

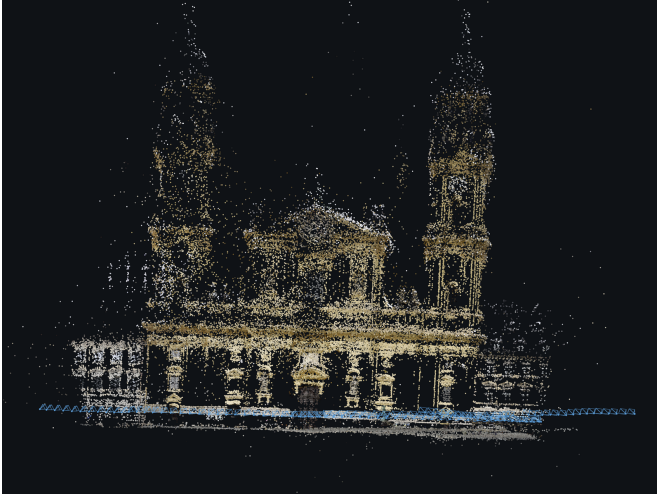
Splatfiction Mapper: Browser-Resident Structure from Motion with WebGPU

Tomáš Sikora

Reaigen s.r.o.

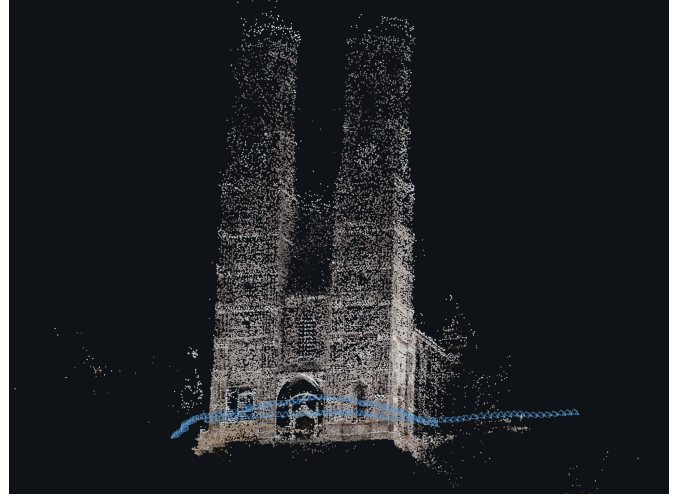
Slovakia

Technical validation and R&D roadmap



(a) Munich Theatine

54,909 points 120 registered cameras



(b) Munich Frauenkirche

53,104 points 123 registered cameras

Figure 1: Sparse structure and recovered cameras from the main benchmark cohort. Original RGB vertices are rendered directly from the supplied PLY files; blue camera centers and orientation glyphs come from the corresponding saved poses. Glyph dimensions are standardized for display, not calibrated fields of view. These are new views of existing reconstructions, not new solves. Framing and full-extent views are documented in the visualization supplement.

Abstract

Splatfiction Mapper brings camera recovery and sparse 3D reconstruction into a browser, moving an essential preparation stage for image-based graphics into a client-side workflow. Developed from a native CUDA SfM system, it combines compact observation storage, a resident matrix-free Schur bundle-adjustment route, explicit WebGPU/WebAssembly execution policies, and geometric and export validation. Its immutable observation payload is reduced from 40 to 16 bytes per factor, a 60% reduction in that payload. A retained same-build, five-scene landmark cohort registers all 521 input images, with per-scene solve times of 27.5–293.1 seconds. Separate retained qualifications report a 920-view synthetic geometry test and a 3,000-view reconstruction with 8.30 million observations and a checked COLMAP export. Run-matched PLY and camera-pose views connect the landmark measurements to inspectable outputs. This research edition separates those results from larger-scale R&D testing: a 5,822-view candidate was not accepted after global-consistency checks detected invalid geometry. The contribution is an integrated browser reconstruction architecture with explicit evidence for storage, execution, and output acceptance. Measurements apply to the tested desktop configurations; broader device coverage, responsive large-capture execution, and matched native comparisons remain validation milestones.

Keywords

Structure from motion, WebGPU, bundle adjustment, browser computing, sparse reconstruction, GPU computing

1 Introduction

Image-based graphics depends on a consistent relationship between photographs and three-dimensional scene structure. Camera poses and sparse points provide the coordinate system for optimizing a radiance field or Gaussian scene. For example, 3D Gaussian Splatting initializes from calibrated images and sparse reconstruction [4]. Browser viewing alone leaves this preparation step outside the browser. Splatfiction Mapper brings it into the client-side workflow: users supply photographs, reconstruct cameras and points, and obtain interoperable reconstruction files.

WebGPU exposes general-purpose GPU computation while making resource limits explicit [11]. SfM must manage correspondences, tracks, camera and lens variables, sparse adjacency, and iterative solver workspace. Compact storage addresses oversized bindings; resident iterative work limits host synchronization. Geometric validation complements both by checking the resulting reconstruction.

Splatfiction Mapper, formerly called Dragon, is one project developed from a native CUDA SfM system. Its browser im-

plementation redesigns storage, execution, and acceptance around the same task. It is distinct from the downstream Gaussian-splatting trainer: the outputs studied here are camera parameters, sparse points, observation tracks, and reconstruction files.

The design addresses three complementary requirements: representing a factor graph within device limits, checking global geometric consistency, and preserving reconstruction relationships through export. Each has its own evidence. Reprojection residuals measure local image agreement; global-consistency checks and serialization audits address different properties of the delivered model.

The contribution combines a compact, gather-oriented factor representation, a resident joint camera/calibration/point solver without a materialized camera-camera Schur matrix, and explicit numerical routing. GPU-eligible optimization and double-precision WebAssembly paths preserve supported camera-model semantics. Staged geometric and artifact checks then make acceptance auditable. This is a systems contribution built from established reconstruction algorithms, rather than a new feature-learning architecture or Schur elimination.

