Active Foveal 3-D Vision

A Fixate–Reconstruct–Saccade Playground
for 360-Degree Scene Reconstruction

A controlled proof of concept and reproducible computational baseline

Luiz Velho
VISGRAF Laboratory, IMPA

Developed in collaboration with
ChatGPT — GPT-5.6 Sol, OpenAI
Claude Code — Claude Opus 5.5, Anthropic

October 2026

TECHNICAL REPORT — REVIEW DRAFT

Abstract

Biological vision does not capture the world at full resolution all at once. A small, high-acuity fovea is moved by saccades, and a persistent percept is assembled over time. We study the computational counterpart of this observation: can a spatially selective binocular observer, with a fixed head, build a persistent metric 3-D representation of its whole surroundings by deciding where to look? We describe a deliberately small architecture built around one loop—fixate, reconstruct everything the fovea can measure, update a global memory, saccade—and a reproducible implementation of it, the Greedy Foveal Explorer. Each fixation measures a 12° binocular foveal core. Correspondence is supplied by a PERFECT / ORACLE service derived from the renderer, and metric points come from spherical epipolar triangulation into a fixed head frame. A persistent surfel map accumulates the geometry, and a 1° cyclopean sphere records which directions have been seen. Gaze selection uses only that sphere and the current fixation's evidence, not the accumulated map. In a static synthetic Blender Classroom, the explorer stopped itself after 542 fixations (322 local and 219 global saccades), with 99.00 % of the sphere seen and 98.28 % with reconstructed depth. Its 16,517,811-surfel map reconstructs 98.32 % of the first-hit reference solid angle within 12 mm; the reference was opened only after the run was frozen. An official reproduction is bitwise identical to the frozen baseline in every control-path product. We present the system as a controlled proof of concept and as an experimental playground, in which individual components can be replaced one at a time and compared against this baseline.

1 Introduction

A conventional camera tries to capture an entire field of view at uniform resolution in a single exposure. Animal vision works differently. Only a small central region of the retina resolves fine detail, and the eyes move several times per second to bring different parts of the scene onto it. What we experience as a stable, detailed, wide visual world is assembled over many brief fixations (Yarbus, 1967). This report asks whether that strategy can be turned into a working computational baseline for 3-D perception.

The question is therefore not

> *How can a camera capture everything at maximal resolution at once?*

but rather

> *Can a spatially selective binocular observer build a persistent 3-D representation by deciding where to look?*

The answer we pursue is an additive one:

$$\text{local high-quality measurement} \;+\; \text{eye movements} \;+\; \text{persistent memory} \\ = \text{virtual wide-field perception over time.}$$

A narrow fovea that is moved deliberately, and whose measurements are remembered in a common frame, behaves over time like a wide-field sensor, even though at no single moment does it see more than 12° of the world in detail.

We built the smallest architecture we could find that makes this concrete. A fixed binocular head sits in a static scene. At each step it fixates, reconstructs everything the fovea can match into metric 3-D, adds the result to one persistent map and to a coarse record of where it has looked, and chooses its next gaze with a simple greedy rule. Stereo correspondence is supplied by a PERFECT / ORACLE service derived from the renderer. This assumption is deliberate: it separates the questions of *exploration* and *accumulation*, which this report addresses, from the question of *natural stereo matching*, which it does not.

Contribution. The contribution is modest and, we hope, useful:

1. a clean active-foveal architecture with explicit persistent state and separable services (Sections 3–8);
2. a demonstration that, in a controlled synthetic setting, a very simple greedy gaze policy suffices to observe 99 % of the sphere and to reconstruct nearly the whole first-hit 360° scene metrically (Sections 10–11);
3. a reproducible computational baseline—official code, frozen run, bitwise reproduction and a truth firewall between control and evaluation—organized as a playground in which one component at a time can be replaced and measured (Sections 9 and 13).

We do not claim optimal gaze control, natural stereo, natural segmentation, or generalization beyond the scene studied here.

How claims are labelled. Numbers in this report are of four kinds. **MEASURED** values come from an identified run; unless stated otherwise, the official reproduction of Section 9.1. **PARAMETER** values are configured settings. **DERIVED** quantities and images are computed from run products. **REFERENCE / EVALUATION** denotes the independent Blender reference used only after a run is frozen. Information that comes from the renderer's internal geometry is called **ORACLE** and is used only where declared: for correspondence, and for visualization.

Outline. Section 2 places the work in the active-vision tradition. Sections 3–8 describe the architecture: the loop and its state, the geometry, a single fixation, the persistent map, the coverage memory and the gaze policy. Section 9 covers the implementation and its reproducibility. Sections 10 and 11 present the Classroom experiment and its results, Section 12 discusses them, Section 13 lays out the playground, and Sections 14 and 15 close.

2 Background and design motivation

Fixation, saccades and persistent perception. Yarbus's recordings of eye movements during picture viewing showed that gaze does not sweep a scene uniformly. It dwells on informative regions, in patterns that depend on the scene and on the observer's task, and perception is built across a sequence of fixations (Yarbus, 1967). Three ingredients of that picture shape the present design: a small region of high-quality sensing, discrete gaze shifts between fixations, and a percept that persists across them.

Active vision. Aloimonos et al. (1988) argued that an observer that controls its own sensor geometry can turn problems that are ill-posed or unstable for a passive observer into well-posed and stable ones. Sensing becomes part of a loop with action. Our explorer is active in exactly this sense: each measurement exists because the system chose to make it, and what it measures depends on where it chose to look.

Animate vision. Ballard (1991) argued that a system able to control its gaze can exploit fixation, and the coordinate frames it induces, to simplify visual computation, instead of reconstructing everything from a passive image stream. We adopt this computational stance. The local measurement is defined relative to the current fixation (a gaze-centred tangent frame and the eye pair's epipolar geometry), and only its result is moved into a common, persistent frame.

Space-variant sensing. Schwartz et al. (1995) reviewed space-variant active vision, in which a sensor trades uniform resolution for a high-resolution centre and a coarse periphery. That trade only makes sense with active gaze control, because the sensor must be pointed at whatever needs detail. The underlying principle—high resolution does not need to exist everywhere at once—is the premise of this project.

What this baseline uses. The present implementation does *not* model a variable-resolution retina. Each eye renders a uniform perspective image, and the central 12° of it serves as a nominal foveal core; peripheral pixels are not used. Variable-resolution retinal sampling belongs to the larger vision of the project, but this baseline does not need it. Its questions are about exploration and accumulation: given a narrow, high-quality binocular measurement, where should it be pointed, and how should the results be remembered?

3 The active foveal playground

Design principle. The whole system is one loop:

$$\text{FIXATE} \rightarrow \text{RECONSTRUCT EVERYTHING LOCALLY} \rightarrow \text{UPDATE GLOBAL } \mathrm{H}_0 \text{ MAP} + \text{CYCLOPEAN COVERAGE} \rightarrow \text{LOCAL OR GLOBAL SACCADE} \rightarrow \text{REPEAT}$$

Each fixation is self-contained. The observer points both eyes at a gaze direction, measures every foveal pixel it can match between the two eyes, and expresses the result in a fixed head frame H_0. It fuses the result into one global map and records the observed directions on a coarse sphere. A simple rule then chooses the next gaze: a short step to a neighbouring view when the fovea's own measurement shows surface continuing in an unexplored direction, or a jump to the middle of the largest unexplored region otherwise. The loop stops when 99 % of the sphere has been seen.

Persistent state. Only four things persist from one fixation to the next (Table 1). Nothing object-centric is needed: there are no object hypotheses, no object list, no per-object maps and no scheduler.

