

Reproduction Guide

This document provides detailed instructions for reproducing the key results and figures of the manuscript “Explainable Generative 3D Design Driven by Social Media Semantics” (under review at Nature Communications).

1. Environment

- Python \geq 3.9
- Install dependencies: `pip install -r requirements.txt`
- Download models/checkpoints: `bash scripts/download_models.sh`

GPU acceleration is recommended but not mandatory for minimal demos.

2. Minimal demo

Reproduce pipeline flow and sample outputs with provided sample data:

```
python tools/demo.py --mode ti2d --config configs/cox3d_demo.yaml
--input_text "retro-futuristic hair dryer with matte finish"
--input_image examples/sample_data/images/ref_01.jpg
--save_cam --multi_view 4
```

Outputs will be under `runs/demo/`.

Expected structure:

```
runs/demo/
├─ images/      # multi-view candidates, heatmaps
├─ meshes/      # OBJ/PLY 3D meshes
└─ reports/     # JSON metrics, logs
```

3. Figure mapping

- **Fig. 3 (Pipeline overview):** Provided as `docs/figures/pipeline.drawio`. Export as PNG/SVG.
 - **Fig. 5 (Explainable optimization):** Use sample data + `--save_cam --multi_view 4`.
 - **Supplementary figures:** See `configs/demo_*` and rerun with varied prompts.
-

4. Table mapping

- **Main metrics table:** Run metrics collector on demo outputs:

```
python tools/collect_metrics.py --runs runs/demo --out runs/demo/report.json
```

- **Ablation study (Table X):**

```
# Example: diffusion only
python tools/demo.py --config configs/ablation/diffusion_only.yaml

# Example: diffusion + NeRF
python tools/demo.py --config configs/ablation/diffusion_nerf.yaml

# Collect results
python tools/collect_metrics.py --runs runs/ablation --out runs/ablation/report.json
```

Requires COCO/KITTI subsets downloaded separately.

5. Seeds & configs

- Default seed: 42
- All configs are stored under configs/
- Each config corresponds to experiments referenced in the paper. Use exactly these configs to minimize variance.

6. Hardware notes

- Development/tested on:
- Windows 10 (Python 3.9, RTX 3080, CUDA 11.8)
- Ubuntu 20.04 (Python 3.10, A100, CUDA 12.0)
- Typical runtime (demo): ~2–3 minutes on RTX 3080, ~10 minutes CPU-only.

7. Known issues

- Missing model weights: ensure scripts/download_models.sh completes.
- Memory errors: reduce batch size in configs/*.yaml.
- CUDA mismatch: check torch and cuda versions with python -m torch.utils.collect_env.

8. Non-reproducible elements

- **User study data:** summarized in manuscript; raw data not redistributed.
- **Full training on COCO/KITTI:** not provided; requires downloading full datasets.

9. Contact

For questions or help reproducing results: - Zhenyu Wang (Tongji University) - Email: 2411673@tongji.edu.cn - ORCID: 0000-0001-7645-5667