FAST AM-FM DEMODULATION IMAGE AND VIDEO ANALYSIS USING SINGLE INSTRUCTION MULTIPLE DATA (SIMD) ARCHITECTURES

Paul Rodríguez V[1]., Ramiro Jordan[2], Marios S. Pattichis[1] and M. B. Goens, MD[3]
[prodrig, rjordan, pattichis]@eece.unm.edu MBGoens@salud.unm.edu
[1]Image and Video Processing and Communication Lab (ivPCL)
[2]Ibero American Science & Technology Education Consortium (ISTEC)
[3]Children's Hospital Heart Center, Department of Pediatrics, UNM School of Medicine
Department of Electrical and Computer Engineering – University of New Mexico
Albuquerque, NM 87131 United States of America
Tel: +01 505 277-1372 Fax: +01 505 277-1381

ABSTRACT

AM-FM demodulation is used for expanding a digital image/video in terms of an AM-FM series expansion. The fundamental AM-FM harmonic is estimated using a collection of bandpass filters which operate over the analytic version of the input image/video. The SIMD architecture was used to develop a fast implementation of the AM-FM decomposition. Preliminary results are shown for a novel fast segmentation of M-mode ultrasound videos, based only on the harmonic estimation.

KEY WORDS

Algorithms and Techniques, AM-FM demodulation, SIMD architecture, ultrasound analysis

1. INTRODUCTION

The primary goal of this paper is to developed a fast implementation of AM-FM demodulation. In order to understand why AM-FM demodulation is very time consuming it should be noted that the AM-FM series describes a digital image/video in terms of:

$$\widehat{X}(n_1,n_2)=\sum_{k=1}^{L} I_{[m(a)=k]}\, a_k(n_1,n_2)\cos(\varphi_k(n_1,n_2)) \tag{1}$$

where $\widehat{X}(n_1,n_2)$ is the reconstructed version of the (real 2D matrix) input $X(n_1,n_2)$, $I_{[.]}$ is the indicator function and

$$a(n_1,n_2)=\max\{a_k(n_1,n_2):k\in[1,L]\} \tag{2}$$

$$m(a(n_1,n_2))=\{k:a_k(n_1,n_2)=a(n_1,n_2)\} \tag{3}$$

Let $X_A(n_1,n_2)=X(n_1,n_2)+j\,\mathbf{H}\{X(n_1,n_2)\}$ (4)

be the analytic version of $X(n_1,n_2)$, where $H\{.\}$ is the discrete hilbert transformer which operates over the rows or columns of $X(n_1,n_2)$ (see section 3 and [2] for details), then the amplitude $a_k(n_1,n_2)$ and phase $\varphi_k(n_1,n_2)$ are estimated from the output of L band-pass (complex) filters: $h_1, h_2, .. h_L$:

$$X_{Ak}=X_A * h_k \tag{5}$$

$$\varphi_k(n_1,n_2)=\arctan\left(\frac{\text{imag}(X_{Ak}(n_1,n_2))}{\text{real}(X_{Ak}(n_1,n_2))}\right) \tag{6}$$

$$a_k(n_1,n_2)=\left|\frac{X_{Ak}(n_1,n_2)}{H_k(\nabla(\varphi_k(n_1,n_2)))}\right| \tag{7}$$

$$\nabla(\varphi_k(n_1,n_2))=\text{real}\left(\frac{\nabla(X_{Ak}(n_1,n_2))}{j\,X_{Ak}(n_1,n_2)}\right) \tag{8}$$

where $\nabla(\varphi_k(n_1,n_2))$ is the instantaneous frequency and H_k is the frequency response of the filter h_k.

At this point it is easy to identify (2), (5)-(8) as the bottleneck operations for a fast AM-FM decomposition. Moreover, all these operations should be repeated L times.

Even though AM-FM demodulation is a computational intensive operation, a fast implementation is possible if it is noted that all of these operations (with the exception of the *arctan*) can take advantage of the Single Instruction Multiple Data (SIMD) architecture, which is embedded in most current microprocessor (i.e: Motorola's PPC 74xx; Intel's PIII, P4, Itanium; AMD's Duron, Athlon, etc.). The SIMD architecture has been already used to boost the time-performance of well-known algorithms [5-8].

