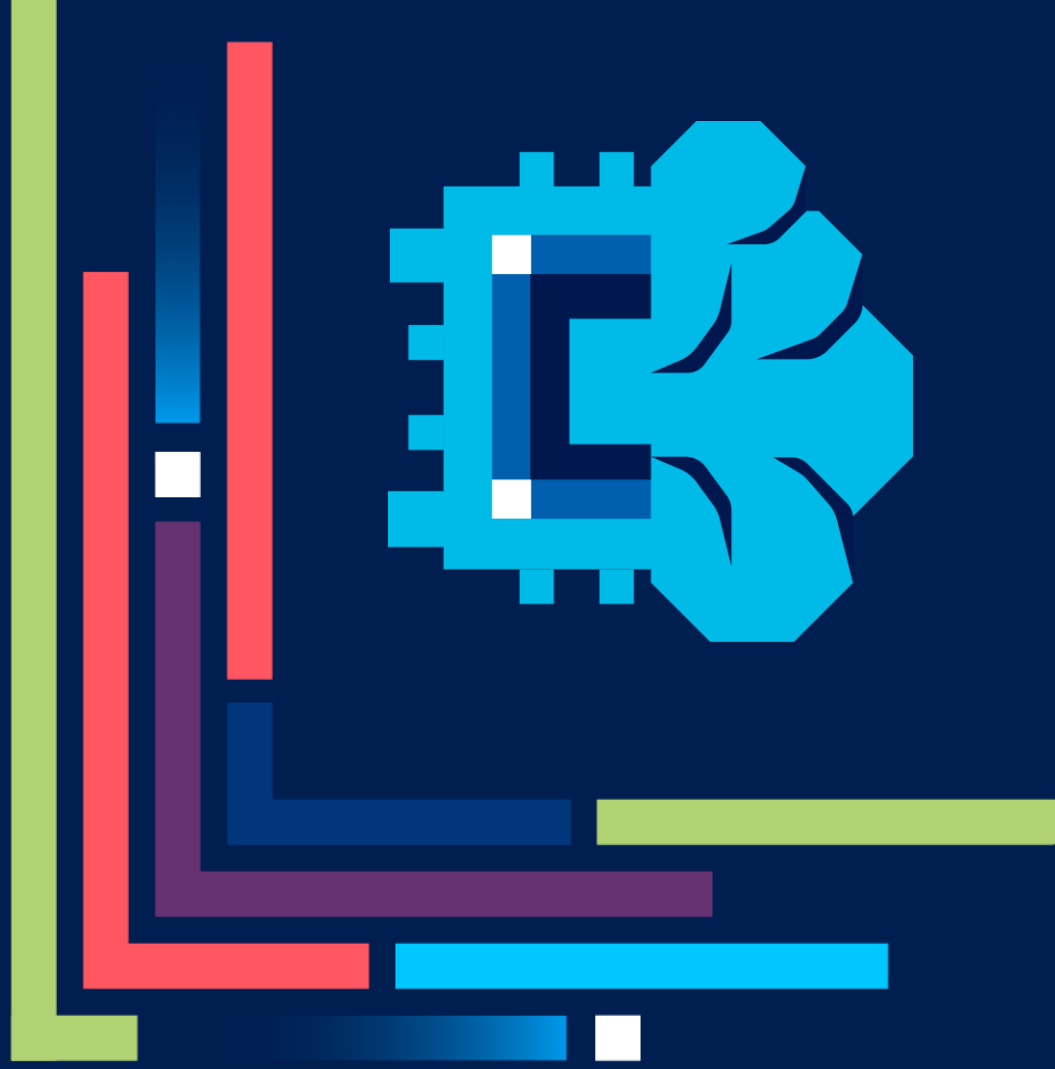
intel ai summit
英特爾 AI 科技論壇

Bringing AI Everywhere



Generative AI Development and Application on Client

Nancy Cheng
Intel Sales Application Engineer
March 27th, 2024

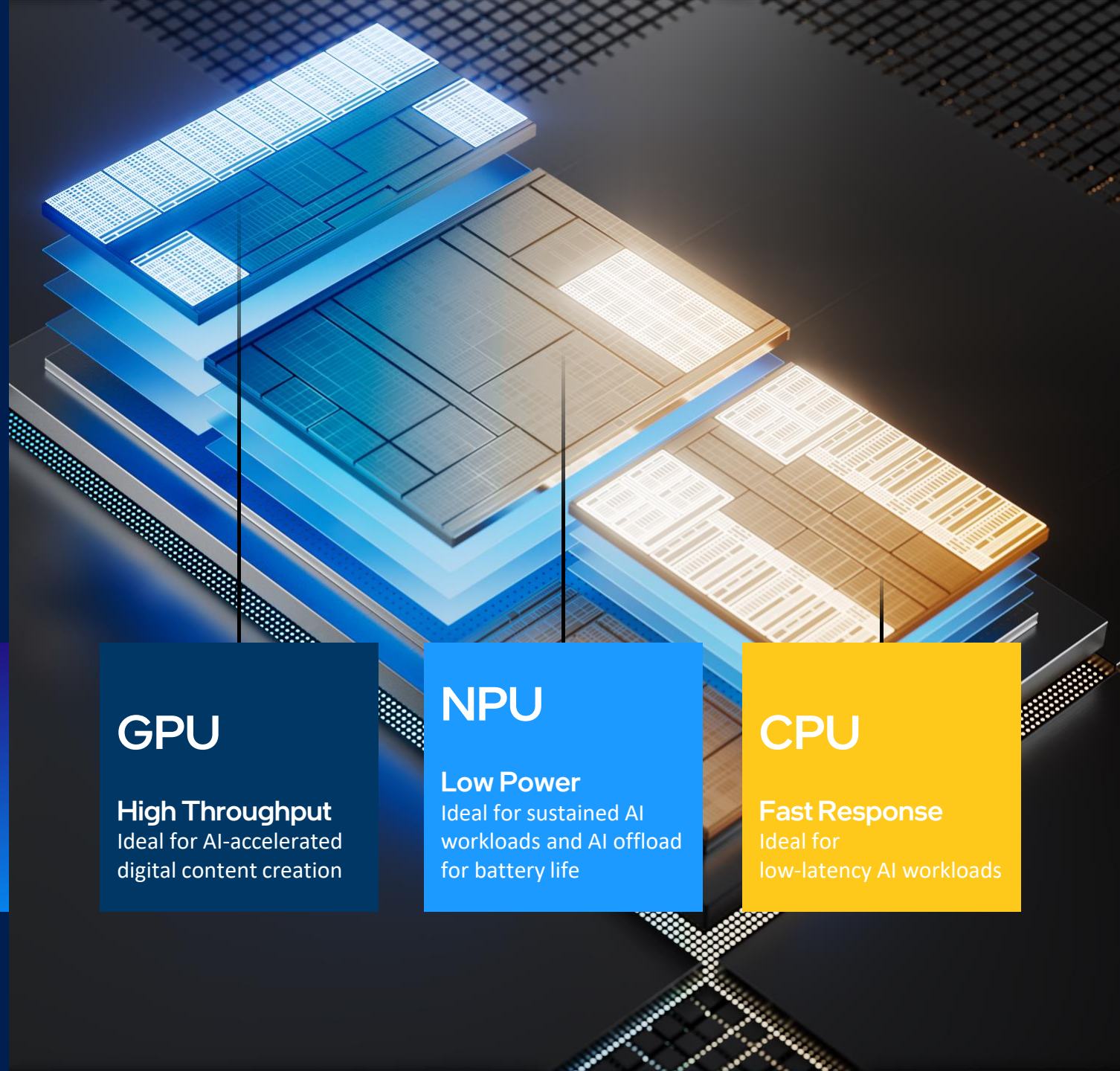
Agenda

- AI PC & What can OpenVINO™ do?
- Riffusion
- Converting/Enabling the models
- Integration into Audacity® + Demo
- Summary

intel CORE ULTRA

Three AI Engines

Heterogenous execution of AI workloads embraces the best practices in AI software design.



intel AI summit

PyTorch TensorFlow Keras TensorFlow Lite ONNX Caffe PaddlePaddle

1 Model

OpenVINO™

2 Optimization

Optimized Performance

CPU

GPU

NPU

FPGA

3 Deployment

Windows Linux macOS

Software Optimizations

Additional Configuration	Autobatching	Default Inference Precision
Threads Scheduling	Shared Memory	Model Caching
Pre- Post-Processing	Asynchronous Mode	Model Compression

OpenVINO™ Notebooks

100+

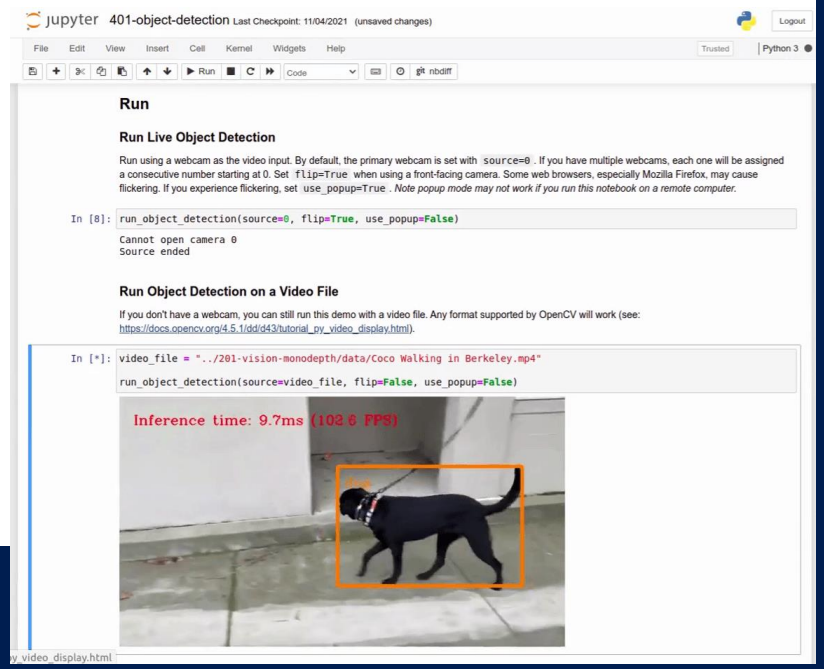
Demos

LLMs, Stable Diffusion, Whisper, GPT, YOLOv5/v8, CLIPS, Object Detection and Segmentation, Image Classification, Human Pose Estimation, and much more!

