

Reaigen-VGGT: Toward Feed-Forward Spatial Reconstruction

RGB-to-Geometry for Spatial Intelligence

Tomáš Sikora • Reaigen s.r.o., Slovakia



Figure 1: Single-scene prototype evidence. Crops from the supplied viewer recording at 00:00.5; image content is unchanged. The colored point display and trajectory are shown beside the selected RGB view and predicted depth. The selected preview is `view_067.png`; the accumulated display contains 214 observed frames. These are qualitative model outputs, not reference geometry or accuracy measurements [1, 2].

214 frames

One recorded inference trace

6.60 frames/s

Mean streaming throughput

5.01 GiB

Peak reserved GPU memory

RTX 4070 Ti | 294×518 RGB inputs | Same training capture | Streaming inference

Executive Overview

Reaigen is developing an RGB-to-geometry system intended to turn ordered images into progressively published cameras, depth, and spatial structure. The product direction is a practical reconstruction layer for spatial intelligence: learn from calibrated indoor data, then infer geometry from RGB without fitting a new model to each deployment scene.

The program already has a recorded execution milestone. One supplied trace processes 214 frames in 54 chunks over 32.44 seconds on an RTX 4070 Ti, with 4.323 GiB peak allocated and 5.010 GiB peak reserved GPU memory [2]. The accompanying recording makes the geometry-publication and visualization path inspectable. This establishes a concrete prototype operating point on the reported machine.

The checkpoint has been fitted to the same capture used in this pilot. Geometric quality has not yet been evaluated or accepted, and unseen-scene performance remains an open research objective. Full training on the intended eight-H200 cluster and evaluation over the full prototype dataset have not been completed. These are the next validation stages, not results implied by the preview.

Technology Thesis

The hypothesis is that **calibrated indoor supervision, causal VGGT-style reconstruction, and resource-aware history management can jointly deliver useful unseen-scene geometry within a practical deployment budget**. The planned study tests data utility, geometric consistency, and sequence-length scalability together.

Reaigen's opportunity is the integration of its data-production capability with a measured reconstruction prototype and explicit deployment controls. The next development phase should convert that integration into scene-disjoint accuracy results, reproducible training, and a characterized operating range. This paper presents the current evidence and the milestones required to reach that point.

Next Milestone

A versioned model-data release, the proposed eight-H200 training study, and a frozen unseen-scene evaluation. Success will be measured through depth and trajectory accuracy, surface coverage, whole-sequence completion, and system-level memory and latency—not through visual density alone.

1 Product Direction and Technical Positioning

1.1 From images to a usable spatial representation

The intended workflow is straightforward: supply an ordered RGB sequence, infer camera and surface geometry, and publish a progressively updated spatial representation. A downstream application could use that representation for scene inspection, measurement, or interaction. Those applications are product directions; the present record evaluates neither semantic understanding nor planning, navigation, or language-grounded reasoning.

The research focus is therefore reconstruction rather than a claim to complete spatial intelligence. A geometry layer must preserve useful relationships across viewpoints, maintain a meaningful scale convention, and remain computationally manageable as observations accumulate. A plausible point cloud is an early output of that process, not its final acceptance criterion.

VGGT provides the research foundation for feed-forward visual geometry [3]. Persistent-state and streaming formulations such as CUT3R and StreamVGGT provide relevant comparison settings [4, 5]. Reaigen’s thesis builds on that foundation. The current evidence does not establish a new backbone or a benchmark advantage over those systems; the planned contribution is a tested combination of calibrated supervision, streaming behavior, and deployment-aware state management.

1.2 Three complementary systems, distinct evidence

Reaigen’s broader program spans data generation, reconstruction, and appearance optimization. Frame Factory supplies an upstream route to reviewed, indexed RGB/depth records and cameras [6, 7]. Reaigen-VGGT targets geometry prediction from RGB. Splatfiction provides a separate browser-resident Gaussian training system for optimizing scene appearance [8].

This division gives the research program a concrete integration path: prepare controlled supervision, train and evaluate a geometric predictor, and explore an optional refinement interface. The supplied artifacts do not demonstrate an end-to-end connection between all three systems. In particular, Frame Factory’s production volume is not the measured training-set size of Reaigen-VGGT, and Splatfiction’s image-quality results are not reconstruction results for this model.

The prospective technical differentiation lies in the quality of these interfaces and the results they enable. Controlled data identity can support reproducible adaptation; explicit state policies can expose the quality–resource trade-off; matched evaluation can show whether the combination is useful. These are testable development advantages; comparative benefit remains to be established under matched evaluation.