The evaluation separates same-build results, synthetic geometric validation, export qualification, and R&D extension tests. Figures 1 and 4 show run-matched geometry. These retained development records document achieved milestones and remaining tests, not production incident rates. Section 8 distinguishes normal solver safeguards from reconstruction outcomes that remain unaccepted.

2 Related Work

Classical and GPU-oriented SfM. COLMAP integrates robust geometric estimation, registration, triangulation, and refinement [10]. GLOMAP revisits global SfM for scalable reconstruction [8]; InstantSfM addresses GPU-native execution in a learning-oriented environment [14]. Our pipeline builds on verified view graphs, global pose estimation, track construction, and bundle adjustment. The emphasis is browser execution with preserved reconstruction and export semantics. Matched comparative timing remains a separate milestone.

Nonlinear optimization on the GPU. Safari [9] studies matrix-free shared-intrinsics bundle adjustment and float32 stability. Our design builds on this algorithmic family through packed, gather-oriented browser execution, explicit numerical routing, and terminal acceptance.

Caspar generates CUDA kernels and GPU nonlinear solvers through symbolic programming [7]. Its published solver uses full normal equations. Our native lineage includes a Caspar-backed visual route whose variable, retraction, and damping conventions inform the browser design. The browser instead uses a matrix-free Schur operator, gather accumulation, and a Huber objective. Schur reduction and Levenberg–Marquardt optimization are established methods; the contribution is their browser-system integration.

Features and matching. ALIKED provides learned keypoints and descriptors [13]; LightGlue provides learned

sparse matching [5]; SIFT is an alternative scale-invariant feature route [6]. Execution records identify the active path: four main-cohort scenes use ALIKED with GPU cosine mutual nearest neighbors, and one uses WebGPU SIFT recovery. A configured model name alone does not establish execution.

SfM in the browser. Splat.js documents browser feature extraction, WebGPU matching, incremental registration, and sparse Schur refinement [1]. Within this existing browser-SfM context, our focus is the combined design of factor residency, camera/sensor identity, numerical routing, and geometric and artifact qualification. Public descriptions provide architectural context, not measured baselines.

3 Problem and System Contract

Let an observation $i \in \mathcal{O}$ associate image frame $f(i)$, physical sensor $s(i)$, calibration identity $k(i)$, point $p(i)$, and observed pixel z_i . We estimate frame poses, calibration parameters, and points. A fixed sensor-from-rig transform allows multiple sensors to share a frame without collapsing their baselines. The projection chain is

$$\mathbf{x}_{r,i} = \mathbf{R}_{f(i)} \mathbf{X}_{p(i)} + \mathbf{t}_{f(i)}, \quad (1)$$

$$\mathbf{x}_{s,i} = \bar{\mathbf{R}}_{s(i)} \mathbf{x}_{r,i} + \bar{\mathbf{t}}_{s(i)}, \quad (2)$$

$$\mathbf{r}_i(\boldsymbol{\theta}) = \pi_{k(i)}(\mathbf{x}_{s,i}) - \mathbf{z}_i. \quad (3)$$

An ordinary single-camera image set uses one frame per image with identity sensor extrinsics. Images that share a COLMAP camera identity share a calibration node. Physical sensor identity and calibration identity remain separate: distinct fixed sensor transforms may reference one shared lens parameter block.

The input contract permits photographs, EXIF information, and explicitly declared source calibration or rig constraints. Evaluation references are recorded separately from authorized solve inputs. The 3,000-view qualification uses an explicit camera prior, whereas the 920-view geometric protocol evaluates a saved reconstruction against synthetic truth. They therefore qualify different capabilities: prior-assisted execution and export at scale, and geometric agreement under a known-reference protocol. The supplied calibration remains a declared input, not a self-calibration result.

For monocular images without a known baseline or scale constraint, reconstruction has a similarity gauge. Anchoring one frame removes rigid coordinate freedom but does not alone determine metric scale. Our comparison procedure therefore aligns estimated and reference camera centers by a similarity transform. Millimeter units are meaningful only where the reference itself has verified metric units.

The system defines success at two levels. Execution completion means that a selected route reaches a terminal state. Publication acceptance means that the output also passes its configured geometric and serialization checks. Here, “publication” means making a model persistable or exportable inside the application, not scientific publication or a proof of geometric correctness.

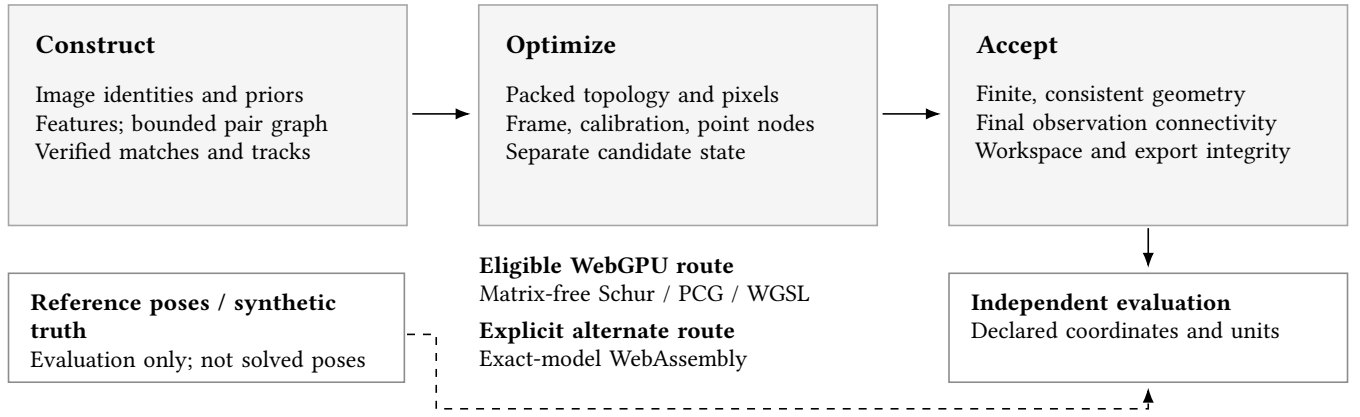


Figure 2: Browser execution and acceptance. Eligible graphs enter the resident WebGPU solver; alternate WebAssembly routes are explicit. Declared source priors are solve inputs, whereas reference poses and synthetic truth enter a separate evaluation path. Output acceptance is distinct from objective decrease.