Figure 1 shows the loop and its data flow. It distinguishes the observation, the derived metric reconstruction, the persistent memory and the action. The global map M is the principal geometric output of the system, and it is *not* an input to gaze selection. The policy reads the coverage sphere S, the visited directions V and the current fixation's valid-geometry evidence. This is intentional. It keeps the baseline minimal, and it leaves open one of the most interesting

Table 1: Persistent state of the baseline. Everything else is rebuilt at every fixation and discarded. The last column is the key architectural fact: the gaze policy does not read the map.

	state	updated by	read by gaze policy
M	global H_0 surfel map: position (float64, metres), blended RGB, support count, first fixation, oracle label (visualization only)	12-mm fusion, every fixation	**no**
S	cyclopean sphere: 180×360 cells of $1°$, each UNSEEN, SEEN or DEPTH	coverage update, every fixation	yes
V	visited gaze directions (unit vectors in H_0)	appended every fixation	yes
g	current gaze (yaw, pitch) in H_0	the saccade decision	yes
—	*transient:* valid-geometry mask of the current foveal core	reconstruction	yes (then discarded)

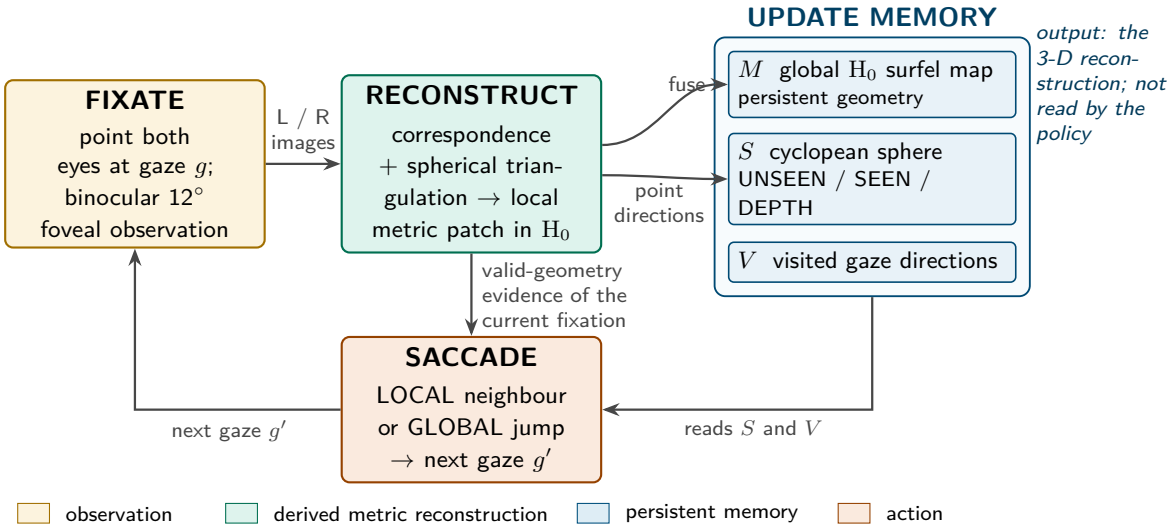


Figure 1: The Fixate–Reconstruct–Saccade loop. One fixation produces a binocular observation (orange), from which a local metric patch is derived (green). The patch is fused into the persistent map M, and its directions update the cyclopean coverage S (blue). The saccade decision (vermilion) reads only S, the visited directions V and the current fixation's valid-geometry evidence. The map M is the system's output; it is not fed back to gaze selection in this baseline.

playground questions: what changes when attention is driven by the reconstructed 3-D map itself (Section 12)?

4 Geometry and coordinate systems

All lengths are in metres. The head never moves, and the eyes rotate about their own centres. Five frames are needed; Table 2 summarizes them.

World and fixed head. The world frame W is Blender's scene frame ($+Z$ up). The fixed head frame H_0 has its origin at the midpoint between the eyes, at $\mathbf{o} = (-0.6, -1.0, 1.2)$ in W, and $\mathbf{p}_H = (\mathbf{p}_W - \mathbf{o})\, R_{wh}$. Its axes are $+X$ to the right (along the eye baseline), $+Y$ up (world $+Z$) and $-Z$ forward (world $+Y$). A gaze is a direction in H_0, parameterized by yaw ψ and

pitch θ:

$$\mathbf{d}(\psi,\theta) = \big(\sin\psi\cos\theta,\ \sin\theta,\ -\cos\psi\cos\theta\big), \qquad \psi = \operatorname{atan2}(d_x, -d_z), \quad \theta = \arcsin d_y. \tag{1}$$

The first gaze is $(0^\circ, 0^\circ)$, straight ahead.

Eye cameras. The left and right eye centres are $(\mp 0.0315, 0, 0)$ in H_0, a 63-mm baseline. Each eye has a rotation R_{hc} from camera to head, in the OpenCV convention (x right, y down, z forward). For a gaze g, both eyes verge on the point 2.1 m along $\mathbf{d}(g)$ from the head origin. Each image's x axis is the head baseline projected into the image plane, so the baseline stays horizontal in every viewing direction. Pixels have integer centres, and rows run top-down.

Current tangent frame. For the current gaze, the tangent frame has $z = \mathbf{d}(g)$, $x =$ the head $+X$ axis projected orthogonally to z and normalized, and $y = z \times x$ (image down). The 12° footprint of a fixation and the eight local saccade candidates are defined in this frame.

Cyclopean sphere. The coverage memory lives on the sphere of directions seen from the H_0 origin, the cyclopean point between the eyes. It is an equirectangular grid of 1° cells: 180 rows by 360 columns, row 0 at the top (pitch $+90^\circ$) and column 0 at yaw -180°. Because each eye is only 31.5 mm from the cyclopean origin, a fixation's cyclopean footprint and the left eye's actual view differ by a small parallax. DEPTH is therefore always intersected with the cyclopean footprint.

Epipolar coordinates. For a head-frame direction $\mathbf{d}$, ϑ is the angle from $+X$ (the baseline), and $\varphi = \operatorname{atan2}(d_y, -d_z)$ is the rotation about X. Rays from the two eyes that see the same point share φ, and their parallax is $\vartheta_R - \vartheta_L > 0$.

Table 2: Coordinate systems. Persistent quantities live in H_0 or on the cyclopean sphere; everything else exists only inside one fixation.

frame	definition	lifetime
world W	Blender scene frame, $+Z$ up, metres	scene only
fixed head H_0	origin between the eyes; $+X$ right (baseline), $+Y$ up, $-Z$ forward	**persistent**: map, gazes
eye cameras L, R	centres $(\mp 31.5, 0, 0)$ mm; OpenCV axes; verge at 2.1 m along the gaze	transient
pixel (u, v)	integer pixel centres, rows top-down; raw raster 640×640	transient
tangent frame	$z =$ gaze; $x =$ head $+X$ projected; $y = z \times x$	transient
epipolar (ϑ, φ)	ϑ from $+X$; $\varphi = \operatorname{atan2}(d_y, -d_z)$	transient
cyclopean sphere	directions from the H_0 origin; 1° grid, 180×360	**persistent**: coverage

5 One fixation: from gaze to metric 3-D

A fixation is a complete measurement transaction:

gaze → binocular render → 12° foveal core → PERFECT correspondence → spherical triangulation → H_0 metric patch → fusion + coverage update

Table 3 lists the sensor and baseline parameters.

Table 3: Sensor and baseline parameters (**PARAMETER**).

baseline (inter-ocular distance)	63 mm; eye centres $(\mp 31.5, 0, 0)$ mm in H_0
vergence	both eyes fixate the point 2.1 m along the gaze
raw raster	640×640 px per eye; focal length 1217.84 px; 29.4° field of view
foveal core	central 256×256 px (65,536 rays) of the left eye, nominally 12°
rendering	Blender Cycles (OPTIX), 64 samples per pixel, box filter, no denoising; fixed seeds L 2111 / R 2112
render passes	Combined (RGB), Position, Object Index
coverage grid	1° cyclopean cells, 180×360
fusion	12-mm association radius, 12-mm hash cells
local saccade	8 tangent-frame neighbours; 7.2° axial step (≈10.1° diagonal)
eligibility	EDGE_SUPPORT ≥ 0.25, UNSEEN_GAIN ≥ 0.25, $\geq 2°$ from visited gazes
stop	SEEN $\geq 99\,\%$ of 4π; safety cap 600 fixations

Binocular render. A persistent Blender process renders both eyes for the requested gaze (Section 9). Each eye produces three images: the RGB that a camera would see, plus the renderer's internal Position and Object Index passes, which only the correspondence oracle uses.

