goddard Space Flight Center (GSFC). The HHT-DPS is the reference system now used around the world. However, the state-of-the-art HHT-DPS works only for 1D data, as designed, and it is not a real-time system. This paper describes the development of the reference HHT Data Processing Real- Time System (HHTPS-RT) with 2D capabilities or HHT2 to process large images as the development goal. This paper describes the methodology of research and development of the new reference HHT2 Empirical Mode Decomposition for 2D (EMD2) system and its algorithms that require high capability computing. It provides this system prototype test results and also introduces the HHT2 spectrum concepts. It concludes with suggested areas for future research.**

I. Overview of HHT1 and HHT2 Development Motivation, Goal and Objectives

The primary motivation for the development of HHT1 and reference HHT2 is the need for a reference spectral analysis engineering tool by newer spaceflight instruments to process on-board a spacecraft instrument 1D and 2D data from non-linear and non-stationary sources. The reference HHT2 system could also be used by other alternative developments of HHT for 2D to compare (refer to) results against the HHT2 reference open source implementation. The goal is to develop a fast reference HHT2 similar to the HHT-DPS [2], which became a reference engineering tool for 1D HHT.

A. HHT1 and HHT2 Development Motivation

The two objectives towards reaching this goal are

- Research and develop a real-time HHT-DPS-RT engineering tool for 1D signal processing (HHT1) based on existing reference HHT-DPS [2].
- Develop the HHT2 for 2D, a completely new HHT system.

The HHT1 will be running on a high-speed microprocessor and reconfigurable hardware comprising a suite of emerging information technologies such as High-Performance Reconfigurable Computing architecture (HPRC). This shall enable the HHT to run on-orbit on future space science instruments as a game-changing spectral analysis technology. For example, HHT1 adapted for spaceflight could be used on-orbit for noise reduction, which, in turn, could reduce the complexity and cost of the Wide Field Infrared Space Telescope (WFIRST) instrument electrical subsystem data processing platform. Furthermore, the HHT1 is extended to 2D using HHT2 separability property and a reference HHT2 version is being developed in an independent from HHT1 path. The computational complexity of the HHT for 2D is of formidable order $O(N^4)$, which for N=2048 is $O(10^{14})$. Having HHT1 run fast enables the development of HHT2 along the methodology of the heritage FFT2. New speedup algorithms were also developed for the reference HHT2 version, which is independent from HHT1. Some of the HHT2 2D applications are to allow on-board instrument image registration, as well as for advanced compression [3], freeing communications downlink for critical telemetry.

The HHT for 1D was first introduced by Huang et al. [1] and was implemented as a powerful engineering tool for analysis of non-linear and non-stationary signals [3], [4]. It was shown in [5] that EMD is similar to selective bank decomposition and that it is efficient for signal de-noising [6]. This motivates the extension of 1D HHT to a real-time HHT1 and to HHT2 to give the HHT .

An important application of HHT1 is on-board a spacecraft noise reduction using sensor reference pixels [7] which, in turn, may greatly simplify the EE subsystem computational platform for future cosmic survey missions such as the

978-1-4673-1916-4/12/$31.00 ©2012 IEEE

WFIRST and also substantially reduce the downlink bandwidth requirements.

HHT1 can also run on small devices to process real-time biological measurements, as part of signal processing and as trend of interest extraction in space medicine.

One of the HHT2 major applications is to facilitate fast on-board image registration allowing this 1st step of image processing to be done in-situ and another application is image compression [3], freeing the communications downlink for more critical telemetry.

B. Goal and Objectives

The title of this paper "The Hilbert-Huang Transform Data Processing Real-Time System With 2D Capabilities" also carries the goal for our research and development. The HHT1 and HHT2 are visualized for space flight applications and must be suitable for running on-board a spacecraft instrument.

The real-time is considered to be the time required for the HHT to ingest the new input vector or image into its memory module and finish processing of the previous input before the new image ingest is completed. This holds even if it is a non-destructive readout during a very long exposure (1000 second). 10 seconds or 10Hz is the acceptable baseline for 2048x2048 sensor image readout if there is enough on-board memory (82 Mega Bytes) for each sensor to store 10 images' 16-bit pixels.