The SIMD execution model operates over *packed data* elements which could be located in memory or in a SIMD register; A *packed data* element is a vector with *S* contiguous elements; let *S*=4 (state of the art for single presicion floating point numbers), and $X = [x_0\ x_1\ x_2\ x_3]$ and $Y = [y_0\ y_1\ y_2\ y_3]$ be two *packed data* elements; also

let ***op*** be a SIMD math operation. Then the SIMD operation is represented by:

$$Z = X \,\mathbf{op}\, Y = [\, x_0 \,\mathbf{op}\, y_0 \;\; x_1 \,\mathbf{op}\, y_1 \;\; x_2 \,\mathbf{op}\, y_2 \;\; x_3 \,\mathbf{op}\, y_3 \,] \quad (9)$$

The capabilities of the SIMD architecture are exploited to implement SIMD compliant (packed versions) of all needed operations such as convolution, maximum, and division to lead a fast AM-FM demodulation.

This paper is organized as follows: in section 2 the scalar SIMD framework is summarized and is used to describe a SIMD compliant AM-FM demodulation algorithm in section 3; in section 4, the experimental results are presented and a novel fast segmentation for M-mode ultrasound videos is shown as an application example. Finally in section 5 the conclusions are listed.

2. SIMD Architecture and Framework

Overall SIMD technology has similar capabilities among different microprocessor manufacturers. Nevertheless the manner a particular SIMD operation is carried out depends deeply on the particular manufacturer's architecture. Well-known manufacturers are Intel with its PIII, P4 and Itanium (SIMD extensions are called SSE or SSE2), Motorola with its PPC 74xx family (SIMD extensions are called AltiVec) and AMD with its Duron and Athlon families (SIMD extensions are called 3DNow!)

The SIMD execution model operates over *packed data* elements (also it can operate over scalar data) which could be located in memory or in a SIMD register. Typical operations are: load/store (16-bit aligned), logic operations, math operations, such as addition, subtraction, multiplication, division, multiply-accumulate (AltiVec only), square root, maximum and minimum; shuffle/permutation operations and Cache-management operations are also provided.

A description of a SIMD math operation was already given in (9). A shuffle/permutation operation performs a reordering of the bits present in a SIMD register; a SIMD register is divided into blocks of bits (SSE considers blocks of 32-bits, SSE2 blocks of 16 and 32-bits and AltiVec considers blocks of 8-bits), and these blocks can be re-ordered in any given fashion.

Memory data access has a great impact on the performance of any SIMD application (load a simd-register with memory data and vice-versa). Addressed memory should be 16-bit aligned and data elements also should be continuous in memory. Moreover the memory addressing mode depends on the architecture. AltiVec uses memory indexing mode for any memory address offset, while SSE specifies an immediate operand to accomplish the same task.

A framework was recently proposed [5-8] to developed SIMD compliant algorithms. This framework is focused on efficient (time-performance as the primary constraint) implementations. It is well-known that non-linear (or pseudo-random) memory data access patterns have a great impact on the implementation's time-performance of a given algorithm. Due to this reason any implementation will be modified (if necessary) in such a way to access input data in a linear fashion. A primitive operation is defined as the core operation needed by a given algorithm (i.e: real or complex addition, multiplication, etc.). In general, any algorithm that can be expressed as a set of vector operations (over contiguous elements) will be able to take advantage of a SIMD architecture; due to this reason, it is very natural to choose a vector/matrix notation to define all primitive operations that are carried-out for a given algorithm. This mathematical notation is architecture independent.

3. SIMD compliant AM-FM demodulation

The AM-FM demodulation is a computational intensive operation, mainly because a collection of 2D Gabor bandpass filters are normally used in (5). An FFT based approach is a partial solution [9] to this problem because the size of the input 2D matrix is restricted to powers of two. In a more general setup a spatial convolution based approach is needed. In order to accomplish this a collection of 2D separable (into 1D) filters is preferred. A 2D Gabor filter can be decomposed into two 1D Gabor filters to speed-up the convolution needed in (5). In general any collection of well-defined 2D (separable) bandpass filters can be used (the optimallity showed by a Gabor filter in a continuous AM-FM demodulation is lose in the discrete case).

