

画像からの3Dポイントクラウド生成

① https://github.com/openai/point-e/tree/main/point_e/examples にアクセス

オープンナイ / ポイント-e (公共)

コード 問題 57 プルリクエスト 13 行動 プロジェクト 安全 洞察

主要 ポイント-e / point_e / 例 /

unixpickle の 最初のコミット c26852e 2022年12月20日 歴史

例のデータ	最初のコミット	6ヶ月前
image2pointcloud.ipynb	最初のコミット	6ヶ月前
ペーパーバナー.gif	最初のコミット	6ヶ月前
pointcloud2mesh.ipynb	最初のコミット	6ヶ月前
text2pointcloud.ipynb	最初のコミット	6ヶ月前

②「例のデータ」クリック

③ コーギー.jpg をクリック

主要 ポイント-e / point_e / 例 / 例のデータ /

unixpickle の 最初のコミット

コーギー.jpg	最初のコミット
コーギーブライ	最初のコミット
キューブ_スタック.jpg	最初のコミット
pc_corgi.npz	最初のコミット
pc_cube_stack.npz	最初のコミット

115 ライン (115 sloc) | 3.25KB

⑤ クリックすると
コードが表示

```
In [ ]: from PIL import Image
import torch
from tqdm.auto import tqdm

from point_e.diffusion.configs import DIFFUSION_CONFIGS, diffusion_from_config
from point_e.diffusion.sampler import PointCloudSampler
from point_e.models.download import load_checkpoint
from point_e.models.configs import MODEL_CONFIGS, model_from_config
from point_e.util.plotting import plot_point_cloud
```

```
In [ ]: device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')

print('creating base model...')
base_name = 'base40M' # use base300M or base1B for better results
base_model = model_from_config(MODEL_CONFIGS[base_name], device)
base_model.eval()
base_diffusion = diffusion_from_config(DIFFUSION_CONFIGS[base_name])

print('creating upsample model...')
upsampler_model = model_from_config(MODEL_CONFIGS['upsample'], device)
upsampler_model.eval()
upsampler_diffusion = diffusion_from_config(DIFFUSION_CONFIGS['upsample'])
```

13.7KB ダウンロード



④ ダウンロード

<https://github.com/openai/point-e> にアクセス

The screenshot shows the GitHub repository page for `openai/point-e`. At the top, there are navigation links for Product, Solutions, Open Source, and Pricing, along with a search bar. Below the repository name, there are tabs for Code, Issues (57), Pull requests (13), Actions, Projects, Security, and Insights. The main content area shows a merge pull request by `heewooj` and a file list with columns for file name, commit message, and time. The file list includes `point_e`, `.gitignore`, `LICENSE`, `README.md`, `model-card.md`, and `setup.py`. Below the file list, the `README.md` content is visible, featuring the title "Point-E" and four 3D point cloud models: a blue cube with a red top, a purple vase, a grey chair, and a purple flower arrangement.

Codeをクリック

The screenshot shows the 'Code' dropdown menu from the GitHub repository page. It has two tabs: 'Local' and 'Codespaces'. Under the 'Local' tab, there are three main options: 'Clone', 'Open with GitHub Desktop', and 'Download ZIP'. The 'Clone' option is expanded, showing 'HTTPS' and 'GitHub CLI' as methods. Below this, the repository URL `https://github.com/openai/point-e.git` is displayed with a copy icon. Below the URL, it says "Use Git or checkout with SVN using the web URL." The 'Download ZIP' option is highlighted with a red arrow.

Zipファイルをダウンロード

① <https://colab.research.google.com/>にアクセス

Colaboratory へようこそ

ようこそ

Colab (正式名称「Colaboratory」) では、ブラウザ上で Python を記述、実行できます。以下の機能を使用できます。

- 環境構築が不要
- GPU に料金なしでアクセス
- 簡単に共有

Colab は、学生からデータサイエンティスト、AI リサーチャーまで、皆さんの作業を効率化します。詳しくは、[Colab の紹介動画](#)をご覧ください。下のリンクからすぐに使ってみることもできます。

はじめに

ご覧になっているこのドキュメントは静的なウェブページではなく、Colab ノートブックという、コードを記述して実行できるインタラクティブな環境です。

② 編集→ノートブック設定→ランチャタイプはpython3、ハードウェアアクセラレータは「GPU」を選択

ノートブックの設定

ランタイムのタイプ

Python 3

ハードウェア アクセラレータ

GPU

GPU のタイプ

T4

プレミアム GPU を利用するには [コンピューティング ユニットを追加購入](#)