The two objectives towards reaching this goal were**:**

a) To research and develop an HHT1 Data Processing Real-Time System prototype for one-dimensional input data signal by adapting the already developed HHT-DPS for 1D [2][4]. The HHT1 will run on a high capability computing microprocessors such as employed by the Graphic Processing Units (GPUs) under a real-time operating system (RTOS or other OS) that has minimum overhead running a computationally intensive application such as HHT1. The microprocessor is assisted by reconfigurable Field Programmable Gate Array (FPGA) hardware comprising a suite of emerging information technologies such as High-Performance Reconfigurable Computing (HPRC) [8]. In HPRC domain computational platform specifications and parallel processing software High Level Languages (HLLs) and RTOS support will facilitate hardware/software co-design of computationally intensive on-board algorithms. The HHT1 relates to the existing HHT-DPS as the 1D Fast Fourier Transform FFT1 relates to the snail pace Digital Fourier Transform or DFT1.

b) Furthermore, the HHT1 is extended to HHT2 for processing two-dimensional data (2D images). The main problem in developing the HHT2 is its computational complexity estimated as $O(N^4) \sim O(10^{14})$ for N=2048 ([9], [10]). The state-of-the art is considering images of up to N=512. In HHT2 N=2048 is a typical image size. It is computationally difficult to build image envelopes in EMD2 process for such large images. It is also difficult to construct envelope prediction points for 2D. These prediction points allow extension of the envelopes over the input image boundary (domain), so that the upper and lower envelopes surfaces can be interpolated beyond the image domain and as a result of this the envelopes median can then be computed over the entire input image. This objective is to

- Find a computationally feasible Prediction Method for extremas beyond input image domain and
- Construct a computationally feasible algorithm for Upper and Lower Envelope Surface Fitting. The envelopes must fit into the image internal local extrema and the predicted extremas outside the image domain. The computational complexity of this process is $O(N^4)$ and this algorithm must alleviate this computational complexity.

II. HHT2 DEVELOPMENT PATHS

The HHT2 prototype has been developed in two ways:

a) By proving the *HHT2 separability*, a property similar to one that allows the 1D Fast Fourier Transform FFT1 to be used in computation of FFT2 to process 2D images. With the HHT2 separability established, we could then analogously use HHT1 for computations in HHT2 along columns or rows, or both, of the pre-processed input image. This would yield the $O(N^3)$ computational complexity compared to $O(N^4)$.

b) By developing an HHT2 prototype that is independent from HHT1 and is along the lines of existing research papers from the University of Alabama USA [9] and the University of Grenoble France [10], and by using novel speedup algorithms specifically developed for HHT2. The experimental approaches reported in [9] and [10] require $\boldsymbol{O(N^4)}$ operations and need substantial performance improvements for images of size **N>256**.

A. Separability Method

Having the fast HHT1 enables the development of HHT2 while using the results from already developed HHT-DPS and the *separability theorem* we have proved recently [11]. Essentially, the way the HHT EMD1 process for 1D works in finding the two sets of local maxima and minima, allows application of HHT1's stand-alone EMD1 to rows or columns of the 2D local extrema map embedded in an NxN zero-matrix. The columns (rows) of such a matrix are eigenvectors to EMD1, while the components of these vectors are still related to the geometry of the input image. This property can be used as a "separability" tool to apply HHT1 stand-alone EMD1 for HHT2 computations.

B. Optimization and EMD2 Partitioning Methods

In developing the HHT2 prototype version independently from HHT1 *we rely on processing speedup techniques we* developed specifically for HHT2. Among these are:

- *Embedding technique for image processing that optimizes computation within large loops.*
- *Partitioning the EMD2 bottleneck computation process for N=2048 into 64 processes of smaller sizes (n=256). This facilitates parallel processing on a few processing elements (PE_1-PE_{64}) or as 64 processes on a single powerful microprocessor core. For example, the partitioning of the bottleneck algorithm that generates*

the shortest adjacent extrema matrices ADMAX, ADMIN could be achieved as follows – the canonical form of LMMAX C(LMMAX) is partitioned in 64 "equal" consecutive vectors C_1, C_2, …, C_{64}. The double loop partition C_i x C is assigned to PE_i.

