ASE, EFM3D & EVL: Datasets, Models & Tools for NBV

Towards Relative Reconstruction Metrics for Next-Best-View

Jan Duchscherer

VCML Seminar WS24/25

Aria Synthetic Environments

Dataset for Egocentric 3D Scene Understanding

Figure 1: [Ave+24]

ASE Dataset Overview

Dataset Content

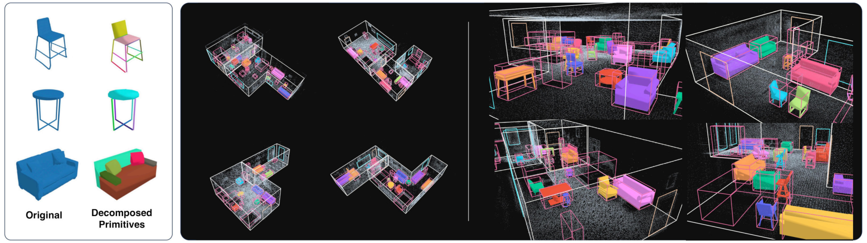
- 100,000 unique multi-room interior scenes
- ~2-min egocentric trajectories per scene
- Populated with 8,000 3D objects
- Aria camera & lens characteristics

Ground Truth Annotations

- 6DoF trajectories
- RGB-D frames
- 2D panoptic segmentation
- Semi-dense SLAM PC w/ visibility info
- 3D floor plan (SceneScript SSL format)
- GT meshes as `.ply` files

Key Resources

- Project Aria Tools for data access
- ASE documentation [Ave+24, Met25a]



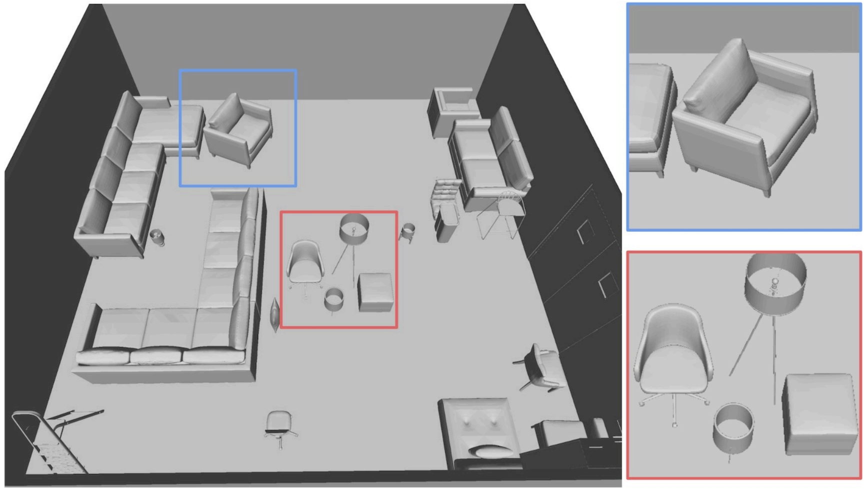


Figure 2: [Ave+24]

ASE Dataset Structure

```
scene_id/
├── ase_scene_language.txt          # Ground truth scene layout in SSL format
├── object_instances_to_classes.json # Mapping from instance IDs to semantic classes
├── trajectory.csv                   # 6DoF camera poses along the egocentric path
├── semidense_points.csv.gz          # Semi-dense 3D point cloud from MPS SLAM
├── semidense_observations.csv.gz    # Point observations (which images see which points)
├── rgb/                             # RGB image frames
│   ├── 000000.png
│   └── ...
├── depth/                           # Ground truth depth maps
│   ├── 000000.png
│   └── ...
└── instances/                       # Instance segmentation masks
    ├── 000000.png
    └── ...
```

EFM3D Benchmark

3D Egocentric Foundation Model:
Egocentric Voxel Lifting (EVL)

