

```
!pip -q install torch
```

```
import torch
print(torch.cuda.device_count())
```

```
0
```

Start coding or [generate](#) with AI.

```
!python /content/sample_data/ \
--silencemap_path /content/sample_data/SilenceMap_final.mat \
--save_figs \
--fig_dir /content/sample_data/figs/
```

```
Device: cpu
Loading SilenceMap file: /content/sample_data/SilenceMap_final.mat
beta length: 1662
coords shape: (1662, 3)
|GT silence| (from MATLAB): 50
|det silence| (MATLAB SilenceMap): 2
mean(beta GT silent) : 0.6312295794487
mean(beta GT active) : 0.8614711165428162
beta min/max after norm: 0.0 1.0
Building k-NN graph...
k_silent (from GT) = 50
```

```
Laplacian baseline: P=0.540 R=0.540 F1=0.540
```

```
Laplacian cluster metrics:
```

Jaccard	:	0.370
ΔCOM (mm)	:	11.151
size_gt	:	50
size_pred	:	50
size_rel_err	:	0.000

```
Training GNN...
[0200] loss=0.21228 data=0.03225 smooth=0.01063 seed=0.25376
[0400] loss=0.19289 data=0.03104 smooth=0.01119 seed=0.21179
[0600] loss=0.18163 data=0.03334 smooth=0.01113 seed=0.18531
[0800] loss=0.17774 data=0.02622 smooth=0.01221 seed=0.18088
[1000] loss=0.17721 data=0.02861 smooth=0.01220 seed=0.17518
[1200] loss=0.17592 data=0.02921 smooth=0.01208 seed=0.17262
[1400] loss=0.17515 data=0.02821 smooth=0.01214 seed=0.17246
```

```
GNN: P=0.740 R=0.740 F1=0.740
```

```
GNN cluster metrics:
```

```
Jaccard      : 0.587
ΔCOM (mm)    : 1.612
size_gt      : 50
size_pred    : 50
size_rel_err : 0.000
```

```
[saved figure → /content/sample_data/figs/1_gt.png]
[saved figure → /content/sample_data/figs/2_laplacian.png]
[saved figure → /content/sample_data/figs/3_gnn.png]
[saved figure → /content/sample_data/figs/4_curves.png]
```

```
!git clone https://github.com/DIKSHAAGARWAL2015/SilenceMap-Localization.git
%cd /content/SilenceMap-Localization
!git status
```

```
Cloning into 'SilenceMap-Localization'...
remote: Enumerating objects: 439, done.
remote: Counting objects: 100% (15/15), done.
remote: Compressing objects: 100% (14/14), done.
remote: Total 439 (delta 5), reused 0 (delta 0), pack-reused 424 (from 1)
Receiving objects: 100% (439/439), 20.41 MiB | 26.26 MiB/s, done.
Resolving deltas: 100% (220/220), done.
/content/SilenceMap-Localization
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
```

```
!pwd
```

```
/content/SilenceMap-Localization
```

```
%cd /content/SilenceMap-Localization
```

```
/content/SilenceMap-Localization
```

```
!git status
```

```
On branch main
```

```
Your branch is up to date with 'origin/main'.
```

```
Untracked files:
```

```
(use "git add <file>..." to include in what will be committed)  
safe_save_run1/
```

```
nothing added to commit but untracked files present (use "git add" to track)
```

```
!git add .
```

```
!git commit -m "safing weights"
```

```
[main 313941a] safing weights  
10 files changed, 10 insertions(+)  
create mode 100644 safe_save_run1/W_col.npy  
create mode 100644 safe_save_run1/W_data.npy  
create mode 100644 safe_save_run1/W_row.npy  
create mode 100644 safe_save_run1/W_shape.npy  
create mode 100644 safe_save_run1/beta.npy  
create mode 100644 safe_save_run1/config.json  
create mode 100644 safe_save_run1/mask.npy  
create mode 100644 safe_save_run1/model_state.pt  
create mode 100644 safe_save_run1/prob.npy  
create mode 100644 safe_save_run1/src_xyz.npy
```

```
!git config --global user.email "dikshaagarwal738@gmail.com"
```

```
!git config --global user.name "DIKSHAAGARWAL2015"
```

```
TOKEN = "ghp_83jNsLwx7uCJEtpFsavZqpekWBxwVa08CUUL"
USER = "DIKSHAAGARWAL2015"
REPO = "SilenceMap-Localization"
!git remote set-url origin https://TOKEN@github.com/USER/{REPO}.git
```

```
!git branch -a
```

```
* main
  remotes/origin/HEAD -> origin/main
  remotes/origin/main
```

```
!git pull
```

```
Already up to date.
