ai 보이스 체인저 조사 및 적용

프로젝트 링크 : https://github.com/w-okada/voice-changer

소개

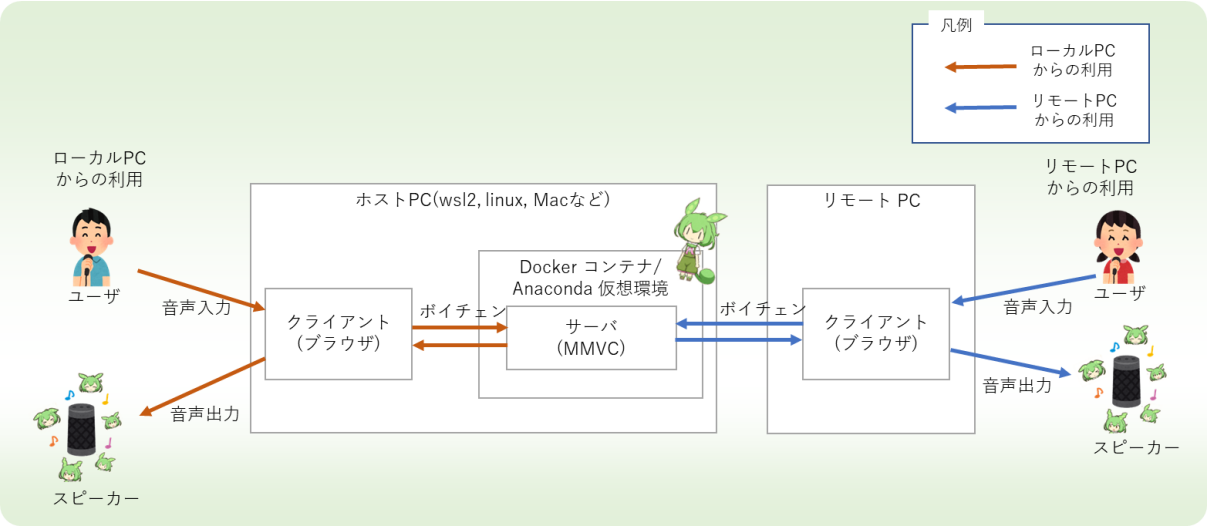
이번에 진행해 볼 프로젝트는 w-okada의 Realtime-voice-changer(RVC) 프로젝트를 소개 및 적용하고 실제로 어느 정도까지 활용 가능한지 테스팅해보는 프로젝트이다.

o-wakada의 RVC 프로젝트는 Hugging face라는 오픈소스 커뮤니티 겸 회사에서 진행하는 오픈소스 프로젝트이다.

Hugging face는 주로 기계학습(ML)의 학습 모델을 구축하고 배포하는 일을 주로 하며 사용자가 직접 ML을 위한 교육과 리소스를 제작하는 것을 진행하기도 한다.

RVC는 말 그대로 사람의 목소리를 학습한 ML model을 PC의 로컬환경 혹은 외부 서버를 세팅해 외부에서 접속 및 처리하여 CPU, GPU 등을 이용해서 내 목소리의 높이, 발음, 억양 등을 Model과 비교하여 Model에 나온 목소리가 실제로 어떻게 말할지를 추론, 그 결과를 추론하는 방식으로 작동한다.

세부적인 작동 방식은 아래 그림과 같다.



최근에는 성능이 어느 정도 보장되는 CPU로도 GPU 이상으로 처리가 가능하도록 수정, 배포되었다는 소식을 듣고 호기심과 흥미를 갖고 진행하게 되었다.

보이스 체인저의 적용

Model을 처리하기 위해서는 보통 CPU 혹은 GPU를 사용하고 처리를 진행하는 방식으로 GPU 버전 혹은 CPU 버전으로 나뉜다.

https://huggingface.co/wok000/vcclient000/tree/main

깃허브 프로젝트 내의 위 링크로 들어가면 각각 v1.5.3의 Mac, DirectML, Cuda 버전이 있다.

Mac은 말 그대로 Macbook 등에 들어가는 M1 실리콘 칩을 말하는 것이다.