PERFECT / ORACLE correspondence. For every pixel centre in the left foveal core, the oracle finds the exact corresponding position in the right image from the renderer's geometry:

- *Catalogued geometry* (positive Object Index). The left pixel's Position is projected into the right camera. The pair is kept when the nearest right pixel sees the same object (a visibility test). The right coordinate is the exact continuous projection.
- *Other geometry* (Object Index 0). The same projection is used. The pair is kept when the nearest right pixel is a geometric hit whose Position lies within 5 mm + 1 % of the range from the left point, so that the right eye sees the same surface point. Otherwise the point is occluded for the right eye.

This is not natural stereo. It is an idealized measurement service that isolates the questions of this report from the difficulty of matching real images.

A truth-stripped interface. The oracle's output deliberately carries no geometry. It is four arrays: `left_core_row`, `left_core_col`, `uv_L` (left pixel centres) and `uv_R` (right image positions). Triangulation sees only these pairs plus the calibration; it never sees Position or Object Index. A natural matcher that produces the same four arrays from the two RGB images can therefore replace the oracle without any change to triangulation, fusion, coverage or the policy.

Spherical triangulation. Each corresponding pair defines two rays, from the left and right eye centres. Expressed in epipolar coordinates, the two rays lie in the same plane through the baseline (shared φ). In that plane, the triangle formed by the baseline b and the two rays gives the range from the left eye: $r_L = b \sin\vartheta_R / \sin(\vartheta_R - \vartheta_L)$. The accepted implementation computes this in an equivalent, numerically guarded form and returns points in H_0. Working in spherical epipolar coordinates rather than in a rectified image plane keeps the whole foveal core usable even when the gaze points close to the baseline direction. There, however, the parallax is small and the range is poorly conditioned.

The metric patch. The valid triangulated points form the fixation's local metric patch in H_0. Each point carries its left-eye RGB (display only) and the oracle Object Index of its pixel

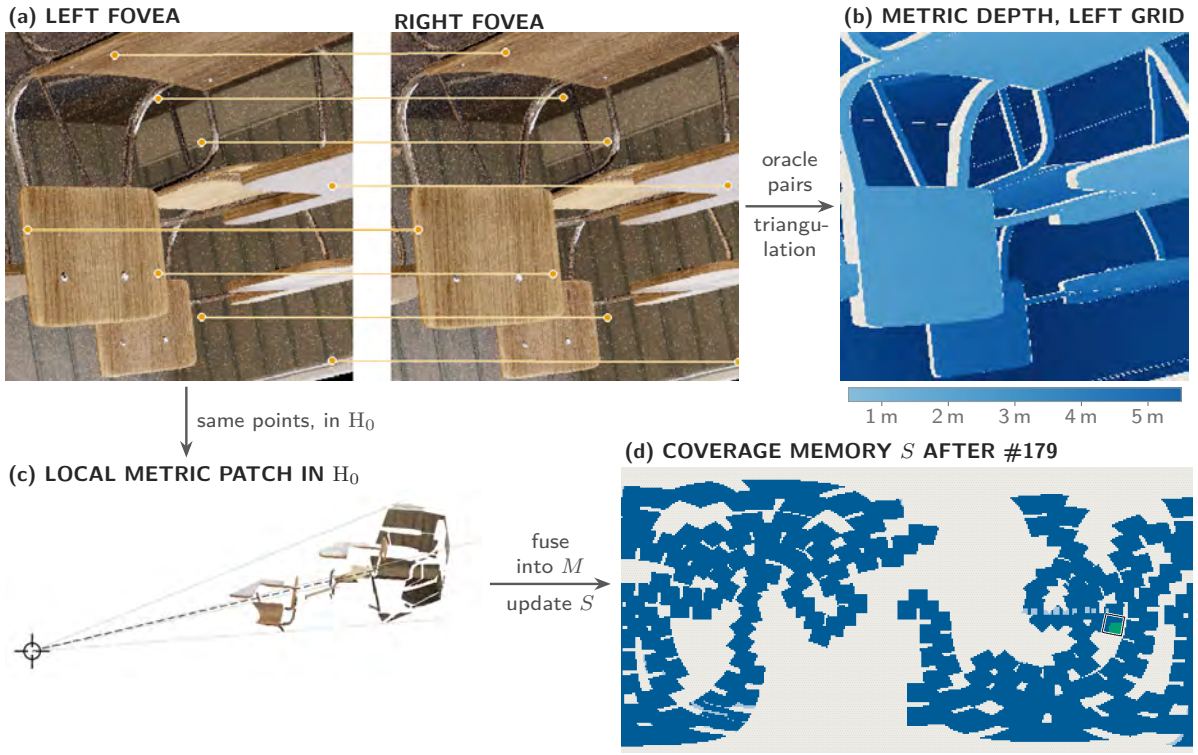


Figure 2: One fixation (#179, gaze yaw 130.0°, pitch −9.3°). (a) The binocular foveal pair (256 × 256 px each; 64-spp render, display RGB), with eight corresponding pixels, one per epipolar row. Corresponding points share a row, and their horizontal offset is the disparity. These are **ORACLE** correspondences, shown by reprojecting the triangulated points (**DERIVED**). (b) Range from the left eye on the left foveal grid (**DERIVED**); 0.5–5.5 m scale. (c) The fixation's 60,778 metric points in H_0, seen obliquely, with the head (crosshair), the gaze ray (dashed) and the 12° core (grey lines) (**DERIVED**). (d) The cyclopean coverage after this fixation (**CONTROLLER-TIME**): DEPTH dark blue, SEEN light blue, UNSEEN stippled grey. Green marks the cells this fixation added; the black outline is its 12° footprint. The fusion matched 36,481 points to existing surfels and created 24,297 new ones; the map grew to 8,836,458 surfels (MEASURED).

(visualization only). MEASURED over the official run, a fixation yields a median of 65,150 points (minimum 18,115, maximum 65,536 = the whole core). As a diagnostic, each point is also compared with the renderer's left Position. This error is recorded, never used. The patch is then fused into M (Section 6), and its directions update the coverage memory S (Section 7).

Figure 2 follows one fixation, #179, through this pipeline.

6 Persistent 3-D reconstruction

One global map. All geometry from all fixations goes into a single surfel map M in H_0. It follows the project's accepted 12-mm association and fusion rule, re-implemented with a vectorized spatial hash and applied to all geometry together, with no per-object partition.

Association. For each new patch, association is done against a *snapshot* of the map as it was before the patch. Each point is matched to the nearest existing surfel strictly within 12 mm. The search covers the 27 neighbouring 12-mm hash cells and is exact; ties go to the earliest surfel. Points of the same patch never match each other.

Fusion. All points of the patch that matched the same surfel are averaged into one contribution $\bar{\mathbf{x}}$, and that contribution is blended into the surfel with its support count s as weight:

$$\mathbf{x}' = \frac{s\,\mathbf{x} + \bar{\mathbf{x}}}{s+1}, \qquad s \leftarrow s+1. \tag{2}$$

RGB is blended in the same way; it serves display only and never enters association, coverage or gaze. Every unmatched point becomes a new surfel, with support 1 and a record of the fixation that created it. Unmatched points are not de-duplicated against each other within a patch. A patch identifier can be fused only once.

Attributes. Each surfel stores its position, blended RGB, support count and first fixation. It also stores the oracle Object Index of its creating point, which is used only for a segmentation-coloured visualization. Support counts and first-fixation provenance make it possible to trace when and how often each part of the scene was measured.

Storage, honestly. The map is deliberately unoptimized. MEASURED for the official run: the final map holds 16,517,811 surfels and its archive is about 1 GB (991 MB); the complete run directory, including every fixation's point patch, is 2.2 GB. Fusion is the largest host-side cost, 257.8 s of the 733 s wall time. It grows with the map, from a mean of 0.23 s per fixation over the first 100 fixations to 0.79 s over the last 92, with a maximum of 2.95 s. Compact representations are an obvious playground experiment (Section 13).