このノートブックを保存する際にコードセルの出力を除外する

キャンセル 保存

③ ファイル→ノートブックを新規作成

Colaboratory へようこそ

ファイル 編集 表示 挿入 ランタイム ツール ヘルプ

ノートブックを新規作成

ノートブックを開く

ノートブックをアップロード

名前の変更

ドライブにコピーを保存

コピーを GitHub Gist として保存

GitHub にコピーを保存

保存

変更履歴

ダウンロード

印刷

Colab と

Untitled1.ipynb

ファイル 編集 表示 挿入 ランタイム ツール ヘルプ

+ コード + テキスト

[] 1

1 |

④ 「#パッケージのインストール」のコードを書き込む

```
1 # パッケージのインストール
2 !git clone https://github.com/openai/point-e.git
3 %cd point-e
4 !pip install .
```

Cloning into 'point-e'...

remote: Enumerating objects: 57, done.

remote: Counting objects: 100% (57/57), done.

remote: Compressing objects: 100% (49/49), done.

remote: Total 57 (delta 7), reused 51 (delta 7), pack-reused 0

Unpacking objects: 100% (57/57), 1.56 MiB | 5.48 MiB/s, done.

/content/point-e/point-e/point-e

Looking in indexes: <https://pypi.org/simple>, <https://us-python.pkg.dev/colab-wheels/public/simple/>

Processing /content/point-e/point-e/point-e

Preparing metadata (setup.py) ... done

Collecting clip@ git+https://github.com/openai/CLIP.git (from point-e==0.0.0)

Cloning https://github.com/openai/CLIP.git to /tmp/pip-install-4mtfgq0r/clip_87de65912e014ecf821e10bd92d5cf1a

Running command git clone --filter=blob:none --quiet https://github.com/openai/CLIP.git /tmp/pip-install-4mtfgq0r/clip_87de65912e014ecf821e10bd92d5cf1a

Resolved https://github.com/openai/CLIP.git to commit a9b1bf5920416aaeac965c25dd9e8f98c864f16

Preparing metadata (setup.py) ... done

Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from point-e==0.0.0) (3.12.0)

Requirement already satisfied: Pillow in /usr/local/lib/python3.10/dist-packages (from point-e==0.0.0) (8.4.0)

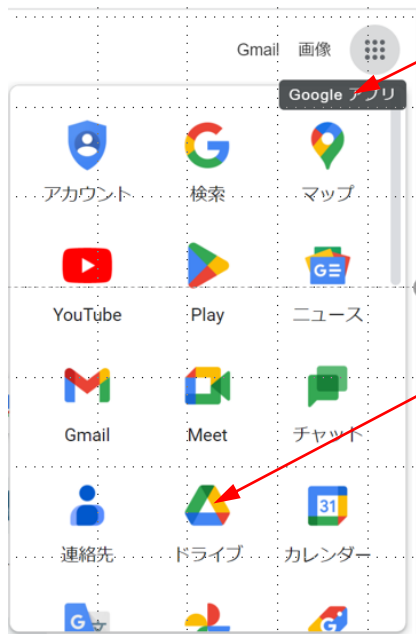
Requirement already satisfied: torch in /usr/local/lib/python3.10/dist-packages (from point-e==0.0.0) (2.0.1+cu118)

Requirement already satisfied: fire in /usr/local/lib/python3.10/dist-packages (from point-e==0.0.0) (0.5.0)

Requirement already satisfied: humanize in /usr/local/lib/python3.10/dist-packages (from point-e==0.0.0) (4.6.0)