2 System Design and Research Hypotheses

2.1 A declared RGB-to-geometry contract

Figure 2 separates the proposed training path from the reported inference path. At deployment, the intended contract keeps model parameters fixed while processing incoming RGB chunks and carrying historical state between them. New geometry is inferred rather than obtained by per-scene gradient fitting.

For an ordered chunk C_k and historical state M_{k-1} , the proposed interface is

$$(\hat{Y}_{C_k}, M_k) = F_\theta(I_{C_k}, M_{k-1}), \quad (1)$$

where θ is fixed at inference. A per-frame output is specified as

$$\hat{Y}_t = (\hat{K}_t, \hat{R}_t, \hat{\mathbf{t}}_t, \hat{D}_t, \hat{c}_t), \quad (2)$$

covering calibration, camera pose, depth, and confidence. This is an operational specification, not a requirement that the implementation store separate tensors with those exact names.

The report names an aggregator, camera head, depth head, learned tensors, and host descriptor history [2]. The supplied source archive documents dataset production and handoff, not the VGGT trainer or inference graph [7]. Equations in this edition therefore remain proposed contracts; exact module internals, trainable tensors, and effective losses require the version-matched model artifact.

Feed-forward operation can be stateful and chunked. The pilot contains 54 forward calls: an initial two-frame chunk and 53 four-frame chunks. The report describes publication as append-only, with earlier frames never refined [2]. That reported policy is consistent with the intended deployment contract, while an independent implementation review remains necessary to establish the complete internal behavior.

2.2 Three hypotheses that guide the program

H1: Calibrated-data utility. Adapting a matched model on calibrated indoor supervision should improve reconstruction on environments excluded from training. The decisive comparison separates new root-scene diversity from additional exposures to familiar views. Gains limited to the training capture would support scene-specific fitting, not the intended generalization claim.

H2: Causal geometric consistency. The model should publish useful camera and surface geometry using only the allowed observation history, while preserving consistency on later revisits. Causality and geometric consistency are separate tests: a causal model can drift, and a consistent reconstruction can have benefited from future observations or revision.

H3: Resource–quality compatibility. The system should complete a declared workload with acceptable geometry, coverage, host memory, GPU memory, and publication latency. A small GPU allocation alone is insufficient when historical state or queueing continues to grow elsewhere. The first target is the full 3,000-frame workload, followed by longer independent streams.

These hypotheses jointly define the intended operating point. A fast but inaccurate output does not meet it; neither does an accurate model that cannot complete the target sequence. Acceptance margins should be established on development data before the final test set is opened.

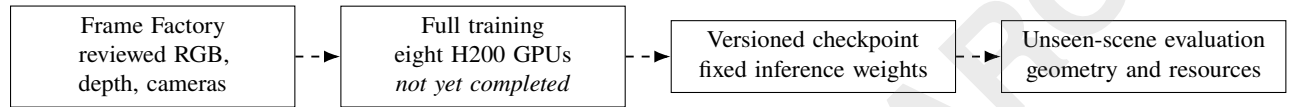
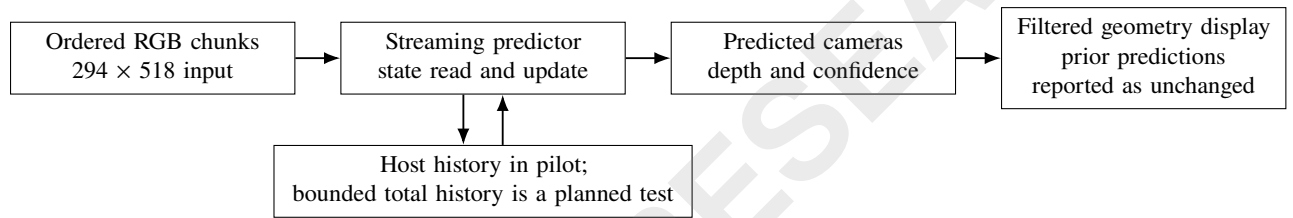
2.3 Geometry, scale, and information boundaries

Under a world-to-camera convention, $\mathbf{x}_c = R_t \mathbf{x}_w + \mathbf{t}_t$. With pixel ray $\mathbf{q} = K_t^{-1} \tilde{\mathbf{u}}$ normalized to $q_z = 1$, axial depth z gives

$$\mathbf{x}_w = R_t^\top (z\mathbf{q} - \mathbf{t}_t). \quad (3)$$

Table 1: Technology maturity at the supplied evidence snapshot. Recorded observations, proposed interfaces, and pending validation are separated. This is a development-status map, not a certification or a completed integrated-system benchmark.