7 Cyclopean coverage memory

The controller's memory of *where it has looked* is separate from the map. It is a 180×360 grid of 1° cells on the cyclopean sphere, each in one of three states:

- UNSEEN: never inside a fixation's footprint;
- SEEN: inside at least one footprint, i.e. looked at;
- DEPTH: looked at, *and* a reconstructed metric point lies in that direction from the H_0 origin.

A fixation's footprint consists of the cells whose centre falls inside its nominal 12° square core, measured in the tangent frame ($|u|, |v| \le \tan 6^\circ$). States only increase: a cell can go from UNSEEN to SEEN to DEPTH, never back. We write

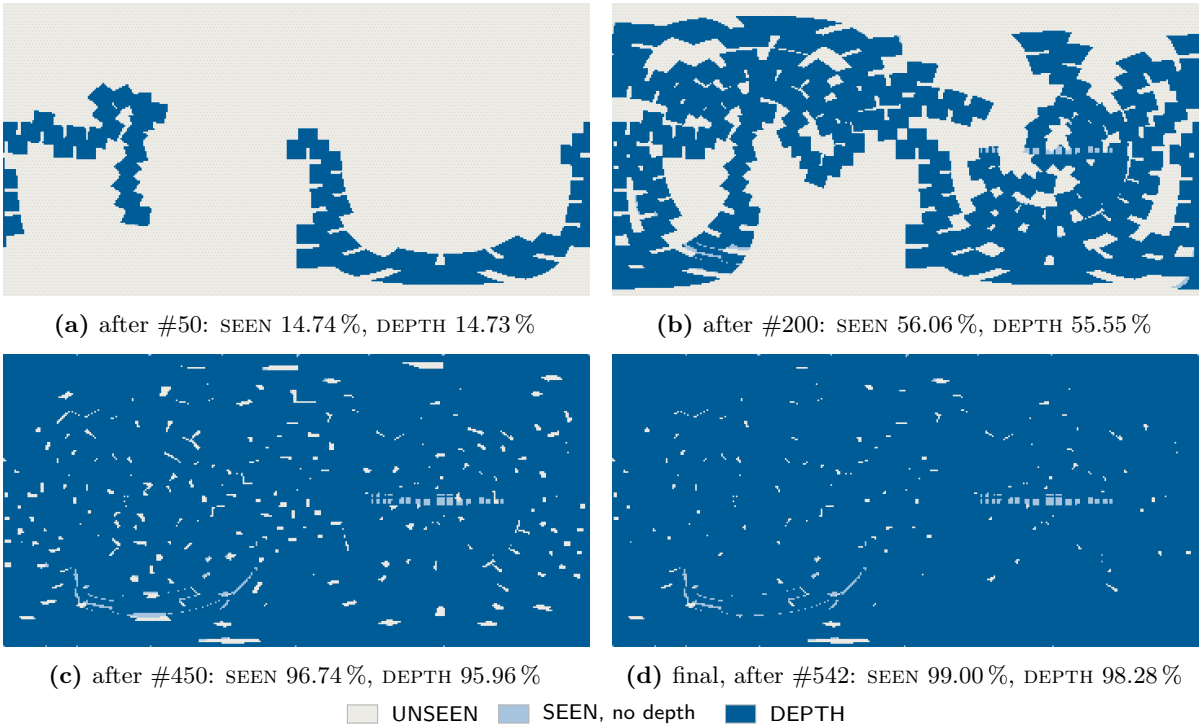
$$S_{\text{seen}} = \{\, c : \text{state}(c) \ge \textsc{seen} \,\}, \qquad S_{\text{depth}} = \{\, c : \text{state}(c) = \textsc{depth} \,\}.$$

Solid angle, not cell counts. An equirectangular grid has the same number of cells in every row, but a row near a pole covers far less of the sphere than a row at the equator. Counting cells would therefore overweight the poles. Every fraction is computed with the exact solid angle of each cell. For a row bounded by latitudes ϕ_{top} and ϕ_{bottom} and a longitude width $\Delta\lambda$,

$$\Delta\Omega = \Delta\lambda\,(\sin\phi_{\text{top}} - \sin\phi_{\text{bottom}}), \tag{3}$$

and the weights of all cells sum to exactly 4π. The SEEN and DEPTH fractions reported below are fractions of 4π.

Observation, measurement and geometry. The three terms are kept apart. SEEN records *observation*: the fovea was pointed there. DEPTH records *measurement*: the fixation produced metric points in that direction. The map M holds the *reconstructed geometry* itself. Coverage



(a) after #50: SEEN 14.74 %, DEPTH 14.73 %

(b) after #200: SEEN 56.06 %, DEPTH 55.55 %

(c) after #450: SEEN 96.74 %, DEPTH 95.96 %

(d) final, after #542: SEEN 99.00 %, DEPTH 98.28 %

Figure 3: Cyclopean coverage memory at four stages (**CONTROLLER-TIME**; the state the policy actually used). Equirectangular, yaw $-180° \to +180°$ from left to right, pitch $+90°$ at the top. Percentages are fractions of 4π (MEASURED, official run). The early walk covers contiguous regions. By #450 only scattered small holes remain, and the final state leaves 1.0 % UNSEEN in 184 small holes plus 0.72 % SEEN without depth (mostly the windows).

is updated from the current fixation's points; it never queries the map. A SEEN cell without depth is a direction that was looked at but returned no measurement, for instance a window with nothing behind it.

Seams and poles. Longitude wraps around, so column indices are taken modulo 360. When unexplored regions are labelled (Section 8), the polar rows are linked across the pole. The footprint is computed from direction dot products and has no seam at all.

Figure 3 shows the coverage memory at four stages of the run.

8 Greedy saccadic exploration

The decision after each fixation is a pure, deterministic function of four inputs: the coverage S, the current fixation's valid-geometry mask, the current gaze g and the visited directions V. The map M is *not* among them.

LOCAL saccades. Eight candidates surround the current gaze in its tangent frame: E, W, N, S, NE, NW, SE and SW. With tangent axes x, y, z and $a, b \in \{-1, 0, 1\}$, a candidate direction is $\mathbf{d} \propto z + \tan(7.2°)\,(a\,x + b\,y)$. Side neighbours are 7.2° away (0.6 of the core width); diagonal neighbours are about 10.1° away. Each candidate gets two scores:

- EDGE_SUPPORT: the fraction of pixels with valid reconstructed geometry in the outer quarter (64 px) of the current 256×256 core on that side, or in the corner block for a diagonal. It asks whether the fovea's own measurement shows surface continuing in that

direction.

- UNSEEN_GAIN: the UNSEEN solid-angle fraction of the candidate's 12° footprint. It asks whether going there would show anything new.

A candidate is eligible if EDGE_SUPPORT ≥ 0.25, UNSEEN_GAIN ≥ 0.25 and it lies at least 2° from every visited gaze. Among eligible candidates, the one with the highest score

$$\text{score} = \text{EDGE_SUPPORT} \times \text{UNSEEN_GAIN} \tag{4}$$

wins; ties go to the first candidate in the fixed order above. The local rule therefore follows measured surface into unexplored territory.

GLOBAL saccades. When no neighbour is eligible, the explorer jumps:

1. It labels the 4-connected components of UNSEEN cells, wrapping in longitude and linking across the poles, and orders them by solid angle.
2. In the largest component, it targets the *deepest* cell: the cell with the largest angular distance to the nearest SEEN cell bordering unexplored territory, i.e. the middle of the hole.
3. It skips cells within 2° of a visited gaze, falling back to the next-deepest cell and then to the next component.

Stopping. After each fixation, the explorer stops if SEEN $\geq 99\,\%$ of 4π (the scientific stop), if the safety cap of 600 fixations is reached, or if no candidate remains. Otherwise it saccades.

What the policy does not use. There is no object identity, no object scheduler, no semantic state and no learned component. The oracle Object Index never reaches the policy, and neither does the 3-D map. Because the decision is deterministic, every saved decision state can be replayed. In the official run, all 541 decisions were re-decided identically (MEASURED, run checks).