```

```
!git config pull.rebase false
```

```
!zip -r /content/sample_data/safe_save_run1.zip /content/sample_data/safe_save_run1
```

```
adding: content/sample_data/safe_save_run1/ (stored 0%)
adding: content/sample_data/safe_save_run1/src_xyz.npy (deflated 38%)
adding: content/sample_data/safe_save_run1/W_data.npy (deflated 53%)
adding: content/sample_data/safe_save_run1/W_col.npy (deflated 72%)
adding: content/sample_data/safe_save_run1/W_shape.npy (deflated 46%)
adding: content/sample_data/safe_save_run1/mask.npy (deflated 91%)
adding: content/sample_data/safe_save_run1/W_row.npy (deflated 72%)
adding: content/sample_data/safe_save_run1/prob.npy (deflated 5%)
adding: content/sample_data/safe_save_run1/model_state.pt (deflated 11%)
adding: content/sample_data/safe_save_run1/config.json (deflated 24%)
adding: content/sample_data/safe_save_run1/beta.npy (deflated 12%)
```

```
!git push
```

```
fatal: could not read Password for 'https://ghp\_83jNsLwx7uCJEtpFsavZqpekWBxwVa08CUUL@github.com': No such device or address
```

```
!pwd  
!git status  
!git remote -v
```

```
/content/SilenceMap-Localization  
On branch main  
Your branch is ahead of 'origin/main' by 1 commit.  
  (use "git push" to publish your local commits)  
  
nothing to commit, working tree clean  
origin https://github.com/DIKSHAAGARWAL2015/SilenceMap-Localization.git (fetch)  
origin https://github.com/DIKSHAAGARWAL2015/SilenceMap-Localization.git (push)
```

```
!git config --global user.email "dikshaagarwal738@gmail.com"  
!git config --global user.name "DIKSHAAGARWAL2015"
```

```
%matplotlib inline
```

```
from mpl_toolkits.mplot3d import Axes3D # noqa: F401 (ensures 3D is registered)  
import matplotlib.pyplot as plt
```

```
!pip -q install mat73
```

```
from IPython.display import Image, display  
import glob  
for fn in sorted(glob.glob("/content/sample_data/figs/*.png")):  
    display(Image(fn))
```

```
!pip install h5py
```

```
Requirement already satisfied: h5py in /usr/local/lib/python3.12/dist-packages (3.15.1)
Requirement already satisfied: numpy>=1.21.2 in /usr/local/lib/python3.12/dist-packages (from h5py) (2.0.2)
```

```
import mat73, pprint
pprint.pprint(mat73.loadmat("/content/sample_data/OT_leadfield_symmetric_1662-128.mat").keys())
pprint.pprint(mat73.loadmat("/content/sample_data/OT_headmodel_symmetric_1662-128.mat").keys())
```

```
dict_keys(['L', 'sensor_locs'])
dict_keys(['Cortex', 'headmodel'])
```

```
!pip -q install mat73 torch tqdm scikit-learn scipy
```

```
import torch
print(torch.__version__)
```

```
2.9.0+cu126
```

```
!pip install torch torchvision torchaudio --index-url https://download.pytorch.org/whl/cu121
```

```
Looking in indexes: https://download.pytorch.org/whl/cu121
Requirement already satisfied: torch in /usr/local/lib/python3.12/dist-packages (2.9.0+cu126)
Requirement already satisfied: torchvision in /usr/local/lib/python3.12/dist-packages (0.24.0+cu126)
Requirement already satisfied: torchaudio in /usr/local/lib/python3.12/dist-packages (2.9.0+cu126)
Requirement already satisfied: filelock in /usr/local/lib/python3.12/dist-packages (from torch) (3.20.0)
Requirement already satisfied: typing-extensions>=4.10.0 in /usr/local/lib/python3.12/dist-packages (from torch) (4.15.0)
Requirement already satisfied: setuptools in /usr/local/lib/python3.12/dist-packages (from torch) (75.2.0)
Requirement already satisfied: sympy>=1.13.3 in /usr/local/lib/python3.12/dist-packages (from torch) (1.14.0)