Workstream	Evidence available	Next qualifying milestone
Calibrated data production	Frame Factory’s separate production record and dataset handoff documentation.	Bind a validated training release to the examples actually consumed by the model.
Learned reconstruction	One same-capture trace: 214 RGB frames, 54 chunks, geometry displayed on an RTX 4070 Ti.	Quantify depth, trajectory, and surface quality on unseen environments.
Streaming state	Host-resident descriptor history; measured device memory and growing within-run latency.	Complete the target sequence and test bounded history against quality and coverage.
Full training	A fitted pilot checkpoint; intended eight-H200 training platform.	Validate distributed execution, complete full training, and retain reproducible artifacts.
Downstream refinement	Splatfiction is documented independently.	Test a defined interface and report geometry before refinement and appearance after it.

A Proposed full-training and validation path**B Reported pilot execution path****Figure 2: Two paths with different maturity.** Dashed arrows denote proposed full-training and evaluation connections. Solid arrows organize the execution reported in the pilot; they are not an audited neural-network graph. A completed Frame Factory-to-model training handoff and a bounded-history policy remain to be demonstrated. The diagram does not imply that the pilot checkpoint came from full eight-H200 training.

If depth instead represents Euclidean range r , the term $\mathbf{zq}$ must become $r\mathbf{q}/\|\mathbf{q}\|_2$. The dataset and prediction schema must specify that convention, units, lens model, validity policy, and treatment of transparent surfaces. The checkpoint’s recorded 3.0625 metres per model unit is a scale conversion, not a measured metric-accuracy result [2].

Image transforms must preserve calibration. For a pixel-coordinate affine transform A , the corresponding intrinsic matrix is $K' = AK$; cropping and pixel-center conventions belong in A . The pilot uses bicubic resizing to 294×518 pixels. Its capture label contains an ultrawide profile, but the supplied report does not establish physical phone acquisition or a validated distortion model.

Strict frame causality requires $\hat{Y}_t = f_\theta(I_{1:t})$. Chunk causality permits access through the last frame of the current chunk and can introduce within-chunk look-ahead. A future-frame perturbation test must distinguish these contracts. The report includes a causal-verification flag and proof hash, but the proof object and masking implementation are not part of the supplied evidence package.

3 Observed Prototype Performance

3.1 One scene, one measured inference trace

The evidence comes from the supplied run report dated 21 September 2026 and the accompanying viewer recording [2, 1].

Table 2 retains the actual measurement boundaries. No new training, inference, or geometric evaluation was performed for this edition.

The checkpoint generation-u00001976 records 1,976 updates and 16 complete exposure epochs over a 3,000-frame capture. The report identifies a same-capture overfit-capability experiment: a scene-specific fitting pilot with an observed execution path. Geometric quality has not been evaluated or accepted in that report.

The trace processes 214 frames, or 7.13% of the 3,000-frame target, on a 12-GB-class RTX 4070 Ti with 11.994 GiB total device memory. The host has approximately 128 GiB of physical RAM. Recorded settings include bfloat16 aggregator autocasting and the named efficient attention backend; the runtime configuration is preserved in Appendix A. These results characterize this machine and workload, not every consumer configuration.

3.2 Streaming-inference throughput

The report records 32.44 seconds of streaming-inference wall time and 6.598 processed frames/s. The final chunk’s elapsed counter is 32.435 seconds; $214/32.435 = 6.5978$ frames/s agrees after rounding. This is image-file processing throughput, not a live-camera acquisition rate or display refresh rate.

Table 2: Recorded pilot operating point. One same-capture trace at 294×518 RGB input resolution. Timing refers to streaming inference. GPU and host quantities use different memory scopes [2].