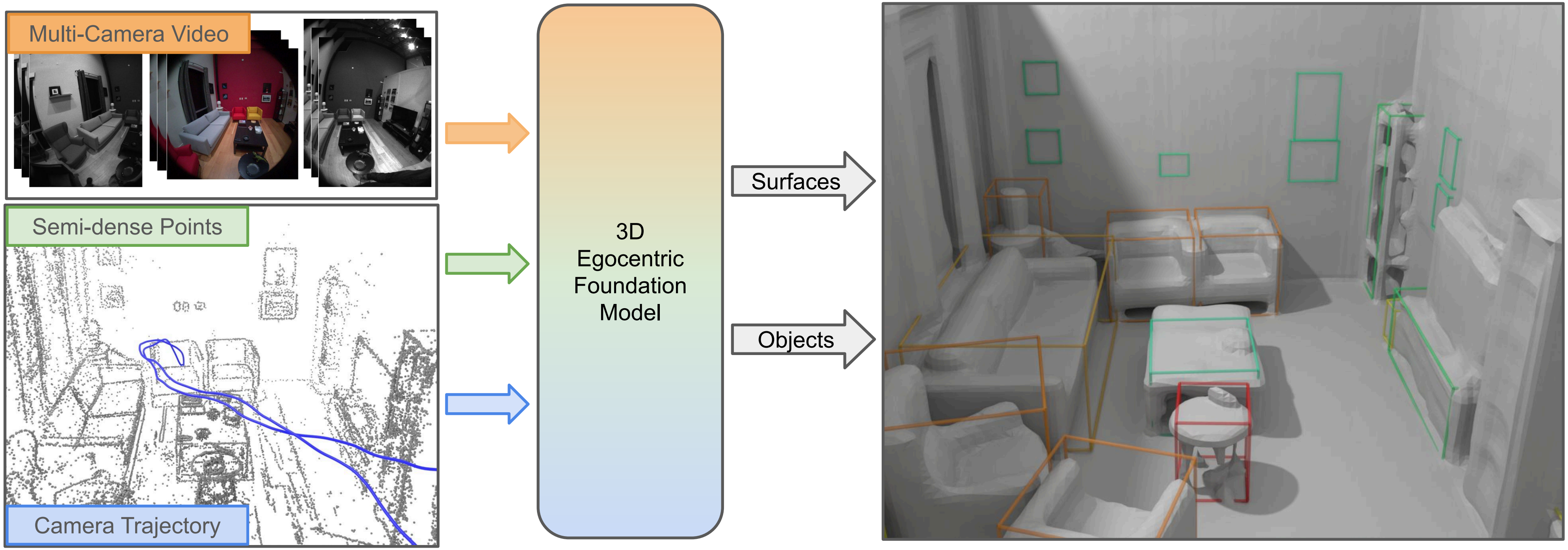


Figure 3: [Str+24]

EFM3D & EVL

EFM3D Tasks

- 3D object detection
- 3D surface regression (occupancy volumes)
  - on ASE, ADT[1], AEO[2] datasets

EVL Architecture

- Utilizes all available egocentric modalities:
  1. multiple (rectified) RGB, grayscale, and semi-dense points inputs
  2. camera intrinsics and extrinsics
- 16.7M trainable + 86.6M frozen params
- Inherits foundational capabilities from frozen 2D model (DinoV2.5) by lifting 2D features to 3D [Str+24]

[1] Aria Digital Twin

[2] Aria Everyday Objects: small-scale, real-world w/ 3D OBBs

EFM3D & EVL

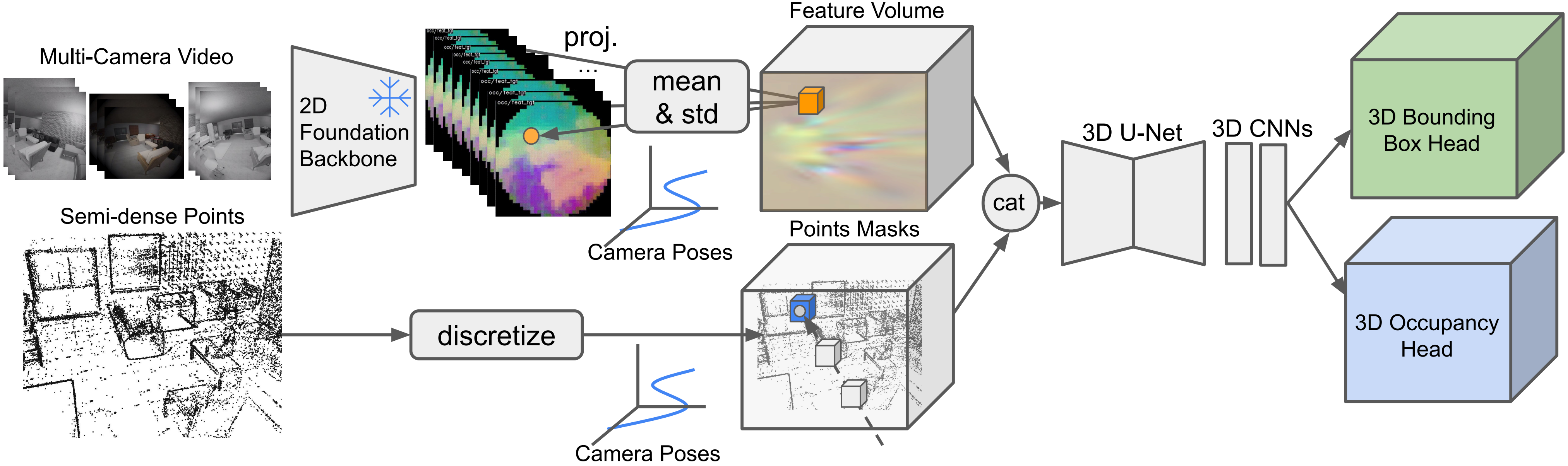


Figure 4: [Str+24]

