



# Intel Vision 2024 の エンタープライズ AI アート展

インテル® Gaudi® AI アクセラレーターによる  
Stable Diffusion のリアルタイム・ファインチューニング

Srinarayan Srikanthan インテル コーポレーション  
Preethi Venkatesh インテル コーポレーション

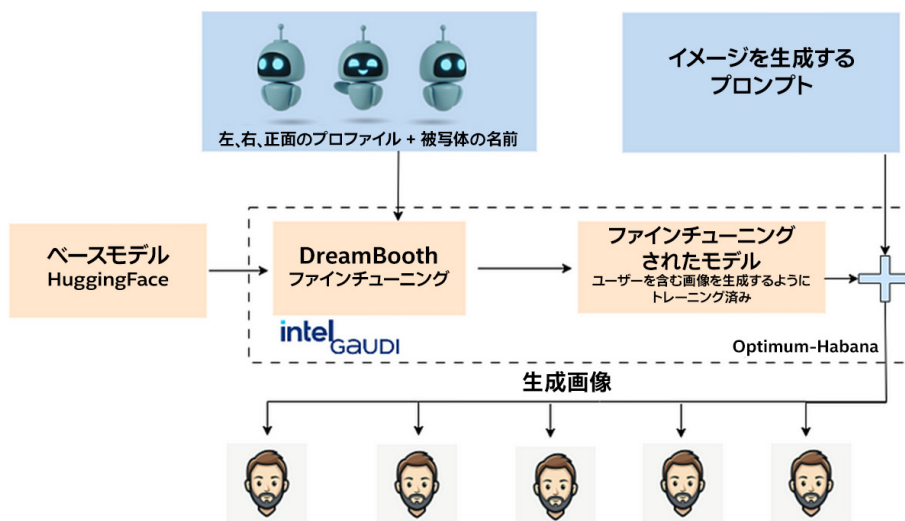
Intel Vision 2024 のエンタープライズ AI アート展では、参加者がインテル® Gaudi® AI アクセラレーターでオープンソースの Stable Diffusion モデルを実行して、独自のアート作品を作成するデモが実施されました。このデモの特筆すべき点は、参加者がリアルタイムで作品を作成する Few-Shot 学習の実践的な応用を通して、創造性と革新性を育むため企業環境で AI をどのように活用できるかを具体的に示したことです。

この記事では、このデモについて説明し、自分で実行する方法も紹介します。

## デモの概要

Stable Diffusion に基づくカスタムアート作品を作成して、インテル® Gaudi® プロセッサーおよびソフトウェア・スタックの機能を紹介しました。デモでは、[DreamBooth](#) (英語) ファインチューニング手法を使用して、限られた画像セットで [Dreamlike Diffusion](#) (英語) と呼ばれる Stable Diffusion 1.5 モデルのバリエーションをファインチューニングし、最小限のデータでパーソナライズされたアートを生成できることを示しました。インテル® Gaudi® プロセッサーと [optimum-habana](#) (英語) により、必要な計算能力が提供され、高速なモデル・トレーニングが行われました。

ベースモデルは、Stable Diffusion 1.5 ベースの Dreamlike Diffusion 1.0 です。DreamBooth ファインチューニングは、Stable Diffusion のようなテキストから画像への変換モデルを、被写体のわずか数枚 (3 ~ 5 枚) の画像でパーソナライズする方法です。このデモでは、Stable Diffusion 1.5 の DreamBooth ファインチューニングにより、独自の画像からカスタムアートを作成できるようにします。



結果として得られるファインチューニングされたモデルは、以下に示すようなユーザープロンプトに基づいて、ユニークなアート作品を生成できます。



**プロンプト :** a dreamlike vision of the universe swirling within close-up side portrait of solo gaudigeekatintel, fluid, constellations and nebulae, dreamlike art, fantasy, star trek aesthetic, vibrant pastel color aesthetic, concept art, sharp focus, flawless skin, pastel colors, digital painting, hd, dramatic lighting, trending in art station (gaudigeekatintel の横顔をクローズアップした肖像画の中に渦巻く幻想的な宇宙、流体、星座と星雲、幻想的なアート、ファンタジー、スタートレックの美学、鮮やかなパステルカラーの美学、コンセプトアート、鮮明、完璧な肌、パステルカラー、デジタル絵画、HD、ドラマチックな照明、ArtStation のトレンド)