- *These and other speedup methodologies make the HHT2 computations tractable on high capability computing platforms for spaceflight.*

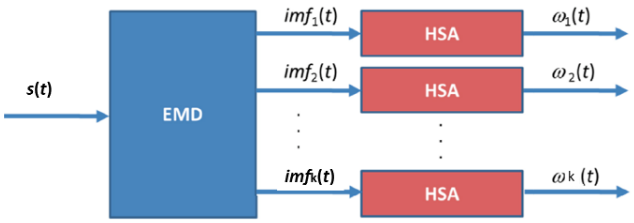
III. HHT1 and HHT2 Concepts and High Capability Computing Architecture

A. HHT1 Architecture

Because the HHT1 and HHT2 engineering tools are intended to be used for spaceflight, the HHT1 implementation is based on platforms that are space qualified. This includes computational devices that are radiation hardened, have low power consumption and low weight. These requirements for spaceflight components preclude state-of-the-art supercomputer ground platforms selection for the HHT1 and HHT2 implementation. It is rather based on powerful computational capabilities of FPGAs that have spaceflight heritage and provide on-board super-computing capabilities. However, the implementation architecture of an application requiring on-board super-computing capabilities is similar to ground implementations and modeling on a super-computer as that, for example, SRC-6 employed by The Catholic University of America (CUA) for HHT research and presented in this section. In view of this the HHT1 and HHT2 developing methodology is making use of the information technologies such as High-Performance Reconfigurable Computing architecture and methodologies under development at CUA and NASA as depicted in the following Fig. 1 and Fig. 2.

B. HHT1 Implementation Data flow for a High Computing Capability Platform

The SRC-6 supercomputer was used to implement the HHT1 for 1D as depicted in Fig. 1 and Fig. 2. This high-computing capability implementation methodology was used for HHT1 prototype development on a flight-alike microprocessor as described in Section IV.A.2.



EMD(s(t)) = {$imf_1(t)$, $imf_2(t)$ … $imf_k(t)$}
HSA – Hilbert Spectrum Analysis
Implemented on SRC-6 HPRC
Xilinx Virtex-II FPGA
CarteTM-C development environment
Floating-point implementation (double precision)
Features Run-Time Reconfiguration

Figure 1. HHT1 Top-Level Architecture.

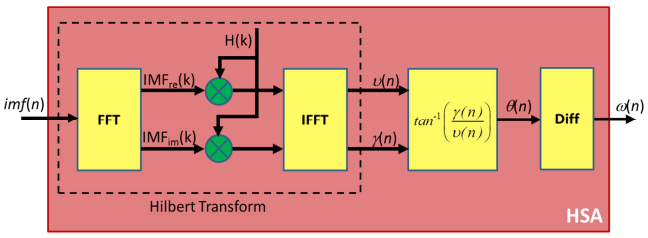


Figure 2. HHT1 Huang-Hilbert Spectrum Top-Level Architecture.

C. HHT2 EMD2 Concepts and Architecture

It would appear that the HHT2 concepts for 2D were going to be similar of that for 1D. Indeed, the HHT2 is envisioned to comprise two parts along the lines of HHT1 – the 2D Empirical Mode Decomposition (EMD2) and the 2D Spectrum Determination similar to the Hilbert Transform Spectrum for 1D (**HTS2**). However, the computational complexity of the well-conceptualized EMD2 is formidable, while the concept for the 2D Hilbert Transform-based Spectrum is in its infancy.

The challenge in HHT2 case is thus twofold:

- Develop a computationally feasible set of algorithms for EMD2 that work on a large image (2048x2048), as opposed to research papers that are using 256x256 images.
- Develop a new a concept of 2D spectrum for HHT2.

We have developed the computationally optimized algorithms' prototypes for EMD2 and we had described in the following sections the EMD2 development methodology, as well as its implementation options and solutions provided by the EMD2 prototype.

IV. HHT2 EMD2 Algorithms Framework and Contributions

A. Algorithms Framework

The HHT fundamentals [1] and their statistical filter implementation methodology [9] were used as a baseline for the *HHT2 prototype algorithms MATLAB reference scripts* development.

1) HHT Fundamentals

The HHT is founded on

(a) Finding the local maxima and local minima points for the input function s(t)

(b) Predicting these (a) sets of points outside the input function domain boundaries

(c) Building the input function upper envelope Uen out of the local maxima and its predicted points

(d) Constructing the lower envelope Len out of the local minima and its predicted points. (like a, b, c)

(e) Computing the Median = (Uen + Len)/2

(f) Finding the first IMF candidate by computing the difference IMF1 = s(t) – Median over the input function domain.

