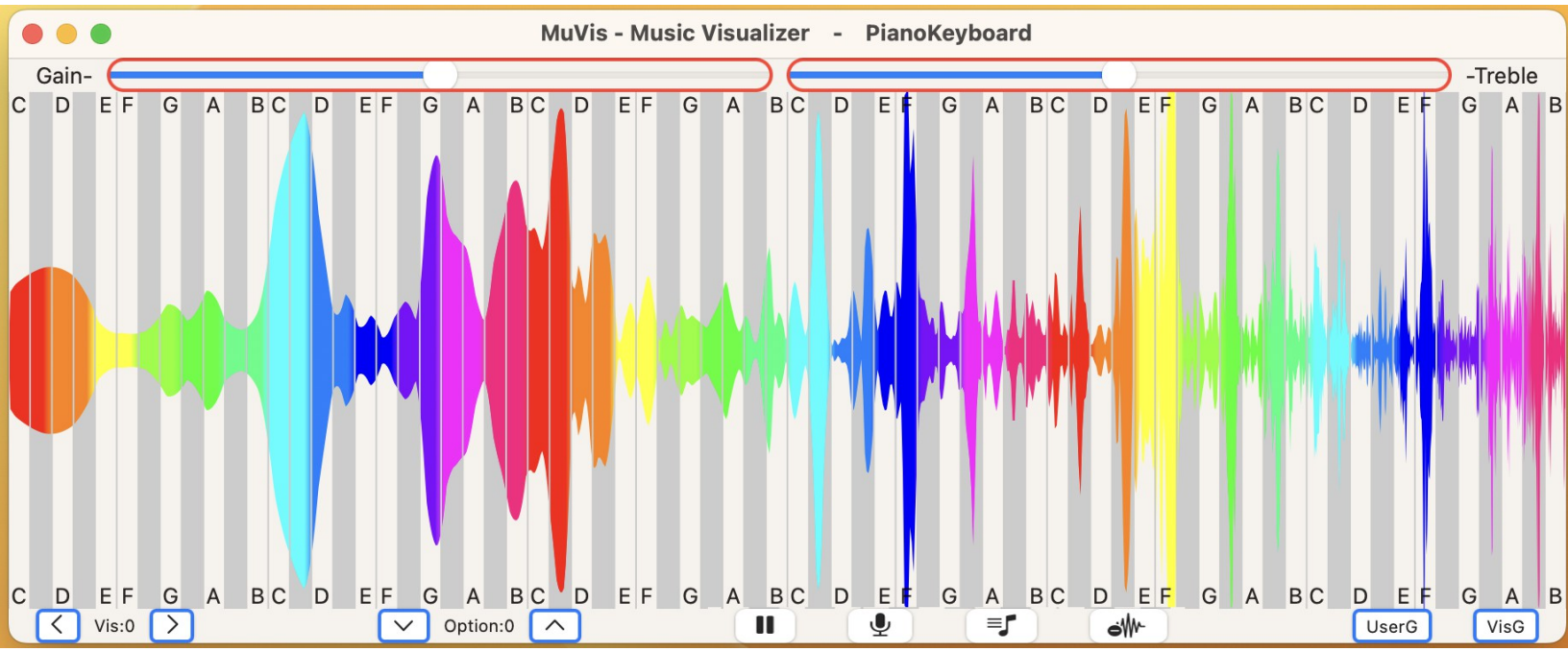
MuVis User Guide
MuVis - an Audio Visualizer app for Music
"See what notes they're playing."
by Keith Bromley, San Diego, CA, USA kbromley@me.com

(updated for MuVis v 2.1 on 11 June 2023)

When the MuVis app first opens, the on-screen display should look somewhat like:



and the computer's speakers should be playing music from a pre-loaded royalty-free music file. (Currently, the song file is "The Elevator Bossa Nova" from Bensound.com.) The on-screen display shows a real-time dynamic visual representation of the audio that you are hearing.

On your Mac, click-and-drag any corner of this window to freely re-size it to any width and height you wish. Some users like tall thin windows; some like short wide windows; while I prefer approximately square windows.

On a Mac, the menu bar at the top shows the app's title "MuVis (Music Visualizer)" followed by the name "Spectrum" of the particular visualization selected for display. On an iPhone or iPad, the menu bar is not displayed.

