WAVE++ : A C++ LIBRARY OF SIGNAL ANALYSIS TOOLS

S. E. FERRANDO, L. A. KOLASA, AND N. KOVAČEVIĆ

1. Introduction

Wave++ is a C++ library of classes and functions designed for the serious programmer wishing to write software or scientific applications which employ the elements of wavelet analysis, time-frequency analysis or Fourier analysis. *Wave++* is a carefully designed, thoroughly tested portable library of reuseable, extensible algorithms and data structures which allow the user to write efficient, fast executing programs in a wide spectrum of signal processing and data analysis applications.

Version 1.0, which has been continuously tested and improved for the past eleven months is available for release at

`http://www.scs.ryerson.ca/~lkolasa/CppWavelets.html`.

Wave++ 1.0 comes with a makefile and will compile under g++ version egcs-2.91.66 19990314/Linux (egcs-1.2.2 release) on most Linux/Unix systems. In the near future we will compile *Wave++* on a Windows/NT machine using Visual C++ and on a Macintosh using Code Warrior.

Wave++ is free software with a very flexible copyright. Please read `/wave++/README`.

2. Mathematical Content

The mathematics underlying wave++ is the subject of *Adaptive Signal Expansions*. This is a relatively new area of mathematical research with many applications. Essentially the possibility of working with adaptive expansions extends the power of available signal analysis tools by incorporating an optimization problem when overcomplete systems of functions are available for *signal analysis*. The book [3] covers very thoroughly some aspects of this field including applications; this reference also includes propietary software with the book. Other references are [2] and [1]. An important contribution of *Wave++* is to offer an implementation of some of the fundamentals tools available for adaptive signal expansions as well as including more conventional (non-adaptive) tools. *Wave++* is built entirely on non-propietary software and users can modify and use the code as they wish subject to a very flexible copyright.

3. Content of *Wave++*

Included in *Wave++* 1.0 is *C++Wavelets* and *C++MP*. *C++Wavelets* is a library which allows one to perform the one-dimensional Discrete Wavelet Transform

Research partially supported by NSERC.

(analysis and synthesis) using orthogonal and bi-orthogonal wavelets in a periodic or aperiodic setting. Discrete Wavelet Packet Analysis/Synthesis, along with the Best Wavelet Basis algorithm, is also available. This too comes in a periodic or an aperiodic version. Full documentation for *C++Wavelets* can be found in the technical report *C++Wavelets: A User's Guide*, available at the above address. *C++MP* contains functions to implement a discrete version of the Matching Pursuit algorithm using the Gabor dictionary on an interval. As described in the technical report *A Flexible Implementation of Matching Pursuit for Gabor Dictionaries*, this version of Mathcing Pursuit allows for a high degree of flexibility when partitioning the underlying dictionary, an important feature unavailable in the other versions. While this version is reasonably fast, *C++MP* also comes with a "fast" implementation of our version of Mathcing Pursuit.

4. Philosophy

Our philosophy at the *Ryerson Computational Signal Analysis Group* is to develop an extensive, integrated library of C++ functions useful to mathematicians, scientists and engineers who write serious data analysis and signal processing applications.

Wave++ has grown out of our own need to have readily available, portable functions which perform the elements of wavelet analysis, time-frequency analysis and Fourier analysis—in short, transform based analysis of data and signals. There are a number of good software packages available, but many of our applications (in financial mathematics, signal de-noising and bio-medical imaging) push the capabilities of the commercial packages beyond their proper limits. The overhead of commercial packages can often prohibit the performance of intensive calculations. Typically, a software package is built to do many things at once while a library of functions can be used to tailor an application exactly to one's needs.

Wave++ is designed with the following features in mind: speed and reliability of execution, ease of user interface and integrability. While there are many freeware packages available, none has been satisfactory.

All components of *Wave++* are written in the C++ programming language. The algorithms which we have written take advantage of C features of C++ (e.g., bit operations, pointer arithmetic) to insure that programs which include *Wave++* library functions execute quickly. The data structures in *Wave++* take advantage of the object oriented (OO) elements of C++. We use the memory management and operator overloading features of C++ to make code writing transparent for the user[1]. Our explicit restriction to use only a subset of C++has the advantage that the user is only required to know/learn a few aspects and idioms of this language. Our main goal is computation: analysis and synthesis. We do, however, understand the need for complex data structures when performing interesting and intensive calculations. We use exactly those features of C++ to make data structure manipulation relatively simple. In order to honor our committment to speed of execution, however, we do not over indulge the OO paradigm.

We have tried to make code writing similar to writing mathematical statements; most of the tedious programming technicalities needed to encode mathematical

[1] See the two technical reports *C++Wavelets: A User's Guide* and *A Flexible Implementation of Matching Pursuit for Gabor Dictionaries* for examples

ideas are done in the background, being built into the class structures included in *Wave++*.

Wave++ is not a software package; *Wave++* is C++ code. It may be compiled into a library archive for easy inclusion into any C++ application. User designed functions using *Wave++* classes can be easily added to the library, and the classes can be modified and reused in other contexts. *Wave++* classes and functions can also be extended.

We have written more software modules that have not been incorporated into this release (*1.0*). These modules are presently being tested and modified to give *Wave++* a coherent interface with the user. Users of *Wave++* can expect to see the following additions to the library in the near future.

- Use of the *namespace* feature of C++ to ease the use of *Wave++* with other libraries.
- One-dimensional Local Trigonometric Transforms, useable with the Best Wavelet Basis Algorithm.
- A graphics package which allows the user to write graphical routines as part of ones code is also soon to be finished. This module will use the free version of OpenGL (mesa) and will compile only under Linux/Unix platforms.

References

[1] S. Chen and D. L. Donoho, *Atomic decomposition by basis pursuit.* IN SPIE International Conference on wavelets, San Diego, July 1995.

[2] M.M. Goodwin, *Adaptive Signal Models, Theory, Algorithms and Audio Applications.* Kluwer Academic Publishers (1998).

[3] Victor Wirckerhauser, *Adapted wavalet analysis from theory to software algorithms.* A.K. Peters (1994).

Department of Mathematics, Physics and Computer Science, Ryerson Polytechnic University, Toronto, Ontario M5B 2K3, Canada.

E-mail address: ferrando@acs.ryerson.ca

E-mail address: lkolasa@acs.ryerson.ca

E-mail address: natasak@home.com