Intel Vision 2024 のデモの録画は[こちら](#)（英語）から視聴できます。

## チュートリアル：独自の Stable Diffusion アートを作成する

### DreamBooth ファインチューニング

デモでは、Stable Diffusion 1.5 の DreamBooth ファインチューニングを使用して、独自の画像（512 x 512 解像度にサイズ変更）のカスタムアートを作成します。4 つの環境変数を設定する必要があります。

1. **MODEL\_NAME** : Dreamlike Diffusion 1.0 モデル (dreamlike-art/dreamlike-diffusion-1.0) に設定します。これは、[dreamlike.art](#)（英語）によって作成された高品質のアートでファインチューニングされた Stable Diffusion 1.5 です。
2. **INSTANCE\_DIR** : 3 ~ 5 枚の入力画像を含むディレクトリーに設定します (例: /home/art\_studio/gaudigeekatintel/)。入力画像は、背景が無地で、適切な照明が当たり、肩の高さで 512 x 512 にサイズ変更された被写体の正面、右側面、左側面の画像です (以下のサンプル画像を参照)。被写体の参照または名前は、できるだけ一意である必要があります (例: 「gaudigeekatintel」)。ここでは、モデルのアーティファクトの作成にこの参照名を使用します。
3. **OUTPUT\_DIR** : ファインチューニングされたモデルの保存先ディレクトリーに設定します (例: /home/art\_studio/dd\_model\_gaudigeekatintel)。
4. **CLASS\_DATA\_DIR** : トレーニングするクラスの画像を含むディレクトリーに設定します。このチュートリアルでは、DreamBooth ファインチューニングで人物画像を生成するために使用する男性と女性のサンプル画像をダウンロードします。[Kaggle データセット](#)（英語）から JPEG 形式の「男性」と「女性」の画像をそれぞれ約 50 枚選択し、/home/art\_studio/person という名前のディレクトリーに保存します。



## インテル® Gaudi® プロセッサ上でのファインチューニング

Gaudi Docker イメージを起動して、上記で作成したすべてのディレクトリーのボリュームをマウントします。

```
cd /home/art_studio
docker run -it --runtime=habana \
-e HABANA_VISIBLE_DEVICES=all \
-e OMPI_MCA_btl_vader_single_copy_mechanism=none \
--cap-add=sys_nice \
--net=host \
--ipc=host \
-v $(pwd) :/home \
vault.habana.ai/gaudi-docker/1.15.1/ubuntu22.04/habanalabs/pytorch-installer-2.2.0:latest
```

optimum-habana をインストールします。

```
pip install optimum-habana==v1.11.0
pip install peft
```

コンテナ内のすべての環境変数を設定します。

```
export MODEL_NAME="dreamlike-art/dreamlike-diffusion-1.0"
export INSTANCE_DIR="/home/gaudigeekatintel/"
export OUTPUT_DIR="/home/dd_model_gaudigeekatintel"
export CLASS_DATA_DIR="/home/person"
```

GitHub\* の optimum-habana リポジトリをクローンします。

```
git clone -b v1.12.0 https://github.com/huggingface/optimum-habana.git
```

DreamBooth ファインチューニング機能を備えた optimum-habana のサポートが [PR](#) (英語) によって追加され、インテル® Gaudi® ベースのアーキテクチャーでファインチューニングを実行できるようになります。

```
cd optimum-habana
cd examples/stable-diffusion/training/
```

インテル® Gaudi® プロセッサで AI モデルをトレーニングする場合、トレーニング・スクリプトを初めて実行すると、optimum-habana は時間をかけてグラフと呼ばれる複雑な計算マップを作成します。このグラフは、効率的な処理を保證するためハードウェアに合わせて調整されます。これは、特定のユースケースで 1 回だけ発生するウォームアップ・コストです。最初の被写体の 3 つの画像に対して 1 回だけ実行する必要があります。その後、グラフを使用して後続の被写体をファインチューニングできます。グラフを作成するには、環境変数 `PT_HPU_RECIPE_CACHE_CONFIG` を使用して `recipe_cache` を制御します。これには、カンマで区切られた次の 3 つの情報が必要です。

1. グラフの保存場所 (マシン上のパス、例: `/tmp/recipe_cache`)。
2. 使用後にグラフを削除するかどうか (保持する場合は `False`、削除する場合は `True`)。
3. キャッシュサイズの上限 (メガバイト単位、例: 1GB の場合は 1024)。