The top and bottom toolbars contains buttons and sliders to allow you to control what you see.

At the bottom left are two buttons labelled by left- and right-chevrons. Clicking the left one changes the display to the previous visualization, and clicking the right one changes the display to the next visualization - which allows you to cycle through the twenty-four visualizations currently provided by the app. On a Mac, as a convenient keyboard-shortcut, you also can simply use the left-arrow and right-arrow keyboard keys to cycle back-and-forth among these visualizations. On your iPhone or iPad, you can simply swipe left or right to change the visualization.

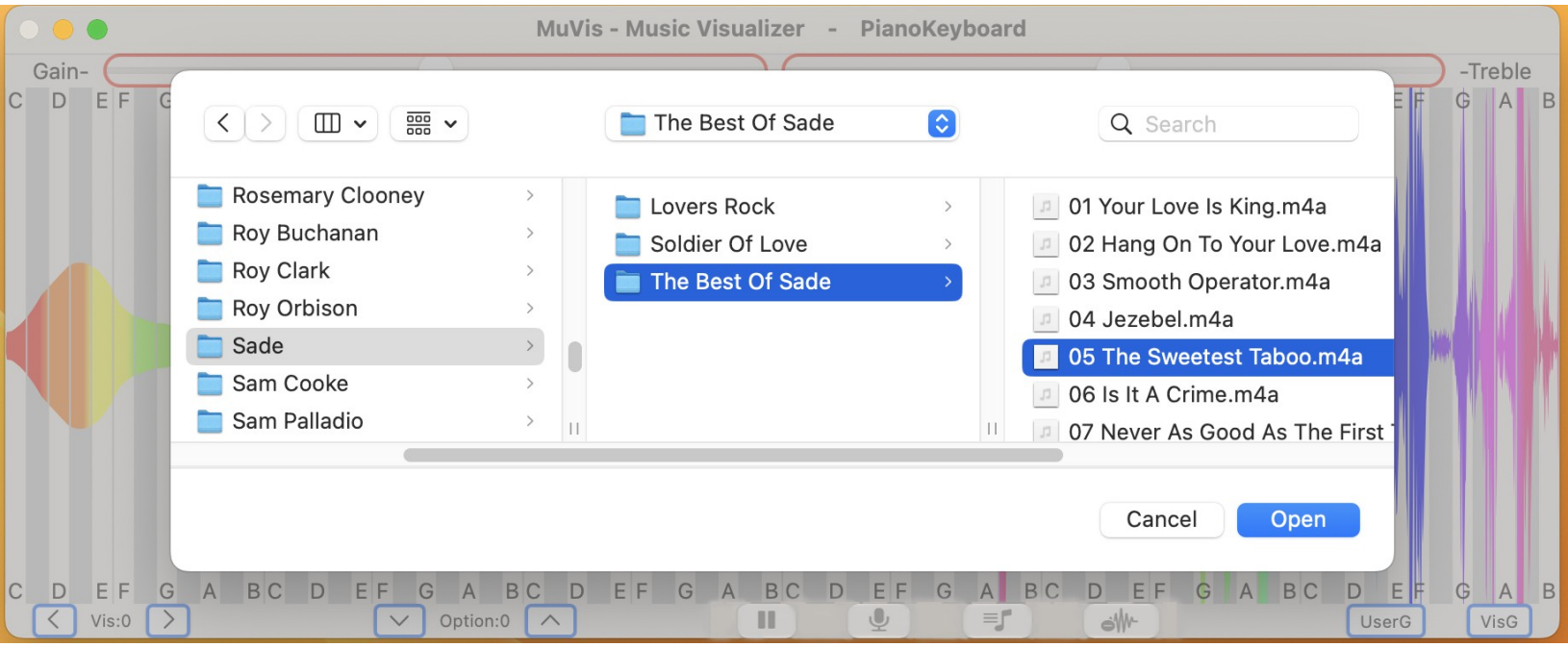
Adjacent to these Visualization chevrons are two Option chevrons. Clicking the downward-pointing one changes the display to the previous option, and clicking the upward-pointing one changes the display to the next option. When programming these visualization algorithms, there are numerous variations and options available to the developer. This button allows the user to select to see any of four variations. The particular variation rendered is different for each visualization. The user should check out the option for each visualization to see which she likes.

