

Reading status. The detailed review below preserves the earlier literature synthesis and its proposed ARIA-NBV adaptations. Recommendations about model inputs, training stages or implementation are research proposals, not a description of the current system or evidence of measured benefit. The active thesis and implementation specify the current choices.

EFM3D and EVL: Egocentric 3D State Substrate

Primary source. EFM3D: A Benchmark for Measuring Progress Towards 3D Egocentric Foundation Models [1].

Local source. `main.tex`, `method.tex`, `experiments.tex`, and `supplemental_text.tex`.

Related ARIA-NBV pages. Project Aria, ASE dataset, EFM3D scene embeddings, VIN model API, and VIN feature contracts.

Core contribution

Egocentric Foundation Model 3D (EFM3D) supplies the actor-visible 3D representation that ARIA-NBV builds on. Its Egocentric Voxel Lifting (EVL) architecture lifts frozen 2D image features from posed, calibrated Project Aria (Project Aria) streams into a local gravity-aligned voxel grid and trains heads for 3D surface regression and Oriented Bounding Box (OBB) detection [1].

The two evidence streams meet in the voxel volume shown in Figure 1.

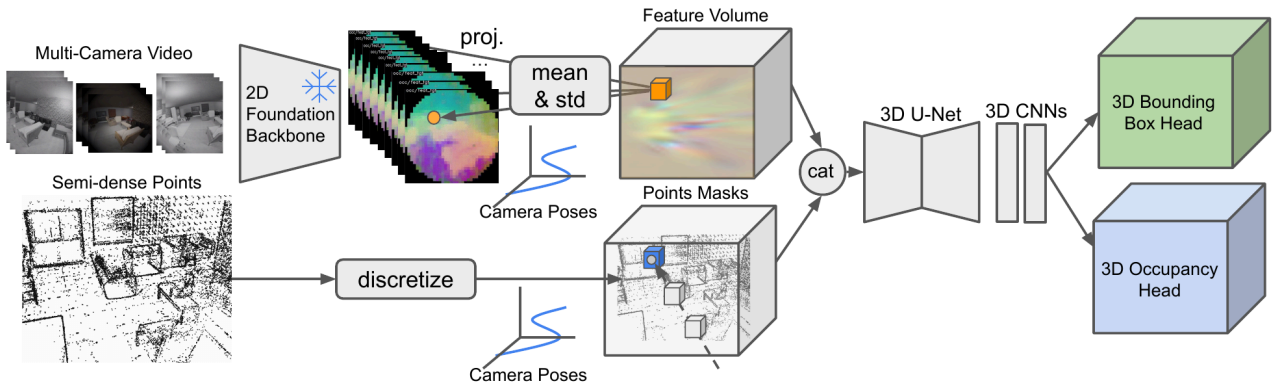


Figure 1: EVL combines lifted image features with point-derived masks before occupancy and box prediction. Reproduced from [1, Fig. 2], with surrounding whitespace removed.

For ARIA-NBV, EFM3D is not a planner and its final voxel heads are not the whole scene memory. It is an actor-visible evidence and target-support provider: local EVL features, head outputs, and predicted OBBs can be combined with broader semidense or fused point evidence for target-conditioned candidate scoring.

The relevant planning invariant is therefore not “dense voxels everywhere.” It is a target-conditioned, actor-visible state that preserves which target is being improved, which finite candidates are valid, what surface/free/unknown evidence exists around the target and each candidate query, and which parts of that evidence were actually observable from the rollout history. Target proposal and scene memory are separate interfaces: EVL provides Aria-native local target proposals, while persistent memory can be a ray-aware sparse map derived from actor-visible observations.

Verified paper signals

signal	source-backed detail	ARIA-NBV relevance
Egocentric inputs	EVL consumes synchronized RGB/SLAM camera snippets, camera calibration, poses, and semi-dense point evidence.	Defines the actor-visible evidence contract for scoring candidate views.
Voxel lifting	Frozen DINOv2.5 features are lifted into a local gravity-aligned voxel grid anchored to the snippet pose.	Provides local evidence and target support, with known local-extent limits.
Geometry priors	Semi-dense point and free-space masks are injected into the voxel representation.	Gives interpretable known-surface, known-free, and observation-support channels for NBV diagnostics.
Surface head	The surface-regression head predicts voxel occupancy/surface evidence.	Useful as a compact reconstruction proxy for scene context.
OBB head	The detection head predicts centerness, seven-parameter gravity-aligned boxes (three dimensions, three centre offsets, and yaw), class scores, and sparse post-processed OBBs.	Supports observed/predicted target selection without exposing GT boxes to the actor.
Dataset scale	The paper trains/evaluates on Aria Synthetic Environments (ASE) and Project Aria-related real datasets, with ASE providing GT meshes/annotations for supervised tasks.	Confirms the substrate for oracle labels and observed-state realism.