intel Ai summit

OpenVINO™ Notebooks Installation

```
# Step 1: Clone the Repository
$ git clone
https://github.com/openvinotoolkit/openvino_notebooks.git
$ cd openvino_notebooks

# Step 2: Create the Environment
$ python3 -m venv openvino_env
$ source openvino_env/bin/activate

# Step 3: Install Requirements
$ python -m pip install --upgrade pip
$ pip install -r requirements.txt

# Step 4: Launch the Notebooks
$ jupyter lab
```



OpenVINO™

intel Ai summit

Installation

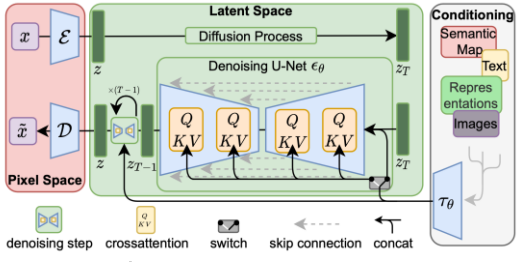
https://docs.openvino.ai/2024/get-started/install-openvino.html

OpenVINO™
NOTEBOOKS

Stable Diffusion

intel Ai summit

Stable Diffusion : Text to Image



Input

Text Source

Prompt: a concept art sketch of an astronaut in white futuristic cybernetic armor in a dark cave,

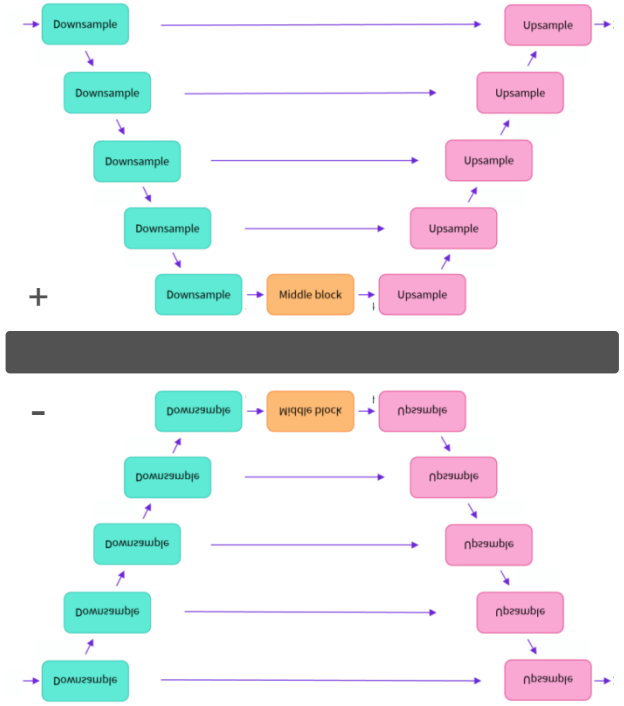
Negative Prompt: Low Resolution, bad quality, monochrome, cropped head, deformed face, bad anatomy

Text Conversion

Text Encoder & Tokenizer

Take the positive and negative prompt and create a tokenized embedding in "Latent Space"

Iterative Noise Reduction



Pixel Conversion

Variation Auto Encoder (VAE)

Turn latent space into pixel space to produce an image

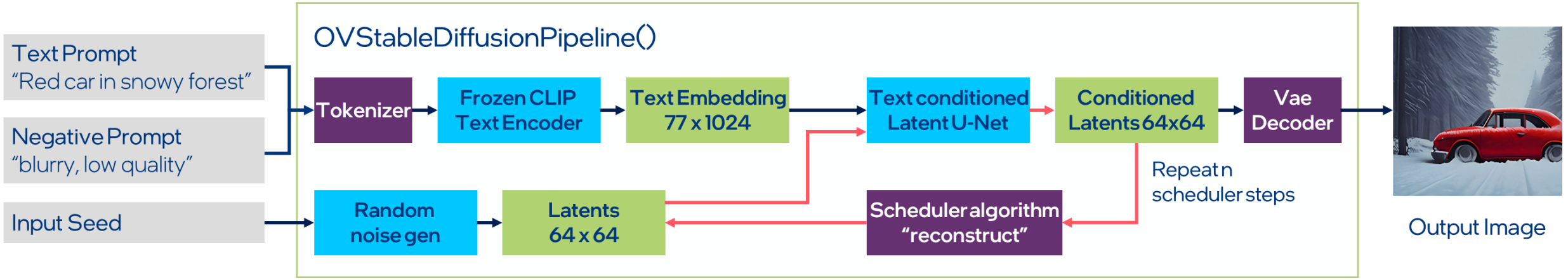
Output

OpenVINO™ Stable Diffusion

High performance inference

SD Pipeline

Inference Pipeline

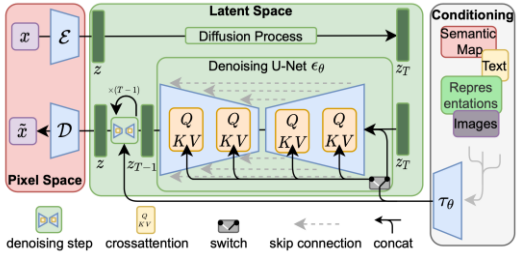


OpenVINO™ NOTEBOOKS

Riffusion
Text-to-music

intel Ai summit

Riffusion : Text to Image (of audio!)



Input

Text Source

Prompt: bubblegum eurodance

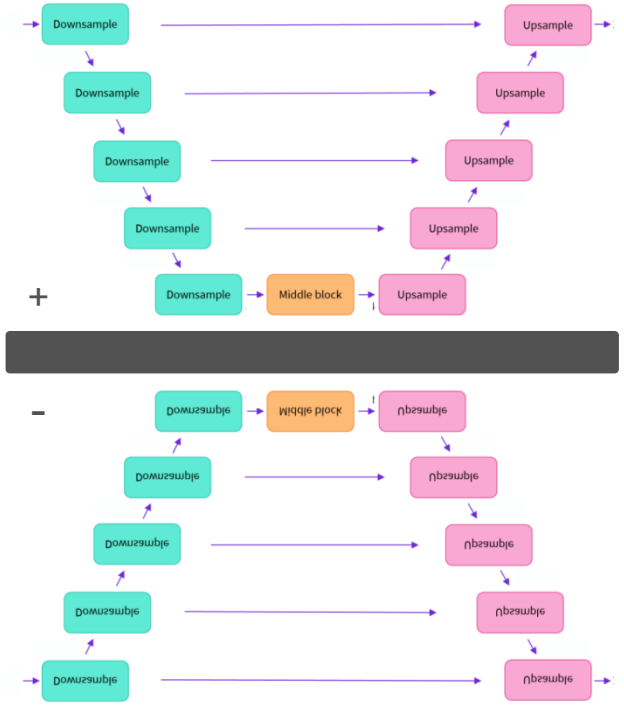
Negative Prompt: bad quality

Text Conversion

Text Encoder & Tokenizer

Take the positive and negative prompt and create a tokenized embedding in "Latent Space"