Also, the pre-computation of the Analytic Image has a great impact in both (i) the speed-up of the AM-FM demodulation, because it allows the use of small length (number of coefficients) filters to compute (5) and (ii) in (4) the so called 'Analytic Image' is needed to avoid ambiguity when estimating the instantaneous amplitude and phase (or frequency) in (6)-(8) (see [2,3] for details). The 'Analytic Image' is obtained by computing the 1D discrete Hilbert transform over rows or columns; this can be done by convolving the FIR Hilbert transformer with the rows or columns; a standard approach is to use the Kaiser window approximation to compute the coefficients of an M order FIR Hilbert transformer as:

$$g_{HT}(n) = (1 - I_{[n=0]}) \frac{2}{n\pi} \sin^2\left(\frac{n\pi}{2}\right) \quad n \in [-M, M] \quad (10)$$

From equations (1)-(8) four math operations are identified: (i) convolution (real and complex) in equations (4-5) and (8); note that the discrete gradient operator can be interpreted as a convolution; (ii) point wise maximum value from a set of L matrices (equation 2); (iii) pointwise multiplication and/or division of two matrices (equations 6-8); and (iv) the arctan operation (eq. 6); All these math operations (with the exclusion of

the arctan function) are built-in functions in any SIMD architecture implementation. Pointwise multiplication/division and maximum of two (or more) arrays of ordered elements are directly mapped into packed operations (note that this condition is true for equations (2) and (6)-(8)). Nevertheless, a SIMD compliant convolution is not.

Let $X = [x_0\ x_1\ ..\ x_{N-1}]$ be a vector with N scalar elements; also let $G = [g_0\ g_1\ ..\ g_{M-1}]$ be a filter with M scalar elements; then the linear convolution Y = X*G, where $Y = [y_0\ y_1\ ..\ y_{N+M-2}]$ is define as

$$y_n = \sum_{k=0}^{M-1} g_k x_{n-k} \tag{11}$$

let $X_M(k) = [\ 0_k\ X\ 0_{M-k-1}\ ]$, where 0_N is a vector of zeros with N elements, then the last equation can be written as:

$$Y = \sum_{k=0}^{M-1} X_M(k)\, g_k \tag{12}$$

note that the convolution, described as in (12), can easily take advantage of packed operations: the vector $X_M(k)$ is linearly accessed and multiplied by a scalar. Then a set of M vectors are accumulated (again they are linearly accessed). Also, the actual implementation does not use the vector X_M explicitly. This procedure will work 'as is' when its applied over contiguous elements (convolution over rows if the input matrix is assumed to be stored in a row-major fashion), and must be modified when applied over columns; this procedure leads to a transposition-free algorithm that can be applied to any higher dimension. Details can be found in [7].

4. Experimental Results and Applications

The SIMD AM-FM demodulation was implemented in the Intel's SIMD architecture (only SSE instruction set was used for compatibility reasons) and tested in a P4 (running at 1.6 Ghz, with 1 Gb of RAM and 512K of L2 cache) and in a Xeon PIII (running at 0.9 Ghz, with 2 Gb of RAM and 2048K L2 cache). The operating system was Linux (kernel 2.4.20 with the preemtible kernel patch) and the code was developed in ANSI C along with inline assembly instructions (to access the SIMD operations).

One of the main reasons to implement a fast AM-FM demodulation was to use it to segment M-mode ultrasound videos; a typical M-mode frame is shown in fig. 1. Two kind of segmentation are needed (i) to identify the medical region of interest (ROI), which is between the near field and the Epicardium [11] (also see figure 4) and (ii) segment the intenal walls of the left ventricle (LV) in order to perform computer aided Echocardiographic measurements such as LV internal dimensions, LV posterior wall thickness, interventricular septal thickness etc. (all of them with high medical impact). In the present work preliminary results are presented for (i).

The present work proposes an FM only demodulation procedure :

$$\hat{X}(n_1,n_2) = \sum_{k=1}^{L} \cos(k\ \varphi(n_1,n_2)) \tag{13.a}$$

$$\hat{X}(n_1,n_2) = -0.5 + \left(\frac{\sin(\frac{1+2L}{2}\varphi(n_1,n_2))}{2\ \sin(\frac{\varphi(n_1,n_2)}{2})} \right) \tag{13.b}$$