2) HHT Fundamentals Implementation for HHT2

Given an input image A of size NxN, we find the two map matrices of local extremas – MMAX, MMIN of size NxN (a). We use these two matrices to build two matrices where the extremas are replaced by the nearest adjacent distance extremas – ADMAX, ADMIN (b). Four values d1, d2, d3, d4 are then determined, where these are the min and max values in ADMAX, ADMIN. The statistical parameter Wen=f(d1, d2, d3, d4) is derived and then a window of size Wen x Wen is used over each pixel in input matrix A to construct the envelope values as the matrix A global max and min within the window (c). Because of the function Wen construction the local extrema points are naturally belonging to the built envelopes. The standard image processing techniques to do the image processing steps are to be found in the classical book on image processing [12].

The HHT2 prototype implementation was done on the HP Pavilion dv6 Notebook platform running Windows 7 Ultimate Operation System (Trademark of Microsoft CO.).

B. Contributions

The contributions of this work include such aspects:

- The concept of an outer matrix of a known flat pixel bias (say, zeroes or (min(min(A)) -1) with an embedded smaller input matrix A and then systematic processing (same for each pixel) on each input pixel without any complicated exceptions. This also allows partitioning the double loop between different processing elements.

- The other contribution aspect is canonical representation of intermediate results also allowing its processing to be partitioned for running many smaller pieces of EMD2 code on many processing elements.

- Proof of HHT2 separability allows direct application of HHT1 EMD script to EMD2 local extrema matrices columns or rows. This preserves the 2D local extrema matrices spatial relations while admitting processing with the computationally effective HHT1 for finding the envelopes meridians/longitudes composition.

- The prototype MATLAB™ (Trademark of MathWorks Inc.) scripts were developed using only transparent commands over which we have full control – those that can eventually be coded in C-language for hardware implementation. This is because we cannot fly in space the 1GB MATLAB module and operating system to control it.

- The N=8 test run input image is depicted in Fig. 3. The run results are depicted in the sub-matrices in Section VI.B and the resulting 2D or bi-dimensional Intrinsic Mode Functions (BIMF) are depicted in Fig. 4 for N=8 and Fig. 5 for N=512 images obtained from references [13][14].

V. Algorithms Implementation Outline

A. HHT1 EMD1 Top-Level Design

The HHT1 application to EMD2 was as follows: Run LMMAX2Kx2K_Optimize and LMMIN2Kx2K_Optimize. This generates LMMAX, LMMIN spatial local extrema maps. Run HHT1 on each column of each map with #sifts = 1 and #IMFs =1 and replace each column with its IMF. LMMAX and LMMIN become the Uen, Len in accordance with the EMD2 separability property. The output BIMF1 = (A – (Uen + Lev)/2) for classical N=512 image elaine.jpg was obtained in 69 seconds and the BIMF1 quality is visibly comparable with that obtained by EMD2 developed independently from HHT1 (Fig. 5).

B. HHT2 EMD2 Algorithms Independent from HHT1

EMD2 algorithms and MATLAB scripts top-level descriptions are as follows. The two supplementary stand-alone scripts are **Write_4_Test_Files.m** and script **ProblemSizeN.m**. In the first script a few large synthetic images are created from the 8x8 seed matrix depicted in Section VI.B. The second script is the control script that contains hard-coded parameters of image size N and parameter W that determines the selection of one of the few algorithms that implement the bottleneck code for EMD2 computations. The main script BIMF2Kx2K.m invokes the few scripts that implement the EMD2 functions. The script results are presented in Section VI.

VI. Test Runs and Results

A. Synthetic Input Test Matrix Construction

The 8x8 seed experimental matrix similar to [2] was used to create a few large synthetic images D16, D32, D64, D512, D1024, D2048. This seed matrix version was used for comparison of results with [2]. We also have full knowledge of these synthetic images derived from transparent seed A (like the number of extremas) and this allows better understanding of their processing computational complexity. The selection of a seed matrix is otherwise arbitrary, as long as matrix A has enough extremas to create large extrema sets in the largest 2048x2048 synthetic image D2048 required to verify EMD2 algorithms computational efficiency. Image D2048 was created as follows: D16 = [A A; A A]; D32 = [D16 D16; D16 D16]; … D2048 = [D1024 D1024; D1024 D1024]. The synthetic image was used because in this paper we are only interested in evaluation of the HHT2 algorithms' computational complexity.