4 Browser-Resident Reconstruction

4.1 Pipeline and native lineage

Photographs are decoded and assigned immutable image identities. The front end extracts features, builds a bounded candidate-pair graph, verifies correspondences, and forms tracks. Geometric stages estimate calibration and relative rotations, recover global positioning, triangulate structure, and attempt supported camera recovery. Joint and point refinement operate on the resulting graph. Terminal stages audit geometry, persist an accepted workspace, and write a COLMAP-compatible model. Figure 2 summarizes this execution and acceptance structure.

The browser uses JavaScript and TypeScript for orchestration, WGSL kernels through WebGPU for selected parallel stages, and WebAssembly for native-derived numerical routines. This hybrid design selects a numerical route appropriate to the supported model and workload. Garden and Salzburg Altstadt identify WebAssembly bundle phases; the other latest registry rows label that phase WebGPU. The resident route in Section 5 keeps eligible joint iterative work on WebGPU. An intentionally selected WebAssembly route is a supported execution policy, not by itself a failed GPU solve. Each timing is attributed to the route that actually ran.

The native implementation contains an ordered CUDA graduated bundle adjuster and a separate Caspar-backed visual corrective route. The browser resident solver derives its variable organization from the latter, not from the ordered CUDA engine. We do not treat these native routes as interchangeable, nor use unmatched native timing records to claim CUDA/WebGPU parity.

4.2 Matching and track construction

The front end caches descriptors on the device and limits work through a selected view graph instead of materializing every possible image pair. Learned descriptors can be matched using GPU cosine mutual nearest neighbors; learned matching and SIFT-based routes are selected where supported by policy. Pairwise geometric verification pre-

cedes track creation. Track assembly resolves competing observation assignments rather than equating a large raw match count with a reliable point graph.

Recovery is consequential because repetitive scenes can produce a connected match graph with an incorrect calibration basin. The system revisits geometric support and may change an eligible feature or initialization route. It must then rebuild or revalidate downstream tracks: registration support before optimization is not sufficient evidence of co-visibility after observation filtering. The larger-scale Small City R&D case in Section 7.6 illustrates the importance of this terminal check.

4.3 Compact observation storage

The prior host-facing observation layout used ten 32-bit values, or 40 bytes per observation. The resident representation separates immutable topology and pixels from changing solver state. Two unsigned 32-bit words encode point, frame, sensor, and flag fields; two 32-bit floats store the observed pixel. The immutable factor payload is therefore 16 bytes, a 60% reduction relative to the prior layout. Figure 3 isolates this comparison. Adjacency, candidate states, normal blocks, reductions, descriptors, and image storage remain additional costs.

Observations are ordered by point, with compressed adjacency for frame and sensor traversal. A point identifier occupies 30 bits with two flag bits; a second word packs a 22-bit frame identifier and a 10-bit sensor identifier. Packing validates these bounds. Calibration nodes are addressed through a separate sensor-to-calibration map, preventing packed sensor identity from accidentally becoming an independent lens variable.

Each reduction gathers its contributing observations. This avoids reliance on portable floating-point atomic addition and supplies a fixed ownership rule for accumulation. High-degree calibration work is segmented into groups of at most 2,048 observations. Rather than create one observation record containing every intermediate quantity, the design maintains separately bound arrays and recomputes se-

Immutable observation payload

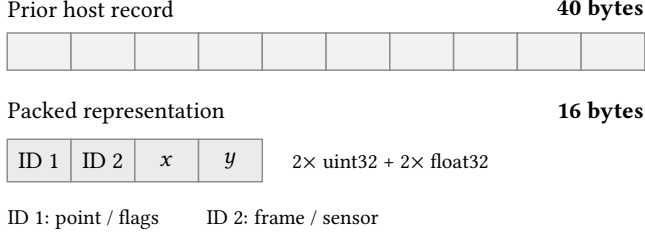


Figure 3: Immutable payload layout. Ten prior 32-bit slots become four words: two packed identities and two observed pixel coordinates. The 40-to-16-byte change describes only the immutable payload, not the complete resident workspace.

lected Jacobian terms as needed. WebGPU limits are checked against the actual requested buffers [11].

A retained intermediate graph contains 7,901,872 observations and 423,314 points. Its identifier and pixel arrays each require 63,214,976 bytes; frame and sensor ordinal arrays each require 31,607,488 bytes. These individual allocations fit below 128 MiB. They do not imply that the application uses less than 128 MiB, or that every capture fits an adapter with that binding limit. With bounded-degree node blocks, the represented factor and working graph scale as $O(|O| + P + F + K)$; image, feature, and matching storage are separate.

5 Resource-Aware Joint Optimization

5.1 Objective and variable semantics

The eligible resident route supports PINHOLE and SIMPLE_RADIAL. Pinhole calibration optimizes independent f_x, f_y while fixing the principal point. Simple radial calibration optimizes tied focal length f and coefficient k_1 , with normalized radius $r^2 = x^2/z^2 + y^2/z^2$ and multiplier $1 + k_1 r^2$. Principal points remain fixed. A model outside this set is not silently replaced by a pinhole approximation.

We minimize the vector-residual Huber objective

$$E(\theta) = \sum_{i \in O} \rho_a(\|r_i(\theta)\|_2), \quad (4)$$

where

$$\rho_a(s) = \begin{cases} \frac{1}{2}s^2, & s \leq a, \\ a(s - \frac{1}{2}a), & s > a. \end{cases} \quad (5)$$

At each linearization, the IRLS weight is one inside the Huber transition and a/s outside it. The same robust objective is evaluated when accepting a candidate, so a weighted local model is not mistaken for a decrease in the nonlinear objective. Frame rotations use right-multiplicative retraction $R \leftarrow R \exp(\delta\omega)$; translations and eligible calibration parameters use additive updates.