In equation 13.a,b only one band-pass filter (with 15 complex coefficients) is used. At this point it must be noted that if $\varphi_k(n_1,n_2)$ is considered to be a random variable ~ U[0 2π] (in eq. 13.a or 13.b) then :

$$P\{\hat{X}(n_1,n_2) \le -0.5\} = \frac{L}{2L+1} \tag{14}$$

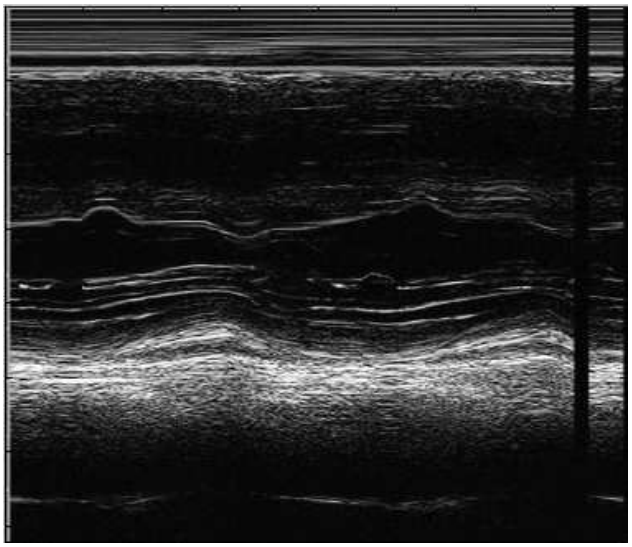

Figure 1 Original M-mode frame.

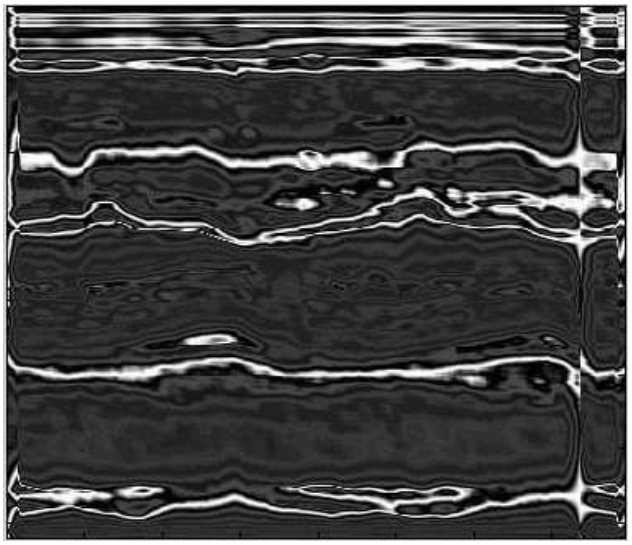

Figure 2.a FM only reconstruction using first 20 harmonics.

Equation 14 means that the reconstructed image tends to have very few pixels with values near L (maximum possible value of in eq. 13.b), when L is increased in eq. 13.b. Also note that the value -0.5 in eq. 14 will be exactly the median value of the reconstructed image (eq. 13.b)as L goes to infinity. A reconstruction using the 20 first harmonics is shown in figure 2.a; note that the wall's structure is well preserved. At this stage, a segmentation of the reconstructed image is easily achieved (see fig. 2.b) by setting a threshold which depends in the current maximum and minimum values present in the reconstructed image. Using this segmented image the boundary identification of the near field and the Epicardium was performed and is shown in figure 3. The boundaries define the medical ROI where all Echocardiographic measurements (still under development) will take place.