ATEK Toolkit

Streamlined ML Workflows for Aria Datasets

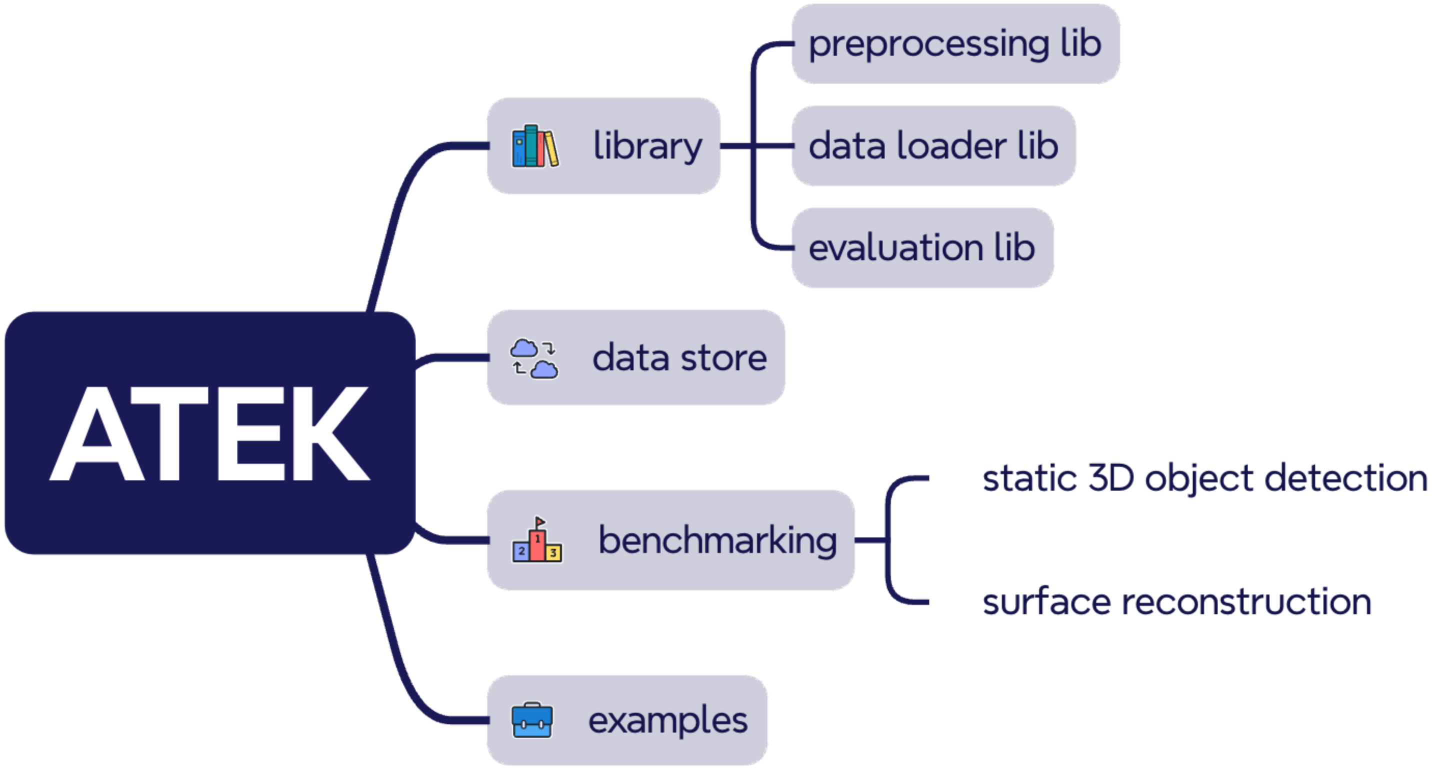


Figure 5: [Met25b]

ATEK Toolkit

ATEK Data Store

- Pre-processed for various tasks $\rightarrow$ ready for PyTorch training
- Local download or cloud streaming
- Eval metrics (accuracy, completeness, F-score) $\rightarrow$ adaptation for RRI
- Integration w/ Meta's MPS
- Various example notebooks

Provided Models

- Cube R-CNN [Bra+23] for OBBs, EVL [Str+24] for OBBs & surface reconstruction

Resources

- ATEK GitHub
- ECCV 2024 Tutorial: Egocentric Research with Project Aria

VIN-NBV: Learning-Based Next-Best-View

Key Innovation [Fra+25]