Requirement already satisfied: networkx>=2.5.1 in /usr/local/lib/python3.12/dist-packages (from torch) (3.6)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.12/dist-packages (from torch) (3.1.6)
Requirement already satisfied: fsspec>=0.8.5 in /usr/local/lib/python3.12/dist-packages (from torch) (2025.3.0)
Requirement already satisfied: nvidia-cuda-nvrtc-cu12==12.6.77 in /usr/local/lib/python3.12/dist-packages (from torch) (12.6.7)
Requirement already satisfied: nvidia-cuda-runtime-cu12==12.6.77 in /usr/local/lib/python3.12/dist-packages (from torch) (12.6)
Requirement already satisfied: nvidia-cuda-cupti-cu12==12.6.80 in /usr/local/lib/python3.12/dist-packages (from torch) (12.6.8)
```

```
Requirement already satisfied: nvidia-cudnn-cu12==9.10.2.21 in /usr/local/lib/python3.12/dist-packages (from torch) (9.10.2.21)
Requirement already satisfied: nvidia-cublas-cu12==12.6.4.1 in /usr/local/lib/python3.12/dist-packages (from torch) (12.6.4.1)
Requirement already satisfied: nvidia-cufft-cu12==11.3.0.4 in /usr/local/lib/python3.12/dist-packages (from torch) (11.3.0.4)
Requirement already satisfied: nvidia-curand-cu12==10.3.7.77 in /usr/local/lib/python3.12/dist-packages (from torch) (10.3.7.77)
Requirement already satisfied: nvidia-cusolver-cu12==11.7.1.2 in /usr/local/lib/python3.12/dist-packages (from torch) (11.7.1.2)
Requirement already satisfied: nvidia-cusparse-cu12==12.5.4.2 in /usr/local/lib/python3.12/dist-packages (from torch) (12.5.4.2)
Requirement already satisfied: nvidia-cusparseelt-cu12==0.7.1 in /usr/local/lib/python3.12/dist-packages (from torch) (0.7.1)
Requirement already satisfied: nvidia-nccl-cu12==2.27.5 in /usr/local/lib/python3.12/dist-packages (from torch) (2.27.5)
Requirement already satisfied: nvidia-nvshmem-cu12==3.3.20 in /usr/local/lib/python3.12/dist-packages (from torch) (3.3.20)
Requirement already satisfied: nvidia-nvtx-cu12==12.6.77 in /usr/local/lib/python3.12/dist-packages (from torch) (12.6.77)
Requirement already satisfied: nvidia-nvjitlink-cu12==12.6.85 in /usr/local/lib/python3.12/dist-packages (from torch) (12.6.85)
Requirement already satisfied: nvidia-cufile-cu12==1.11.1.6 in /usr/local/lib/python3.12/dist-packages (from torch) (1.11.1.6)
Requirement already satisfied: triton==3.5.0 in /usr/local/lib/python3.12/dist-packages (from torch) (3.5.0)
Requirement already satisfied: numpy in /usr/local/lib/python3.12/dist-packages (from torchvision) (2.0.2)
Requirement already satisfied: pillow!=8.3.*,>=5.3.0 in /usr/local/lib/python3.12/dist-packages (from torchvision) (11.3.0)
Requirement already satisfied: mpmath<1.4,>=1.1.0 in /usr/local/lib/python3.12/dist-packages (from sympy>=1.13.3->torch) (1.3.0)
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.12/dist-packages (from jinja2->torch) (3.0.3)
```

```
!pip install mat73
```

```
Collecting mat73
  Downloading mat73-0.65-py3-none-any.whl.metadata (3.6 kB)
Requirement already satisfied: h5py in /usr/local/lib/python3.12/dist-packages (from mat73) (3.15.1)
Requirement already satisfied: numpy in /usr/local/lib/python3.12/dist-packages (from mat73) (2.0.2)
  Downloading mat73-0.65-py3-none-any.whl (19 kB)
Installing collected packages: mat73
Successfully installed mat73-0.65
```

```
!pip install plotly --quiet # "--quiet" hides installation logs
```

```
!python /content/sample_data/debugger.py --save_figs --use_mat \
--leadfield_path /content/sample_data/OT_leadfield_symmetric_1662-128.mat \
--headmodel_path /content/sample_data/OT_headmodel_symmetric_1662-128.mat
```

```
Device: cpu
Loading .mat leadfield/headmodel...