Iterative Noise Reduction

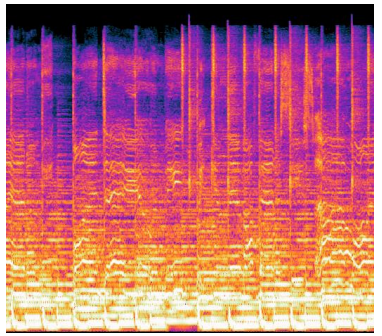


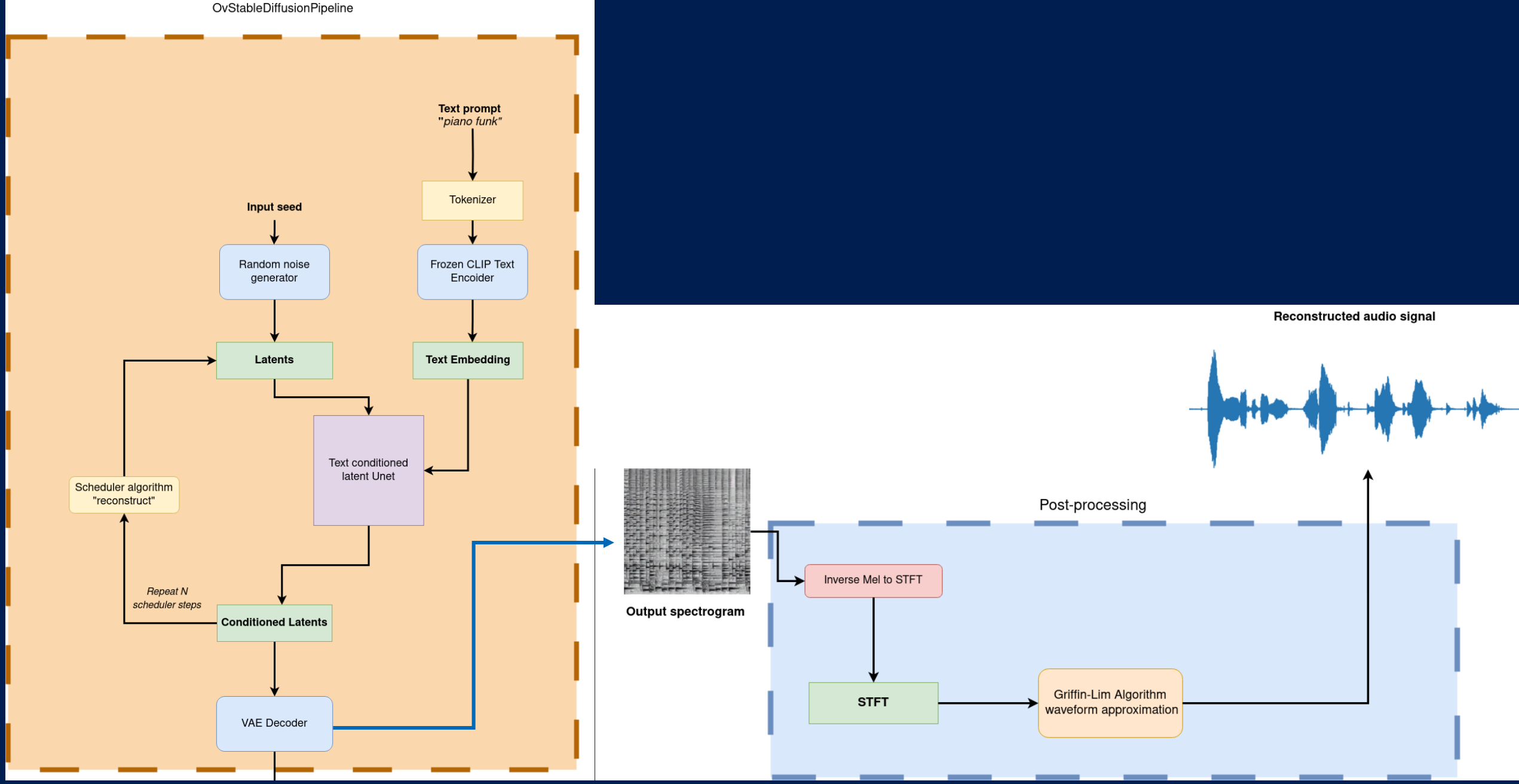
Conversion

Variation Auto Encoder (VAE)

Turn latent space into pixel space to produce a spectrogram

Output




OvStableDiffusionPipeline
Text prompt
"piano funk"
Tokenizer
Input seed
Random noise generator
Frozen CLIP Text Encoider
Latents
Text Embedding
Text conditioned latent Unet
Scheduler algorithm "reconstruct"
Repeat N scheduler steps
Conditioned Latents
VAE Decoder
Output spectrogram
Post-processing
Inverse Mel to STFT
STFT
Griffin-Lim Algorithm waveform approximation
Reconstructed audio signal

Audacity®

Audacity®

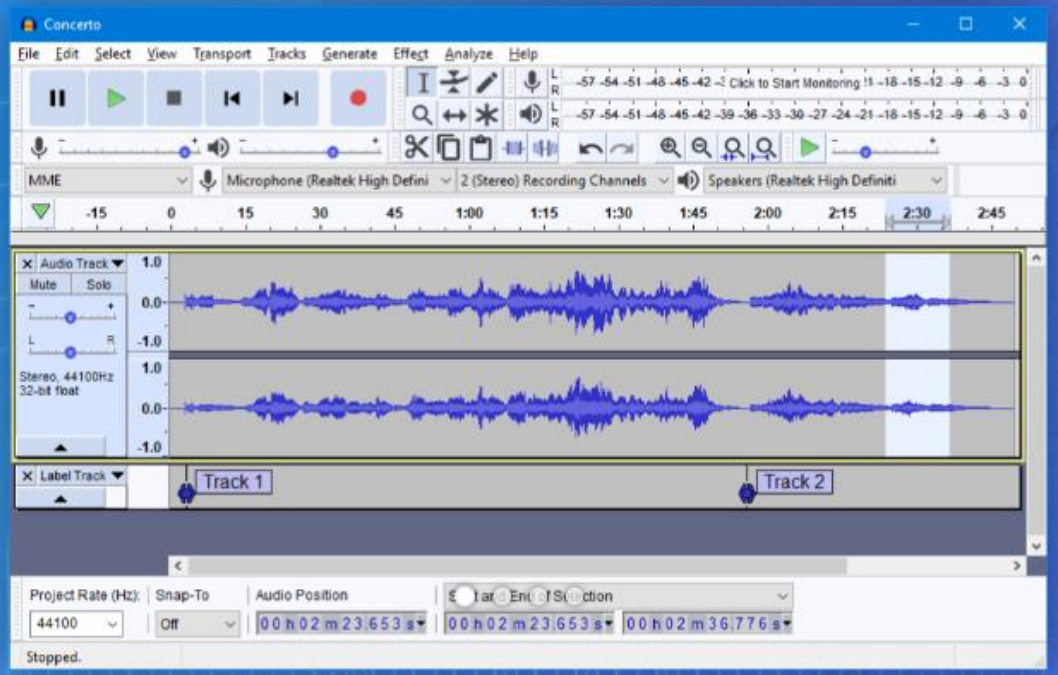
Search ...

HOME ABOUT DOWNLOAD HELP CONTACT GET INVOLVED COPYRIGHT

Free, open source, cross-platform audio software

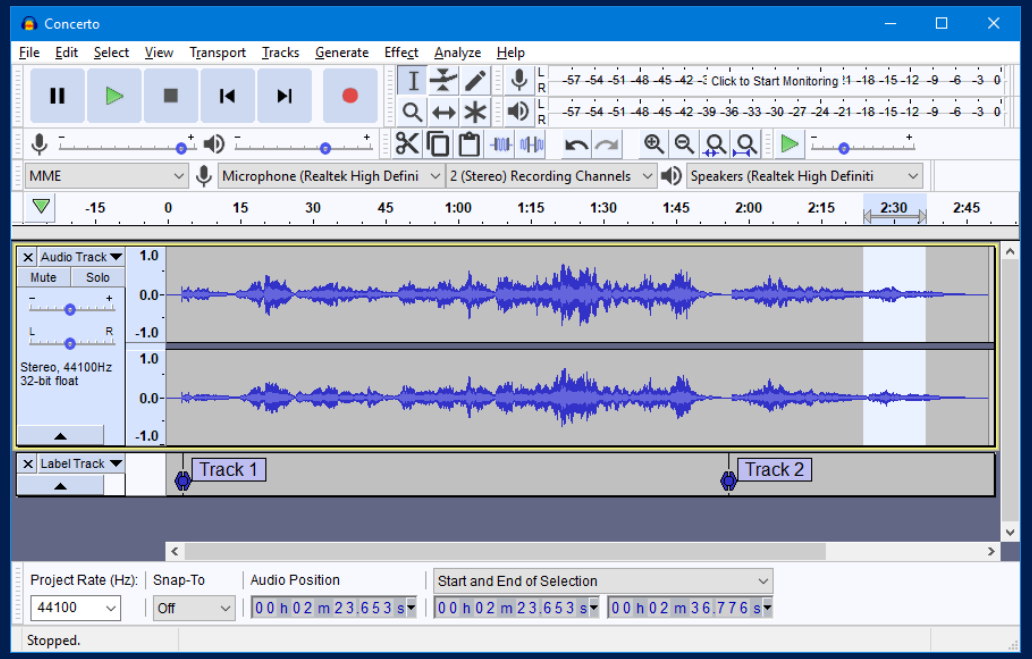
Audacity is an easy-to-use, multi-track audio editor and recorder for Windows, macOS, GNU/Linux and other operating systems.

Audacity is free, open source software.



intel Ai summit

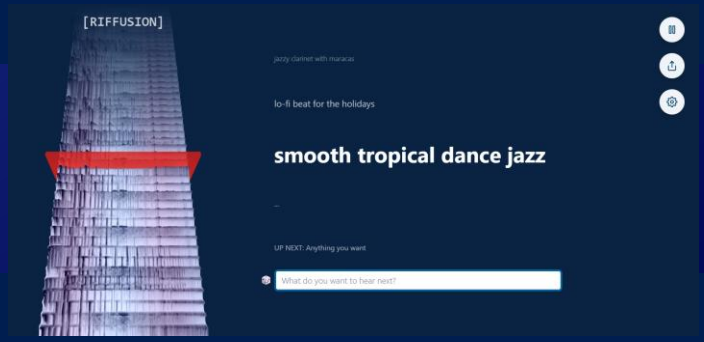
Riffusion in Audacity®



intel Ai summit

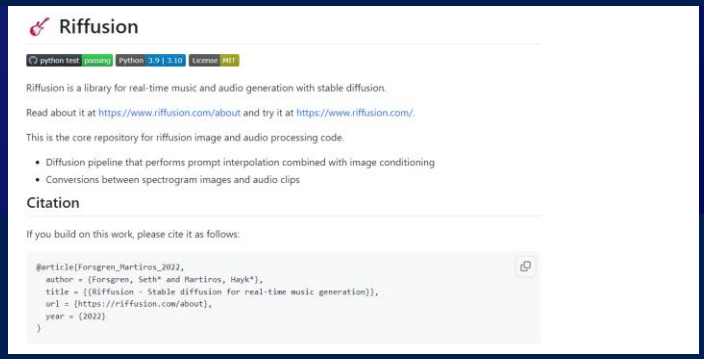
Acknowledgment to Riffusion Project

https://www.riffusion.com



https://www.riffusion.com/about

https://www.github.com/riffusion/riffusion



intel AI summit

Integrating into Audacity® – Python to C++

Python APIs clearly map to C++ APIs, making the transition straightforward

Python APIs from Notebooks

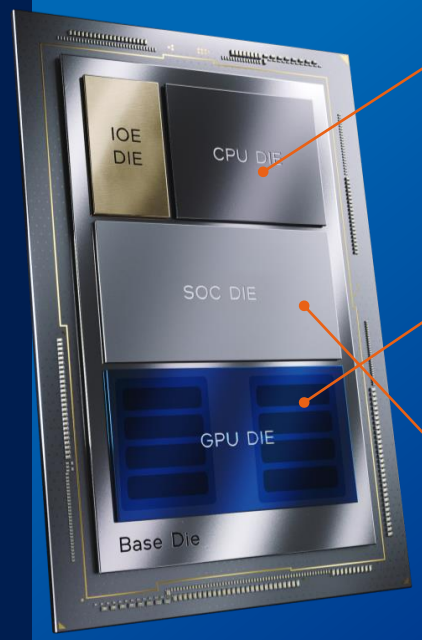
```
from openvino.runtime import Core, Model

self.core = Core()

# Device to target such as CPU, GPU and now NPU
device = “NPU”

# Setup Caching to improve model load in future
self.core.set_property({'CACHE_DIR':
os.path.join(cache_dir, 'cache’)})

# Produce a compiled-model object for a given device
self.compiled_model =
self.core.compile_model(model_path, device)

# Create the inference request
self.infer_request =
self.compiled_model.create_infer_request()
```