- First NBV method to directly optimize **reconstruction quality** (not coverage)
- Predicts **Relative Reconstruction Improvement (RRI)** without capturing new images
- 30% improvement over coverage-based baselines
- Trained 24h on 4 A6000 GPUs (no pre-trained backbone)

Relative Reconstruction Improvement (RRI)

For a candidate view q, RRI quantifies expected improvement:

$$\mathrm{RRI}(q) = \frac{\mathrm{CD}(\mathcal{P}_{\text{base}}, \mathcal{P}_{\text{GT}}) - \mathrm{CD}\big(\mathcal{P}_{\text{base} \cup q}, \mathcal{P}_{\text{GT}}\big)}{\mathrm{CD}(\mathcal{P}_{\text{base}}, \mathcal{P}_{\text{GT}})}$$

- Range: $[0, 1]$ where higher = better view
- Normalized by current error $\longrightarrow$ scale-independent
- Chamfer Distance (CD) measures reconstruction quality

VIN Architecture

Predicts RRI from current reconstruction state:

VIN-NBV: Learning-Based Next-Best-View

$$\widehat{\mathrm{RRI}}(q) = \mathrm{VIN}_\theta\left(\mathcal{P}_{\text{base}}, C_{\text{base}}, C_q\right)$$

- Input: RGB sequence, partial point cloud + camera poses
- Features: Surface normals, visibility counts, depth
- Main Idea: Project features to candidate view q, compute fitness score for each candidate
- Output: RRI ranking via ordinal classification

RRI-based NBV with ASE, EFM3D & ATEK

- Use ASE visibility data + GT meshes for oracle RRI and visibility count
- Maybe compute RRI separately for each entity (walls, doors, objects) to allow semantic weighting
- Use EVL as scene encoder

Pipeline

1. **Scene Encoding**:
   - Sample random point t in ASE trajectory as starting pose
   - Get partial PC, camera poses and RGB-D frames $(C, \mathcal{P}_{\text{base}}, I_{\text{RGB}-D})^{1:t}$ up to t from historical trajectory
   - Use EVL to encode current scene observation

3. **Sample**: Generate candidate viewpoint pool around last pose

4. **Predict**: Use scene encodings + candidate view encoding to predict RRI per candidate
   - freeze EVL weights, only train VIN head

RRI-based NBV with ASE, EFM3D & ATEK

ATEK Integration

- GT meshes enable oracle RRI computation (training labels)
- Mesh-based metrics (accuracy, completeness, F-score) for evaluation
- Pre-processed data splits for model training

Key Challenges

- Ray-casting from candidate views to compute visibility and $\mathscr{P}_{\text{base} \cup q}$ from GT meshes
- Multi-entity scenes vs. VIN-NBV's single-object focus $\Rightarrow$ compute?
- Projection of features to candidate views? Is this explicit $\mathrm{SE}(3)$ tf actually necesary?

Next Steps & TODOs

Literature Review

- Read Project Aria paper [Met25a]
- Study EFM3D & EVL in depth [Str+24]
- Reread VIN-NBV and GenNBV approach to get in-depth understanding of potential metrics and loss functions
- Mesh to distance field conversion / Distance to mesh surface as as metric?
- Is CD dependent on the density of the point cloud?

Technical Exploration

- Explore GT meshes (`.ply` files) in ASE dataset
- Get familiar with ATEK and ATEK Data Store
- Test mesh-based evaluation metrics

Implementation Goals

- Implement ray-casting/rendering for candidate views
- Develop RRI computation pipeline using GT meshes

Bibliography

[Ave+24] A. Avetisyan et al., "SceneScript: Reconstructing Scenes With An Autoregressive Structured Language Model." [Online]. Available: https://arxiv.org/abs/2403.13064

[Met25] Meta Platforms Inc., "Aria Synthetic Environments Dataset." [Online]. Available: https://facebookresearch.github.io/projectaria_tools/docs/open_datasets/aria_synthetic_environments_dataset

[Str+24] J. Straub, D. DeTone, T. Shen, N. Yang, C. Sweeney, and R. Newcombe, "EFM3D: A Benchmark for Measuring Progress Towards 3D Egocentric Foundation Models." [Online]. Available: https://arxiv.org/abs/2406.10224

[Met25] Meta Platforms Inc., "Aria Training and Evaluation toolkit (ATEK)." [Online]. Available: https://github.com/facebookresearch/ATEK

[Bra+23] G. Brazil, A. Kumar, J. Straub, N. Ravi, J. Johnson, and G. Gkioxari, "Omni3D: A Large Benchmark and Model for 3D Object Detection in the Wild." [Online]. Available: https://arxiv.org/abs/2207.10660

[Fra+25] N. Frahm et al., "VIN-NBV: A View Introspection Network for Next-Best-View Selection." [Online]. Available: https://arxiv.org/abs/2505.06219