WGSL arithmetic on this route is float32, with optional float16 use elsewhere in feature matching. The browser retains explicit double-precision WebAssembly paths for numerical tasks and unsupported camera models. This division

follows available execution types; it is not a claim of bit-identical CUDA, WebAssembly, and WGSL results [12]. Our Huber policy also differs from the raw least-squares factors in the referenced Caspar route.

5.2 Resident matrix-free Schur system

Partition the increment into camera-side variables δc (frame and calibration nodes) and points δp . The damped normal system is

$$\begin{bmatrix} B & E \\ E^T & C \end{bmatrix} \begin{bmatrix} \delta c \\ \delta p \end{bmatrix} = \begin{bmatrix} b_c \\ b_p \end{bmatrix}, \quad (6)$$

where C is block diagonal over points and the right-hand side uses the negative weighted gradient. Eliminating points gives

$$S\delta c = b_c - EC^{-1}b_p, \quad (7)$$

$$Sv = Bv - E[C^{-1}(E^T v)]. \quad (8)$$

The operator is evaluated from observation adjacencies rather than a stored camera-camera matrix. Point contributions are gathered in point order, transformed by their 3×3 block inverse, and gathered back into frame and calibration nodes. The reduced operator retains cross-node effects even though the preconditioner uses small independent blocks. Points are back-substituted after the solve:

$$\delta p = C^{-1}(b_p - E^T \delta c). \quad (9)$$

Block preconditioning uses 6×6 frame blocks and 2×2 calibration blocks, with a guarded diagonal fallback when block factorization is unusable. Recomputing Jacobians trades arithmetic for lower resident storage and avoids synchronizing a complete factor Jacobian through JavaScript. This is an implementation tradeoff, not an experimentally established optimum for every GPU.

5.3 Bounded submission and guarded updates

The retained resident policy sets nonlinear and PCG iteration caps to 20. GPU scalar flags suppress further updates after convergence or invalid arithmetic. The default PCG exit uses a squared-residual ratio threshold of 10^{-4} , with tighter caller policies possible. A squared-norm ratio of 10^{-4} is not a relative norm tolerance of 10^{-4} ; the benchmark export does not independently establish the initial-residual and zero-residual handling. Two PCG iterations are submitted per chunk; immutable uploads are split into writes of at most 16 MiB, with yield points between bounded units of host work.

Current and candidate states are separate. Retraction, objective evaluation, and finite-value checks precede a state swap. A candidate is committed only when it has a finite, strictly lower robust objective. A non-improving trial step increases damping while preserving the current state: this is normal safeguarded optimization, not a failed reconstruction. Nonfinite arithmetic remains a numerical diagnostic, distinct from convergence. Selected damping conventions follow the native visual route, while browser execution and the reduced linear operator differ. A partially written candidate cannot replace the current reconstruction.

Capacity validation checks storage binding sizes, buffer sizes, packed-index limits, and dispatch bounds before admitting a graph to the resident route. Models ineligible by camera type use an explicitly identified exact-model, fixed-intrinsic WebAssembly route. Once a graph selects the GPU route, an unexpected shader or capacity failure remains an explicit execution issue; it is not relabeled as successful alternate-route selection or hidden behind an unexpectedly long single-threaded solve.

Chunking provides scheduling and cancellation opportunities within the resident loop. Whole-application responsiveness is a separate productization milestone: the retained 3,000-view development build records a 117.4-second synchronous blocked phase. That observed interruption remains relevant to interactive use; bounded solver submissions alone do not resolve it.

5.4 Inspecting points and cameras

The inspection surface is separate from the reconstruction kernels. It projects stored points and cameras for scene review and exposes image-pair correspondences. Display grouping and glyph sizing affect inspection, not saved poses. A plausible viewport does not replace geometric or artifact checks.

For the paper figures, original RGB vertices are read from the supplied ASCII PLY exports and joined to camera records by exact run ID. The accompanying viewer documents the quaternion order as (q_w, q_x, q_y, q_z) and uses world-to-camera extrinsics. Thus

$$\mathbf{x}_c = R\mathbf{x}_w + \mathbf{t}, \quad C_w = -R^T\mathbf{t}. \quad (10)$$

The forward and up directions are $R^T(0, 0, 1)^T$ and $R^T(0, -1, 0)^T$. We render every saved camera center and orientation. Camera glyphs use one standardized template per scene because the visualization manifest does not contain lens intrinsics. Their outlines are not calibrated fields of view, and no line connecting cameras is added as an inferred trajectory.

The paper views use orthographic projection, original point colors, nearest-depth point rasterization, and overlaid camera glyphs. Points and cameras share one viewing transform. Display framing does not delete or filter source geometry; off-frame point counts, exact view transforms, glyph scales, source hashes, and full-extent views are retained in the supplement. Neither these figures nor their visual density establish new geometric accuracy or surface completeness.

6 Geometric and Artifact Acceptance

Output acceptance combines three layers of evidence. The first checks finite values and objective decrease during optimization. The second examines the terminal reconstruction without ground truth: reprojection residuals, positive-depth and camera-model checks, pairwise rotational or epipolar consistency, and connectivity in the final observation graph. The third is an external evaluator comparing the saved result with a declared reference or synthetic truth. Together,

these separate local numerical progress from the geometric properties required of an accepted output.

Connectivity is evaluated on the point-camera incidence structure surviving terminal filtering. A view-graph edge retained before bundle adjustment need not imply meaningful shared geometry afterward. In one retained gate, camera covisibility is constructed from at least five common points. A disconnected terminal graph is rejected even when every image possesses a pose record. These checks reduce particular failure modes; they do not prove the absence of every global deformation or erroneous correspondence.

