Sound Vision 2.0: Bark Scale Visualizer

Eli Stine
Composition & Computer Technologies, University of Virginia
ems5te@virginia.edu

ABSTRACT
There exist a great deal of visual real-time signal analysis tools and music visualizers ([1], [2], and [3]) whose purpose is to allow a user to gain information that can’t be determined (or can’t be determined easily) through auditory analysis alone. This information can be used in a variety of ways: to notate, mix, and/or inform the composition of music, for example.

It was my purpose in the creation of the Sound Vision software, and the subsequent addition of its Bark scale visualizer module described in this paper, to create a music visualizer that would be useful for the analysis of electroacoustic sound art, music that has unique characteristics (dynamic spatialization, large timbral shifts), and thus requires unique tools to analyze.

A brief background on the Bark psychoacoustical scale and music visualization is followed by a description of the mechanics of the Bark scale visualizer, its parameters, and successes and issues with the system. The paper concludes with examples of potential future work.

Keywords
Bark psychoacoustical scale, critical bands, music visualization, MaxMSP, Jitter.

1. INTRODUCTION
1.1 Project Purpose
The purpose of this project was to extend the Sound Vision software, whose goal is two-fold: to aid in the analysis of electroacoustic music, with an emphasis on clarifying spatialization, mixing choices, and spectral design, and also to function as a responsive visual accompaniment to music in live performance. While a stereo FFT visualizer (the component that existed in Sound Vision before this project) is helpful to visualize spectral details and spatialization, it tends to be difficult to visually follow sound sources (melodic instruments, for example) using it. The Bark scale visualization component, being a “coarser” measure of spectral activity, was intended to visualize general areas of the spectrum and to reveal different information about the input signal than the stereo FFT could, with the intention that in conjunction they could reveal more information about inputted music than they could separately.

1.2 Bark Scale Critical Bands
Within the cochlea, the sense organ of hearing within the inner ear, there exists a set of “critical bands”: bands of audio frequencies within which a second tone will interfere with the perception of the first tone by auditory masking [4]. The Bark scale, devised by Eberhard Zwicker in 1961, divides the auditory range of the ear into areas that roughly correspond to the cochlea’s critical bands, resulting in a non-linear scale with most resolution around 2kHz or so, where the critical bands are closest [5]. Thus, the Bark scale offers a much more psychoacoustically pertinent scale than dividing the spectrum into equally spaced Hertz distances for the analysis of musical signals.

1.3 Music Visualization
1.3.1 History & Similar Projects
The first music visualizer was the Atari Video Music, which was entirely analog, and used simple heuristics about an incoming stereo signal to display dynamically colored diamonds [6]. A more recent, better known development in music visualization is the Advanced Visualization Studio, which is a programmable scripting language used to create visualizers in the popular WinAmp and iTunes softwares in the 2000s [7]. Another more modern visualizer aimed less at consumers and more at musicologists and electronic music composers is Sonic Visualizer, which offers a suite of different visualization tools to analyze digital audio [8].
1.3.2 Atomic Techniques

All of the music visualizers mentioned in 1.3.1 take advantage of some of the atomic methods for visualizing (audio) signals: oscilloscopes, spectrograms, vectorscopes, and EQ visualizers.

Oscilloscopes display amplitude over time, either the value of samples in the digital domain or voltage in the analog domain. Spectrograms display frequency and amplitude over time, and tend to have frequency displayed on the Y-axis, time on the X-axis, and color or opacity used to display amplitude/energy. Vectorscopes display the difference between signals (for example, a correlation meter, or goniometer, compares left and right channels within a stereo signal, and is used to determine monophonic compatibility) [9]. EQ visualizers pass a signal through a series of filters and then create a real-time bar graph, with the height of the bars in the graph corresponding to the amplitude of the signal coming out of each of the filters. The Bark scale visualizer implemented for this project is a combination of the EQ visualizer and vectorscope.

2. MECHANICS

2.1 Outline

Very roughly, this program was created with the MaxMSP/Jitter [10] program and displays Bark bands on the Y-axis, panning on the X-axis, and amplitude of the Bark bands through color and opacity.