Loaded leadfield: (128, 1662) vertices: (1662, 3)
Avg SNR ≈ 12.05 dB
```

```
[0.51313055 0.4518721 0.32164642 ... 0.56000257 0.3839895 0.7893636 ]
(1662,)

[auto] q_silent set to 3.01% for |S|=50/1662
[0200] loss=0.68013 cls=0.47360 smooth=0.04131
[0400] loss=0.58991 cls=0.37552 smooth=0.04288
[0600] loss=0.57662 cls=0.31665 smooth=0.05199
[0800] loss=0.55375 cls=0.35100 smooth=0.04055
[1000] loss=0.51240 cls=0.29145 smooth=0.04419
[1200] loss=0.49845 cls=0.27096 smooth=0.04550
[1400] loss=0.48491 cls=0.24795 smooth=0.04739
[1600] loss=0.48629 cls=0.23389 smooth=0.05048
[1800] loss=0.49877 cls=0.27293 smooth=0.04517
[2000] loss=0.46888 cls=0.21916 smooth=0.04994

GNN(BCE): P=0.820 R=0.820 F1=0.820
```

```
===== GNN Component Report =====
Vertex-set Jaccard (IoU): 0.695
#Components: GT=5 Pred=5
Matched comp IoU: mean=0.705 median=0.692
Size error abs: mean=1.20 median=1.00
Size error rel: mean=0.120 median=0.100
```

- ✓ EVERYTHING SAVED SAFELY
- 📁 Folder: /content/SilenceMap-Localization/safe\_save\_run1
- ➡️ You can now move to MONKEY (NHP) data using these weights.

```
[saved figure → /content/sample_data/figs/1_gt.png]
[saved figure → /content/sample_data/figs/3_gnn.png]
```

```
import numpy as np
import scipy.io as sio
import h5py

# --- paths (use the SAME path your feature_gnn.py is trying to load) ---
OLD_LEADFIELD = "/content/sample_data/OT_leadfield_symmetric_3002-128.mat" # current (non-v7.3) file
CORTEX_V73     = "/content/sample_data/cortex_downsampled_v73.mat"           # v7.3 file with DownsampledCortex/V
NEW_LEADFIELD  = "/content/sample_data/OT_leadfield_symmetric_3002-128.mat" # overwrite as v7.3

# 1) load L and sensor_locs from the old file (MATLAB v7)
m = sio.loadmat(OLD_LEADFIELD)
L = m["L"]
sensor_locs = m["sensor_locs"] if "sensor_locs" in m else m["sensor_locs".encode()] if "sensor_locs".encode() in m else m["se
```

```
# 2) load cortex vertices from v7.3 file: DownsampledCortex/V
with h5py.File(CORTEX_V73, "r") as f:
    V = np.array(f["DownsampledCortex"]["V"], dtype=np.float32)
    if V.shape[0] == 3 and V.shape[1] != 3:
        V = V.T # ensure (P,3)

# sanity: L columns must match #vertices
assert L.shape[1] == V.shape[0], f"Mismatch: L is {L.shape} but V is {V.shape}"

# 3) write MATLAB v7.3 (HDF5) leadfield file with keys your loader needs
with h5py.File(NEW_LEADFIELD, "w") as f:
    f.create_dataset("L", data=np.asarray(L, dtype=np.float32))
    f.create_dataset("sensor_locs", data=np.asarray(sensor_locs, dtype=np.float32))
    f.create_dataset("src_xyz", data=np.asarray(V, dtype=np.float32))

# quick verify
with h5py.File(NEW_LEADFIELD, "r") as f:
    print("Saved v7.3 keys:", list(f.keys()))
    print("L:", f["L"].shape, "sensor_locs:", f["sensor_locs"].shape, "src_xyz:", f["src_xyz"].shape)
```

Saved v7.3 keys: ['L', 'sensor\_locs', 'src\_xyz']  
L: (128, 3004) sensor\_locs: (128, 3) src\_xyz: (3004, 3)

```
!python /content/sample_data/feature_gnn.py --save_figs --use_mat \
--leadfield_path /content/sample_data/OT_leadfield_symmetric_3002-128_mm.mat \
--headmodel_path /content/sample_data/OT_headmodel_symmetric_3002-128_mm.mat
```

Device: cpu  
Loading .mat leadfield/headmodel...  