Ranked backbone requirements

The backbone for finite-candidate Q_H should be selected by the invariants it preserves, not by whether it looks like a full 3D reconstruction.

rank	requirement	representation consequence
1	Actor-visible target identity and support.	The state must expose a target hypothesis from observed or predicted OBBs, class/confidence, projected support, semidense support, and EVL support. GT OBBs, meshes, and target crops remain label/evaluation surfaces.
2	Candidate-row permutation equivariance with hard mask isolation.	Candidate scoring should treat the finite candidate table as an unordered masked action set. The first value model should score each physical candidate through shared candidate-to-state queries; Deep Sets or masked Set Transformer context are ablations once independent calibration is stable [2], [3].
3	Local relative geometry, not global pose memorization.	Candidate, target, and support tokens should encode target-local, current-camera-local, and query-local relative poses, gravity alignment, ranges, bearings, and frustum/OBB intersections. Full SE(3) invariance is not the right constraint because gravity, egocentric camera history, and object orientation are meaningful.
4	Scene support beyond EVL’s fixed local cube.	EVL should be a local read and evidence source; broad scene memory should come from semidense/fused observations accumulated into surface, free-space, and unknown evidence so valid far candidates are not discarded merely because they lie outside the current EVL volume.
5	Candidate visibility, not only frustum membership.	Candidate features should be produced by a ray- or render-like query over the current actor-visible map. A point inside a candidate frustum is not necessarily visible, and a valid projection into a logged image is not by itself an observation.
6	Logged visual semantics without future-observation leakage.	DINO-derived descriptors can be attached only from logged frames, native observation lineage, or depth-consistent logged projections. Counterfactual candidate states do not get freshly rendered RGB/DINO features unless a separate synthetic-rendering protocol is explicitly introduced.
7	Directional history and uncertainty.	The value model needs to know which directions around the target have already been observed, which candidate rays are supported, and which evidence is low-support or out-of-extent. This separates “unknown” from “known bad” and prevents EVL coverage gaps from becoming false invalidity.
8	Provenance and compression.	Feature banks must carry checkpoint/config/source hashes, point or voxel lineage, support counts, and descriptor dimensionality. Raw high-dimensional DINO features are an ablation surface, not the default offline-store contract.

Representation ladder for ARIA-NBV

The recommended first serious representation is a sparse scene and target-pooled state:

$$s_t^{\text{actor}} = \left(P_t^{\text{semi/fused}}, M_t^{\text{ray}}, F_t^{\text{DINO@pt}}, V_t^{\text{EVL}}, O_t^{\text{pred}}, \phi_e, \mathcal{Q}_t, m_t, \rho_t, h_t, b_t \right),$$

where $P_t^{\text{semi/fused}}$ is broad scene geometry, M_t^{ray} is a derived sparse occupied/free/unknown evidence map, $F_t^{\text{DINO@pt}}$ is an optional compressed descriptor bank attached to genuinely observed logged semidense or fused points, V_t^{EVL} is the local EVL evidence field, O_t^{pred} are actor-visible detections, ϕ_e is the selected target descriptor, $\mathcal{Q}_t$ is the finite candidate set, m_t are validity masks, ρ_t are invalidity or low-support reasons, h_t is selected rollout history, and b_t records backbone/provenance metadata.

For each target e and candidate i , the useful pooled tokens are:

- **Target support pool g_e^{tgt} :** semidense/fused points inside or near the actor-visible OBB, OBB geometry, class/confidence, projected area, observed support, EVL support, and optional compressed DINO descriptors sampled from logged frames.

- **Candidate-frustum support pool $g_{t,i}^{\text{fr}}$** : a support baseline over points and evidence inside the candidate frustum, not a visibility model by itself.
- **Target-candidate intersection pool $g_{t,e,i}^{\cap}$** : support and descriptors in the candidate frustum that intersect the target OBB or target-local neighborhood.
- **Ray-aware candidate query**: nearest observed surface, free-space length, unknown-space length, hit/empty mask, target-membership weights, and uncertainty rendered from the current actor-visible map into the candidate camera.
- **EVL local read**: voxel/head/neck/crop summaries near the target and candidate rays when the query is inside the current EVL support volume; otherwise an explicit out-of-extent or low-coverage flag.