例:

```
PT_HPU_RECIPE_CACHE_CONFIG=/tmp/recipe_cache,False,1024 python <script.py>
```

これで、後で再利用できるように指定された場所にグラフが作成され保存されます。



次のパラメーターで `train_dreambooth.py` スクリプトを 5 ステップ実行し、グラフを作成します。この手順は、指定されたベアメタルシステムまたはコンテナで 1 回だけ実行する必要があることに注意してください。グラフがキャッシュに保存されたら、被写体と同じ場合でも、異なる場合でも、後続の実行でこのプロセスを繰り返す必要はありません。

```
PT_HPU_RECIPE_CACHE_CONFIG=/tmp/dld_recipe_cache,False,1024 \
python train_dreambooth.py \
--pretrained_model_name_or_path=$MODEL_NAME \
--instance_data_dir $INSTANCE_DIR \
--output_dir=$OUTPUT_DIR \
--class_data_dir=$CLASS_DATA_DIR \
--with_prior_preservation \
--prior_loss_weight=1.0 \
--instance_prompt="gaudigeekatintel" \
--class_prompt="person" \
--train_batch_size=1 \
--gradient_accumulation_steps=1 \
--learning_rate=2e-6 \
--lr_scheduler="constant" \
--lr_warmup_steps=0 \
--max_train_steps=5 \
--gaudi_config_name Habana/stable-diffusion \
--train_text_encoder \
--center_crop \
--num_class_images=12 \
--seed=0 \
--prior_generation_precision bf16 full
```

次に、`train_dreambooth.py` スクリプトを 350 ステップ実行し、被写体の画像で完全にファインチューニングします。このステップでは、`/tmp/dld_recipe_cache` に保存されたグラフを再利用します。

```
PT_HPU_RECIPE_CACHE_CONFIG=/tmp/dld_recipe_cache,False,1024 \
python train_dreambooth.py \
--pretrained_model_name_or_path=$MODEL_NAME \
--instance_data_dir $INSTANCE_DIR \
--output_dir=$OUTPUT_DIR \
--class_data_dir=$CLASS_DATA_DIR \
--with_prior_preservation \
--prior_loss_weight=1.0 \
--instance_prompt="gaudigeekatintel" \
--class_prompt="person" \
--train_batch_size=1 \
--gradient_accumulation_steps=1 \
--learning_rate=2e-6 \
--lr_scheduler="constant" \
--lr_warmup_steps=0 \
--max_train_steps=350 \
--gaudi_config_name Habana/stable-diffusion \
--train_text_encoder \
--center_crop \
--num_class_images=12 \
--seed=0 \
--prior_generation_precision bf16 full
```

これにより、保存されたグラフと最適なパラメーターを使用して、`INSTANCE_DIR` に保存された画像でモデルがファインチューニングされます。ファインチューニングされたモデルは `OUTPUT_DIR` に保存されます。保存されたグラフを使用したファインチューニングの実行には、1 枚のインテル® Gaudi® カードで約 3 ~ 4 分かかります。このデモ設定では 1 枚のカードを対象としていますが、現在 8 枚のカードで実行できるようにワークロードを最適化しています。