ERROR:root:ERROR: not a MATLAB datatype: <HDF5 dataset "L": shape (128, 3004), type "<f4">, (float32)  
ERROR:root:ERROR: not a MATLAB datatype: <HDF5 dataset "sensor\_locs": shape (128, 3), type "<f4">, (float32)  
ERROR:root:ERROR: not a MATLAB datatype: <HDF5 dataset "src\_xyz": shape (3004, 3), type "<f4">, (float32)  
Loaded leadfield: (128, 3004) vertices: (3004, 3)  
Avg SNR ≈ 10.00 dB (target=10 dB)  
[-0.04012999 -0.01967404 0.00861103 ... 0.24074838 0.05484879  
 0.64439959]  
(3004,)  
[auto] q\_silent set to 1.60% for |S|=48/3004

```
[0200] loss=0.52863 cls=0.31463 smooth=0.04280
[0400] loss=0.41801 cls=0.20633 smooth=0.04234
[0600] loss=0.40837 cls=0.17592 smooth=0.04649
[0800] loss=0.40063 cls=0.16484 smooth=0.04716
[1000] loss=0.36307 cls=0.15965 smooth=0.04069
[1200] loss=0.33983 cls=0.13205 smooth=0.04156
[1400] loss=0.34671 cls=0.13369 smooth=0.04260
[1600] loss=0.34178 cls=0.13176 smooth=0.04200
[1800] loss=0.32940 cls=0.12619 smooth=0.04064
[2000] loss=0.32413 cls=0.11821 smooth=0.04118
GNN(BCE): P=0.917 R=0.917 F1=0.917
```

```
===== GNN Component Report =====
Vertex-set Jaccard (IoU): 0.846
#Components: GT=4 Pred=4
Matched comp IoU: mean=0.851 median=0.920
Size error abs: mean=1.00 median=1.00
Size error rel: mean=0.083 median=0.083
```

- ✓ EVERYTHING SAVED SAFELY
- 📁 Folder: /content/sample\_data/safe\_save\_run1
- ➡ You can now move to MONKEY (NHP) data using these weights.

```
[saved figure → /content/sample_data/figs/1_gt.png]
[saved figure → /content/sample_data/figs/3_gnn.png]
[saved figure → /content/sample_data/figs/1_gt_overlay.png]
[saved figure → /content/sample_data/figs/3_gnn_overlay.png]
```

```
!python /content/sample_data/gnn_silencemap.gnn.py \
--use_mat \
--headmodel_path /content/sample_data/cortex_downsampled_v73.mat \
--save_figs \
--fig_dir /content/sample_data/figs_monkey/
```

```
Device: cpu
Loading cortex vertices from .mat using smart_loadmat ...
[smart_loadmat] Loading /content/sample_data/cortex_downsampled_v73.mat ...
[smart_loadmat] SciPy cannot read this file (likely v7.3): Please use HDF reader for matlab v7.3 files, e.g. h5py
[smart_loadmat] mat73 is not installed; skipping mat73.
[smart_loadmat] Falling back to raw h5py read ...
[smart_loadmat] Loaded keys (first 10): ['#refs#/a', 'DownsampledCortex/F', 'DownsampledCortex/V', 'DownsampledCortex/labels']
Loaded vertices: (3004, 3)
```

```
Simulating 3 silent regions on cortex
Total silent nodes ≈ 30 (1.00% of 3004)
Target per-region size ≈ 10
Region 0: center index = 268
Region 0: picked 10 vertices
Region 1: center index = 2327
Region 1: picked 10 vertices
Region 2: center index = 1963
Region 2: picked 10 vertices
Final |S| = 30 silent nodes (1.00% of cortex)
mean(beta silent): 0.0
mean(beta active): 1.0
beta shape: (3004,)
beta min/max: 0.0 1.0
Building kNN graph ...