C++ APIs for integration into Application

```
#include <openvino/openvino.hpp>

ov::Core core;

// Device to target such as CPU, GPU and now NPU
std::string device = “NPU”

// Setup Caching to improve model load in future
core.set_property(ov::cache_dir(“cache”));

// Produce a compiled-model object for a given device
ov::CompiledModel compiled_model =
core.compile_model(model_path, device);

// Create the inference request
ov::InferRequest infer_request =
compiled_model.create_infer_request();
```

Integrating into Audacity® – Using Different Device Types

OpenVINO™ supports multiple devices easily



Intel® Core™ Ultra

CPU

Fast Response, DL Boost (VNNI)

GPU

16 Vector Engines for AI acceleration

NPU

New Dedicated AI Engine

C++ APIs for integration into Application

```
#include <openvino/openvino.hpp>

ov::Core core;

//Find all supported devices on this system
std::vector<std::string> devices =
core.get_available_devices();

// Device to target such as CPU, GPU and now NPU
std::string device = “NPU”

// Setup Caching to improve model load in future
core.set_property(ov::cache_dir(“cache”));

// Produce a compiled-model object for a given device
ov::CompiledModel compiled_model =
core.compile_model(model_path, device);
```

intel Ai summit

Integrating into Audacity® – Using Different Model Types

OpenVINO™ supports many formats directly, avoiding the need for conversion

C++ APIs for integration into Application

```
#include <openvino/openvino.hpp>

ov::Core core;

// Device to target such as CPU, GPU and now NPU
std::string device = “NPU”

// We can choose between a bunch of different model formats
//std::string model_path = "path/to/text_encoder.xml";
//std::string model_path = "path/to/text_encoder.tflite";
std::string model_path = "path/to/text_encoder.onnx";

// Produce a compiled-model object for a given device
ov::CompiledModel compiled_model = core.compile_model(model_path, device);

// Create the inference request
ov::InferRequest infer_request = compiled_model.create_infer_request();
```

ONNX

Caffe

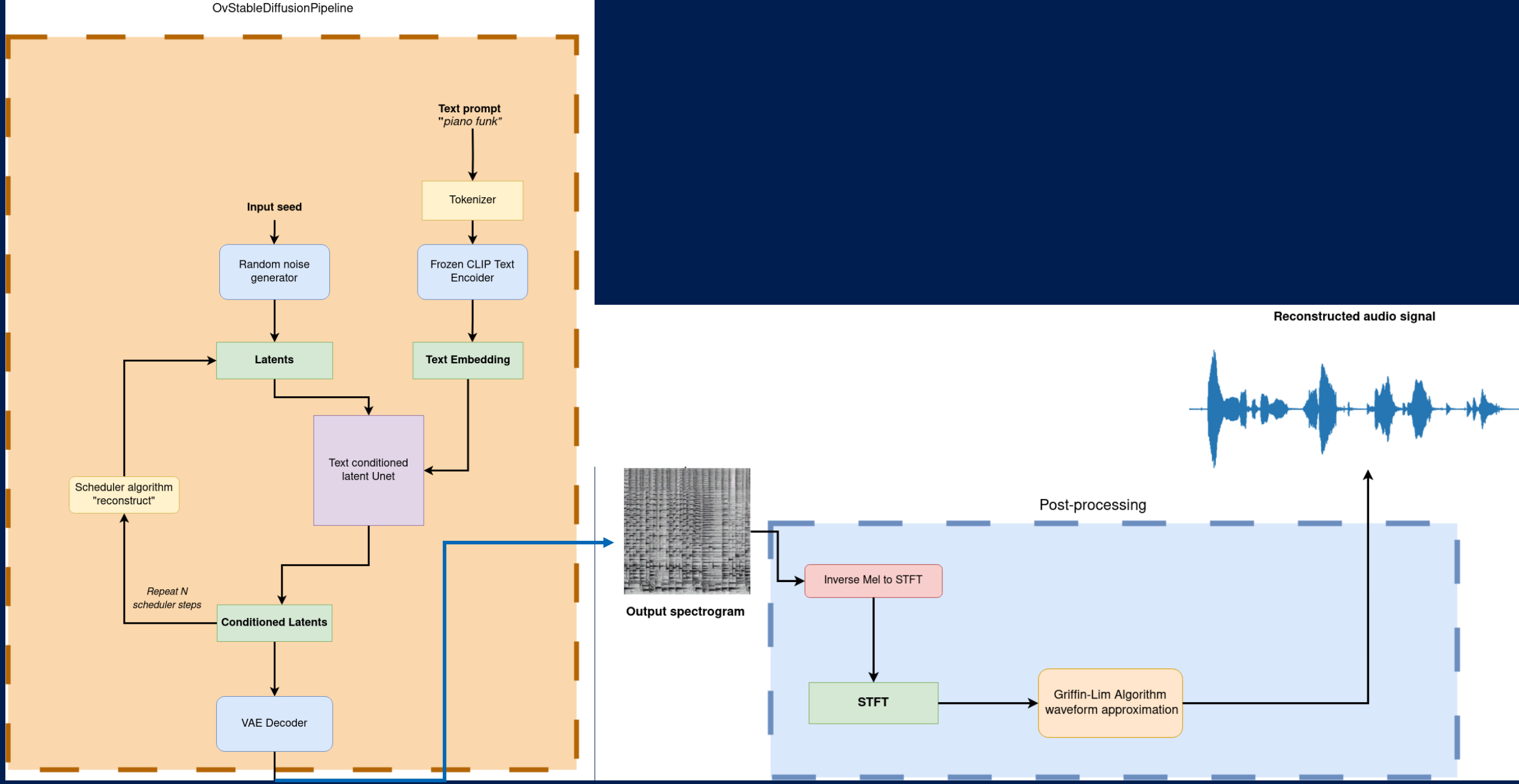
PaddlePaddle

TensorFlow Lite

TensorFlow

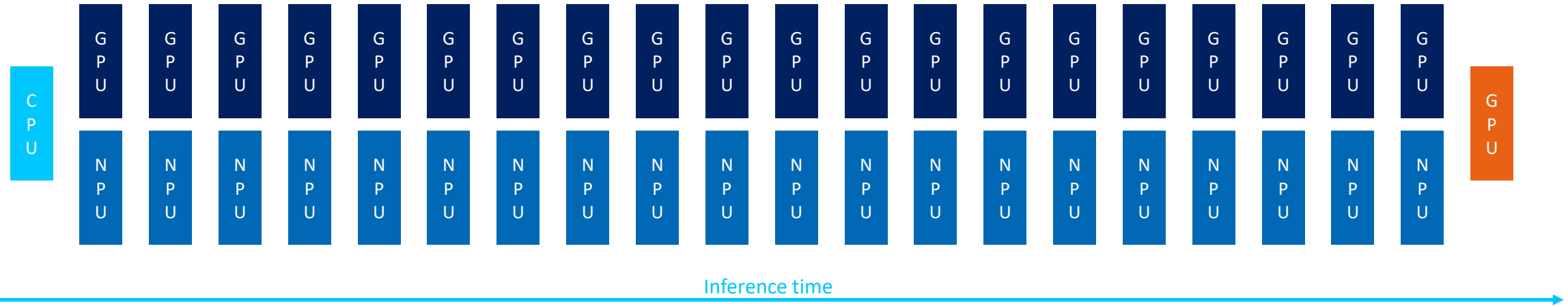
PyTorch

intel Ai summit


OvStableDiffusionPipeline
Text prompt
"piano funk"
Tokenizer
Input seed
Random noise generator
Frozen CLIP Text Encoider
Latents
Text Embedding
Text conditioned latent Unet
Scheduler algorithm "reconstruct"
Repeat N scheduler steps
Conditioned Latents
VAE Decoder
Output spectrogram
Post-processing
Inverse Mel to STFT
STFT
Griffin-Lim Algorithm waveform approximation
Reconstructed audio signal