Export is a further contract. Camera IDs, frame IDs, image names, point IDs, and bidirectional observation references must survive serialization. A point track referring to an image observation must agree with the image’s point assignment. The 3,000-view qualification includes an independently parsed COLMAP text archive with cameras, rigs, frames, images, and points. This is stronger than asserting that an export button completed, but remains separate from independent geometric accuracy.

For reference evaluation, matched camera centers are aligned by a least-squares similarity transform. The registry multiplies the center discrepancy by 1,000 and stores it under a field ending in Mm. The field name does not establish the underlying reference unit. TerraSky3D supplies COLMAP reconstructions with pre-calibrated cameras [3]; those centers are not treated here as an independently surveyed metric trajectory. We report rotation discrepancy in degrees and preserve the center fields as reference-coordinate discrepancies in the supplement. Synthetic AP01 has known metric geometry, but its similarity-aligned pose error still does not demonstrate recovery of absolute monocular scale.

7 Technical Validation

7.1 Evidence and protocol

The primary registry export, E1, was generated on 22 September 2026 and contains 16 completed runs from 8–10 September across eight scenes and seven build labels. All 16 are marked accepted. This selected completed-record archive supports inspection of those outcomes; it is not a complete attempt history and cannot establish a general success or production reliability rate. The supplemental ledger retains every record. Table 1 selects the latest timestamp per scene, not the fastest run. Its eight numerical rows reproduce the supplied export at the displayed precision.

The five-scene main cohort uses build 1.31.0, commit 6f4ca0e, and an NVIDIA RTX PRO 4500 Blackwell. Host CPUs vary between AMD EPYC 7443 and 7663, so this is a same-build/GPU-model cohort, not an identical-host experiment. Three older latest-per-scene runs are shown separately: two TerraSky3D scenes on that GPU and Garden from Mip-NeRF 360 [2] on an RTX 4090.

The registry harness uses Linux Chromium with Vulkan and non-default flags, including unsafe WebGPU and disabled sandbox options. It is not a qualification of every stock browser or default security configuration. Browser flags, hardware labels, build identities, and timing fields are

Table 1: Latest retained run per scene (E1). The first five rows form the main same-build cohort (1.31.0, commit **6f4ca0e, RTX PRO 4500 Blackwell); the last three are historical context. Every row registers all N input images. Reprojection statistics use the recorded evaluation raster. Rotation P95 is agreement with a COLMAP reference after similarity alignment, not surveyed accuracy. Solve time is not upload-to-export latency.**

Scene	N	Points	Observations	Solve (s)	Median (px)	P95 (px)	Rot. P95 (°)
Vienna State Opera	106	46,429	224,814	293.139	0.349	1.294	0.155
Munich Theatine	120	54,909	304,921	174.544	0.752	2.308	0.237
Munich Frauenkirche	123	53,104	398,756	238.907	0.870	2.113	0.123
Salzburg Andrakirche	114	51,158	295,799	189.138	0.854	2.349	0.238
Salzburg Altstadt	58	16,110	78,095	27.512	0.561	1.373	0.186
<i>Historical context: Graz 1.30.96 / Blackwell; Garden 1.30.68 / RTX 4090</i>							
Graz Church	239	93,537	651,241	227.919	0.934	2.312	0.148
Graz Townhall	335	132,057	762,230	446.903	0.983	1.542	0.166
Garden	185	78,115	391,348	92.285	0.835	1.651	0.124

retained in the sanitized supplement. The solve interval is the producer’s elapsed reconstruction time, not upload-to-download latency; it does not establish complete data-staging and export time. There are no repeated trials under one frozen configuration, so we report neither confidence intervals nor release-to-release speedups.

E2–E5 are separate retained qualifications with their own protocols and hardware. Their summaries and the implementation description follow the supplied manuscript [15]; the original E2–E5 source reports and solver source are not part of the present evidence packet. These qualifications are therefore retained reported results, not independently revalidated by the E1/PLY audit. Table 2 keeps their scopes separate. No reconstruction was rerun for this manuscript. **Visual evidence.** The reference bundle contains 62 PLY files, of which 15 have entries in its current camera manifest. All 15 manifest-linked vertex counts agree with their recorded point counts. We use the seven run-matched, latest-per-scene landmark entries, including the five main-cohort runs. Every selected PLY vertex count and camera count agrees with its E1 record. Unlinked PLYs are not treated as additional accepted results.

Figures 1 and 4 show the five-scene main cohort. The supplement adds the two historical Graz reconstructions, camera-only views, and full-extent point views. The historical Room and Garden inspection captures are preserved separately: Room has 311 registered images and 94,041 points; Garden has 185 registered images and 79,517 points. Neither has a matching PLY/camera pair in this reference bundle. In particular, the 79,517-point historical Garden capture is not the 78,115-point registry run in Table 1.

7.2 Landmarks and historical context

All 521 images in the five-scene main cohort are registered. Per-scene solve time ranges from 27.512 seconds for Salzburg Altstadt to 293.139 seconds for Vienna State Opera. Median reprojection error ranges from 0.349 to 0.870 pixels; the largest P95 is 2.349 pixels. P95 rotation discrepancy against the aligned COLMAP references ranges from 0.123° to 0.238°. These measurements quantify agreement with the aligned COLMAP references; independently surveyed pose accuracy is a separate evaluation target.

The eight latest-per-scene records collectively cover 1,280 distinct input-image instances across the selected scenes. Complete registration and subpixel median residuals show useful browser reconstruction outcomes. Hardware and release differences prevent interpreting these rows as a controlled performance ranking. Vienna executes SIFT matching/recovery; the other selected registry rows report cosine mutual-nearest-neighbor matching of learned descriptors.