This makes the global dense field optional. The first persistent memory should be a sparse ray-aware evidence map, because it preserves occupied, free, and unknown space before any appearance descriptor is added. Point and sparse backbones such as KPConv, Minkowski sparse convolutions, or Point Transformer variants become natural later ablations for encoding the point/descriptor bank [4], [5], [6]. They should compete against simple pooled descriptors first, because the immediate research risk is leakage, support mismatch, visibility ambiguity, and candidate masking rather than insufficient network capacity.

Salvageable EFM3D internals

EFM3D is most useful as a feature/evidence source with a known local support envelope:

- **Keep as mandatory actor-visible substrate**: predicted/observed OBBs, class scores, centerness, detection confidence, local EVL extent, camera calibration, pose lineage, and support metadata. These define target proposals and local evidence without using oracle GT.
- **Harvest as high-ROI scene features**: semidense point and free-space masks, support counts, local voxel occupancy/surface logits, and EVL coverage flags. These are interpretable and directly diagnose why a candidate is uncertain or unsupported.
- **Build ray-aware memory before appearance memory**: accumulate actor-visible surface, free-space, unknown, support-count, uncertainty, and directional-history channels before treating DINO as the missing scene representation.
- **Harvest DINO only through observation visibility**: sample frozen DINO maps or upsampled feature maps only from logged frames where a point has native observation lineage or depth-consistent visibility, attach compressed descriptors to semidense or fused world points, and pool those descriptors by target OBB, candidate query, and their intersection. This is the cleanest way to use EFM3D’s image-foundation signal outside the fixed EVL cube without confusing projection validity with visibility.
- **Use pre-head EVL features as ablations**: pooled reads from lifted voxel features, neck occupancy features, OBB-head features, and target/candidate crops can test whether EVL internals outperform simple point support. They should carry extent/coverage metadata so out-of-bounds queries are explicit.
- **Defer global radiance fields and Gaussian splats**: they may become useful visualization or future scene-memory baselines, but the EFM3D supplemental evidence already flags dynamic-scene artifacts and false surfaces. They do not solve the immediate actor-visible Q-state contract by themselves.

The limited EVL voxel extent should be handled by a three-tier state rather than by pretending the local cube is global: EVL provides local high-quality target/support evidence, a sparse ray-aware map provides persistent occupied/free/unknown memory, and logged DINO-on-point descriptors add optional appearance evidence after visibility gating. The projection and pooling contract for those logged descriptors is defined in the EFM3D scene-embedding theory note. Out-of-EVL does not imply invalid; it implies missing EVL support and should be encoded as coverage/reason metadata unless the evaluator proves the candidate is physically impossible.

Tractable scene-encoding optimum

The tractable optimum for ARIA-NBV is not to store the largest possible EVL tensor. It is to store the smallest actor-visible state that preserves the task invariants:

1. target identity and target support;
2. broad occupied/free/unknown geometry;
3. candidate-conditioned visibility and directional history;
4. local EVL evidence and extent diagnostics;
5. optional compressed logged appearance descriptors with provenance.

The resulting state has three layers. First, EVL remains the Aria-native local evidence and target-proposal substrate: predicted OBBs, class scores, centerness, occupancy evidence, lifted voxel/neck reads, support counts, voxel pose, and finite extent. Second, a sparse ray-aware map accumulates semidense or fused actor-visible observations into surface, free-space, unknown, uncertainty, support-count, last-seen, and directional-history channels. Third, a point or cell feature bank may attach compressed DINO descriptors sampled only from logged frames after visibility gating.

This design is cheaper and better matched to the value problem than naively using only final EVL heads. The head fields `occ_pr`, `cent_pr`, `bbox_pr`, and `clas_pr` are a good baseline, but they are local, task-collapsed predictions. They do not preserve enough history, free/unknown evidence, visibility, target-candidate relations, or appearance ambiguity to be the only persistent state for finite-horizon Q_H .