## テキストプロンプトを使用してファインチューニングされた画像を生成する

このデモの出力例を以下に示します。次のサンプルスクリプトでは、`optimum-habana` の `diffusers` クラスを使用して、Stable Diffusion パイプラインとスケジューラーを実行します。ファインチューニングされたモデルを読み込み、入力プロンプトに基づいて肖像画を生成します。ここでは、「`portrait of solo gaudigeekatintel in a multiverse universe with planets, moons and solar flares, star trek, pastel colors, blue and purple tone background, dramatic lighting, trending in art station.` (惑星、月、太陽フレア、スタートレック、パステルカラー、青と紫のトーンの背景、ドラマチックな照明、ArtStation のトレンドを含む多元宇宙を背景とした `gaudigeekatintel` の肖像画)」というプロンプトを試します。独自の肖像画を作成するには、DreamBooth ファインチューニングで使ったインスタンスプロンプトを入力してください。

```

import torch
import os
from optimum.habana.diffusers import GaudiStableDiffusionPipeline, GaudiDDIMScheduler

# スケジューラーの設定
scheduler = GaudiDDIMScheduler.from_pretrained(
    "dreamlike-art/dreamlike-diffusion-1.0",
    subfolder="scheduler"
)

# インテル(R) Gaudi(R) プロセッサの Stable Diffusion パイプラインの使用
pipeline = GaudiStableDiffusionPipeline.from_pretrained(
    '/home/dd_model_gaudigeekatintel',
    scheduler=scheduler,
    use_habana=True,
    use_hpu_graphs=True,
    gaudi_config="Habana/stable-diffusion"
)

negative_prompt="easynegative, head covered, face covered, helmet, not indoors, no flashy
jewelry, not indian bride, two men, two women, two persons, two heads, two bodies, duplicate
person, multiple person, mirror reflection, eye color change, low quality, worst quality:1.4,
bad anatomy, bad composition, out of frame, ugly, old person with wrinkles, morbid,
mutilated, out of frame, extra fingers, mutated hands, poorly drawn hands, poorly drawn face,
mutation, deformed, ugly, blurry, bad anatomy, bad proportions, extra limbs, cloned face,
disfigured, out of frame, ugly, extra limbs, bad anatomy, gross proportions, malformed limbs,
missing arms, missing legs, extra arms, extra legs, mutated hands, fused fingers, too many
fingers, long neck, watermark, signature, text, deformed nose, deformed lips"

# "gaudigeekatintel" の代わりに DreamBooth ファインチューニングの --instance_prompt フラグで使ったインス
タンス・ラベルを使用

prompt=" portrait of solo gaudigeekatintel in a multiverse universe with planets, moons and
solar flares, star trek, pastel colors, blue and purple tone background, dramatic lighting,
trending in art station"

image = pipeline(prompt, height=536, width=960, negative_prompt= negative_prompt, num_
inference_steps=150, guidance_scale=7).images[0]

output_path= '/home/output/'
os.makedirs(output_path)
filename='gaudigeekatintel.png'
os.path.join(output_path, filename)

```



プロンプト : portrait of solo gaudigeekatintel in a multiverse universe with planets, moons and solar flares, star trek, pastel colors, blue and purple tone background, dramatic lighting, trending in art station  
 (惑星、月、太陽フレア、スタートレック、パステルカラー、青と紫のトーンの背景、ドラマチックな照明、ArtStation のトレンドを含む多元宇宙を背景とした gaudigeekatintel の肖像画)

## 追加のファインチューニングとプロンプトの例



プロンプト : create a captivating book cover of Patatintel as an astronaut in space, face not covered, bring illuminating large planet, dramatic lighting, 4k, trending in art station, sharp focus, flawless skin, photorealistic, dreamart, hd  
 (宇宙飛行士 Patatintel の魅力的な本の表紙を作成、顔は覆われていない、照らされた大きな惑星、ドラマチックな照明、4K、ArtStation のトレンド、鮮やか、完璧な肌、フォトリアリスティック、幻想的、HD)





プロンプト : photorealistic close-up portrait of solo Christophatintel with suit, futuristic cityscape dominated with skyscraper with electronic gadgets, dramatic lighting, trending in art station  
 (スーツを着た Christophatintel の写真のようなクローズアップの肖像画、電子機器、ドラマチックな照明、ArtStation のトレンドを含む高層ビルが目立つ未来的な都市の風景)



プロンプト : portrait of solo Sriatintel in the interstellar space, dressed in the space suit from the movie interstellar, black hole, moons, universe, make it look like a movie poster, dramatic lighting, trending in art station  
 (星間空間にいる Sriatintel の肖像画、映画「インターステラー」の宇宙服を着ている、ブラックホール、月、宇宙、映画のポスターのように見せる、ドラマチックな照明、ArtStation のトレンド)

このチュートリアルは、[インテル® Tiber™ デベロッパー・クラウド](#)<sup>1</sup> または [AWS\\*](#) のインテル® Gaudi® AI アクセラレーター・インスタンスで試すことができます。詳細は、[インテルの AI フレームワーク](#)（英語）サポートまでお問い合わせください。

<sup>1</sup> 旧インテル® デベロッパー・クラウド

[ **訳者注:** 2024 年 8 月現在、インテル® Tiber™ デベロッパー・クラウドでは、インテル® Gaudi® 2 AI アクセラレーターへのアクセスを Enterprise ELA プランの利用者向けに承認制で提供しています。 ]