Z. Yingfang, P. Yi, Z. Junren. "Study on Application of Audio Visualization in New Media Art."
IOP Conf. Series: Journal of Physics: Conf. Series 1098. 2018.
doi :10.1088/1742-6596/1098/1/012003.

The work focuses on some of the principles behind audio visualization, explores some techniques used, and shows some examples of audio visualization. This is useful, as it gives me some idea on how to make my own audio visualizations, and gives me inspiration to make some interesting visualizations. The work gives some examples of how visuals can be linked to emotions and mental imagery, which is what I am aiming for. The authors are professors at a university. The authors conclude that audio visualization can be designed in a growing number of ways. I personally discovered that audio visualization is dependent on the fast fourier transform algorithm which transforms the audio data into an array of frequencies that can be used on the graphic.

Yannakopoulos, Yannis. "Creative Audio Visualizers". *Tympanus,*
https://tympanus.net/codrops/2018/03/06/creative-audio-visualizers/. 2018.

The work explores how to create a basic music visualization using p5.js. This is useful, as it can get me started with creating music visualizations in p5.js. The work gives specific code, so I can study it and figure out how everything works. The author is a freelance developer exploring generative art, WebGL, GLSL and web audio. The author observes that p5.js can split the audio into separate frequencies, so you can make selective parts of your graphic respond to certain frequencies. That was probably also the most important thing I learned from the article.

Bhardwaj, Prakhar. "Music Visualizer with THREE.js & Web Audio API". *Medium*,
https://medium.com/@mag_ops/music-visualiser-with-three-js-web-audio-api-b30175e7b5ba. 2018.

The article explores how to create a music visualization using three.js and the web audio API. This is useful since it goes into how the Web Audio API works under the hood, and since I am using that API, it could be useful to know. It also talks about some three.js basics, such as creating a scene, a light, camera, render, etc. I couldn't find the author's credentials, but judging from the article, they seem to know what they're talking about, and this isn't a research article, so I think it's okay. The author observes that in the array returned from getByteFrequencyData, higher frequencies are in the upper half, and the lower frequencies are in the lower half. I observed that sometimes it is necessary to normalize the data returned.