Audacity® – Riffusion Model Device Assignments

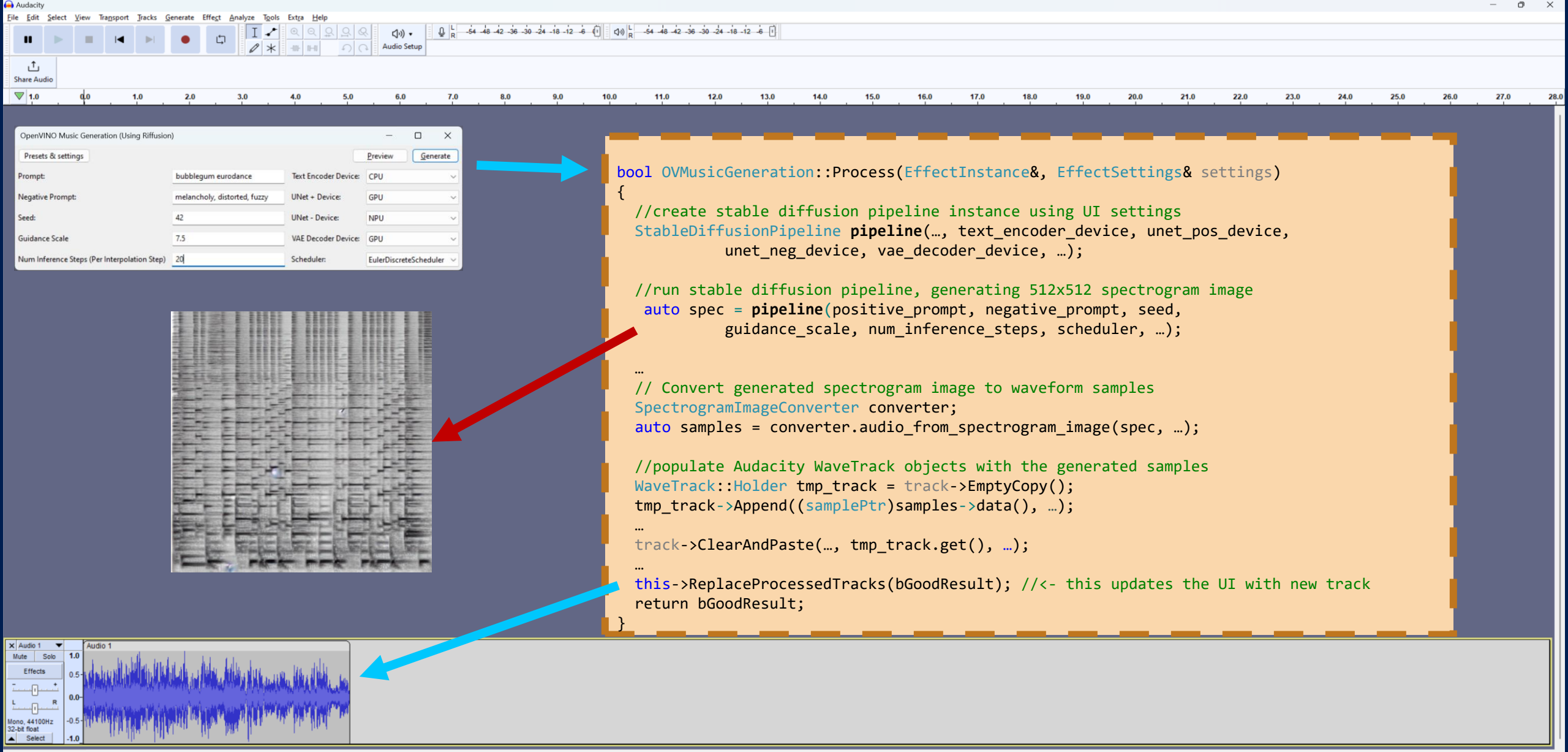
Utilizing NPU in parallel with GPU maximizes throughput



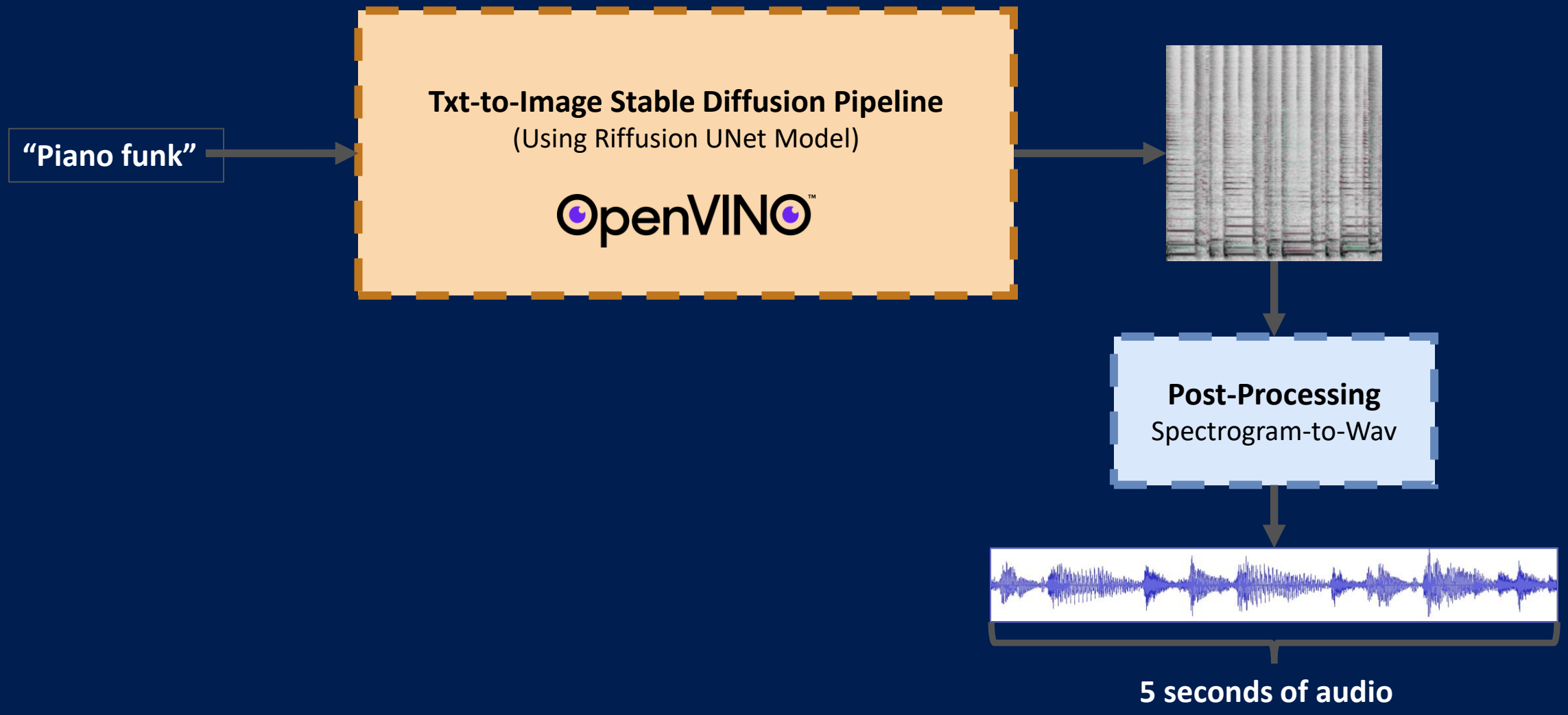

```
// Text Encoder on Core CPU
std::string device = “CPU”
std::string model_path = “path\to\text_encoder.onnx”
ov::CompiledModel text_enc_model = core.compile_model(model_path, device);


// VAE decoder on integrated GPU
device = “GPU”
model_path = “path\to\vae_decoder.onnx”
ov::CompiledModel vae_enc_model = core.compile_model(model_path, device);
```

```
// Unet Positive Prompt on integrated GPU
device = “GPU”
model_path = “path\to\unet.xml”
ov::CompiledModel unet_pos_model = core.compile_model(model_path, device);


// Unet Negative Prompt on new integrated Neural Processor
device = “NPU”
model_path = “path\to\unet.xml”
ov::CompiledModel unet_neg_model = core.compile_model(model_path, device);
```