Solving Laplacian smoother ...
[auto] q_silent set to 1.00% for |S|=30/3004
Laplacian: P=0.967 R=0.967 F1=0.967
Building GNN ...
Training supervised GNN ...
[0200] loss=0.02213 cls=0.01723 smooth=0.00490
[0400] loss=0.02055 cls=0.01669 smooth=0.00386
[0600] loss=0.02025 cls=0.01596 smooth=0.00430
[0800] loss=0.02019 cls=0.01607 smooth=0.00412
[1000] loss=0.02022 cls=0.01606 smooth=0.00417
[1200] loss=0.02028 cls=0.01666 smooth=0.00362
[1400] loss=0.02009 cls=0.01561 smooth=0.00448
[1600] loss=0.02008 cls=0.01597 smooth=0.00411
[1800] loss=0.02059 cls=0.01651 smooth=0.00408
[2000] loss=0.02005 cls=0.01583 smooth=0.00422
GNN:      P=0.800 R=0.800 F1=0.800

[beta]
mean(silent): 0.0
mean(active): 1.0
corr with X_act: -1.0

[g_lap]
mean(silent): 0.23411522805690765
mean(active): 0.9662275910377502
corr with X_act: -0.7584455203423163

[g_gnn_prob]
```

```
mean(silent): 0.2554951310157776
mean(active): 0.001517578843049705
corr with X_act: 0.8234944564547522
[saved figure → /content/sample_data/figs_monkey/1_gt.png]
[saved figure → /content/sample_data/figs_monkey/2_laplacian.png]
[saved figure → /content/sample_data/figs_monkey/3_nn.png]
```

```
import h5py
import numpy as np
from pathlib import Path

leadfield_in = "/content/sample_data/OT_leadfield_symmetric_3002-128.mat" # change
headmodel_in = "/content/sample_data/OT_headmodel_symmetric_3002-128.mat" # or your OT_headmodel_*.mat
scale = 1000.0 # meters -> millimeters

def copy_with_mm(in_path):
    in_path = Path(in_path)
    out_path = in_path.with_name(in_path.stem + "_mm" + in_path.suffix)

    def needs_scale(arr):
        # heuristic: meters ~ 0.05-0.2, mm ~ 50-200
        m = np.nanmax(np.abs(arr))
        return m < 5.0

    with h5py.File(in_path, "r") as fin, h5py.File(out_path, "w") as fout:

        def recurse_copy(name, obj):
            if isinstance(obj, h5py.Group):
                fout.require_group(name)
            elif isinstance(obj, h5py.Dataset):
                data = obj[...]
                # scale only geometry datasets (never scale L)
                if name in ["src_xyz", "sensor_locs",
                            "Cortex/vertices", "DownsampledCortex/V"] and needs_scale(data):
                    data = (data * scale).astype(obj.dtype)
                    print(f"[scaled] {in_path.name}: {name}")
                fout.create_dataset(name, data=data, dtype=obj.dtype)

        fin.visititems(recurse_copy)
```

```
print("Wrote:", out_path)
return str(out_path)

leadfield_mm = copy_with_mm(leadfield_in)
headmodel_mm = copy_with_mm(headmodel_in)

print("Use these paths now:")
print("leadfield:", leadfield_mm)
print("headmodel:", headmodel_mm)

[scaled] OT_leadfield_symmetric_3002-128.mat: sensor_locs
[scaled] OT_leadfield_symmetric_3002-128.mat: src_xyz
Wrote: /content/sample_data/OT_leadfield_symmetric_3002-128_mm.mat
[scaled] OT_headmodel_symmetric_3002-128.mat: Cortex/vertices
Wrote: /content/sample_data/OT_headmodel_symmetric_3002-128_mm.mat
Use these paths now:
leadfield: /content/sample_data/OT_leadfield_symmetric_3002-128_mm.mat
headmodel: /content/sample_data/OT_headmodel_symmetric_3002-128_mm.mat
```