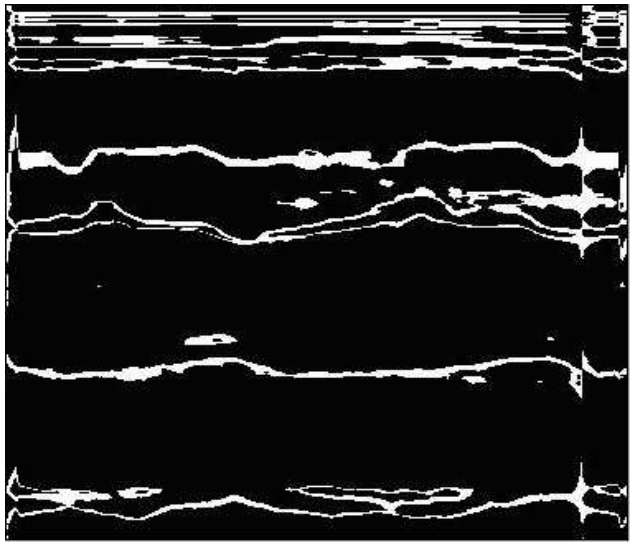

Figure 2.b Segmented version of fig. 2.a

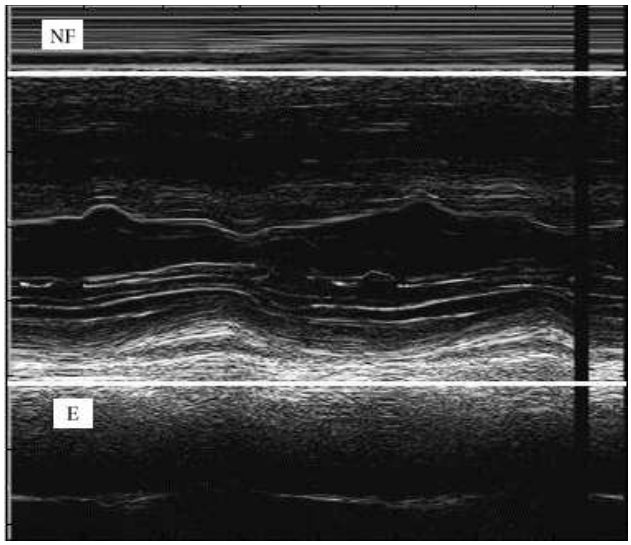

Figure 3 Medical ROI is shown. **NF** stands for "Near Fileld" and **E** stands for "Epicardium"

The M-mode video was digitized using a Matrox Meteor-II framegrabber; the original video (640x480 pixels per frame) contains text data (irrelevant for the present application) and the ultrasound video itself, which has a constant size of 396x360 pixels per frame. The overall time needed to calculate the reconstructed image (eq. 13.b) from the input image was 0.074 sec. in the Xeon PIII system and 0.048 sec. in the P4 system; theses results leads to a frame processing speed of 13.5 and 20.4 frames per second respectively.

5. Conclusions

A fast AM-FM demodulation implementationis possible. Moreover, specific problems could be suitable for a real-time processing. Ongoing work: authors are seeking an efficient parallel implementation (shared memory computers), and will apply the techniques described in this paper to develop a computer aided Echocardiographic measurements system.

ACKNOWLEDGEMENT

The authors would like to acknowledge that the computer equipments of the Hewllett Packard grant "Wireless Education" (WED2002) were used in the present work.

REFERENCES

[1] M. S. Pattichis "AM-FM Transform with Applications" Ph.D. diss. The University of Texas at Austin 1998

[2] J. P. Havlicek, J. W. Havlicek, N. D. Mamuya and A. C. Bovik "Skewed 2D Hilbert Transforms and Computed AM-FM Models" Proc. IEEE ICIP. 1998

[3] J. P. Havlicek, J. W. Havlicek, and A. C. Bovik "The Analytic Image" Proc. IEEE ICIP 1997

[4] J. P. Havlicek, A. C. Bovik Ch. 4.4 "Handbook of Image and Video Processing" Academic Press 2000

[5] P. Rodríguez V., M. Pattichis and R. Jordan "Computational SIMD Framework: Split-Radix SIMD FFT Algorithm, Derivation, Implementation and Performance" Proc. IEEE DSP2002.

[6] P. Rodríguez V. "A Radix-2 FFT Algorithm for Modern Single Instruction Multiple Data (SIMD) Architectures" Proc. IEEE ICASSP 2002.

[7] P. Rodríguez V. "A Radix-2 Multidimensional Transposition-free FFT algorithm for Modern Single Instruction Multiple Data (SIMD) Architectures" Proc. IEEE EUSIPCO 2002.

[8] P. Rodríguez, M. Pattichis, R. Jordan "Parallel Single Instruction Multiple Data (SIMD) FFT: Algorithm and Implementation" submitted to ICIP 2003

[9] P. Rodríguez V. and M. S. Pattichis "Adaptive Sampling and Processing of Ultrasound Images" Proc. IEEE ASILOMAR 2000.

[10] "IA-32 Intel Architecture Software Developer's Manual" Vol. 2, No. 245471, 2001

[11] A. Snider, G. Serwer, S. Ritter "Echocardiography in Peadiatric Heart Desease" 2nd ed. 1997 Mosby-Year