On a Mac, as a convenient keyboard-shortcut, you also can simply use the up-arrow and down-arrow keyboard keys to cycle among these options. On your iPhone or iPad, you can simply swipe up or down to change the option rendered.

Adjacent to these Option chevrons is a “Pause” button. When clicked, it stops the music playback and freezes the visualization display. The button icon changes to “Play” which, when clicked, causes the music playback and the visualization display to continue.

The adjacent button has a “Microphone” icon. When pressed it stops playing the audio from the selected song file (and also stops it's visualization), and starts playing the live audio from the microphone (and starts it's visualization). The icon then becomes a slashed microphone - which, when pressed, performs the reverse actions and changes the icon back to "Microphone".

The next button has music-list icon. (It is disabled and grayed-out if the microphone is enabled.) When the button is clicked, a sub-window pops up looking somewhat like:



On a Mac, it allows you to select whatever songs you want from your own Music folder. Selecting a song (by navigating your Artist and Album structures and clicking on its name) causes that song to play - and the visualization to render its dynamic attributes. Unfortunately, on an iPhone or iPad, the pop-up window allows access only to songs loaded into your iCloud Drive folder (called “Files” in iOS.). (Apple allows only its own Music app to have access to songs in the iPhone or iPad music library.)

The adjacent button has a Waveform icon with a minus sign. When clicked, it passes the audio signal through a “spectral enhancer” filter which reduces the display of noise (usually percussive effects which smear spectral energy over a large frequency range) and enhances the display of spectral lines (usually the harmonics of musical notes). The icon then becomes a normal Waveform which, when clicked, removes the enhancing filter. (Note that this enhancement applies only to the visual display, not to the audio you are hearing.)

Previous versions of MuVis had a button labelled “Dark” to toggles the app's window between Light and Dark modes. Now, on an iPhone or iPad, the user can select Light/Dark mode in the Settings | Display & Brightness pane. On a Mac, the user can select Light/Dark mode in the System Settings' Appearance pane. (I personally use the free app “Nightfall” from github.com/r-thomson/Nightfall/releases to put an icon in my Mac's menu bar to easily switch between modes.) Different visualizations look subjectively better or worse in either Light or Dark

Mode. Feel free to experiment and pick your favorites.

Finally, at the right end of the bottom toolbar are two buttons labelled UserG (for User Guide) and VisG (for Visualizations Guide). Clicking either button activates a “Quick Look” popup window to display the app's documentation. The User Guide (the document you are currently reading) explains the functionality of the app's user interface. The Visualizations Guide describes the purpose of each visualization - probably in more detail than you want.

The top toolbar contains two sliders - one controlling "Visualization Gain" and the other controlling "Visualization Treble Boost". As their names imply, sliding them adjusts the gain and slope of the visual data representation. (Note that they do NOT adjust the audio levels of the music your are hearing.) They allow subjectively adjusting the display to be the most pleasing - despite whether the musical passage is very quiet or very loud.

Note that, whenever the “Mic On” button is clicked, the “Song” button is disabled (grayed-out).

Note also that, whenever the “Pause” button is clicked, the “Microphone”, “Music List”, “Waveform”, “Visualization Previous/Next”, and “Option Previous/Next” buttons are disabled (grayed-out).

On an iPhone or iPad, for convenience, I have enabled swipe-action shortcuts for some of these buttons:
swiping left on your screen advances the display to the Next Visualization;
swiping right on your screen advances the display to the Previous Visualization;
swiping up on your screen advances the “Option” button the the next higher setting; and
swiping down on your screen advances the “Option” button the the next lower setting.

Also, triple-tapping the screen removes (or puts back) the top- and bottom-toolbars to allow the visualization to utilize the entire screen.

On a Mac, for convenience, I have enabled keyboard shortcuts for some of these buttons:
pressing the keyboard's left-arrow key presents the Previous Visualization;
pressing the right-arrow key presents the Next Visualization;
pressing the down-arrow key advances the “Option” button the the next lower setting;
pressing the up-arrow key advances the “Option” button the the next higher setting.

