

Audio source separation using snakes, crabs and torches

GStreamer Conference 2025

23 October 2025

Sebastian Dröge <sebastian@centricular.com>

Audio source separation

- Split an already mixed audio stream into its individual logical sources
- Obviously a probabilistic approach
- Use cases: karaoke, music practice, improving text-to-speech, ...

demucs

- NN model for audio source separation
 - Developed by Facebook research
- Reference implementation (MIT) in Python using PyTorch
- Various pre-trained models on music available
 - Different sources (e.g. vocals, drums, bass, other)
- Good performance & fast

GStreamer element

- Written in Rust (of course)
- Can use demucs in 2 ways
 - WebSocket-based Python service
 - In-process usage of Python via PyO3
 - Both use custom streaming Python class around demucs
- Configurable chunk sizes and per-chunk overlap
 - Quality / latency trade-off
 - Default: 3s + 25% overlap => 9s latency

PyO3

- Rust crate for Python interop
 - **Much** more convenient than Python C API
- Can be used for
 - Calling / embedding Python code from Rust
 - And for writing Python modules in Rust
- Provides control over GIL, or using free threading

PyO3

- Used here to embed a Python class
 - `include_str!()` and `PyModule::from_code(s)`
- Call its methods
 - `pyobj.call1(py, ("foo", 123))`
- Provide a Rust closure as Python callable
 - Custom Python class defined in Rust
 - Provides a `__call__` method to Python

Example karaoke pipeline

```
gst-launch-1.0 uridecodebin uri=file:///path/to/music/file ! audioconvert ! tee name=t ! \  
    queue max-size-time=0 max-size-bytes=0 max-size-buffers=2 ! demucs name=demucs model-name=htdemucs \  
    demucs.src_vocals ! queue ! audioamplify amplification=-1 ! mixer.sink_0 \  
    t. ! queue max-size-time=9000000000 max-size-bytes=0 max-size-buffers=0 ! mixer.sink_1 \  
    audiomixer name=mixer ! audioconvert ! autoaudiosink
```

Questions? Comments?

Code available at

https://gitlab.freedesktop.org/gstreamer/gst-plugins-rs/-/merge_requests/2554