Figure 4 shows the complete gaze trajectory. Panel (a) shows the local walks, each starting where a global jump landed; panel (b) shows the 219 jumps themselves.

9 Software architecture and reproducibility

Code. The official implementation is small: five source files plus an equivalence checker, in the repository directory `tools/greedy_foveal/` (Table 4). The canonical technical description is the architecture document `docs/architecture/greedy-foveal-playground.md`.

A persistent observation service. Starting Blender and loading the scene for every fixation would dominate the run time. The render server instead loads the Classroom once and then only moves the two eye cameras. MEASURED, a 64-spp binocular pair takes about 0.76 s. The host and the server talk through an atomic file protocol. The host writes a request `req-NNNN.json` (fixation number, yaw, pitch, output directory). The server writes the calibration and the two images, then a `done-NNNN.json` marker. A failure writes `fail-NNNN.json` and exits nonzero, and a `stop` file ends the loop. At start-up the server checks that the scene's eye pose equals the accepted fixed head, and every fixation re-checks that the head has not moved and that the rendered gaze equals the requested one.

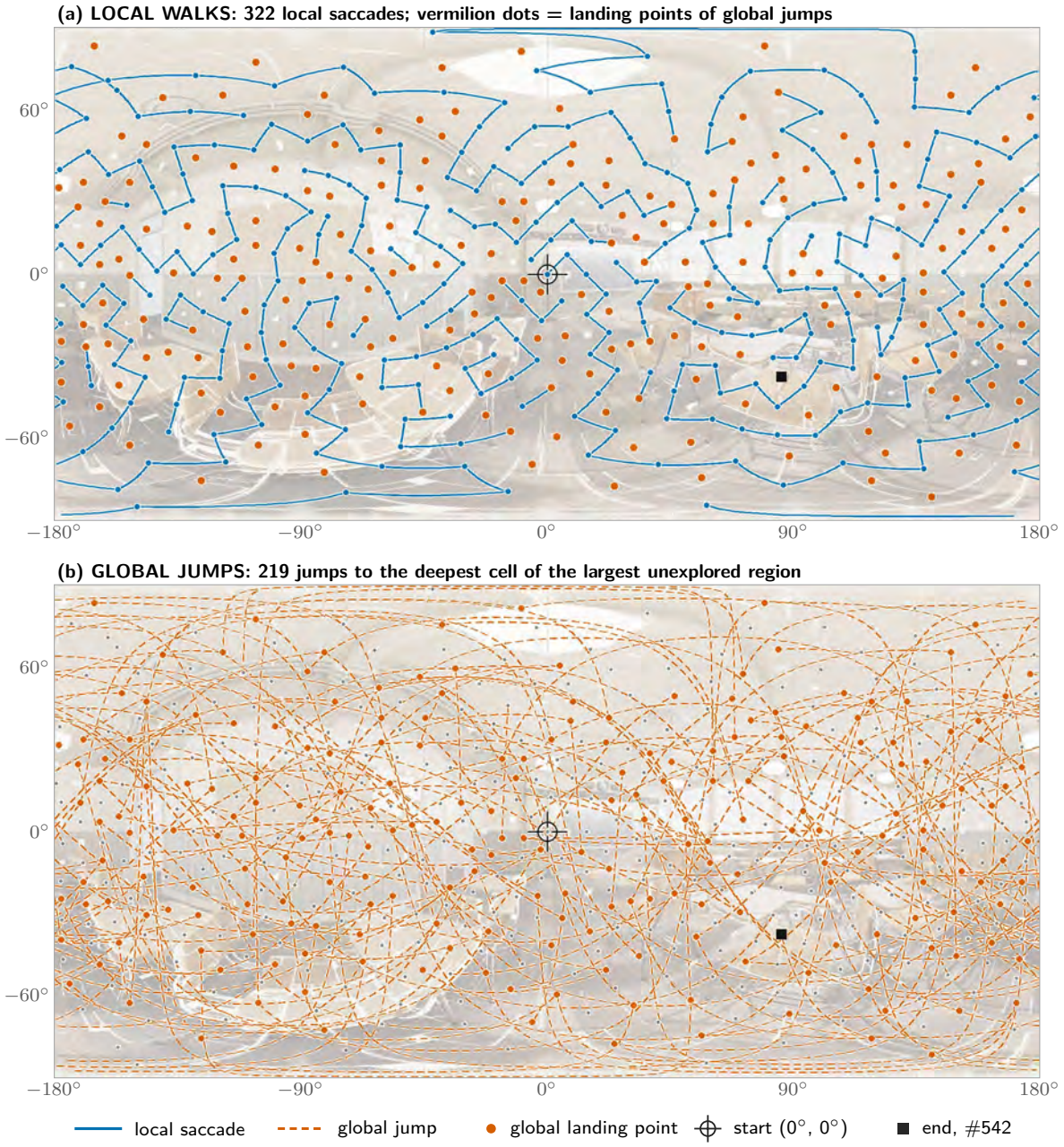


Figure 4: The complete 542-fixation gaze trajectory over the 360° sphere (**CONTROLLER-TIME**; gazes and decisions from the official run). Equirectangular, yaw left to right, pitch up. The background is the final reconstructed RGB panorama, lightened (**DERIVED**). (a) Local saccades (solid blue) form short walks that follow measured surface into unexplored territory. (b) Global jumps (dashed vermilion) carry the gaze to the middle of the largest remaining hole; late in the run, most of them land in small isolated holes. Grey dots in (b) are the other fixations.

Table 4: The official implementation (`tools/greedy_foveal/`).

file	runs in	role
`explorer.py`	host Python	sphere grid and coverage; one fixation's correspondence and reconstruction; the global map and fusion; the greedy policy
`run.py`	host Python	the active loop (`explore`), freeze, post-hoc `evaluate`, `checks`, `visualize`
`render_server.py`	Blender	persistent binocular observation service
`visuals.py`	host Python	overview figure; display colours for the exported point clouds
`demo.py`	host Python	offline presentation replay of the frozen run (video, poster, stills)
`check_equivalence.py`	host Python	source, run and self-test equivalence checks against the frozen baseline

Run stages. A run proceeds as

$$\mathsf{explore} \rightarrow \mathsf{freeze} \rightarrow \mathsf{evaluate} \rightarrow \mathsf{checks} \rightarrow \mathsf{visualize}\ .$$

`explore` runs the loop and saves, per fixation, the calibration, the point patch and the decision state. At the end it writes the trajectory, the coverage record, the final map and two point-cloud exports, and *freezes* them by recording their SHA-256 hashes. `evaluate` refuses to run unless the frozen products are intact. `checks` verifies invariants: the head stayed fixed; every point is finite; no fixation is repeated; SEEN never decreased; the map reloads identically; every decision re-decides identically; and the fusion code reproduces the accepted fusion rule bitwise on a known-answer test.

The truth firewall. The evaluation reference is *unavailable during control*. An audit hook records every file the host opens during `explore`, and the freeze record states that none of them belongs to the reference. Only after verifying the freeze does `evaluate` open the reference. MEASURED for the official run: 3,813 files were opened during control, none of them under the reference.

Dependencies. The explorer reuses accepted project modules without copying them. By role, these are: sensor geometry and spherical epipolar triangulation (*core runtime*); the oracle correspondence, an EXR reader and the Blender acquisition helpers (*synthetic observation*); the full-sphere reference extraction and the 12-mm coverage measure (*post-hoc evaluation*); and the project's visual-language style module (*presentation*). Three data dependencies sit outside version control: the Blender scene, the accepted head-pose calibration and the reference image (the last two hash-checked). The architecture document gives the full map.

9.1 Official baseline and bitwise reproduction

The baseline was first run as a prototype and frozen: run, implementation and demo were each recorded by hash and version-control tag. It was then promoted into an official, tagged code baseline (tag `greedy-foveal-playground-baseline`). The promotion changed only documentation strings, and it was mechanically checked: each promoted file is identical to its frozen version once docstrings are removed. A clean run of the official code was then compared with the frozen run by an equivalence checker. Its rules were declared before the run: control-path products must be *bitwise* equal, and only presentation RGB may differ. GPU path tracing is not bitwise reproducible from run to run, and RGB never enters gaze, association, coverage or stopping.