Personally, I enjoy running the MuVis app on a Mac while keeping my right-hand fingers on the keyboard's up / down / left / right arrows – to facilitate easy toggling through the various options.

The simplest way to use MuVis is to manually select each song you want to play (and visualize) using the “Song Selection” pop-up window described above. However, for Mac users, the following alternative way is available:

1. Click on the “Microphone” button so that MuVis displays the visualization of whatever audio the microphone hears.

2. Run the Apple Music app (or any other music player app) to select and play any song from your music library (or from any music subscription service). This audio will be played through the Mac's speakers – which will be “heard” by the microphone – resulting in the screen rendering a visualization of the music playing.

This “speaker to microphone” acoustic process may result in undesirable audio feedback and the introduction of

background noises. But these problems can be eliminated by using the free BlackHole app from existential.audio/blackhole . BlackHole allows the Mac's microphone-input circuitry to "electronically tap into" the audio driving the computer's speakers such that the audio signal being fed into MuVis is the pristine digital audio signal being supplied by your music-player app. I use this process to "get the best of both worlds" - that is, the convenience of using the Apple Music app to select songs and playlists – as well as the visual renderings provided by MuVis. The following section provides details of how to set this up.

The Mac version the "MuVis - Music Visualizer" app can be used in conjunction with the BlackHole audio driver. (Both are free.) BlackHole is a modern macOS virtual audio loopback driver that allows audio to be passed between applications. Once this combination is installed on your Mac, you will be able to use Apple Music (or any other music player app) (e.g., Colibri, Pine Player, Spotify, Amazon Music, GarageBand) to play your desired songs or playlists and the MuVis window will display real-time graphics depicting interesting musical features of whatever is being played.

Apple does not allow the distribution of drivers via their App Store, so I am not able to include BlackHole as part of the MuVis app. You will have to perform a little tricky initial setup. But this only requires a few minutes, and once it is done, it never has to be done again. Here are the steps to installing and setting up the BlackHole driver:

1. Download the BlackHole Installer from existential.audio/blackhole . Pay the desired donation amount (currently $10) to keep this free software supported, and on the subsequent screen, select Download BlackHole 2ch. This puts "BlackHole2ch.v0.5.0.pkg" into your Downloads folder. Double-click on this package, and then run the pop-up Installer app to install the driver into your macOS system. (Note that, unlike installing an app, you will not see BlackHole in your Applications folder.)

2. To give you comfort that this driver was installed properly, you can go to System Settings | Sound and confirm that "BlackHole 2ch" is listed as a virtual device in both the Input and Output sections.

3. In the System Settings | Sounds window, select "BlackHole 2 ch" as both the input and output virtual devices. At this point, whatever you play from the media player (e.g., Apple Music), will play into the MuVis app (using it's micOn button) and display real-time graphics. But you won't be able to hear anything - because the audio output is going to a virtual device and not to the desired output speakers. So you have to set up what is called a "Multi-Output Device" To do this, you need to open the "Audio MIDI Setup" app which is in the Utilities folder within your Applications folder. (If the "Audio Devices" window does not immediately appear when opening "Audio MIDI Setup" select Audio Devices from the Windows drop down menu in Audio MIDI Setup.) In the lower left hand corner of the Audio Devices pane, click the "+" icon, and select "Create Multi-Output Device". In the resultant pop-up window, select "BlackHole 2ch" as your Primary Device, and select 44.1 kHz as your Sample Rate. And click on the "Use" boxes in front of "BlackHole 2ch" and "External Headphones" (or whatever output speakers you want the audio to go to). Check the "Drift Correction" box after "External Headphones".

4. Now, when you go to System Settings | Sound, you will see "Multi-Output Device" listed as an option for the Output of your system's audio - which sends the audio from the Apple Music player to both the output speakers AND the BlackHole virtual Device (which in turn sends it to the MuVis app).

5. If you change the settings in the System Settings | Sound panel (for example, to use other audio-producing apps), then you will need to remember to come back and select "Multi-OutputDevice" when you want to use the MuVis app.

For more information on the BlackHole loopback driver, see their Wiki site at github.com/ExistentialAudio/BlackHole/wiki .

Once you've performed this installation then, in the MuVis app, simply triple-tapping the screen puts MuVis into the mode of using BlackHole input instead of file/microphone input.
