Faster than an FFT, the audio-oriented QRST (spectral transform)

Richard Fobes

Author of ***The Creative Problem Solver's Toolbox***

Presentation to Westside Proggers

2011-January-27

Quick Rolling Spectral Transform™ (QRST)

Richard Fobes

Author of ***The Creative Problem Solver's Toolbox***

Quick Rolling Spectral Transform™

* Spectral Transform: Like human ear, converts vibrations into wavelength/amplitude info
* Rolling: Does a few calculations as each waveform sample becomes available
* Quick: Faster than an FFT; low latency/delay for streaming

Quick Rolling Spectral Transform™

* Still in development (interested in helping?)
* Will put on GitHub as open-source software
* Perl Artistic license
* Name trademarked —
  to prevent the name from being co-opted

Transform of audio sample through QRST algorithm

Fast playback

* No change in pitch
* No dropout
* Important for voice playback
  (people listen faster than can speak)

Slow playback

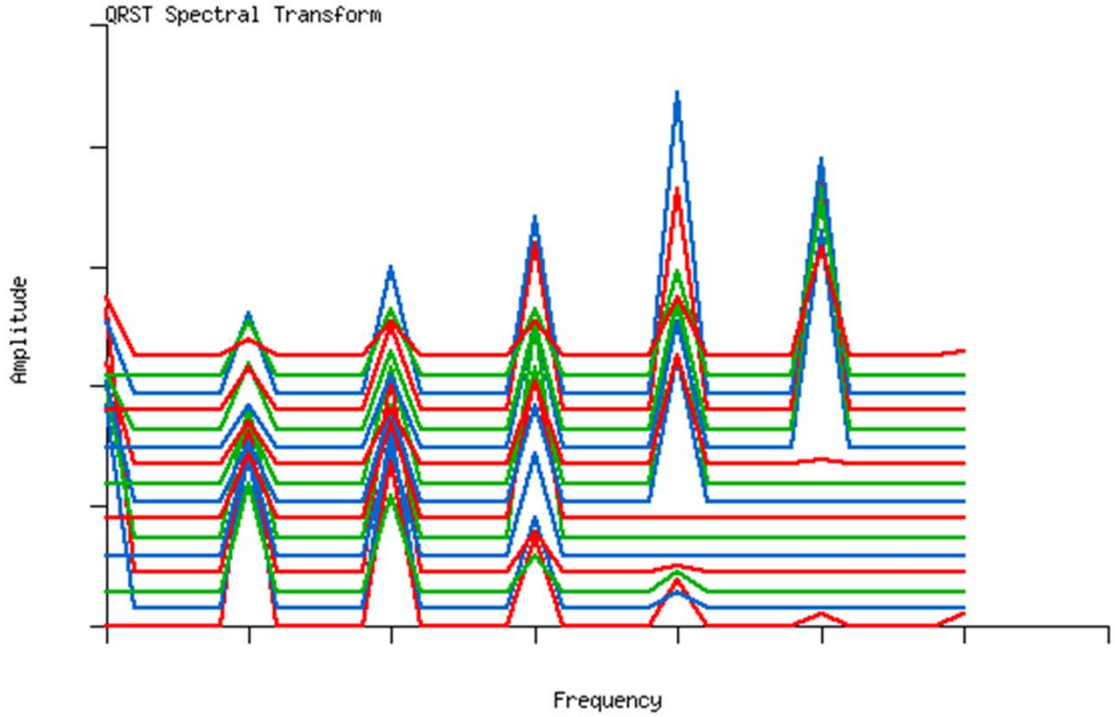
* No change in pitch
* No glitchy transitions
* Useful for writing down phone numbers, etc.

Equalization-adjusted playback

* Just requires a few lines of code
* Noise reduction

QRST graphic output of changing-wavelength sine wave

Bottom spectrum is earliest time, top is latest time



Applications for Quick Rolling Spectral Transform™

* Voice-to-visual technology for deaf
* Voice messaging – big gap in digital technology
* Improvements in voice recognition
* Streaming, efficient, voice/audio compression

Voice-to-visual methods long overdue for deaf

* Ideally involves designing a “Fully Phonetic Sign Language” (FPSL)
* (FPSL also useful for non-deaf to communicate underwater, through window, in quiet hall, at noisy party)
* This is the project for which I originally designed the QRST algorithm

Missing: Voice messaging

* **Voice-based** version of text/instant messaging
* Sort of like email, but **talking** instead of typing, and **listening** instead of reading
* **Microphone** better than keyboard, especially on **mobile devices**
* **Not** a **replacement** for: text/instant messaging, voice-mail, or telephone
* **Requires** speeded-up playback — because we **listen faster** than we speak
* This is a **very big gap** in digital technology
* Open-source **opportunity** (using email infrastructure)?