The result (MEASURED): **23 of 23 strong-equivalence checks pass**. Every one of the 542 gazes, decisions and local-candidate scores; every per-fixation point patch and oracle label; every fusion count; all 541 decision states; the coverage history; the final map's positions, support counts and provenance; and the post-hoc evaluation are bitwise identical to the frozen run. The checker's own comparators were shown to fail on deliberately perturbed inputs (a one-ULP change in a coordinate, a flipped decision, a changed constant). A 75-fixation run compared against the full baseline fails it, as it should. For a baseline that is meant to be compared against, this is the property that matters: any later difference is caused by the component that was changed, not by drift in the baseline.

10 The Classroom proof of concept

Setting. The scene is the Blender Classroom, a furnished room with desks, chairs, walls, windows, a blackboard and ceiling fixtures. Everything is static. The head is fixed at the room position given in Section 4. Observations are rendered at 64 samples per pixel. Correspondence is PERFECT / ORACLE (Section 5).

The run. The explorer starts looking straight ahead and runs until SEEN $\geq 99\,\%$ of 4π. A safety cap of 600 fixations would have stopped it otherwise; the cap was not reached. Nothing in the run was tuned to this scene beyond the parameters of Table 3.

Reference and metric. The reconstruction is evaluated against an independent full-sphere reference: one 720×360 equirectangular render (0.5° cells) from the head origin. It records, for every direction, the first surface hit (**REFERENCE / EVALUATION; NOT AVAILABLE DURING CONTROL**).

- Of its 259,200 cells, 255,758 have a first hit; the other 3,442 look through windows into empty space.
- A reference cell counts as *reconstructed* if some surfel of the final map lies within 12 mm of its first-hit point.
- Coverage is reported per cell and, with exact 0.5° solid-angle weights, as a fraction of the first-hit reference solid angle.

The reference contains only what is visible from the head origin. Surfaces hidden behind furniture are not reconstruction targets, and no claim is made about them.

11 Results

Table 5 gives the headline numbers of the official reproduction, and Figure 5 shows how coverage grew.

Three phases. The growth curve has three distinct regimes (Table 6).

- *Local exploration.* For the first 68 fixations the explorer never needs to jump: local walks alone cover 19.6 % of the sphere. Through #200 it makes 195 local and only 4 global saccades, and SEEN rises almost linearly to 56.06 %.
- *Mixed exploration.* Between #201 and #450, local walks increasingly run into explored territory, and global jumps make up nearly half the moves (123 of 250). SEEN reaches 96.74 %.

Table 5: Final result of the official reproduction (MEASURED; bitwise identical to the frozen baseline in every control-path value). Coverage percentages refer to the first-hit reference, not to all scene surfaces.

final fixation	542
stop reason	SEEN $\geq 99\,\%$ of 4π (cap 600 not reached)
SEEN (observed fraction of the sphere)	99.00 %
DEPTH (with reconstructed depth)	98.28 %
local / global saccades	322 / 219
surfels in the final map	16,517,811
first-hit reference cells within 12 mm	98.52 % (251,966 / 255,758)
first-hit reference solid angle within 12 mm	**98.32 %**
authored objects only (Object Index > 0)	98.25 %
distance to nearest surfel, covered cells	median 0.73 mm, 90th percentile 2.24 mm
wall time	733 s
render / fusion / correspondence / geometry / policy	413.5 / 257.8 / 18.2 / 12.6 / 3.3 s
post-hoc evaluation	26.3 s

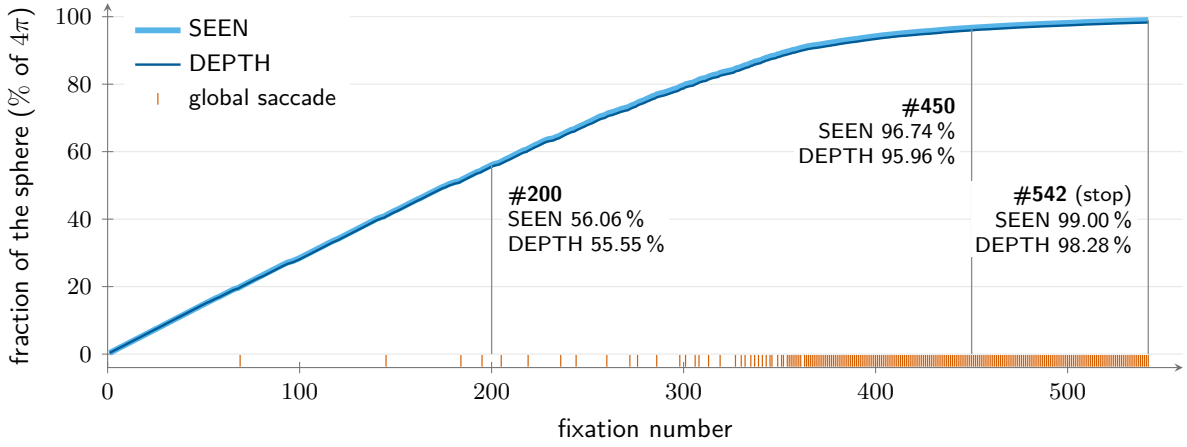


Figure 5: Growth of SEEN and DEPTH with fixation number (**CONTROLLER-TIME**, MEASURED, official run). Ticks along the bottom mark the 219 global saccades. The two curves nearly coincide because almost every direction that was looked at also returned depth. Three phases are visible: fast local exploration (few global jumps before #200); a mixed phase in which global jumps become frequent (#201–#450); and a long tail of single global jumps into small holes (#451–#542).

- *Hole filling.* After #450 all 92 moves are global, each into one small isolated hole. The mean SEEN gain falls to 0.025 % per fixation (minimum 0.015 %).

The final reconstruction. Figure 6 compares the reconstruction with the independent reference. Projected back onto the sphere, the final map reproduces the room's appearance (RGB), its metric layout (depth) and, for visualization only, its object structure (oracle instance labels). The 3-D view shows the room as one coherent metric model. Post hoc, 98.32 % of the first-hit reference solid angle lies within 12 mm of the map (98.52 % of reference cells). Covered cells are typically much closer than the threshold: the median distance to the nearest surfel is 0.73 mm. What remains has two causes. One is the 184 small UNSEEN holes, each smaller than 0.002 sr. The other is thin slivers along depth edges, where the reference ray from the head origin meets a surface that the binocular measurement did not reach.

