WaveGenSynth

WaveGenSynth is a unique synthesizer that leverages diffusion techniques to create playable sounds. It can seamlessly integrate into your musical workflow as a VST, AU, standalone application, or run directly from the source code. The digital instrument is the result of my bachelor thesis Text-to-Playable-Sound: synthesizers based on latent diffusion technology.

Text-to-Playable-Sound: synthesizers based on latent diffusion technology

In this work, the integration and applicability of generative artificial intelligence in the field of music production was analyzed through the introduction of a digital instrument. Using selected diffusion models, users can define sounds through textual descriptions and play and manipulate them with standardized music production tools. The diffusion models used were evaluated for their suitability in the given context and modified for integration into a digital instrument. Using certain frameworks, the digital instrument was created, resulting in a user interface. This allows users to edit model and instrument-specific parameters. The analysis showed that the models used do not always adequately respond to the context of music production, resulting in unexpected sound patterns or abstract artifacts. Currently, available text-to-audio models do not provide high-quality reproduction of familiar sounds but offer opportunities for experimental applications. The implementation of a prototype of the digital instrument allows for such experiments and the exploration of innovative sound synthesis methods. However, functions are currently missing to reproduce selected areas of the generated sounds or to play them indefinitely. Nevertheless, interesting and unusual soundscapes can already be produced, which could find application in musical compositions.

Table of Contents

Installation

!!! Installer and Server-Application files are currently unsigned for mac. To open right click and click on open. Double click won't work

Binary Installations (VST/AU/Standalone)

Download the latest stable release from the releases page, and follow the instructions for your specific platform.

Both Installer and Server are available for Intel and Apple Silicon Macs.

After installation, run the server and connect to it through the plugin. Note that the server executable may exhibit unstable performance on Intel Macs. If you face issues with the server executable or prefer running it as a script, download `server_light.zip` and follow the instructions in step 4 below.

Building from Source

If you prefer to build from source, follow these steps:

1. **Clone the Repository**:
   - Clone the repository along with its submodules using the command: `git clone --recurse-submodules <REPOSITORY_URL>`
   - If cloned normally, initialize and update the submodules with:

     ```
     git submodule init
     git submodule update
     ```
2. **Load the Project**: Open the project using the provided CMake file.
3. **Configure Run/Debug Settings**: Create a Run/Debug Configuration by selecting one of the available executables to run the project.
4. **Set Up the Server**:
   - Update the environment: `conda env update --file environment.yml` (located in the FastAPI folder)
   - Activate the environment: `conda activate WaveGenSynthAPI_env`
   - Start the server: `uvicorn server:app --reload`

Mac On Mac, Xcode is required for C++ support.

Windows Not yet tested.

Usage

Operating WaveGenSynth is straightforward: 1. Launch the Server and plugin. 2. Initialize a model. This step may take some time during the first run, causing the plugin to freeze momentarily. Monitor the server console until the download and setup are complete. If a timeout occurs in the plugin, hit refresh after the setup is complete. (Select 'cuda' if running on hardware with a Nvidia GPU, 'mps' if running on Apple Silicon, and 'cpu' as a fallback) 3. Enter a prompt in the designated field. 4. Click the "Generate" button. 5. Select the appropriate

MIDI input device. If using as a standalone click option and then select your MIDI device 6. Enjoy creating and playing sounds!

Modelevaluation

The effect of different torch-devices, parameters and prompts on the available pre-trained models are displayed here

Credits

WaveGenSynth was crafted using the following technologies: - **GUI Framework**: JUCE - **GUI Module**: foleys_gui_magic - **Diffusion Model**: AudioLDM, integrated via the pipeline and AudioLDM2, integrated via the pipeline from HuggingFace

A special thank you to the developers of these tools that made WaveGenSynth possible.