intel Ai summit

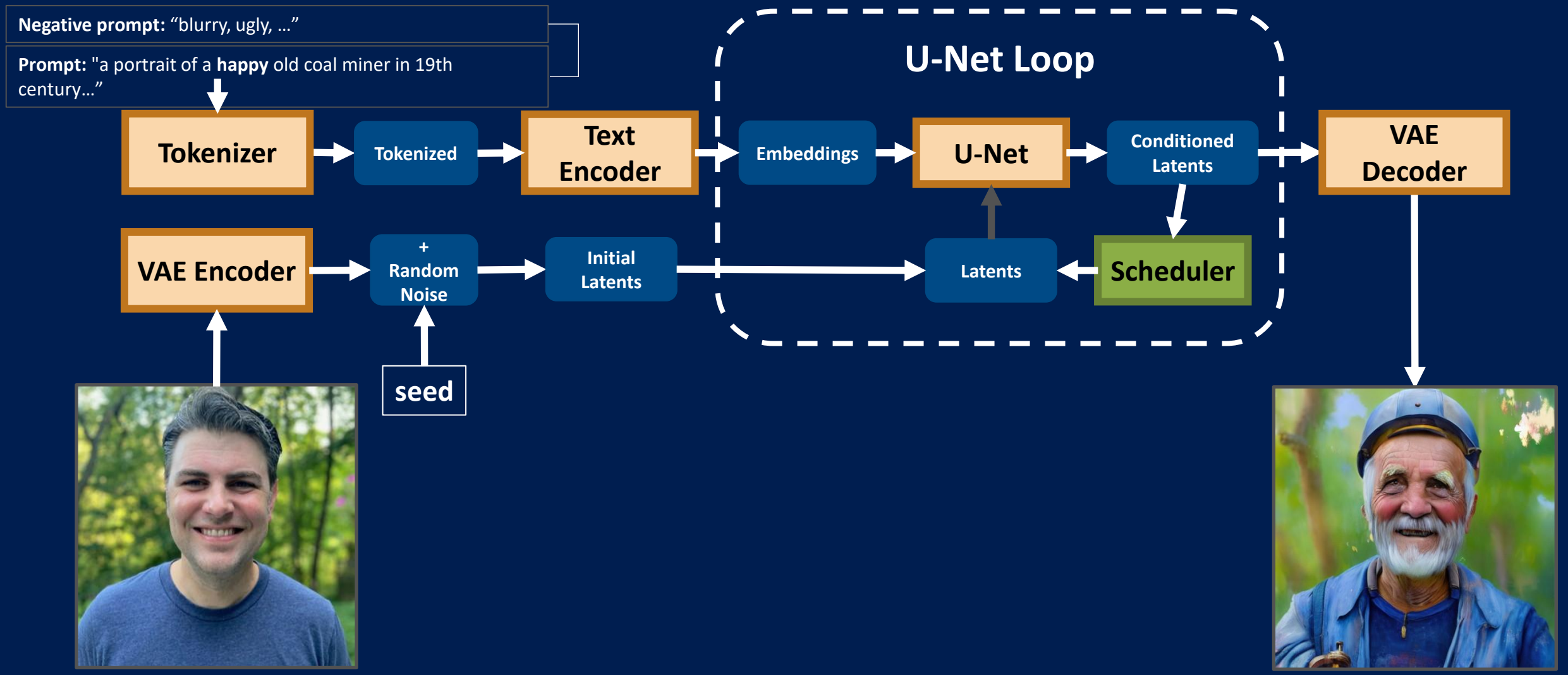



```
bool OVMusicGeneration::Process(EffectInstance&, EffectSettings& settings)
{
  //create stable diffusion pipeline instance using UI settings
  StableDiffusionPipeline pipeline(…, text_encoder_device, unet_pos_device,
            unet_neg_device, vae_decoder_device, …);

  //run stable diffusion pipeline, generating 512x512 spectrogram image
   auto spec = pipeline(positive_prompt, negative_prompt, seed,
            guidance_scale, num_inference_steps, scheduler, …);


  …
  // Convert generated spectrogram image to waveform samples
  SpectrogramImageConverter converter;
  auto samples = converter.audio_from_spectrogram_image(spec, …);

  //populate Audacity WaveTrack objects with the generated samples
  WaveTrack::Holder tmp_track = track->EmptyCopy();
  tmp_track->Append((samplePtr)samples->data(), …);
  …
  track->ClearAndPaste(…, tmp_track.get(), …);
  …
  this->ReplaceProcessedTracks(bGoodResult); //<- this updates the UI with new track
  return bGoodResult;
}
```

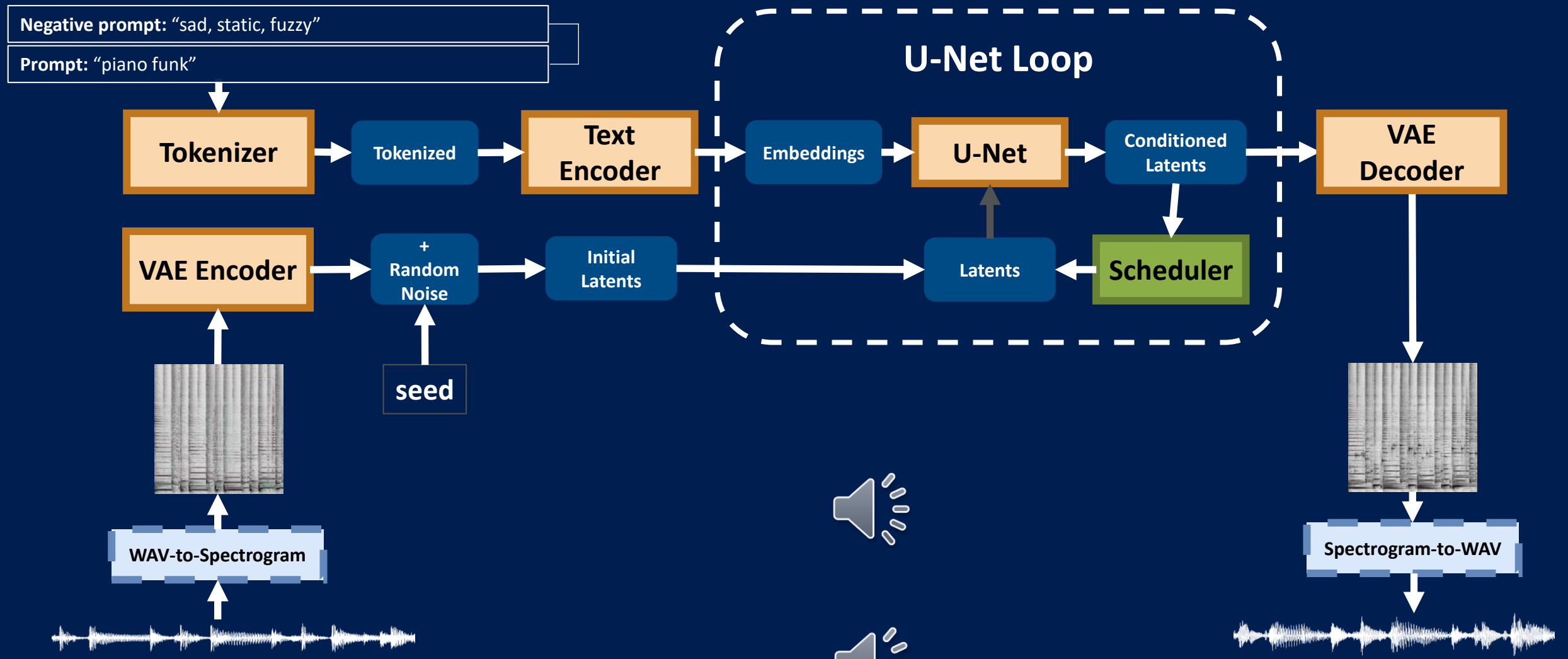


How can we generate audio segments longer than 5 seconds?