The registry makes the hybrid execution policy explicit. Garden spends 19.153 seconds in extraction, 18.926 in matching, 13.056 in geometry, and 39.167 in its WebAssembly-labeled bundle phase, within a 92.285-second solve. Salzburg Altstadt spends 6.343, 4.109, 5.428, and 11.042 seconds in those phases, respectively. These two records measure WebAssembly bundle routes, not the resident Schur solver.

Vienna, Frauenkirche, and Andrakirche explicitly record the `webgpu-caspar-matrix-free-schur-pcg` backend in bundle epochs. Munich Theatine also records that backend with float32 precision in final calibration/recovery rounds, although its main bundle-epoch list is empty. This is route-specific evidence of execution, not attribution of every reported bundle-phase second to an identical graph or optimization schedule.

Figure 5 summarizes recorded phase accounting. Repeated geometry/recovery entries are summed; the remaining categories include pairing, setup, and quality phases. The producer solve interval differs slightly from the phase sum, and model setup is labeled overlapped. The figure is consequently an accounting of reported phases, not a verified disjoint partition of wall-clock time.

7.3 Synthetic geometry: 920 views

E2 qualifies AP01-920 at build 1.30.63, commit 7dab689, on physical Chrome 152 under Windows 11 with an NVIDIA discrete adapter and 32 GiB of host memory. The sequence contains 46 camera centers and 20 headings per center. Synthetic camera and surface truth evaluate the saved solve rather than being supplied as solved poses. All 920 images are registered in 548.576 seconds.

The strict sparse output contains 5,980 points and 76,083 observations. Its reprojection median is 0.382 pixels and P95 is 1.129 pixels. After similarity alignment to the synthetic

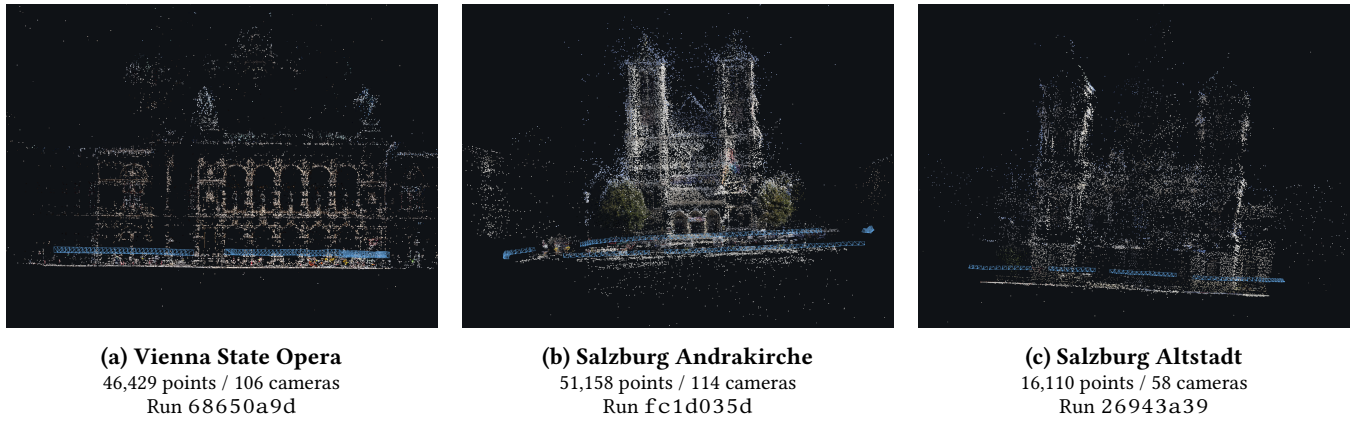


Figure 4: The remaining three main-cohort reconstructions, complementing Figure 1. Original RGB PLY points and all saved camera poses share the same viewing transform. Blue glyphs indicate camera placement and orientation; no connecting trajectory is synthesized. Views are independently framed and are not at a common scale. The visualization supplement provides larger versions, camera-only views, off-frame vertex counts, and full-extent views.

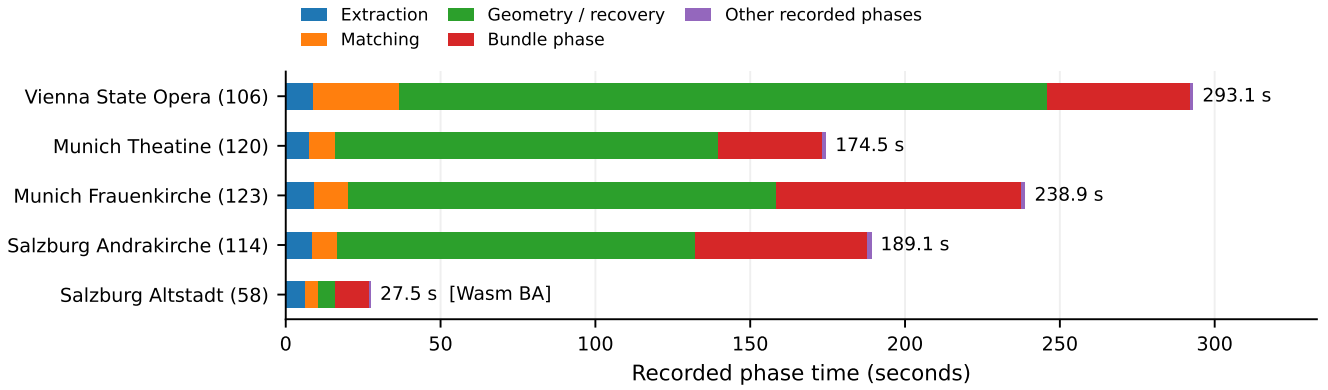


Figure 5: Recorded phase accounting for the five-scene 1.31.0 cohort, in Table 1 order. Repeated geometry/recovery entries are summed; “other” includes pairing, setup, and quality phases. Labels give the producer’s solve interval. Model setup is reported as overlapped, so these bars are not a verified disjoint wall-clock partition. Altstadt’s bundle phase is WebAssembly; the other phase labels are WebGPU. Host and route differences prevent a controlled speed ranking.

trajectory, camera-center median error is 0.617 mm and P95 is 1.966 mm; rotation median is 0.0209° and P95 is 0.0340°. The estimated focal length is 873.031 pixels versus a reference of 873.067 pixels. Against rendered depth geometry, point-to-surface median discrepancy is 16.549 mm and P95 is 42.107 mm.