More concretely, the program starts by passing a stereo audio signal through 25 filters, as specified by Bark critical bands (using Tristan Jehan’s analyzer~ [11] Max/MSP object), the output of which is monitored and stored as 25 x 2 Bark band levels (for each channel). Next, the stored band levels of the same band are compared and their difference is used to generate 25 panning locations (1 - 5, where 3 is center) for the Bark bands. Next, each band is stored in a 25 X 5 matrix, with color and opacity dependent on band level* and horizontal location dependent on panning location. Lastly, the video is lightly processed, with gating (blacking out a band if it is below a certain luminance threshold), video feedback (for visual smoothness), and dynamic opacity adjustment (makes quieter bands more transparent and louder bands more opaque).

2.2 Parameters

While using the Bark scale visualizer, a user has control over a number of parameters that will manipulate the visualization and fine-tune it to remove unneeded information and isolate spectral areas or volume levels of interest.

The available parameters (as seen above in Table 1) are:

1. Gating Threshold - blacks out visualized bands below the set threshold (0. - 1.). Allows the removal of quieter, less important sounds within a texture to reduce visual clutter.
2. Number of Visualized Bands – displays only the n loudest bands (n ∈ [1, 2, ..., 25]). Can be useful to track the loudest or several of the loudest spectral bands within a texture (e.g. vocals, pitch centers of melodies, etc.).
3. Panning Width – enhances or attenuates the effect on horizontal placement of bands as a result of stereo channel differences. Ranges from mono (filling entire horizontal space) to enhanced stereo, where small channel differences are exaggerated.
4. Opacity Effect – determines how much the relative levels of bands affect the overall opacity range of the image. For example, if Opacity Effect is low, all bands will be nearly opaque, if Opacity Effect is high, the loudest

* Color is determined by mixing three matrices corresponding to red, green, and blue channels individually: the red matrix only displays bands above 0.66 luminance (hot), the green matrix displays bands between 0.33 and 0.8 luminance (warm), and the blue matrix displays bands below 0.55 luminance (cool). Opacity is created by finally multiplying the color matrix by the same matrix in greyscale, effectively increasing the range between the most opaque and most transparent pixels.
bands will be completely opaque and the quietest will be nearly transparent.

5. Video Feedback – smooths the visualization by not entirely erasing previous frames. Ranges from no feedback (mechanical, precise) to very high feedback (very slow changes within visualization, with repeated spectral activity emphasized).

Other parameters that exist internally to the program but are not revealed to the user are FFT window size and coloring schemes.

3. RESULTS

3.1 Successes

Overall, the Bark scale visualizer shows areas of activity and relative volume levels of parts of the spectrum in real time effectively, through the use of both color and opacity to delineate volume differences. A user can track multiple pitch lines or instruments (voice, bass, drums, etc.) in pitch space effectively, but not extremely accurately: if a frequency within a band goes up in pitch this is visualized as a color and opacity change between the band the frequency is within and the band immediately above it, revealing relative pitch changes but not precise, absolute frequency changes.

The Bark scale visualizer can also clarify mixing techniques or the spectral coverage of a texture or instrument: for example, it revealed to me the area of high frequencies that are activated to get a “breathiness” with recorded vocal material.

Table 2. Image from visualization of Jean-Claude Risset’s *Mutations* [12]

3.2 Issues

There are some issues within the Bark scale visualizer. Visualizing panning is only effective in simpler textures and when extreme. Results are clearest with music having a high dynamic range (less compressed) and only a few instruments within different ranges of the spectrum. For example, Table 4., which displays an image from the visualization of Megadeth’s *Peace Sells*, has a large yellow middle section which is saturated by drums, guitar, and vocals, and it is very difficult to pick them out as separate sources. Ultimately, the effectiveness of the Bark scale visualizer depends on the music it is visualizing.

4. FUTURE WORK

A number of future additions/modifications to this system suggested themselves during its creation. Unlike the stereo FFT that previously existed in Sound Vision, the small number of bands in the Bark scale visualizer could lend itself to being activated in the world of hardware, through a DMX-controlled lighting matrix. With the panning width and video
feedback parameters turned up, the Bark scale visualizer is also effective as a more ambient, less informative visualizer: live music eye candy (see Table 5). Another possible development is the integration of the Bark scale visualizer within the stereo FFT visualization of Sound Vision, with the vertical dimensions of the stereo FFT warped to match that of the Bark scale.

Table 5. Image from visualization of a recording of Haydn’s Piano Sonata No. 59 in E Flat Major [15]

5. REFERENCES