Stable Diffusion Image-to-Image Pipeline

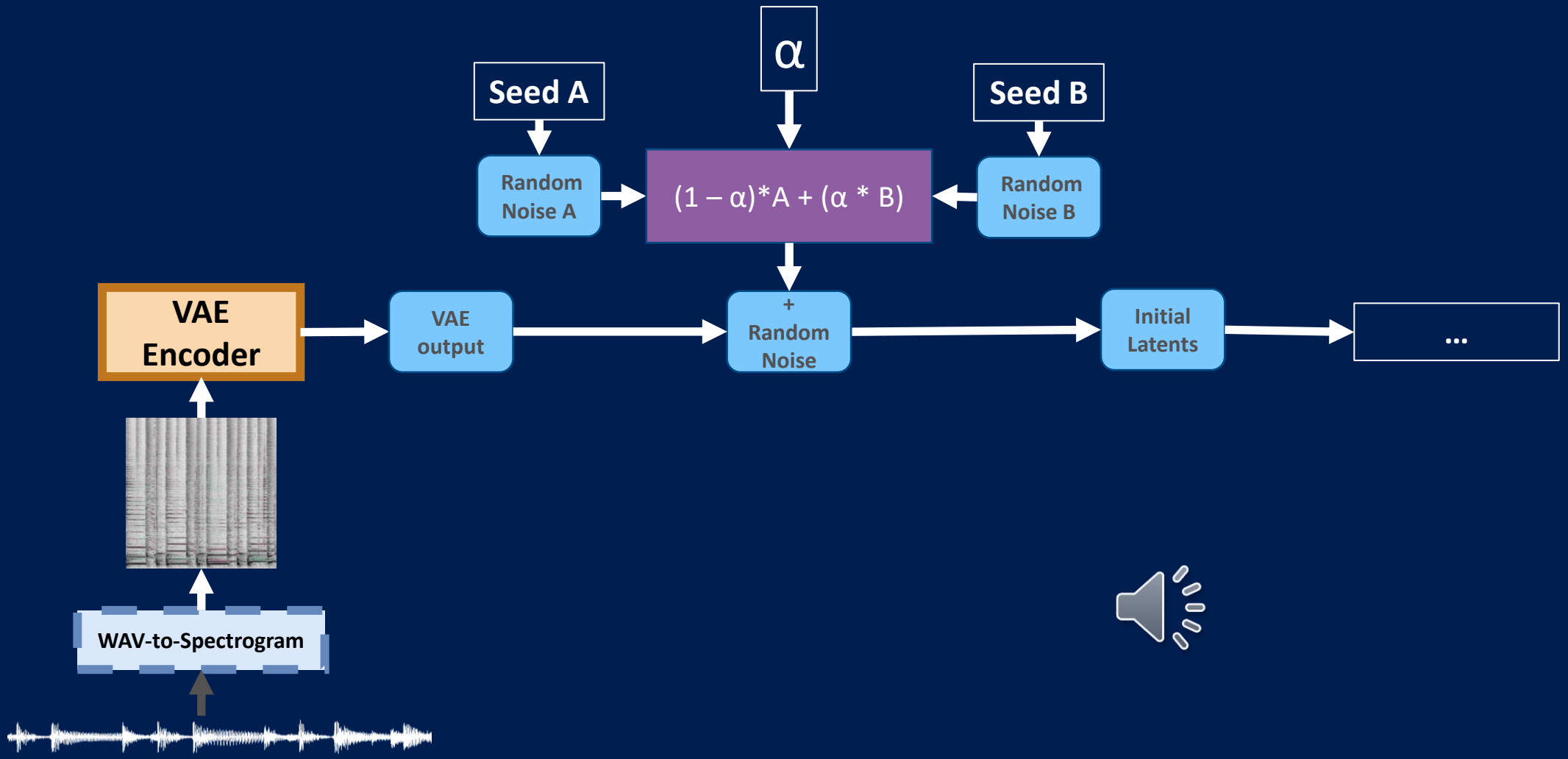


Stable Diffusion Image-to-Image Pipeline

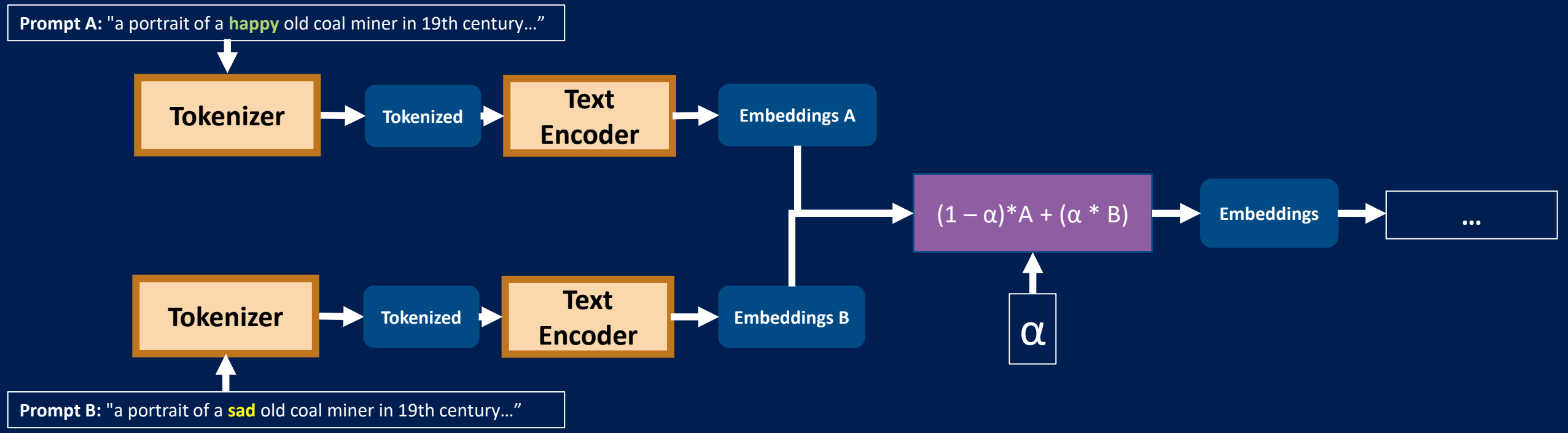


intel AI summit

Creating Smooth Transitions – Seed Interpolation



Creating Smooth Transitions – Prompt Interpolation



Creating Smooth Transitions (In Images!)

Image-to-Image Stable Diffusion 'Latent Space Interpolation' Pipeline

?

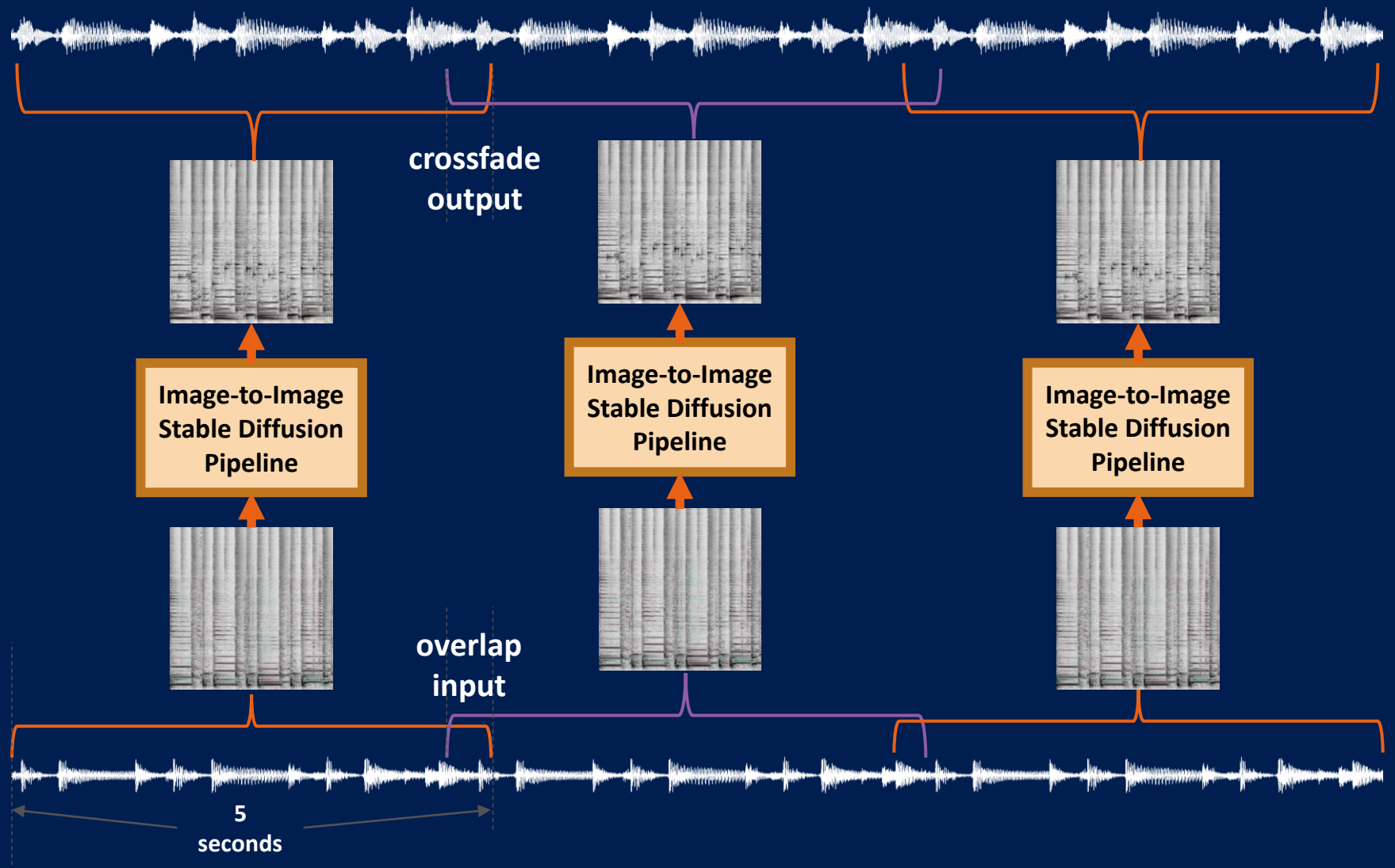
Prompt A: "a portrait of a **happy** coal miner in 19th century…"

Prompt B: "a portrait of a **sad** coal miner in 19th century…"

α

α = 0. 0.25 0.5 0.75 1.