B. Benchmark Input Image

The benchmark image elaine512 was also used as input to the HHT2 prototype system resulting in Fig. 5 BIMF1.

C. Test Runs, Results and Timing

>> **BIMF2Kx2K**

BIMF2Kx2K_s_start time is 24-Aug-2011 10:45:29.
Problem size N and adjacent distance parameter search width W are set to N=8 and W=0.

Input matrix upper left 8x8 corner.

Seed Matrix A:

8	8	4	1	5	2	6	3
6	3	2	3	7	3	9	3
7	8	3	2	1	4	3	7
4	1	2	4	3	5	7	8
6	4	2	1	2	5	3	4
1	3	7	9	9	8	7	8
9	2	6	7	6	8	7	7
8	2	1	9	7	9	1	1

Wen_Determination_s_start is 24-Aug-2011 10:45:29
AMAXDA Upper Corner Seed of Ls is:

0	0	0	0	0	0	0	0
0	0	0	0	9.8995	0	9.8995	0
0	9.8995	0	0	0	0	0	0
0	0	0	9.8995	0	0	0	9.8995
9.8995	0	0	0	0	0	0	0
0	0	0	0	0	0	0	9.8995
9.8995	0	0	0	0	0	0	0
0	0	0	9.8995	0	9.8995	0	0

AMAXDA Upper Corner Shortest Distances:

0	0	0	0	0	0	0	0
0	0	0	0	2.0000	0	2.0000	0
0	2.2361	0	0	0	0	0	0
0	0	0	2.2361	0	0	0	2.0000
2.0000	0	0	0	0	0	0	0
0	0	0	0	0	0	0	2.0000
2.0000	0	0	0	0	0	0	0
0	0	0	2.0000	0	2.0000	0	0

Median_Display

5.2500	5.0000	4.5000	4.1667	4.4167	4.9167	5.5833	5.7500
5.0000	4.8333	4.4444	4.1667	4.3889	4.9444	5.6111	5.8333
4.7500	4.6667	4.2222	3.8889	3.9444	4.6667	5.3889	5.7500
4.2500	4.3889	4.2778	4.1667	4.2222	4.8333	5.3889	5.6667
4.4167	4.5556	4.5000	4.4444	4.5000	4.9444	5.3333	5.5000
4.5833	4.7222	4.8889	5.2778	5.3889	5.5000	5.2778	5.2500
5.0833	5.0556	5.0000	5.5556	5.6111	5.6667	5.0000	4.9167
5.1250	5.0833	5.0000	5.8333	5.8333	5.8333	4.7500	4.6250

BIMF1_Display

2.7500	3.0000	-0.5000	-3.1667	0.5833	-2.9167	0.4167	-2.7500
1.0000	-1.8333	-2.4444	-1.1667	2.6111	-1.9444	3.3889	-2.8333
2.2500	3.3333	-1.2222	-1.8889	-2.9444	-0.6667	-2.3889	1.2500
-0.2500	-3.3889	-2.2778	-0.1667	-1.2222	0.1667	1.6111	2.3333
1.5833	-0.5556	-2.5000	-3.4444	-2.5000	0.0556	-2.3333	-1.5000
-3.5833	-1.7222	2.1111	3.7222	3.6111	2.5000	1.7222	2.7500
3.9167	-3.0556	1.0000	1.4444	0.3889	2.3333	2.0000	2.0833
2.8750	-3.0833	-4.0000	3.1667	1.1667	3.1667	-3.7500	-3.6250

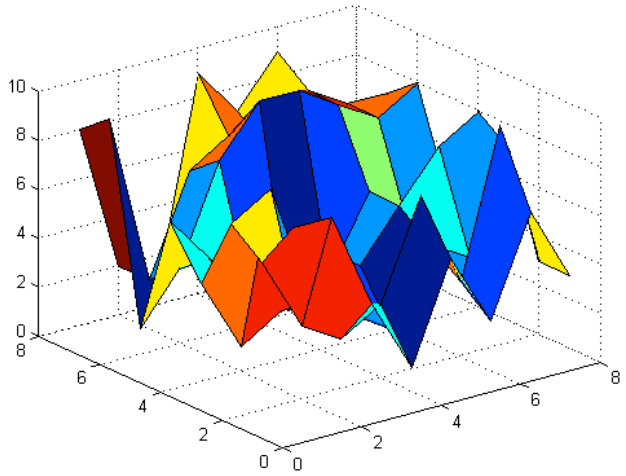


Figure 3. N=8 input image.