DirectML은 x86, x64 CPU를 말하는 것이고, 흔히 우리가 사용하는 Intel, AMD의 CPU를 말하는 것이다.

Cuda 는 GPU를 ML의 병렬처리에 사용하도록 하는 API 모델이며, 흔히 말하는 NVIDIA의 GPU에 적용한다.
(AMD사에서 개발하는 GPU에는 아직 적용하는 버전이 없다.)

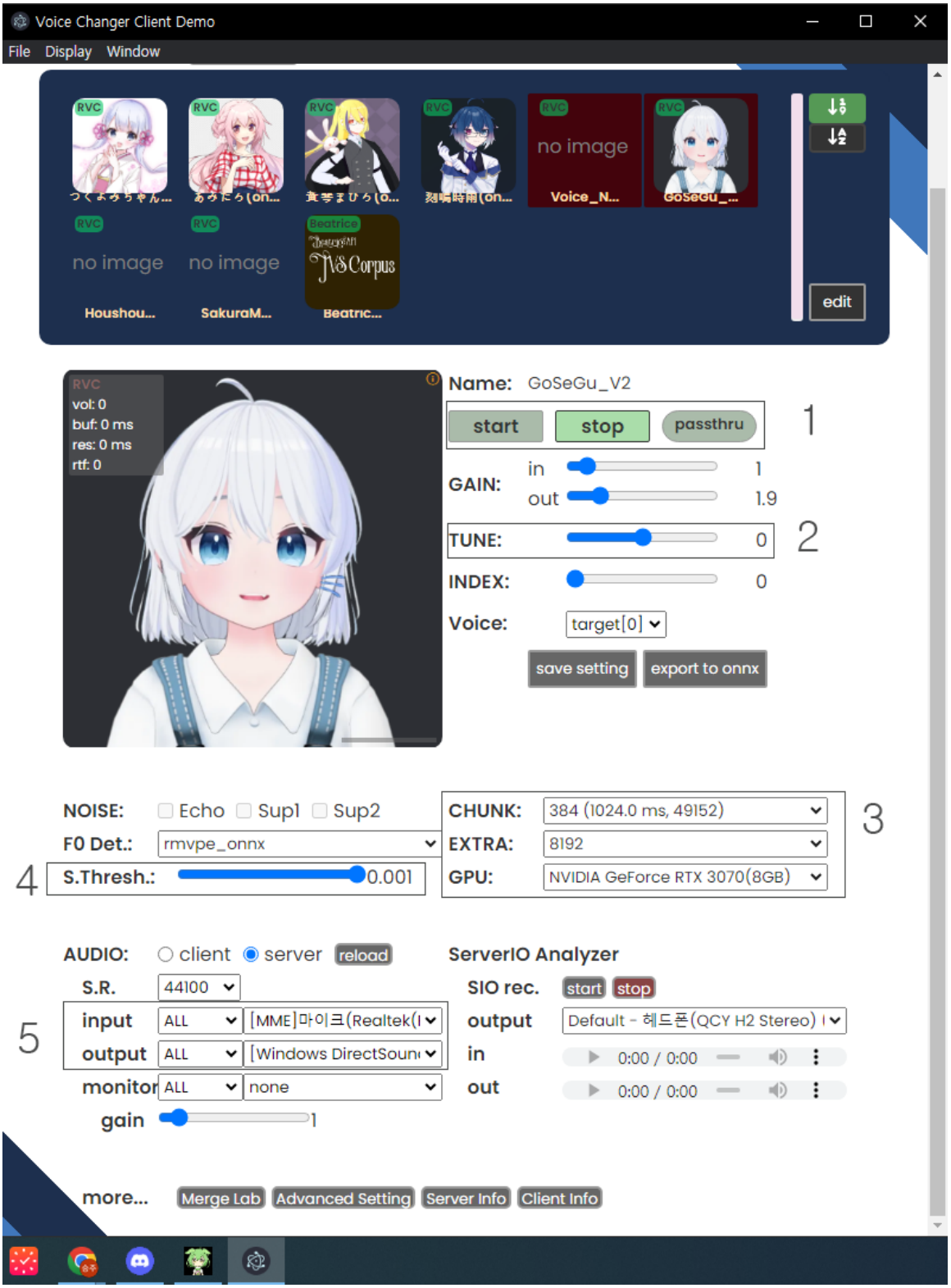
GPU 송출 버전 (v1.5.3)