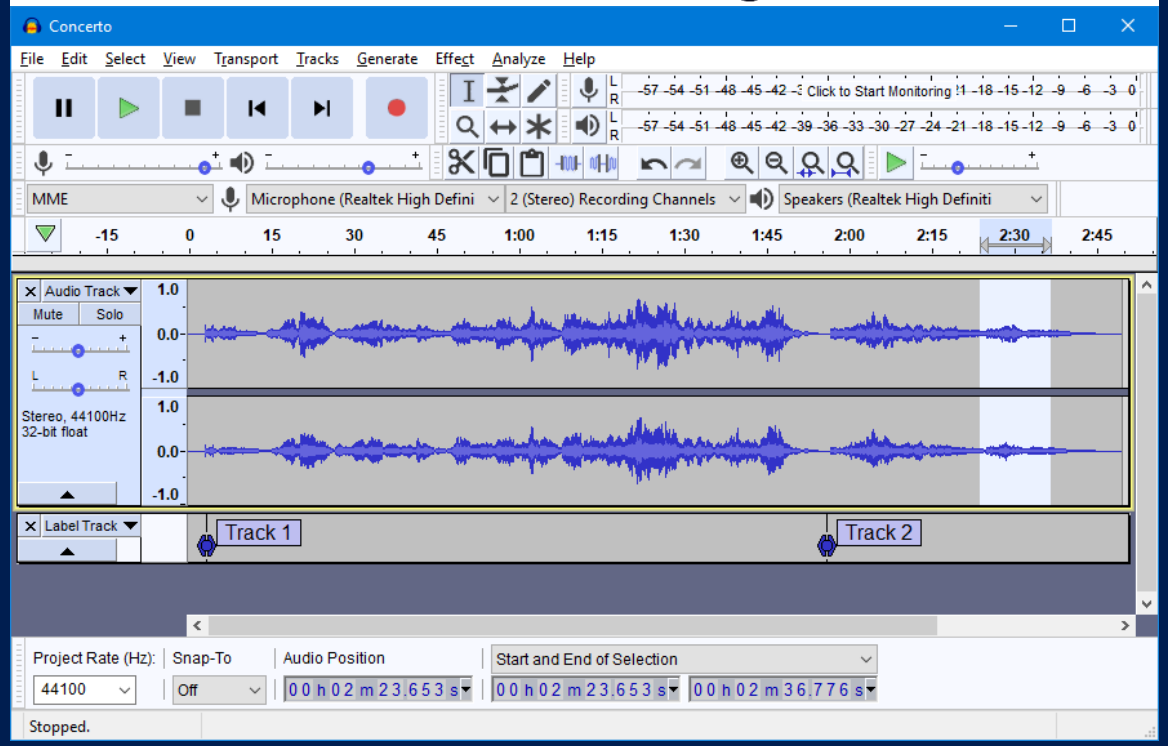
Music Style Remix



intel Ai summit

OpenVINO™ AI Plugins for Audacity®

Audacity®



OpenVINO™ AI Plugins

Noise Suppression

Music Separation

Transcription

wHISper C++

Music Generation & Style Remix

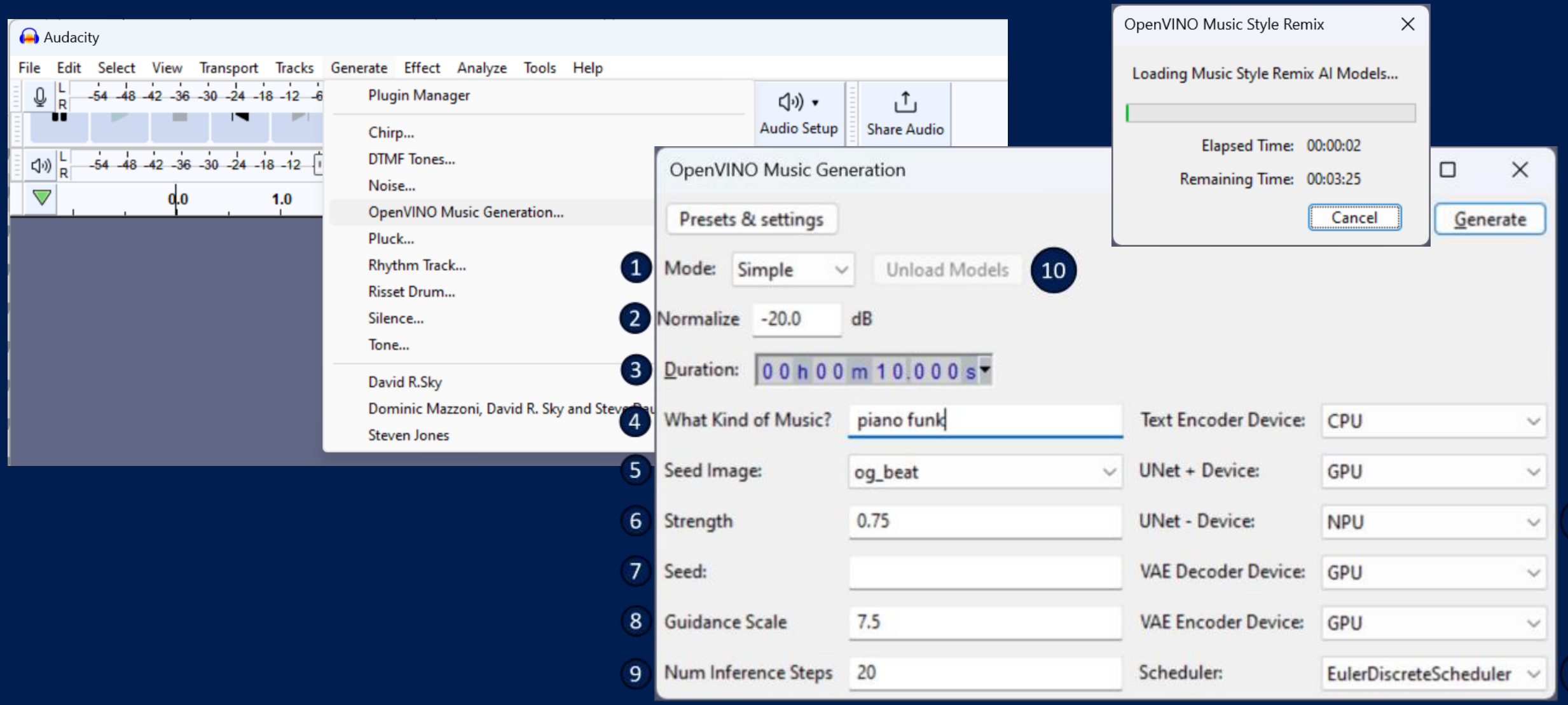
[RIFFUSION]

"We at Audacity are thrilled to be partnering with Intel to help bring powerful, open, and most importantly free AI tools to a mass audience. With time, we expect these kinds of initiatives to produce a new kind of creative environment for musicians, podcasters and audiophiles - a worthy successor to the traditional audio tools that have typified the last 20 years."

–Martin Keary, Head of Product, Audacity®

intel AI summit

OpenVINO™ AI Plugins for Audacity® Music Generation



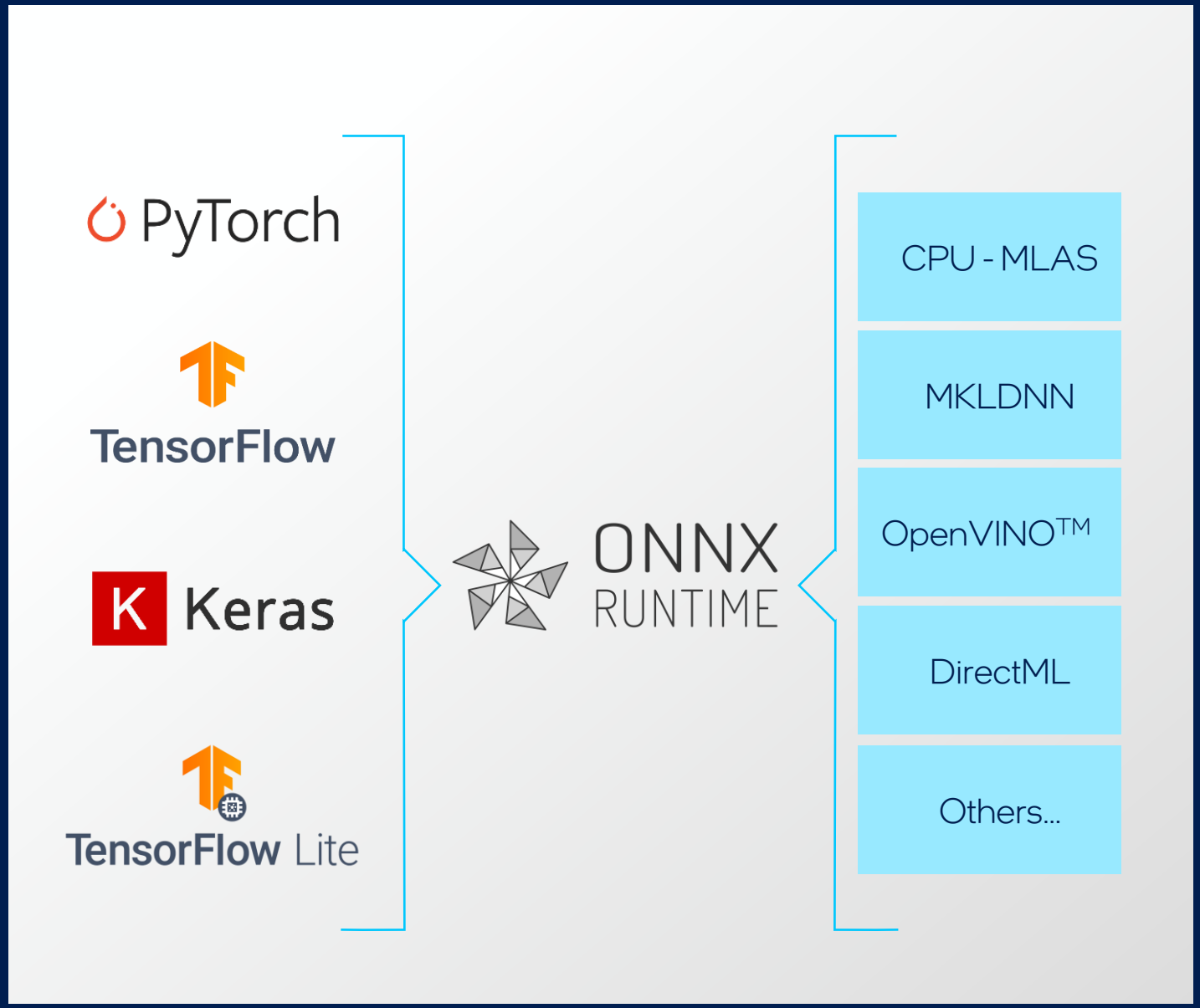
https://github.com/intel/openvino-plugins-ai-audacity/releases

Using ONNX* with OpenVINO™ EP

ONNX* enables cross platform development, utilizing execution providers for multiple hardware backends

OpenVINO™ Execution Provider for ONNX* provides optimized performance on Intel® Platforms

DirectML Execution provider enables cross platform scalability



intel AI summit

Transforming the PC Experience through AI

Strong set of AI capabilities, including the new NPU for power efficient AI offload

Optimized Performance using OpenVINO™, ONNX*, and DirectML Frameworks

Transition from PoC to Productization easily using OpenVINO™ Notebooks

Intel® Core™ Ultra

CPU

Fast Response, DL Boost (VNNI)

Ideal for light-weight, single inference low-latency AI tasks

GPU

Performance Parallelism & Throughput, 16 Vector Engines

Ideal for AI infused in Media/3D/Render pipeline

NPU

Dedicated AI Engine for New & Improved Experiences

Ideal for power-efficient, sustained AI and AI offload

SW Tools

ONNX OpenVINO DirectML

intel AI summit

What's Next?

Download and try it for yourself!
https://github.com/openvinotoolkit/openvino_notebooks

Audacity® plugins to be published:
https://github.com/intel/openvino-plugins-ai-audacity

Explore the many Intel supported open-source projects on Github:

- Stable Diffusion for GIMP
  https://github.com/intel/openvino-ai-plugins-gimp
- Plugins for OBS Studio
  https://github.com/intel/openvino-plugins-for-obs-studio
- Many more:
  https://github.com/intel

OpenVINO™ Documentation:
https://docs.openvino.ai/

The Art of Performance: Inference Optimization of Gen AI and LLMs with OpenVINO™ on AI PC

Zhuo Wu

AI Software Evangelist

intel.

OpenVINO

intel.

Notices and Disclaimers

For notices, disclaimers, and details about performance claims, visit www.intel.com/PerformanceIndex or scan the QR code:

intel Ai summit

Thank You!