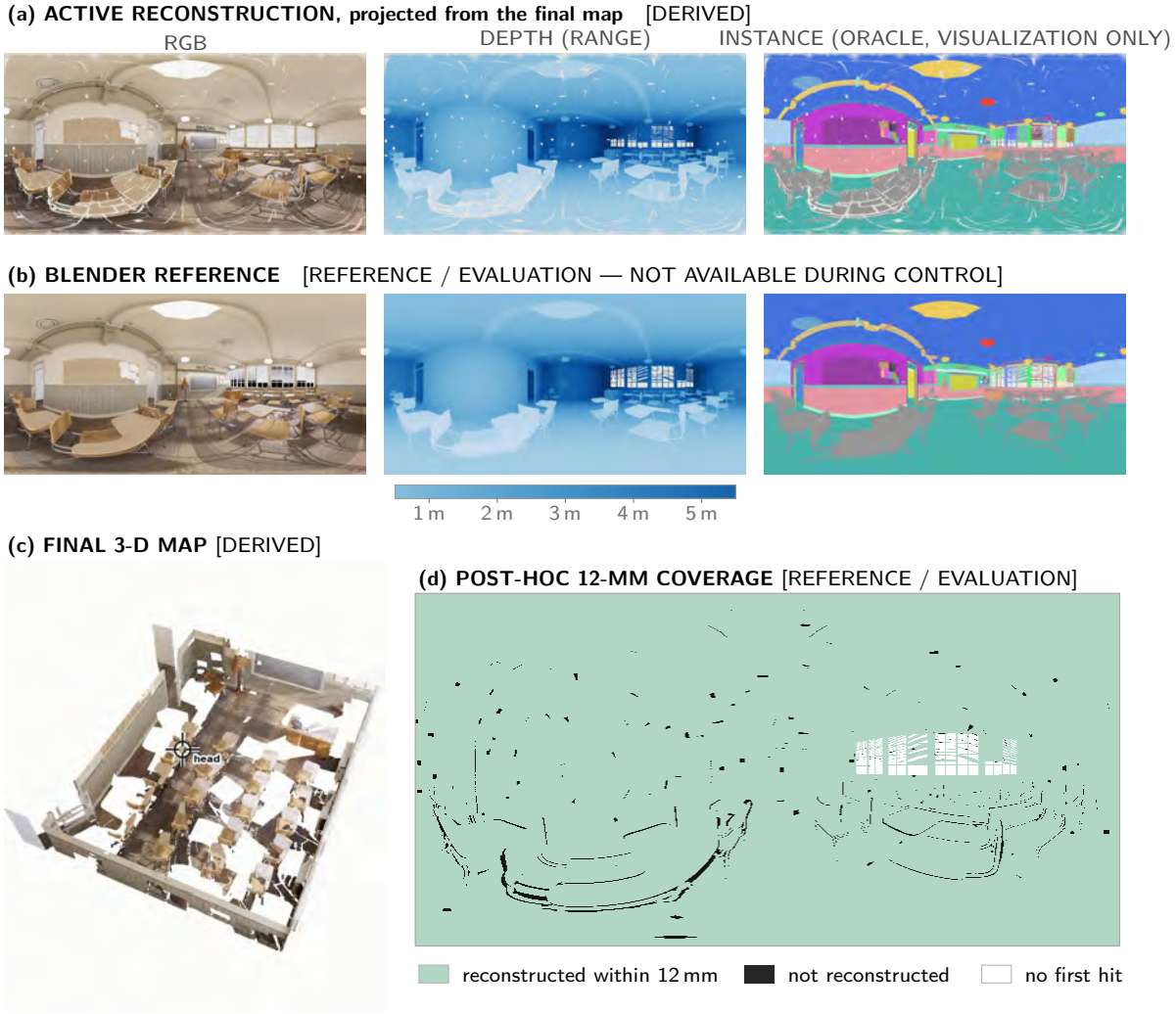


Figure 6: Final reconstruction and reference. (a) The final map projected onto the sphere from the head origin, nearest surfel per pixel: display RGB, range, and oracle instance labels (**DERIVED**; instance colours are **ORACLE / VISUALIZATION ONLY** and played no role in the run). Stippled grey = nothing reconstructed in that direction. (b) The independent Blender reference at 0.5° (**REFERENCE / EVALUATION**, never available during control). (c) The 16.5-million-surfel map seen from above and behind the head, with the ceiling removed for display (**DERIVED**). (d) Post-hoc evaluation over the reference sphere: 98.32 % of the first-hit reference solid angle is reconstructed within 12 mm. The misses are small holes and thin depth-edge slivers; the windows (white) have no first hit. Panels (a)–(c) are images from the frozen demonstration, whose control path the official reproduction matches bitwise.

Table 6: Exploration phases (MEASURED, official run; values at the end of each phase).

fixations	local	global	SEEN	DEPTH
1–200	195	4	56.06 %	55.55 %
201–450	127	123	96.74 %	95.96 %
451–542	0	92	99.00 %	98.28 %

12 Discussion

What the result establishes. Within its controlled setting, the experiment supports five observations.

1. *Narrow measurements accumulate into a whole scene.* Repeated 12° binocular measurements, each expressed in one fixed frame and fused into one map, add up to a nearly complete metric representation of the first-hit 360° scene: 98.32 % of the reference solid angle within 12 mm, after 542 fixations.
2. *A very simple policy is enough here.* The greedy rule has one score and two thresholds, and it uses no model of the scene beyond a 1° coverage grid. It stopped on its own at 99 % coverage, without stalling.
3. *Local and global gaze are complementary.* Local saccades exploit what the current fixation measured: they follow surfaces. Global saccades exploit what the memory knows: they go where nothing has been seen. Neither alone would suffice. Local walks stall at boundaries of explored territory, and pure global jumps would ignore the continuity that makes local walks efficient early on.
4. *Memory turns a narrow fovea into a wide-field observer.* At no moment does the system see more than 12° in detail. Persistent memory—the map for geometry, the coverage sphere for attention—is what makes the accumulated percept wide.
5. *The architecture is separable.* Observation, correspondence, triangulation, fusion, coverage and policy meet at narrow, explicit interfaces. The truth-stripped correspondence interface is the clearest example, and each of these services can be replaced independently (Section 13).

What it does not establish. The experiment says nothing about natural correspondence: matching was perfect by construction. Nor does it show natural segmentation; the oracle labels only colour figures. It shows no optimal or even efficient gaze selection, no perception with a moving head, no dynamic scenes, and no generalization beyond one synthetic room. The reconstruction target is the first-hit scene seen from one point, so hidden surfaces are outside it.

The map is not in the control loop. The most important architectural observation is what the policy does *not* read. Gaze is driven by a coverage record (*was this direction looked at?*) and by the current fixation's valid-geometry mask (*does measured surface continue this way?*). The accumulated 3-D map—the system's best knowledge of the scene—is never consulted. This kept the baseline minimal, and the result shows that such a loop can already reconstruct the first-hit scene almost completely. It also frames a sharp next question: what changes when attention is driven by the reconstructed 3-D map? Map-driven frontiers could target depth discontinuities, low-support regions or surfaces seen only at grazing angles. Whether that reduces the fixation count, improves the geometry, or both, is exactly the kind of question the playground is built to answer.

The late global tail. After #450, the explorer spends 92 fixations clearing small holes one at a time, each jump gaining on average 0.025 % of the sphere. Each hole is the size of a fraction of the fovea, so most of each fixation re-measures known territory. This is not a defect to explain away. It is useful evidence about the simple policy. A global rule that picks the *largest* hole regardless of distance spends many long saccades on small gains. Ordering the remaining holes into a short tour, or covering several holes per fixation, are obvious alternatives—and they are measurable against this baseline.

On the oracle assumption. PERFECT correspondence removes matching errors and missed matches. This report does not estimate how natural stereo would change the measured coverage or precision. The design does guarantee, though, that the comparison can be made cleanly: a natural matcher plugs into the same interface, and everything downstream stays fixed.

13 The playground

The accepted baseline is intentionally not optimal. Its value is that it is complete, simple, explicit and exactly reproducible, so that the effect of changing one component can be measured. The research method it supports is:

change ONE component → compare against the official baseline → measure the causal effect

Table 7 lists the replaceable components and Figure 7 shows where they sit in the loop.