RVC 모델에서 가장 흔하게 사용하는 버전이며 흔히 실시간 음성 변조를 사용할 때 사용하는 클라이언트다.

hugging face의 파일을 다운받았으면 압축해제 후 MMVCServerSIO.exe 혹은 start_http·bat(이쪽을 더 권장한다)을 실행한다.

실행하면 로컬 RVC 서버를 실행하기 위한 기초 모듈을 다운받고 예제 모델들을 설치 받는다.

이후 프로그램을 종료하고 재실행하면 아래 이미지와 같은 클라이언트를 확인할 수 있다.



위의 이미지에서 직접적으로 만지는 섹션을 나누면 숫자가 적힌 5개로 나눌 수 있다.

1. 변조 시작, 종료, 마이크 입력 그대로 출력

말 그대로 시작, 종료를 말한다. pass thru는 pass through를 줄인 말로 현재 마이크에 입력되는 소리를 그대로 출력하도록 하는 것을 말한다.

2. TUNE

입력한 목소리에 비해서 얼마나 높게 소리를 올릴 것인지 설정하는 섹션이다.

3. 목소리 처리 단위 정리

한 번에 처리할 양을 정하는 chunk, 처리한 데이터를 임시 저장할 extra, 처리에 사용할 GPU 선택 탭이 각각 있다.

4. Sound Thresh

마이크에 들어가는 목소리 외의 잡음을 얼마나 잡을지를 의미한다.

5. audio

목소리의 입력, 출력을 어느 장치로 할지 설정할 수 있다.

각 섹션에서 핵심적인 부분 및 자세한 설명은 아래와 같다.

- Tune은 올리면 올릴수록 출력되는 목소리가 입력된 목소리에 비해 높아진다. 파라미터를 조정하면 Model이 인식한 높이에 비해 수치를 조정한 더 높은 목소리 혹은 더 낮은 목소리를 출력한다.
- 입력받은 음성을 처리할 때 GPU의 vram 크기에 따라 한 번에 처리 가능한 양, 처리 속도가 늘어난다.
  여기서 Chunk의 크기를 줄일수록 한 번에 처리하는 양이 줄어들면서 좀 더 실시간에 가깝게 처리가 진행되나, vram의 크기가 받아주지 않는다면 처리를 다 하지 못하고 끊어져서 출력을 진행하게 된다.

쉽게 요약하자면, GPU 처리하기 위해서는 VRam이 더 높은 GPU를 사용하면 좀 더 실시간에 가깝게 처리가 가능하므로 성능이 상승한다.