Simply extending the EVL voxel area is therefore a diagnostic ablation, not the default solution. EFM3D supports configurable voxel volumes and the external repo includes inference-side OBB tracking and volumetric fusion, but dense 3D cost grows cubically, a larger root cube still lacks counterfactual RGB/DINO evidence, and global fusion still needs to be filtered into target/candidate/provenance queries. Larger EVL support should answer whether the current local cube is the bottleneck; it should not replace the sparse actor-visible memory contract.

The computational pipeline is:

1. run or load EVL on the logged root snippet;
2. extract actor-visible target hypotheses and local evidence with extent/support metadata;
3. collapse semidense or fused points with uncertainty, observation counts, and lineage;
4. build a sparse ray-aware occupied/free/unknown memory;
5. optionally sample logged DINO features at visible points or cells and compress them;
6. query target pools, candidate-frustum support, target-frustum intersections, ray-aware candidate observations, and local EVL reads;
7. feed typed pooled summaries into candidate-to-state Q_H rows with hard validity masks and explicit low-support reasons.

Default stores should persist sparse geometry and pooled summaries. Raw dense EVL feature volumes, raw 768-dimensional DINO per point, all-candidate renderings, larger EVL cubes, neural fields, and Gaussian splats are heavier ablation artifacts unless they are shown to improve the actor-visible target-RRI objective under the same leakage policy.

Cube R-CNN role

Cube R-CNN is useful to ARIA-NBV as a detector and target-proposal baseline, not as the primary scene representation [7]. Its strengths are practical:

- It predicts 3D OBBs from single RGB images, so it can test how far a simpler detector-only target proposal pipeline can go without EVL.
- It is easier to isolate as an RGB detector ablation, including through the ATEK Cube R-CNN preprocessing/adaptor surface.
- Its ROI and detection features can be pooled into a target descriptor baseline when the question is target identity/support, not full scene memory.
- It avoids EVL’s local voxel extent by construction, which makes it a useful stress test for target proposal coverage.

Its limits are equally important. Single-frame Cube R-CNN predictions do not give persistent scene memory, semidense/free-space support, local voxel evidence, or logged-history descriptors. Accumulating per-frame boxes still needs tracking and duplicate/noise handling. For this thesis, Cube R-CNN is therefore an auxiliary probe for actor-visible OBB proposals and ROI features; it is not a replacement for the EFM3D-plus-semidense scene substrate.

ARIA-NBV adoption

- **Core substrate:** use EVL/EFM3D outputs as local actor-visible evidence and target support for one-step RRI scoring, rollout state summaries, and target-conditioned candidate ranking.

- **Scene memory hypothesis:** first test semidense/fused observations and sparse ray-aware occupied/free/unknown evidence as the broader queryable scene memory; then add compressed logged-DINO descriptors as an appearance ablation.
- **Target contract:** use predicted or observed OBB support as actor-visible target input; keep GT meshes and GT target crops for offline labels/evaluation only.
- **Candidate contract:** score finite candidate rows with hard validity masks, target-local geometry, ray-aware candidate queries, target-candidate support pools, and explicit support/coverage reasons.
- **Backbone ladder:** start with pooled support/descriptor tokens and independent candidate-to-state queries, then add Deep Sets, masked Set Transformer, sparse/point backbones, or richer cross-candidate attention only if simpler controls fail.
- **Diagnostics:** preserve frame lineage, voxel extent, camera stream identity, calibration, and Machine Perception Services (MPS) / ASE point-source metadata in offline stores.
- **Storage policy:** keep full EVL internals, raw logged visual descriptors, dense voxel/world grids, and class-volume tensors for named ablations or reproducibility-critical diagnostics; use pooled summaries, sparse geometry, and compressed point descriptors as the default tractable substrate.
- **Implementation boundary:** code-path anchors and cache-reader details live in the EFM3D scene-embedding implementation notes.
- **Free-space policy:** native `voxel/free_input` is useful when present, but current VIN code can derive a proxy from observed counts and occupied input. Do not report derived free space as native EVL free-space evidence without a provenance flag.
- **Extent policy:** low EVL support, missing descriptors, and out-of-EVL extent are maskable support features or diagnostic reasons. They are hard invalidity only when they block physical evaluation, state transition, or target-crop scoring.