Table 7: Playground extension points. Each row can be replaced independently; the architecture document gives each component's interface.

component	current baseline	replaceable by (examples)
renderer	Blender Cycles, persistent server, 64 spp	other scenes; other sample counts; a fast rasterizer; real cameras
stereo matcher	PERFECT / ORACLE correspondence	natural stereo (block matching, learned matchers)
local measurement	spherical epipolar triangulation	per-point uncertainty; surface normals; variable fovea
map representation	flat float64 surfel arrays	compact surfels; voxel / TSDF / octree maps
fusion	12-mm snapshot association, support-weighted mean	confidence weighting; outlier rejection; de-duplication
coverage representation	1° UNSEEN/SEEN/DEPTH sphere	finer or equal-area grids; graded confidence
local gaze policy	8 neighbours, edge support × unseen gain	map-driven 3-D frontiers; information gain
global gaze policy	deepest cell of the largest unexplored region	nearest-hole or tour ordering of holes
stopping rule	SEEN $\geq$ 99 %, cap, no candidate	depth- or quality-based stop; time budget
segmentation / semantics	none in control (oracle labels for figures)	natural segmentation; object-aware attention
head motion	none: fixed head	head pose as an action; better-conditioned stereo
visualization	overview figure, offline demo replay	interactive viewer; web presentation

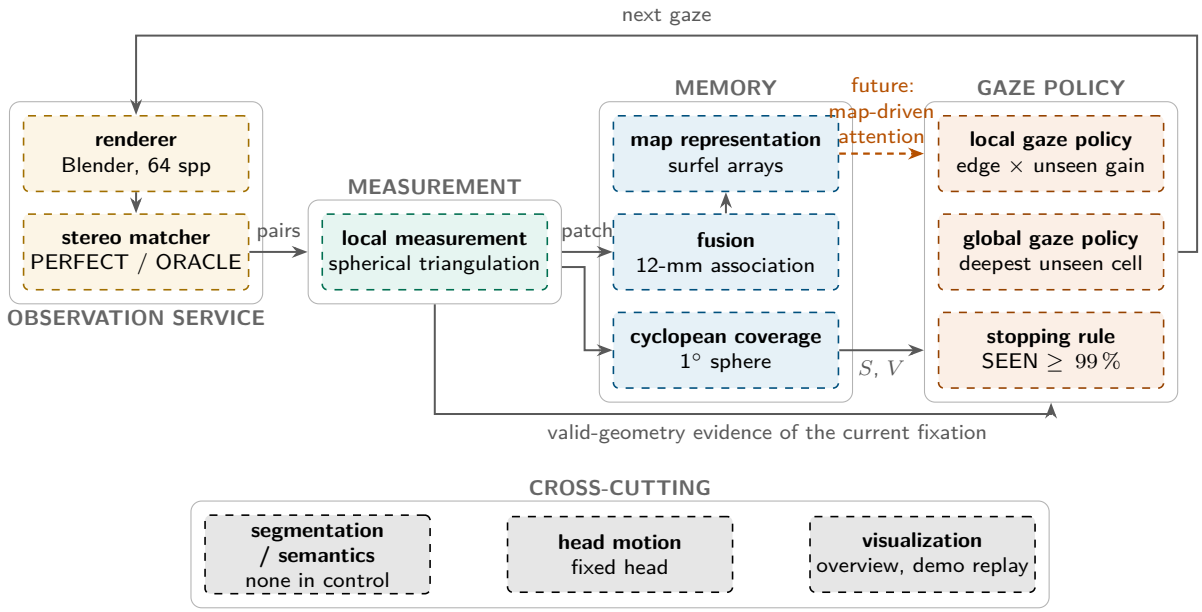


Figure 7: Playground extension map. Every dashed box is a replaceable component, coloured by role: observation (orange), measurement (green), memory (blue), gaze policy (vermilion), cross-cutting (grey). Solid arrows are the data flow of the baseline. The dashed vermilion arrow from the map to the policy is *not* part of the baseline: map-driven attention is a future experiment, and in the baseline the policy reads coverage, visited directions and the current fixation's evidence only.

Possible directions. Without selecting the next experiment, the playground invites several:

- *natural stereo*, through the truth-stripped correspondence interface;
- *map-driven frontier exploration*, closing the loop from M to the policy;
- *smarter global saccade ordering* for the hole-filling tail;
- *compact map representations* that keep fusion semantics while cutting memory and time;
- *natural segmentation and semantics*, to give attention an object-level vocabulary without oracles;
- a *moving head*, so that stereo can be re-centred where it is poorly conditioned;
- *new scenes*, to test whether the baseline's behaviour transfers;
- *actual foveated, variable-resolution sensing*, replacing the uniform render with a retina-like sampling.

Each of these changes one row of Table 7. Each can be judged against the same frozen numbers: fixations to 99 % SEEN, the growth curve, first-hit coverage, map size and time.

14 Limitations

The baseline is defined by its limitations as much as by its results. Stating them precisely is what makes it usable as a reference.

- **Fixed head.** Only the eyes move. The head position is a parameter, and poorly conditioned near-baseline gazes cannot be avoided by moving it.
- **Static synthetic scene.** One Blender room, with no motion, no lighting change and no sensor noise model beyond Monte Carlo render noise.
- **PERFECT / ORACLE correspondence.** Matches come from the renderer's geometry; no natural stereo is involved.

- **Oracle labels only where declared.** Object Index is used by the correspondence oracle and for visualization. It never drives gaze, fusion or stopping.
- **64-spp synthetic observation.** RGB is noisy and not bitwise reproducible. Under PERFECT correspondence it affects display only.
- **First-hit evaluation.** The target is what is visible from the head origin. Hidden and occluded surfaces are outside the target set.
- **An unsophisticated greedy trajectory**, with a diagonal bias in local walks, triangular holes and a long hole-filling tail.
- **The map is not used for control.** The policy never consults the reconstruction.
- **Storage is not optimized**: about 1 GB for the final map and 2.2 GB per run.
- **One demonstrated scene.** Nothing here shows transfer to other environments.

15 Conclusion

We set out to ask whether a narrow, spatially selective binocular observer could build a persistent 3-D representation of its surroundings by choosing where to look. In a controlled synthetic setting it can. A loop of fixation, local metric reconstruction, persistent memory and a simple greedy saccade rule observed 99 % of the sphere in 542 fixations, and reconstructed 98.32 % of the first-hit reference solid angle within 12 mm. The implementation reproduces this result bitwise.

The important result is not that a greedy policy is optimal; it plainly is not. It is that a small, explicit active-foveal loop is enough to produce a complete and reproducible computational baseline. That changes the question. It is no longer *whether* active foveal reconstruction can work, but *which component to replace next*, and by how much the change helps.

References

John Aloimonos, Isaac Weiss, and Amit Bandyopadhyay. Active vision. *International Journal of Computer Vision*, 1(4):333–356, 1988. doi: 10.1007/BF00133571.

Dana H. Ballard. Animate vision. *Artificial Intelligence*, 48(1):57–86, 1991. doi: 10.1016/0004-3702(91)90080-4.

Eric L. Schwartz, Douglas N. Greve, and Giorgio Bonmassar. Space-variant active vision: Definition, overview and examples. *Neural Networks*, 8(7–8):1297–1308, 1995. doi: 10.1016/0893-6080(95)00092-5.

Alfred L. Yarbus. *Eye Movements and Vision*. Plenum Press, 1967.

A Reproduction and provenance

Repository and tags.

- repository: `https://github.com/visgraf/f3d-vision`
- official baseline: annotated tag `greedy-foveal-playground-baseline` (commit `203a70c`)
- frozen prototype implementation: tag `greedy-foveal-explorer-v0-impl`
- frozen demonstration: tag `greedy-foveal-explorer-v0-demo`

Canonical documents.

- architecture (the canonical technical description): `docs/architecture/greedy-foveal-playground.md`
- official reproduction and equivalence evidence: `docs/engineering/greedy-foveal-playground-baseline-report.md`
- proof-of-concept closure, with the hashes of the frozen run and demo: `docs/prototype/greedy-foveal-explorer-v0-closure.md`

Commands. From the repository root (Blender 5.2.1 with an OPTIX GPU; about 12 minutes on an RTX 4090):

```
.venv/bin/python tools/greedy_foveal/run.py explore  --run RUN --max-fix 600 --spp 64
.venv/bin/python tools/greedy_foveal/run.py evaluate --run RUN
.venv/bin/python tools/greedy_foveal/run.py checks   --run RUN
.venv/bin/python tools/greedy_foveal/check_equivalence.py run --run RUN \
    --baseline previews/greedy-foveal-explorer-v0-grow600 --known-answer
```

Measured source of this report. All MEASURED values come from the official reproduction run (`previews/greedy-foveal-official-baseline/`, freeze record SHA-256 `6bc9b190...`). It was made from commit `b77b1d9` and is bitwise equivalent to the frozen run in every control-path product. The figures were produced by `docs/technical-report/make_figures.py` from that run and from the frozen demonstration's images. No scene was rendered for this report. `figures/figure-sources.json` records the hash of every input and output.