This qualification targets sparse camera-and-point recovery in one structured, repeated-center apartment capture. It does not target a complete dense surface: reverse surface recall at 100 mm is 14.42%. A separate loose rendering seed contains 58,513 points and is distinct from the strict model carrying the reported accuracy. Broader indoor generalization and surface completeness therefore require their own evaluations.

7.4 Scale and export: 3,000 views

E3 uses the same 1.30.63 commit on the corresponding Windows and Chrome qualification setup, but with an explicitly supplied pinhole calibration prior for 540 × 960 images. It registers all 3,000 images and produces 826,824 points with 8,297,892 observations. The solve takes 2,460.966 seconds,

or 41.0 minutes. Reprojection median is 0.687 pixels and P95 is 2.493 pixels. This is a different capture and protocol from E2.

The exported archive is 576,619,147 bytes and contains 3,005 entries: 3,000 images plus five COLMAP text model files. Its retained parser report validates observation relationships and the one-rig, one-camera, 3,000-frame structure. The complete qualification, including export, takes approximately 42.8 minutes. This demonstrates execution and artifact consistency on the tested host. No independent 3,000-view trajectory or surface truth is supplied, so it does not inherit the millimeter-level AP01-920 result.

The development build completes without a reported device loss or page crash, with a sampled JavaScript heap of approximately 4.47 GB and a 117.4-second synchronous phase that blocks interactive use. Host-memory demand and responsiveness remain productization tasks. The record does not qualify lower-memory deployment or any later scheduling fix. Whole-application heap demand is distinct from the per-buffer analysis in Section 4.3.

Table 2: Retained qualifications and R&D status (E2–E5). E2 uses synthetic metric truth; E3 establishes prior-assisted execution and export, without independent geometric truth at that scale. E4 identifies a global-consistency limit: its reconstruction is not accepted. E5 supplies earlier solver-stage evidence, not an extra same-configuration E3 trial. Values follow the supplied manuscript; original E2–E5 reports are outside this packet.

Record	Build	Registered	Points	Observations	Med./P95 (px)	Supported outcome
AP01-920 (E2)	1.30.63	920/920	5,980	76,083	0.382/1.129	Geometric qualification
AP01-3000 (E3)	1.30.63	3,000/3,000	826,824	8,297,892	0.687/2.493	Execution and export
Small City (E4)	1.31.2	5,822/5,822	662,533	4,524,228	0.931/1.683	Not accepted
AP01-3000 (E5)	1.30.61	—	330,157	4,369,032	—	Resident-stage evidence

E5 point and observation counts describe its terminal strict model; its registration total is not independently tabulated in the retained summary. Its 12.316-second joint-stage time belongs to an earlier incoming graph with 226,188 points and 1,241,749 observations; the full solve takes 1,767.183 seconds. E2 reports the strict sparse model, not the separate loose rendering seed.

7.5 Direct evidence of resident solver execution

E5 is an earlier AP01-3000 qualification at build 1.30.61, not a new dataset or an additional trial of E3 under the same configuration. It explicitly records the float32 matrix-free Schur/PCG backend. Three joint epochs take 12.316 seconds in total; the recorded incoming graph contains 226,188 points and 1,241,749 observations. Subsequent pruning and recovery change topology, and the terminal strict model contains 330,157 points and 4,369,032 observations. The complete solve takes 1,767.183 seconds.

The 12.316-second value therefore verifies a resident joint-solver stage on its recorded graph. It is neither an end-to-end 3,000-image time nor a solve of E3’s final 8.30-million-observation graph. It also cannot establish a speedup over E3, whose topology and recovery work differ.

7.6 Large-scale R&D validation: 5,822 views

E4 records the Small City development qualification at build 1.31.2 on Chrome 153 with an RTX PRO 4500 Blackwell. It tests a larger capture than the retained 3,000-view execution-and-export qualification. All 5,822 images acquire poses, with 662,533 points and 4,524,228 observations. Median/P95 reprojection errors are 0.931/1.683 pixels. These local measures describe numerical completion; the reconstruction does not meet global-consistency acceptance criteria.

Terminal checks identify epipolar disagreement in 2,536 of 28,171 tested edges, or 9.002%, above the configured 8% limit. Final five-shared-point covisibility separates into components of 4,276 and 1,546 cameras. The focal trajectory enters an incorrect basin, and reference rotation P95 reaches 159.686° . The model is rejected and no valid export is established. A later harness failure is a separate recorded event; it does not explain away the geometric inconsistency.

The engineering outcome is a detected global-consistency limit in a larger-scale R&D test, not a validated 5,822-view reconstruction. The case illustrates why acceptance checks extend beyond image registration and local residuals. The retained manuscript describes subsequent transactional repair changes, but their effectiveness awaits a physical rerun. The next milestone is an accepted reconstruction under the same declared geometric and export checks; source changes alone do not qualify that outcome.

8 Technology Maturity and R&D Roadmap

8.1 Demonstrated capability and status

The records demonstrate browser-resident SfM on the tested discrete-GPU desktops. The E1 cohort connects complete registration and measured solve times to inspectable PLY geometry and cameras. Separate retained qualifications address known-reference sparse geometry, 3,000-view execution and export, and resident-solver execution. These are distinct milestones, not interchangeable measures of accuracy, scale, or speed.

Development outcomes also differ. Declining a finite, non-improving trial step is normal optimizer behavior; selecting an intended WebAssembly route is part of the hybrid design. Neither is itself a failed reconstruction. In contrast, E4 contains geometrically inconsistent output that was not accepted, and E3 records a substantial synchronous interruption. These define large-scale consistency and responsiveness work, respectively. Their R&D context does not establish that they have been resolved. The records do not measure production incident rates.

