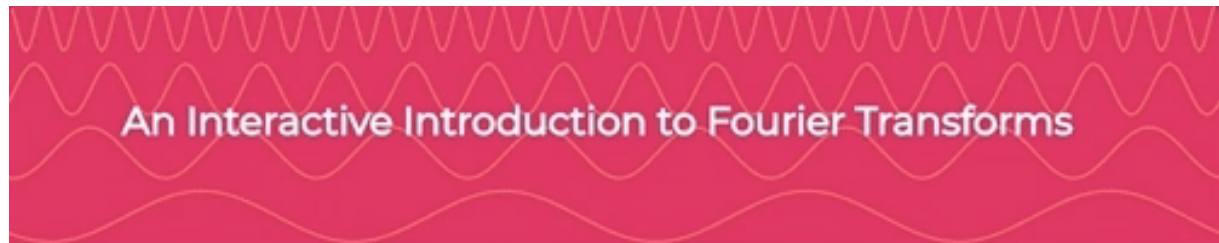
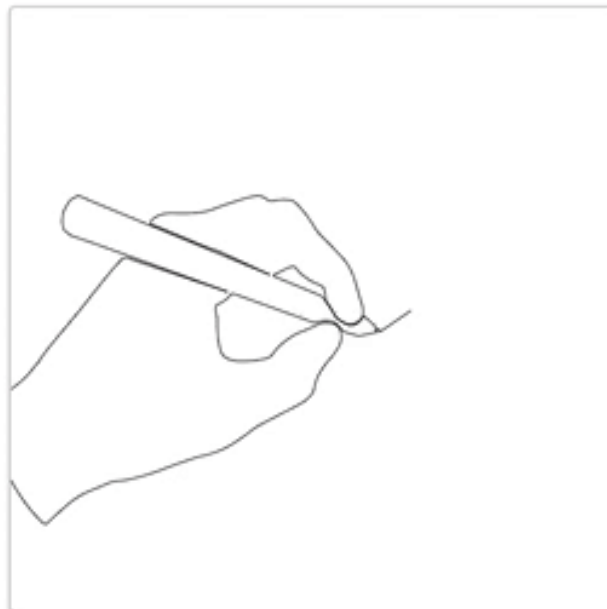

An Interactive Introduction to Fourier Transforms

This is the source code for jezzamon.com/fourier



Fourier transforms are a tool used in a whole bunch of different things. This is a explanation of what a Fourier transform does, and some different ways it can useful. And how you can make pretty things with it, like this thing:



I'm going to explain how that animation works, and along the way explain Fourier transforms!

By the end you should have a good idea about

- [What a Fourier transform does](#)

Understanding this code

This webpage is coded in JavaScript, using Webpack, a tool that merges all the JS files into one and translates it into something compatible with older browsers. It uses npm to install dependencies and run the build scripts.

It uses some features of modern JavaScript like classes that might not be too familiar to you if you've only done intro JavaScript, but don't let that phase you too much! All the interactivity and graphics is just done with raw JS, no graphic libraries or anything like that.

Where the interesting files are

The content of the page is written in Markdown, in `content/content.md`. That file gets converted into HTML when the website is built in `script/make-html.js`. (If you would like to create a translation for a new language, there are instructions in `content/`!)

The code for all the interactive elements on the page is under the `js/` folder. Each interactive element has a 'controller' that handles updating and rendering that element (e.g. `EpicyclesController`). There's something that I called a conductor that handles events and passes them on to each controller. Then they're all created and linked to elements on the page in `main.js`.

The actual Fourier transform is done in `just-fourier-things.js`, which is really a wrapper over the `fft.js` library. Sound synth is done in `synth.js`, using the Web Audio API.

To generate the images used for the JPEG section, I used Python and a Jupyter notebook. That's in `python/dct.ipynb`, and it's fairly readable itself.

Installing stuff

If you're curious about playing with the code yourself, you'll need to install npm. You can then run `npm install` to install all the dependencies. Once that's done, you can run `npm run watch` to start a script that will listen for file changes and rebuild everything, and `npm run reload` to launch a webserver that reloads the page whenever anything changes :) (warning though: I haven't tried to build things on Windows so not sure how some of the scripts will run there)

If you want to mess with the python stuff, you'll need jupyter, as well as NumPy, PIL, SciPy and Matplotlib. I recommend using Anaconda to install all that.