Quantity	Recorded value
Observed frames / published chunks	214 / 54
Available / target capture frames	3,000
Fitting updates / exposure epochs	1,976 / 16
Streaming wall time	32.44 s
Mean processed-frame throughput	6.598 frames/s
Peak GPU allocated / reserved	4.323 / 5.010 GiB
Host history at 214 frames	1.823 GiB
Peak recorded process RSS	4.383 GiB
Last / first full-chunk time per frame	1.78×
Displayed points	1,439,812
Geometric quality evaluated / accepted	No / No

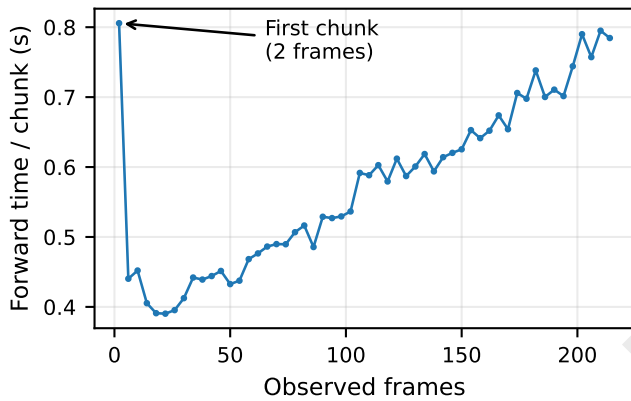


Figure 3: Observed forward time as history grows. All 54 recorded chunks are shown in order. The first contains two frames; the remaining 53 contain four. This within-run trace identifies a long-sequence optimization target; it is not a scaling forecast or a set of independent repeated trials.

3.3 A concrete long-sequence engineering target

The first full chunk takes 0.4403 seconds, approximately 0.1101 seconds per frame. The last takes 0.7845 seconds, approximately 0.1961 seconds per frame. The 1.78× increase makes history cost a measurable development priority (Figure 3). The recorded median and p95 chunk-forward times are 0.5877 and 0.7864 seconds, respectively, including the initial two-frame chunk.

Peak GPU allocation and reservation are 4.323 and 5.010 GiB. The logged device series remains nearly flat over the observed prefix (Figure 4). Historical descriptors are reported as resident outside GPU memory. The current run therefore demonstrates fit in the stated device configuration while leaving total-system scaling to further evaluation.

Host descriptor history reaches 1.823 GiB and process resident memory reaches 4.383 GiB (Figure 5). Descriptor bytes are only part of the process footprint. The report’s 25.56-GiB history estimate for 3,000 frames is a projection, not a measured endpoint. No full-target completion, bounded-memory guarantee, or training-memory result is inferred from this prefix.

Moving history off the GPU changes where state is stored;

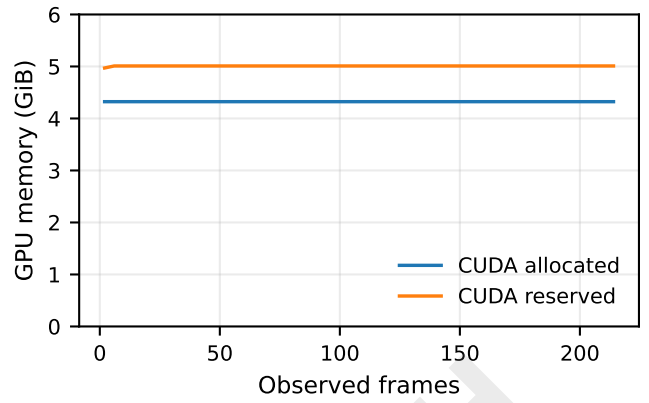


Figure 4: Reported device-memory footprint. Logged CUDA allocation is repeated at 4.323 GiB; reservation reaches 5.010 GiB. Without the logger implementation, repeated fields are not treated as a complete instantaneous allocation timeline. Host history is accounted for separately.

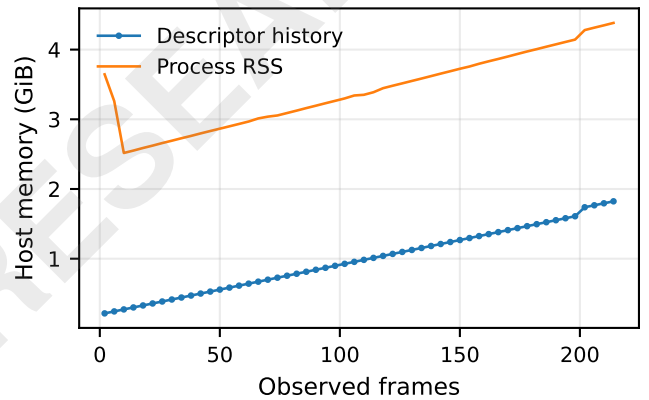


Figure 5: Host-state growth defines the next operating-range study. Descriptor history and process RSS are separate logged series. The additional increment near frame 202 is preserved. Only the observed 214-frame prefix is plotted; there is no extrapolated continuation.

it need not eliminate growth in total memory or compute. A bounded window, fixed descriptor budget, or persistent-anchor policy is therefore a separate intervention to test against geometry and coverage. The metadata setting `compression_ratio=4` does not establish a measured fourfold saving or a validated compression method.