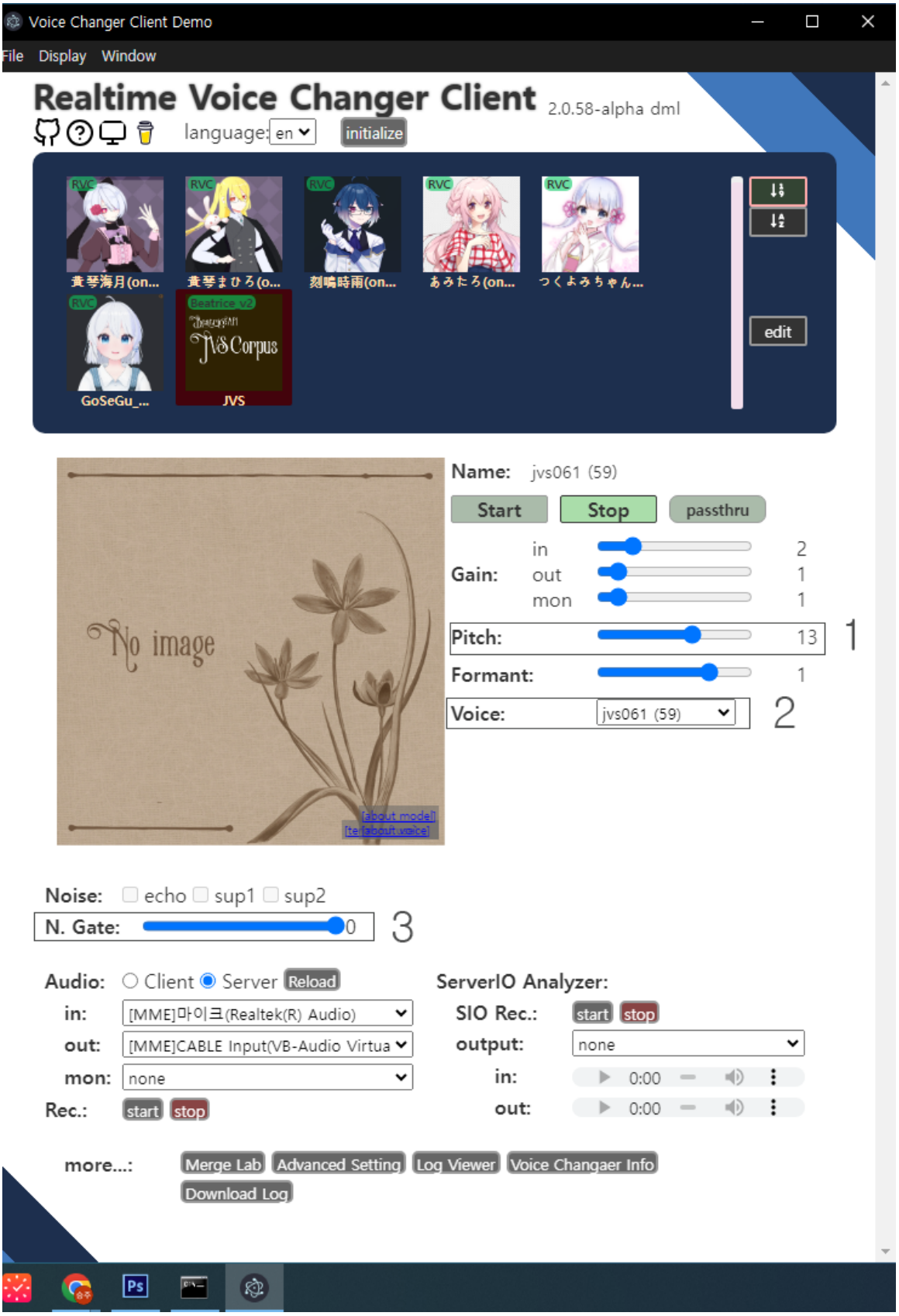
현재 내 사양은 이미지에 나온 데로 GTX 3070(8gb)이라 대략 1초 정도의 딜레이를 두고 있기 때문에

CPU 송출 버전 (v2, Beatrice)

GPU 송출과 유사하게 동일한 링크에서 v2를 다운받고 이를 압축 해제한다.

GPU 송출과는 다르게 start_http·bat만이 정상적으로 작동하므로 이를 실행한다.

기초 모듈 및 예제가 모두 설치되고 종료한 뒤 다시 실행하면 아래와 같은 화면이 나올 것이다.



GPU 버전과의 다른 점만을 설명하면 아래와 같다

- Tune에서 pitch로 단어가 알기 쉽게 변경되었다.
- 기존의 처리 - 변경 형식이 아닌 플러그인의 응용으로 Voice를 선택하는 란이 생겼으며 실제로 음원 편집 프로그램의 주파수처럼 Model의 형식을 변경할 수 있다.
- 이미지에서는 생략되었으나 GPU의 chunk, extra 등과 다르게 실제 처리량과 딜레이 속도만을 표시하도록 변형되었다.
- sound thread에서 Negative Gate로 잡음 삭제를 음원 프로그램과 유사하게 바꾸었다.

CPU 송출은 구버전이 아닌 최신 v2 버전에서 혁신적인 변화가 있었다.