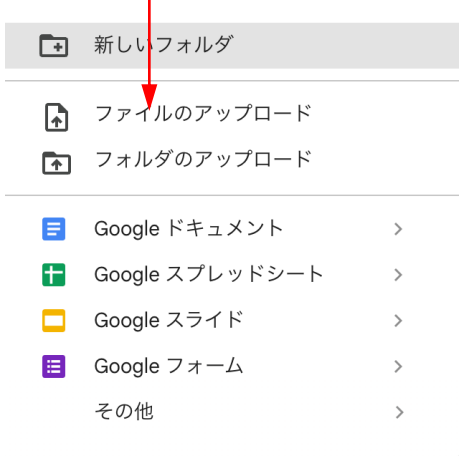
Google で検索または URL を入力



①「Googleアプリ」をクリック

②ドライブをクリック

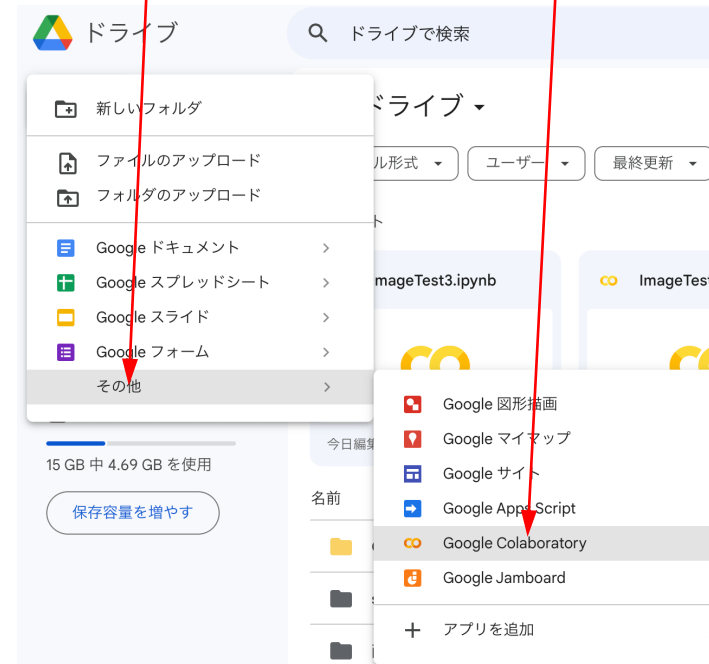
④ファイルのアップロード



③「+新規」をクリック



⑥その他 → Google Colaboratory クリック



⑤corgi.jpgをアップロード



⑦ Colaboratory 起動



⑧ Colaboratory に以下コードを書き込む

```
1 from google.colab import drive
2 drive.mount('/content/drive')
```

←Googleドライブのファイルを
Colaboratoryにマウントする



```
1 # パッケージのインポート
2 from PIL import Image
3 import torch
4 from tqdm.auto import tqdm
5 from point_e.diffusion.configs import DIFFUSION_CONFIGS, diffusion_from_config
6 from point_e.diffusion.sampler import PointCloudSampler
7 from point_e.models.download import load_checkpoint
8 from point_e.models.configs import MODEL_CONFIGS, model_from_config
9 from point_e.util.plotting import plot_point_cloud
```



```
1 # モデルの準備
2 device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')
3
4 print('creating base model...')
5 base_name = 'base40M' # use base300M or base1B for better results
6 base_model = model_from_config(MODEL_CONFIGS[base_name], device)
7 base_model.eval()
8 base_diffusion = diffusion_from_config(DIFFUSION_CONFIGS[base_name])
9
10 print('creating upsample model...')
11 upsampler_model = model_from_config(MODEL_CONFIGS['upsample'], device)
12 upsampler_model.eval()
13 upsampler_diffusion = diffusion_from_config(DIFFUSION_CONFIGS['upsample'])
14
15 print('downloading base checkpoint...')
16 base_model.load_state_dict(load_checkpoint(base_name, device))
17
18 print('downloading upsampler checkpoint...')
19 upsampler_model.load_state_dict(load_checkpoint('upsample', device))
```



```
1 # サンプラーの準備
2 sampler = PointCloudSampler(
3     device=device,
4     models=[base_model, upsampler_model],
5     diffusions=[base_diffusion, upsampler_diffusion],
6     num_points=[1024, 4096 - 1024],
7     aux_channels=['R', 'G', 'B'],
8     guidance_scale=[3.0, 3.0],
9 )
```



```
1 #画像の準備
2 img = Image.open('/content/drive/My Drive/corgi.jpg')
3
4 # 推論の実行
5 samples = None
6 for x in tqdm(sampler.sample_batch_progressive(batch_size=1, model_kwargs=dict(images=[img]))):
7     samples = x
```

Googleドライブのファイルを
読み込む

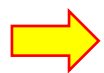


⑨全てのコードを実行

```
1 # ポイントクラウドの表示  
2 pc = sampler.output_to_point_clouds(samples)[0]  
3 fig = plot_point_cloud(pc, grid_size=3, fixed_bounds=((-0.75, -0.75, -0.75),(0.75, 0.75, 0.75)))
```

実行結果

2次元



3次元