Voice-recognition software not yet robust

* Dragon Naturally Speaking quite good, but other proprietary **voice-recognition** software (e.g. IBM & Microsoft) not robust
* Use FFT & "Hidden Markov models" (primitive)
* Try QRST & identify independent dimensions of vocal sounds (including singing)
* No usable **open-source voice-recognition** software
* (Database of written words and their pronunciation already open-source available)

Opinion: Current audio compression methods are inefficient

* **Wasted bandwidth and storage**, especially on mobile devices
* Video compression better than audio compression (which is half of "video")
* **Fast, low-latency voice** compression needed for non-entertainment uses
* (Music compression OK as-is?)

QRST compression

* Raw audio file: 110 KB
* QRST compressed file: 16 KB (?)
* MP3 compressed file: 21 KB
* Ogg/Vorbis compressed file: 19 KB

Seduced by cleverness of Fast Fourier Transform (FFT)

* Fourier transform not a good match for **streaming** and **real-time** applications (e.g. iPhone)
* **Resource hog**, especially for mobile devices
* Computes **wrong information** for real-time voice applications
* Input samples **overlap** when attempt to increase time resolution
* FFT actually slow, because **Fourier transform is slow**

Features & benefits of Quick Rolling Spectral Transform ™

QRST Features	QRST Benefits
Fast!	Suitable for **real-time applications**
Only simple calculations (e.g. divide by power of 2) on relatively few integers	Can implement in **hardware** (e.g. FPGAs and IP cores) and **mobile devices**
Low latency/delay	Good for **streaming** applications
Small size of QRST compressed data	Consumes less **bandwidth,** less **storage**
Variable-speed without disruptions	Needed for **listening quickly**

Historical perspective

* Writing text arose because **ink** on **paper** was the only practical "storage device" for people's words
* Eventually, all documents will be accessible using **fully phonetic writing systems** not yet invented
* ("International Phonetic Alphabet" is not fully phonetic)
* In future, each person will **view words** in whatever writing system they grew up learning

How does the QRST algorithm work?

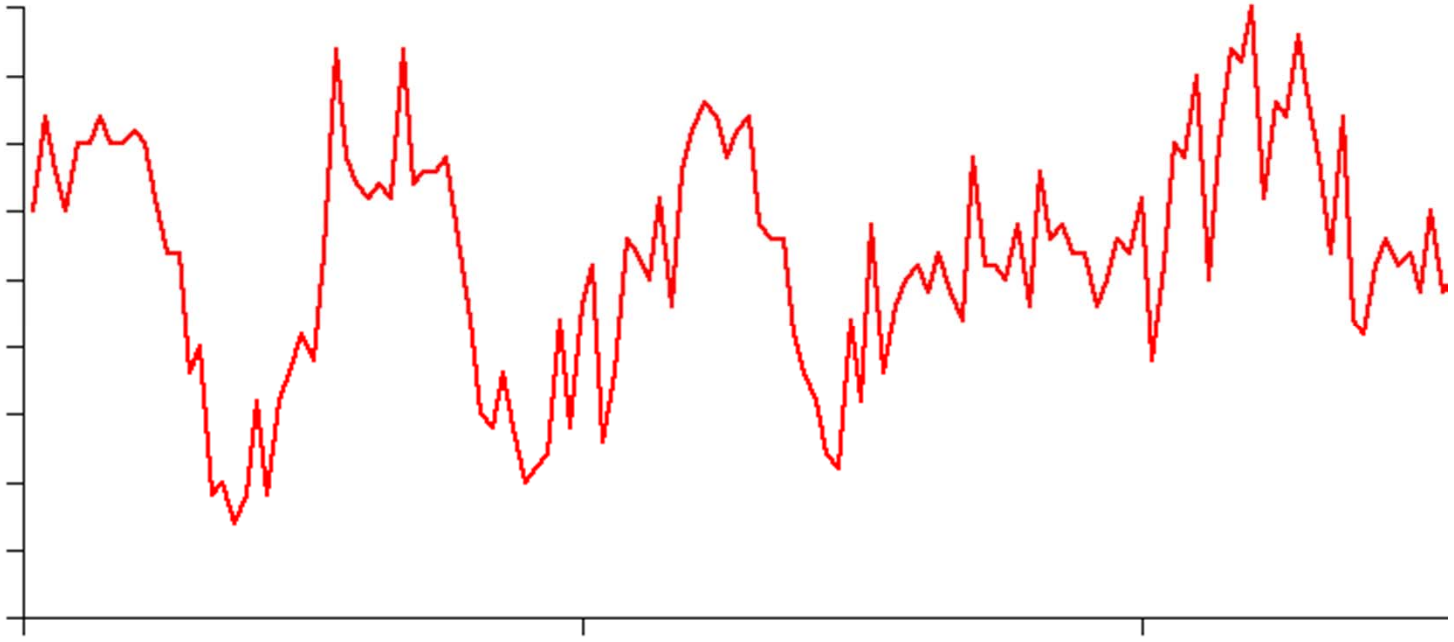
1. Low-pass filtered by octave
2. High-pass filtered by octave
3. Peaks and troughs identified — within each octave
4. Wavelength measurements from peak-to-peak distances
5. Amplitude is sum of up-and-down movements
6. Higher-frequency results produced more often — and sooner — than low-frequency results