Do not adopt

- Do not treat EVL predictions as ground truth; they are actor-visible evidence.
- Do not treat EVL’s fixed local voxel extent as the complete scene extent. Candidates can be partially out of bounds even when valid in the scene.
- Do not treat a larger EVL cube as the main scene-memory fix; use it as an ablation against sparse ray-aware memory.
- Do not mix offline GT meshes/OBBs into actor-facing target selection, target descriptors, or scorer inputs for the main protocol. GT assets may appear only in offline target-task identity gates, label generation, privileged teachers, or evaluation products with explicit provenance.
- Do not generate fresh DINO descriptors for arbitrary counterfactual candidate poses unless that becomes an explicit synthetic-rendering protocol.
- Do not treat the current logged-feature sampling helper as a completed feature-cache/training system; persisted DINO-on-point and EVL-internal feature banks remain representation ablations until backed by stored artifacts and readers.
- Do not treat Cube R-CNN as a full scene encoder; it is a detector/ROI-feature baseline.
- Do not keep implementation debug dictionaries in this literature page; feature-selection details belong in generated VIN API docs and code docstrings.

Open risks / caveats

- Local voxel anchoring and coordinate-frame conventions can silently corrupt projection features and RRI labels.
- Dynamic objects, reflections, distant surfaces, and incomplete semi-dense support can weaken EVL evidence.
- EVL features should be logged with explicit config/checkpoint/source hashes because rollout/Q datasets will otherwise mix incomparable backbone states.
- Descriptor compression may erase small-object or hard-turn target cues; keep raw-feature probes as bounded diagnostics before committing a low-dimensional store format.

Terminology

Dataset

ASE – Aria Synthetic Environments: Synthetic indoor dataset with simulated Aria observations, trajectories, and scene annotations. The thesis uses ASE reference meshes and rendered depth to measure target

reconstruction improvement. Semidense actor evidence is a separate input, not the default oracle reconstruction source.

MPS – Machine Perception Services: Project Aria services that derive poses, mapping, and other perception outputs from recordings. MPS-style poses and semidense points describe available sensing products. They should not be confused with the ASE-depth reconstruction used by the current oracle evaluation.

Project Aria – Project Aria: Meta's egocentric research-device and tooling ecosystem for calibrated, time-aligned multimodal sensing. ARIA-NBV treats Project Aria and its MPS-style products as the actor-visible sensing contract, while ASE meshes remain offline supervision and evaluation assets.

Geometry

OBV – Oriented Bounding Box: Box described by a center, orientation, and three extents. It approximates an object extent and defines a local target frame. Semantic class and confidence can accompany the detection but are not geometric properties of the box itself.

Model

EFM3D – Egocentric Foundation Model 3D: Egocentric 3D perception framework used to extract spatial evidence from observations. Its EVL architecture supplies frozen features and geometric predictions to the candidate-value model; the perception backbone and the learned planning head serve different roles.

EVL – Egocentric Voxel Lifting: Architecture that lifts synchronized egocentric image evidence into a gravity-aligned voxel feature volume. EVL belongs to the EFM3D framework. Its features and prediction heads provide observed spatial evidence; they do not by themselves supply observations from unvisited candidate poses.

Bibliography

- [1] 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>
- [2] M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. Salakhutdinov, and A. J. Smola, “Deep Sets,” in *Advances in Neural Information Processing Systems*, 2017. [Online]. Available: https://papers.nips.cc/paper_files/paper/2017/hash/f22e4747da1aa27e363d86d40ff442fe-Abstract.html
- [3] J. Lee, Y. Lee, J. Kim, A. R. Kosiorek, S. Choi, and Y. W. Teh, “Set Transformer: A Framework for Attention-based Permutation-Invariant Neural Networks,” in *Proceedings of the 36th International Conference on Machine Learning*, in Proceedings of Machine Learning Research, vol. 97. PMLR, 2019, pp. 3744–3753. [Online]. Available: <https://proceedings.mlr.press/v97/lee19d.html>
- [4] H. Thomas, C. R. Qi, J.-E. Deschaud, B. Marcotegui, F. Goulette, and L. J. Guibas, “KPConv: Flexible and Deformable Convolution for Point Clouds,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019. [Online]. Available: https://openaccess.thecvf.com/content_ICCV_2019/html/Thomas_KPConv_Flexible_and_Deformable_Convolution_for_Point_Clouds_ICCV_2019_paper.html
- [5] C. Choy, J. Gwak, and S. Savarese, “4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019. [Online]. Available: <https://arxiv.org/abs/1904.08755>
- [6] X. Wu *et al.*, “Point Transformer V3: Simpler, Faster, Stronger,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. [Online]. Available: <https://arxiv.org/abs/2312.10035>
- [7] 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>