Beatrice라는 기능을 도입한 것인데, vst 라는 음원 편집 프로그램들의 플러그인을 이용하는 원리로 기존 GPU 처리에 수많은 리소스가 들어가는 데 비해 CPU의 thread 1개 정도로 실시간에 가깝게 처리가 가능하도록 만들어진 것이다.

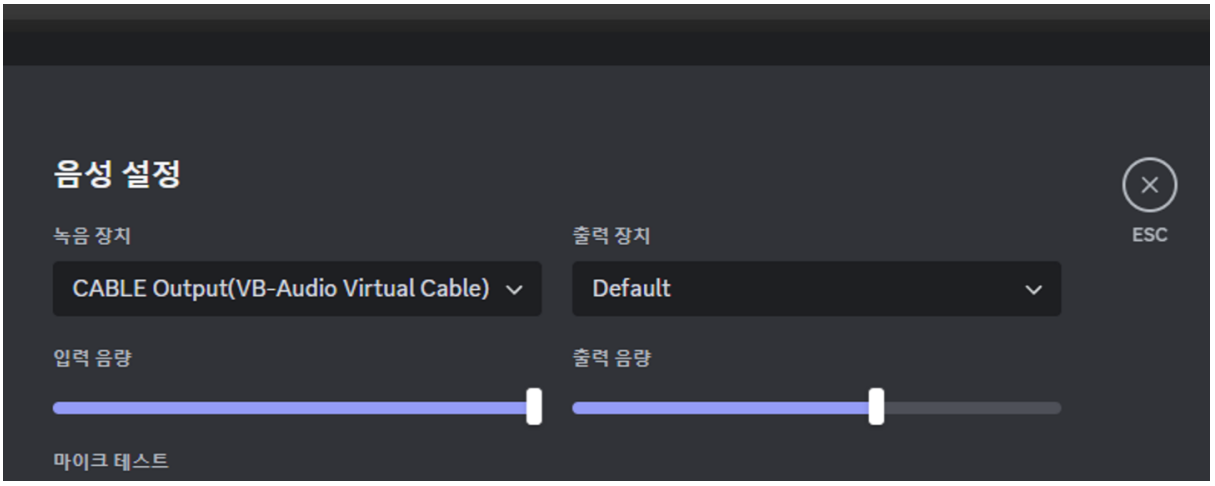
물론 기능 자체가 생긴 지 1주일 정도 지난 시점이라 관련 모델이 아주 부족한 상황이지만 이후 학습한 모델이 나온다면 아마 저사양으로 딜레이가 없다시피 RVC가 가능하다는 점에서 기존의 모델과 크게 차이를 둘 것이다.

외부 프로그램에서 사용하는 방법

변형한 목소리는 기본 세팅으로는 내 기기에서만 실행이 되고 외부 프로그램으로는 송출이 어렵다.