One sine wave per octave

* Currently tracks just one wave within each octave
* Might need to double that number (interlaced) for better quality

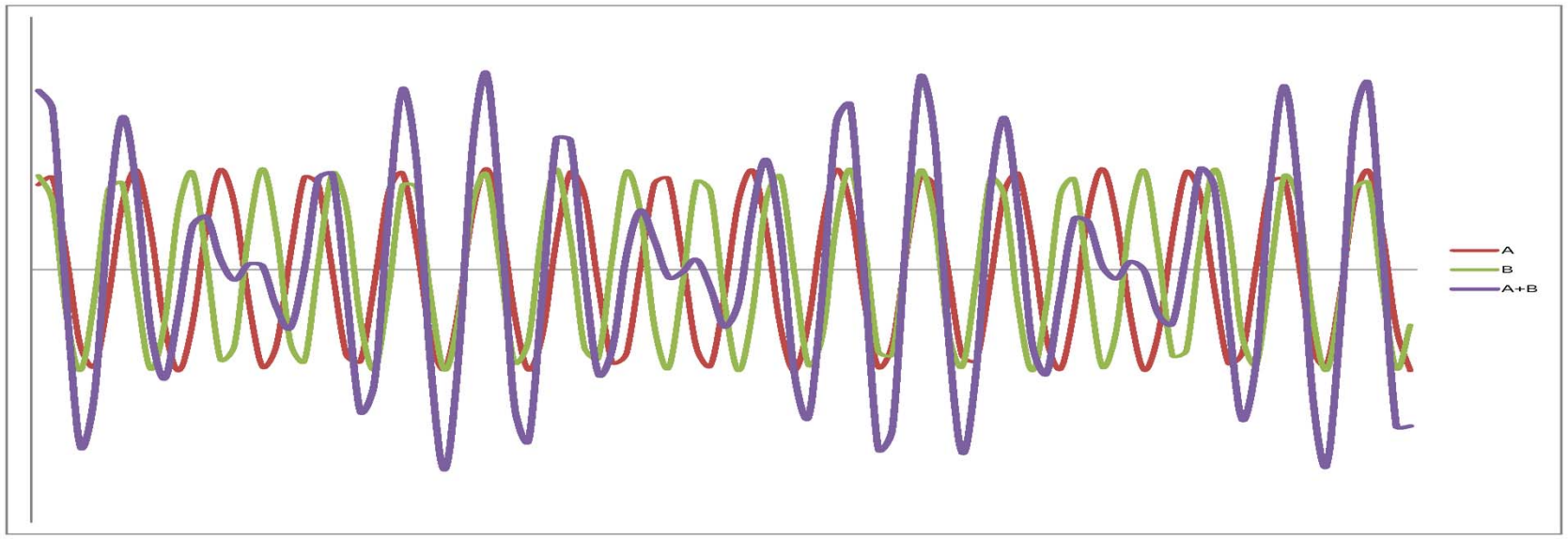
Voice waveform

* When view waveform of voice, don't see lots of frequencies as Fourier transform would suggest