8.2 Next validation milestones

Large-capture consistency. Requalify the described transactional repair on Small City using final connectivity, epipolar consistency, calibration behavior, reference agreement, and export checks. An accepted physical rerun is the milestone; source changes alone do not replace the non-accepted E4 result.

Responsiveness and device coverage. Reduce long synchronous host work and measure memory demand across complete workflows. The 117.4-second pause and approximately 4.47 GB sampled heap are starting observations, not resolved conditions. Broader qualification should cover stock-security browser configurations, other GPU vendors, integrated GPUs, and lower-memory or mobile devices. The current registry uses non-default Chromium; large qualifications use desktop Windows/NVIDIA hardware.

Generalization and comparative performance. Extend reference evaluation beyond the structured AP01 apartment, separating sparse accuracy from dense surface coverage. Compare native/browser execution with matched images, resize policies, calibration inputs, feature budgets, repeated trials, and timing boundaries. Distinguish end-to-end from solver-stage timing when graph topology differs. A runtime

benefit attributable to packing also requires a controlled study. These are proposed tests, not completed qualifications or delivery commitments.

8.3 Evidence for technical due diligence

The package retains original selected PLYs, camera records, run identities, source hashes, rendering parameters, the sanitized E1 ledger, and figure/audit scripts. Figure assets are unchanged from v2: original colors and duplicate vertices are preserved, without surface fitting, altered poses, or invented trajectories. Glyphs are inspection overlays, not calibrated fields of view. The atlas discloses off-frame counts and full extents; historical Room/Garden captures remain separate.

Method details and E2–E5 measurements remain attributed to the retained manuscript: original qualification reports and solver source are outside this packet. E1 is separately re-audited. Build-matched source, original reports, reference geometry, and controlled reruns are needed to close those verification gaps.

9 Conclusion

Splatfiction Mapper combines compact factor storage, resident joint optimization, explicit hybrid numerical routes, and geometric and artifact acceptance in a browser-resident SfM system. The retained record includes all 521 images across five same-build landmark scenes, a synthetic 920-view geometric qualification, and a 3,000-view execution and checked-export qualification. Direct PLY and saved-pose views make the landmark outputs inspectable.

The next stage is to qualify broader device coverage, improve whole-application responsiveness, and demonstrate accepted geometry on larger captures. The 5,822-view R&D result remains outside the demonstrated accepted-reconstruction scale. Progress is measured by geometric and export acceptance, not registration counts alone.

Authoring disclosure

A language model assisted with manuscript editing, evidence-processing scripts, figure preparation, and consistency checks. Reconstruction measurements originate from retained system records. The new scene figures are deterministic renderings of supplied PLY vertices and saved camera poses, not generated reconstruction content or new SfM experiments. Technical authorship, final verification, and publication decisions remain with the author.

References

- [1] Arrival.Space. Splat.js. Software repository, 2026. <https://github.com/arrival-space/splat.js>. Browser SfM implementation; Accessed 22 September 2026.

- [2] Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. Mip-NeRF 360: Unbounded Anti-Aliased Neural Radiance Fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.
- [3] Mattia D’Urso, Yuxi Hu, Christian Sormann, Mattia Rossi, and Friedrich Fraundorfer. TerraSky3D: Multi-View Reconstructions of European Landmarks in 4K. arXiv:2603.28287v1, 2026.
- [4] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Transactions on Graphics*, 42(4), Article 139, 2023.
- [5] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. Light-Glue: Local Feature Matching at Light Speed. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.
- [6] David G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. doi:10.1023/B:VISI.0000029664.99615.94.
- [7] Emil Martens, Aaron Miller, Matias Varnum, and Annette Stahl. Caspar: CUDA Accelerator for Symbolic Programming with Adaptive Reordering. arXiv:2605.30583, 2026.
- [8] Linfei Pan, Dániel Baráth, Marc Pollefeys, and Johannes L. Schönberger. Global Structure-from-Motion Revisited. In *European Conference on Computer Vision*, 2024. arXiv:2407.20219.
- [9] Daniel Safari. Matrix-Free Shared Intrinsic Bundle Adjustment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27017–27026, 2025. doi:10.1109/CVPR52734.2025.02516.
- [10] Johannes L. Schönberger and Jan-Michael Frahm. Structure-from-Motion Revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4104–4113, 2016. doi:10.1109/CVPR.2016.445.
- [11] W3C GPU for the Web Working Group. WebGPU. W3C specification, 2026. <https://www.w3.org/TR/webgpu/>. Accessed 22 September 2026.
- [12] W3C GPU for the Web Working Group. WebGPU Shading Language. W3C specification, 2026. <https://www.w3.org/TR/WGSL/>. Accessed 22 September 2026.
- [13] Xiaoming Zhao, Xingming Wu, Weihai Chen, Peter C. Y. Chen, Qingsong Xu, and Zhengguo Li. ALIKED: A Lighter Keypoint and Descriptor Extraction Network via Deformable Transformation. *IEEE Transactions on Instrumentation and Measurement*, 72:1–16, 2023. doi:10.1109/TIM.2023.3271000.
- [14] Jiankun Zhong, Zitong Zhan, Quankai Gao, Ziyu Chen, Haozhe Lou, Jiageng Mao, Ulrich Neumann, Chen Wang, and Yue Wang. InstantSfM: Towards GPU-Native SfM for the Deep Learning Era. arXiv:2510.13310v3, 2026.
- [15] Tomáš Sikora. *Splatfiction Mapper: Browser-Resident Structure from Motion under WebGPU Resource Constraints*. Retained nine-page manuscript, supplied as `Splatfiction_Mapper_Visual_Correction_Preview.pdf`, 2026. Source of the method description and E2–E5 qualification summaries; companion E1 registry and point-cloud reference bundles exported 22 September 2026.