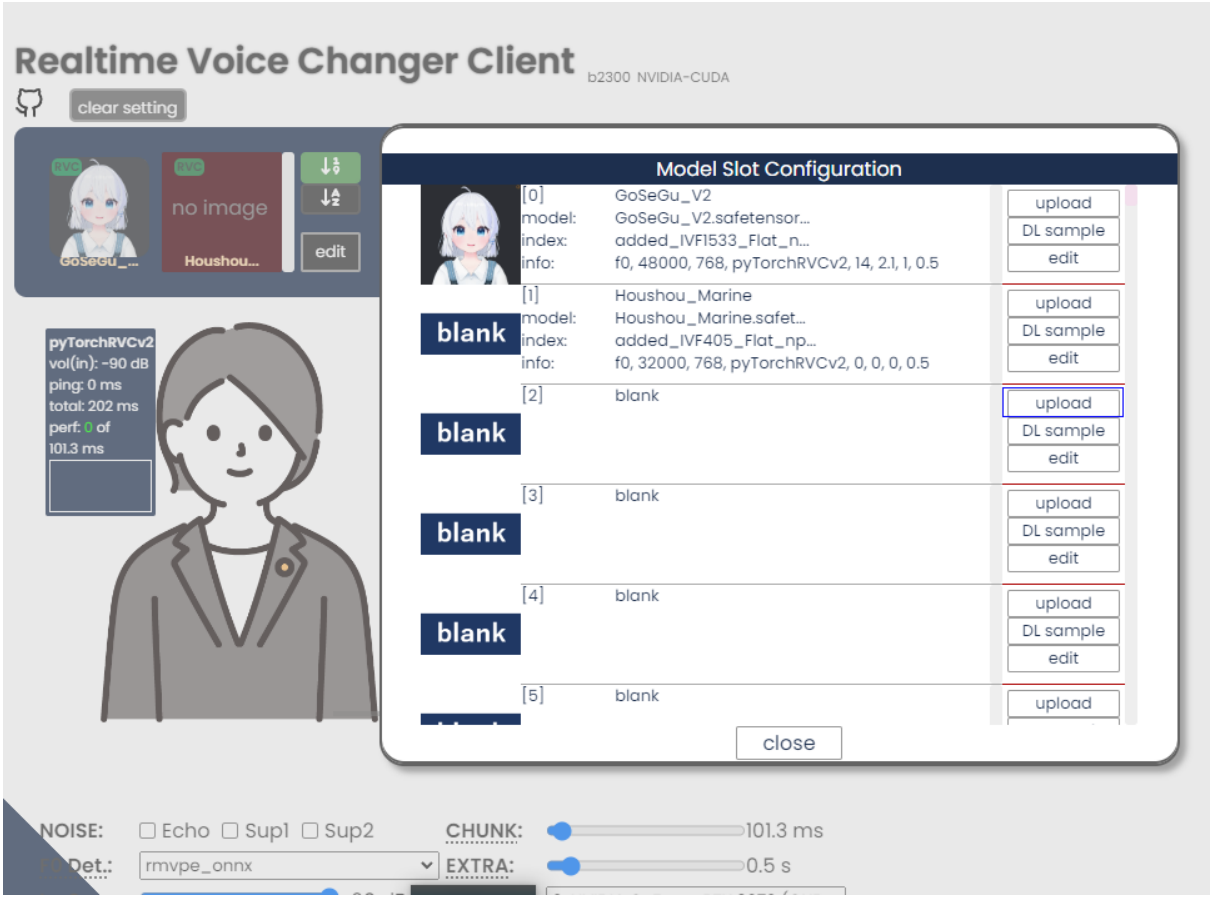
이를 해결하기 위해서는 시스템 내부에서 가상 input과 output을 설정해서 출력할 수 있도록 해야 한다.

vb-audio를 검색하면 나오는 여러 프로그램이 있다. 이를 설치하면 virtual input, output이 사용할 수 있는 오디오에 추가되는데, 이를 RVC client의 output에 vb input을 넣고, 외부 프로그램의 오디오 입력에 vb output을 적용하면 내가 출력하는 소리가 가상 오디오에 입력 - 가상 오디오를 원하는 프로그램에 출력하는 방식으로 진행이 가능하다.