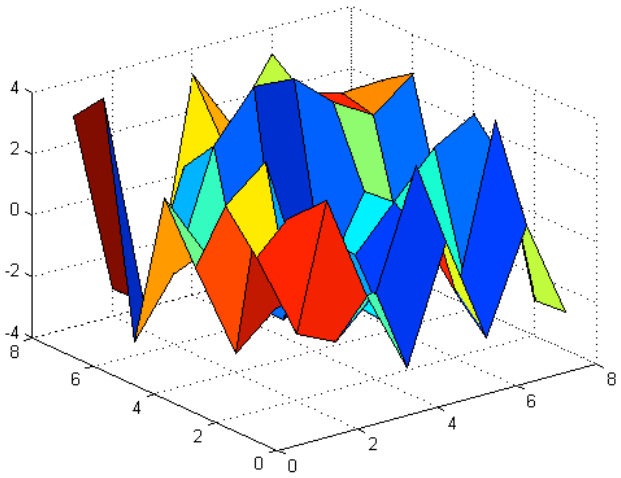


Figure 4. N=8 image BIMF1, T= 1.06 seconds.

The BIMF1 image depicted below in Fig. 5 was obtained by EMD2 from benchmark image elaine512.jpg.

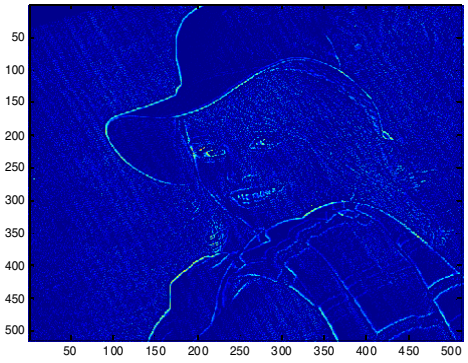


Figure 5. BIMF1 for benchmark image elaine512.

The test run timing with different input image s(t) sizes N and configuration parameters {s, N, W, PE}, as well as the computed number of extrema maximas and minimas, run elapsed time T in seconds and statistical filter window size Wen {MAX, MIN, Tsec, Wen} are shown below in Table I.

TABLE I. RUN TIME PARAMETERS AND MEASUREMENTS
Input images s: S (Synthetic) and E (elaine512)

s	N	MAX	MIN	Tsec	W	Wen	PE
S	8	10	9	1.04	0	3	-
S	256	8256	6240	9.80	0	3	-
S	512	32896	24768	99.30	0	3	-
E	512	17541	17467	16.02	2	3	1
S	1024	131328	98688	56.96	1	3	1
S	2048	524800	393984	217.4	15	3	-

VII. HHT2 SPECTRUM CONCEPTS

The concept of the 2D Hilbert Transform-based spectrum is presently in its infancy stage (practically it is non-existent). However, we can elaborate upon the HHT2 Spectrum in a few different ways and research alternative and *new concepts of HHT2 Spectrum (**HHT2S**)*, as follows:

- Firstly, we know that *2D input data (images) contain a spectrum* – you only have to look at the surface of a lake with two boats passing in two perpendicular directions, one moving across the lake from shore to shore and the other moving along the lake in the middle of it between the two shores. We can see two separate wave patterns moving across and along the lake and then merging into a complex 2D wave pattern. We also know that heritage transforms such as FFT2 can produce the 2D spectrum for linear and stationary data when FFT2 is applied to real or synthetic images. Synthetic NxN image can be produced in MATLAB by combining two sinusoids – one along one dimension axis and the other along the perpendicular dimension axis. We also used an 8x8 baseline image similar to [2] and replicate it a few times to obtain larger images, such as a 2048x2o48 synthetic image.

- Secondly, we can just apply the FFT2 to the EMD2 produced BIMFs surface functions.

- Thirdly, we can also research the feasibility of developing a 2D Hilbert Transform and then apply it to the EMD2 generated BIMF surfaces to yield the 2D functions' Hilbert Spectrum of instantaneous frequencies.

- We can also attempt to apply the HHT1 Hilbert Transform Spectrum to the meridians and longitude curves on the BIMF surfaces, as the first approximation to the 2D Hilbert Transform Spectrum.