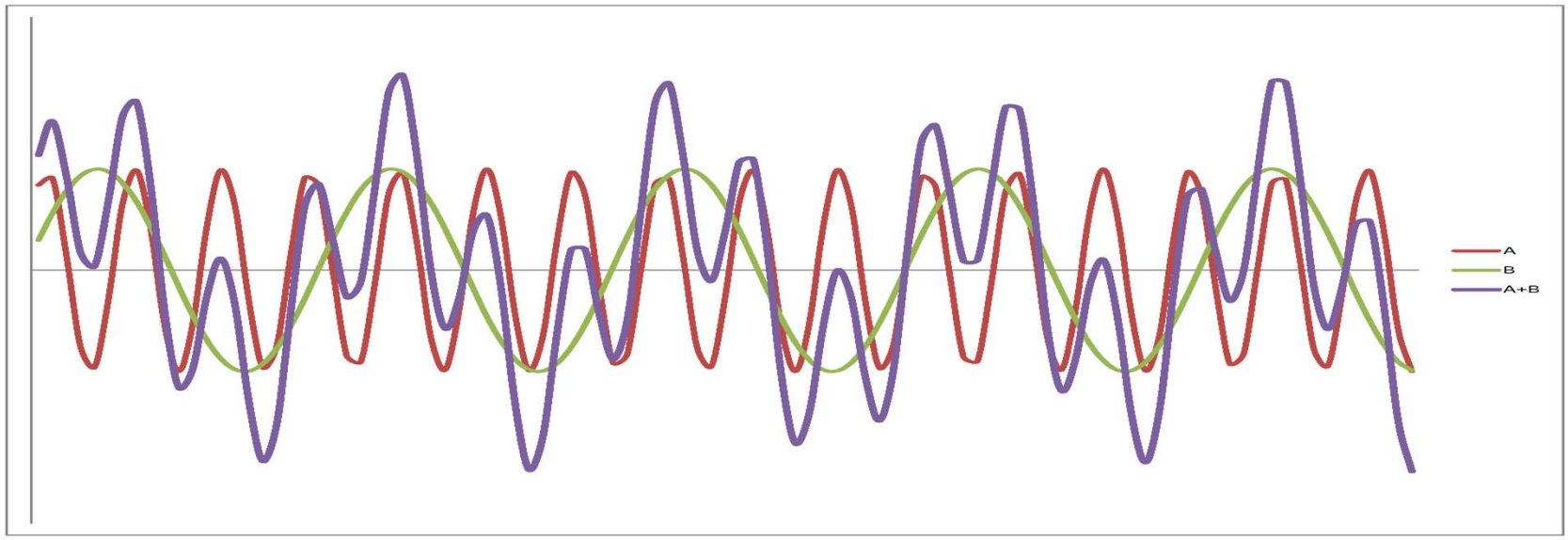
Two sine waves with similar wavelengths

* When two sine waves within same octave, result is **single sine wave** that changes (smoothly pulses) in amplitude



Two sine waves in different octaves

* When two sine waves in **different octaves**, result is **higher-frequency** sine wave moves up and down at **lower frequency**



QRST 4-byte compression format

* Time delay (8 bits) (255 indicates time extension…)
* Channel (4 bits)
* Octave (4 bits) (15 is always highest octave)
* Wavelength within octave (8 bits)
  (256 notes per octave, compares to 12 on piano)
* Amplitude (8 bits)
  (256 volume settings, not analog levels)

Is it patentable?

* No (for small business) and yes (for unethical corporation)
* Most software patents not really patentable, but cheaper to pay for license than fight rich corporation in court
* Patent system is dysfunctional; small business cannot defend patent from infringement
* Long-term solution: "open patent" – for sharing instead of monopolizing
* Short-term solution: put on GitHub as open-source software

What's Richard's motivation?

* Attention for ***The Creative Problem Solver's Toolbox***, which teaches what should be taught in schools
* **Invigorate economy** with useful new things to buy
* (Soapbox: Government should reduce taxes on businesses that **create** what has **value** and previously did not exist, and slightly increase taxes on businesses such as banking, insurance, lawyers, etc. that primarily shift what already exists)
* Credibility/attention for **VoteFair ranking** and **NegotiationTool.com** and **Dashrep** language

What is the opportunity for you?

* Contribute to development of open-source QRST on GitHub (tests, improvements, to-do's, etc.)
* Get experience with vocal processing, which is ripe for increased business opportunities
* Possibly implement refined QRST in hardware as business opportunity (talk to me)
* Pioneer a new digital frontier

QRST removes barriers to overdue applications

* **Voice messaging**
* Robust **voice-recognition** software
* **Microphones** instead of tiny keyboards for **mobile devices**
* Dictating notes, editing notes, and other **new voice-based** applications
* Better **audio compression** (for download speed and storage size), and for videos
* **Deaf** can "see" voices in meaningful ways
* And probably **much more**

Thank you

Richard Fobes

www.Fovationz.com

Westside Proggers Google group

prog@SolutionsCreative.com