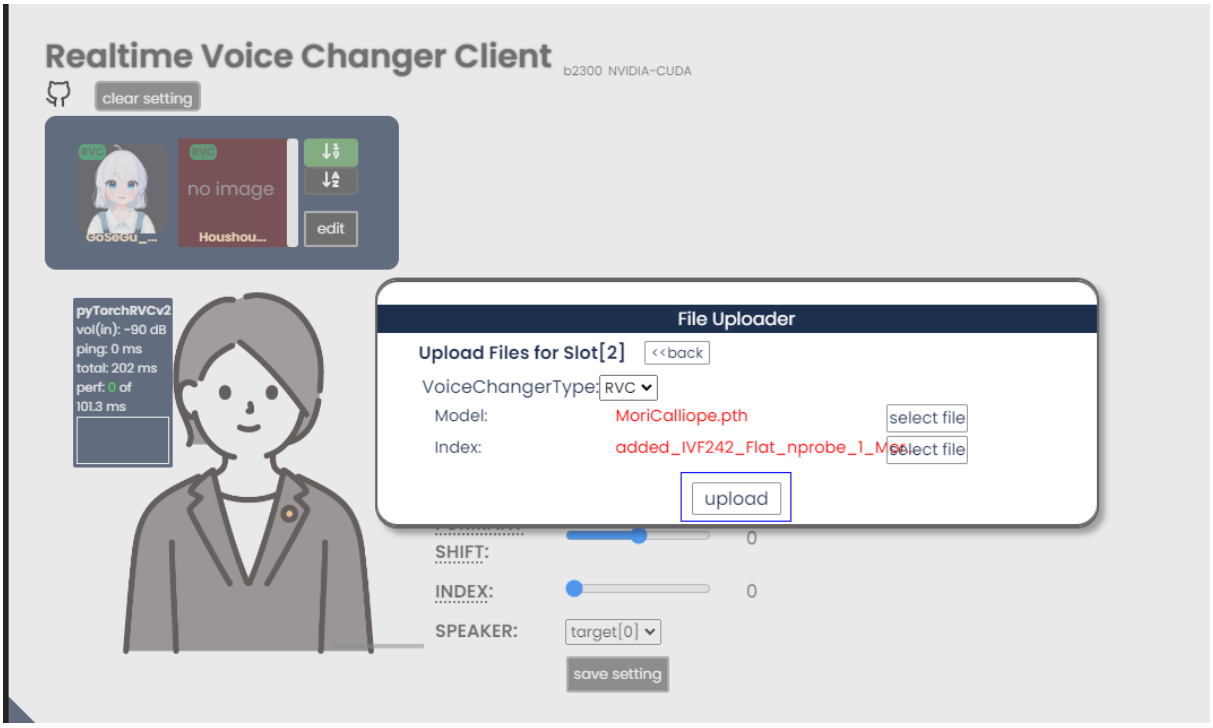


새로운 Model을 추가하는 방법

직접 Model을 제작했거나 외부에서 제작한 모델을 다운로드 받아서 예시 모델이 아닌 다른 모델을 사용하고 싶은 경우 아래와 같이 edit 탭에서 빈 공간 혹은 기존 공간에 upload를 진행한다.



.pth 파일과 별도의 index가 있는 경우 아래 그림과 같이 model에는 .pth 파일을 선택하고 별도로 index가 있는 경우에는 하단 index에 .index 파일을 선택한 뒤 업로드하면 된다.



그 외

Voice Model을 직접 학습시켜서 만든다거나, 유튜브 등에서 볼 수 있는 AI Cover 등 Hugging face에서는 여러가지 프로젝트들을 진행하고 있다.

본문에서는 beatrice를 혁신적이라고 표현하고 있지만, Model에 입력할 데이터의 전처리, 사전학습에 사용할 model, 데이터의 양 등이 아직 정확하게 연구된 자료로 나온 것이 없어서 실제로는 오픈소스로 진행된 프로젝트 중 여러 개발자가 모여서 만든 기존 gpu를 사용하는 버전을 크게 개선시킨 버전을 주로 사용하고 있다.
(https://github.com/deiteris/voice-changer/releases)

또한 이를 개인적인 게임 음성 시스템이나 디스코드 등 통화를 하는 사용자를 위주로 설명하였으나 OBS 등 스트리밍 프로그램이나 REAPER 등의 음원 편집 프로그램에서도 직접 활용이 가능하며, 클라이언트에서 직접 변조한 목소리를 파일로 저장하는 등 여러가지 가능성이 있는 프로젝트이므로 많은 관심 바란다.

GPU 버전과의 다른 점만을 설명하면 아래와 같다

- Tune에서 pitch로 단어가 알기 쉽게 변경되었다.

- 기존의 처리 - 변경 형식이 아닌 플러그인의 응용으로 Voice를 선택하는 란이 생겼으며 실제로 음원 편집 프로그램의 주파수처럼 Model의 형식을 변경할 수 있다.
- 이미지에서는 생략되었으나 GPU의 chunk, extra 등과 다르게 실제 처리량과 딜레이 속도만을 표시하도록 변형되었다.
- sound thread에서 Negative Gate로 잡음 삭제를 음원 프로그램과 유사하게 바꾸었다.