VIII. CONCLUSIONS

We have developed a reference engineering real-time tool prototype for HHT2 EMD2 and elaborated on the HHT spectrum concept for the 2D case. We have found and evaluated the EMD2 computationally intensive process' complexity as $O(10^{14})$ for N=2048. We have introduced the concepts of embedding matrices and computationally intensive process partitioning into many smaller processes allowing them to run on many processing elements reducing the run time by an order of two magnitudes. We have demonstrated the feasibility of running the EMD2 for N<=512. We developed the partitioning of the problem for N=2048 and tested the EMD2 on N=2048 size images in real-time, compared to the state-of-the-art N=256 size images limitation. We have run the HHT2 EMD2 for N=512 using a synthetic image and benchmark input image elaine512 in real-time and performed simulation runs for N=2048 sub-process of size 2048x64 to determine the prototype system performance. The HHT2 EMD2 prototype MATLAB scripts were developed using basic commands that could be later implemented in C-Language under full user control. Much work remains to be done to realize this HHT2 prototype into an engineering tool production product with sift numbers larger than 1 and number of IMFs larger than 1 as well as developing the HHT2 theoretical background.

ACKNOWLEDGEMENTS

The authors are thankful to Dr. Norden E. Huang for originally introducing us to HHT and for encouragement in this work.

REFERENCES

1. N.E. Huang, Z. Shen, S.R. Long, M.C. Wu, H.H. Shih, Q. Zheng, N.-C. Yen, C.C. Tung, and H.H. Liu, "The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis," Royal Society of London Proceedings Series A, vol. 454, pp. 903-95, 1998.
2. S. Kizhner, T.P. Flatley, N.E. Huang, K. Blank, and E. Conwell, "On the Hilbert-Huang transform data processing system development," 2004 IEEE Aerospace Conference, pp. 1961-1979, Big Sky, MT, USA, 6-13 March 2004.
3. A. Linderhed, "2D empirical mode decompositions in the spirit of image compression," SPIE International symposium on Aerosense, Orlando, Florida, USA, 1-5 April, 2002 and Proceeding of SPIE, Wavelet and Independent Component Analysis Applications IXI, Volume 4738, ISBN 0-8194-4488-X.
4. S. Kizhner, K. Blank, T. Flatley, N.E. Huang, D. Petrick, and P. Hestnes, "On certain theoretical developments underlying the Hilbert-Huang transform," 2006 IEEE Aerospace Conference, Big Sky, MT, USA, March 2006.
5. P. Flandrin, G. Rilling, and P. Goncalves, "Empirical mode decomposition as a filter bank," IEEE Signal Processing Letters, vol. 11, no. 2, pp. 112- 114, Feb. 2004.
6. G. Rilling, P. Flandrin, and P. Goncalves, "Detrending and denoising with empirical mode decomposition," 12th European Signal Processing Conference (EUSIPCO-04), pp 1581–1584, Wien, Austria, 2004.
7. T. Kong, R. Medicherla, M. Vootukuru, and S. Kizhner, "Hardware implementation of a novel method for reducing sensor and readout electronic circuitry noise in digital domain," Proceedings of Military and Aerospace Programmable Logic Devices Conference (MAPLD), Greenbelt, MD, USA, Aug. 31 - Sep. 3, 2009.
8. E. El-Araby, "Virtualizing and sharing resources in hardware accelerated high-performance computing: An HPRC perspective," Presentation at NASA GSFC, June 2011.
9. S.M.A. Bhuiyan, R.R. Adhami, and J.F. Khan, "Fast and adaptive bidimensional empirical mode decomposition using order statistics filter-based envelope estimation," EURASIP Journal on Advances in Signal Processing, Volume 2008, Article ID 728356, doi:10.1155/2008/728356
10. C. Damerval, S. Meignen, and V. Perrier, "A fast algorithm for bidimensional EMD," IEEE Signal Processing Letters, vol. 12, no. 10, pp. 701-704, Oct. 2005.
11. NASA Goddard Tech Trends, Volume 7, Issue 4, Summer 2011.
12. R.C. Gonzalez and R.E. Woods, Digital Image Processing, Pearson Education, Upper Saddle River, NJ, USA, 2nd edition, 2002.
13. Source images used in this paper: http://www.ux.uis.no/~tranden/brodatz/D18.gif
14. Source images used in the paper (from Dr. Jacqueline Le Moigne NASA Goddard Space Flight Center): http://imageseer.nasa.gov/