3.4 What the visual milestone establishes

Figure 1 shows indoor structure, a trajectory overlay, an RGB preview, and predicted depth. It also contains dispersed and layered samples. The report records 1,439,812 displayed points under a stride-four, confidence-filtered display with a 12-metre depth cap and a two-million-point budget [2]. Point count measures displayed output, not surface accuracy or completeness.

The visual milestone is the publication and inspection of model output through a working viewer. Converting that milestone into a reconstruction-quality claim requires raw predictions paired with reference depth, cameras, and surfaces. Such pairs and a raw point-cloud export are not included in the supplied evidence. The viewer recording’s later reset and

progress states are not pooled with this trace or counted as additional evaluated trials.

4 Technology Maturity and R&D Roadmap

4.1 The next development phase

The next phase should turn the observed execution milestone into an evaluated reconstruction system. Table 3 organizes the work as evidence-producing milestones. The eight-H200 configuration is the intended full-training platform; it is not a completed experiment, a guaranteed accuracy improvement, or a basis for an unmeasured training-time estimate.

1. Establish the model–data contract. Prepare an immutable training manifest binding root environment, scene variant, lighting, camera trajectory, frame identity, RGB/depth pairing, calibration, units, transforms, and hashes. Validate the files actually decoded by the learner. Frame Factory’s depth-transcode check preserves input samples through compression; it does not itself certify source geometry or the model’s depth convention [6].

Assign splits by root environment before selecting frames. Adjacent views and alternate variants of the same layout belong in one split. Keep the pilot capture in development. Record asset reuse so that familiar layouts are not mistaken for new environments. The model-training release must be established independently of upstream rendered or landed-frame counts.

2. Qualify training and distributed execution. Start with a small, versioned shard and known geometric targets. Verify finite losses and gradients, camera conventions, causal masking, optimizer updates, and checkpoint restoration. Then test two-, four-, and eight-GPU execution with the precision, sharding, optimizer-state placement, and activation-memory settings recorded.

For data-parallel training with per-device batch b and accumulation factor a , the planned eight-device effective batch is $B_{\text{eff}} = 8ba$. Sequence length and tokens per sample must also be recorded. Scaling comparisons should name whether global work or per-device work is fixed. An inference trace on a consumer GPU cannot establish this training configuration’s memory fit or distributed efficiency.

3. Execute scene-diverse training with controls. Use a pre-declared update or compute budget and retain matched controls that separate new supervision from extra optimization. The proposed supervised objective combines depth, pose, intrinsics, and geometric consistency:

$$\mathcal{L}_{\text{prop}} = \lambda_d \mathcal{L}_d + \lambda_p \mathcal{L}_p + \lambda_K \mathcal{L}_K + \lambda_g \mathcal{L}_g. \quad (4)$$

This remains a full-study proposal, not a recovered objective for the pilot. Component definitions, weights, curriculum, trainable modules, and any teacher supervision must be fixed and logged. Geometry terms require consistent units and reference visibility; confidence filtering must not silently remove difficult examples from the evaluation denominator.

4. Evaluate quality and operating range together. Freeze the selected checkpoint and perform scene-disjoint evaluation without per-scene gradient updates. Complete the 3,000-frame target before making longer-horizon claims. Compare history

policies at matched quality and coverage, not solely at the smallest reported CUDA footprint. Preserve failures and incomplete runs in the evaluation record.

4.2 What this phase is intended to deliver

The useful output is an evidence package linking a known training release to a reproducible model and a declared deployment envelope. It should explain what the model predicts accurately, where consistency degrades, how long streams can run, and how much host and device capacity they require.

A positive result for one hypothesis should be reported at its actual scope. Better held-out depth with failed sequence completion supports a quality improvement but not a deployment claim. Lower memory accompanied by excessive coverage loss identifies a trade-off, not an unconditional optimization. This approach makes development decisions depend on measured outcomes rather than on a larger training cluster or a denser visualization alone.

This technical roadmap does not set a compute budget or delivery schedule. Resourcing should follow a defined workload and the application-specific acceptance targets.

5 Validation and Deployment Readiness

5.1 Measure geometry, not only output volume

The primary accuracy study should report depth error, camera-trajectory error, and surface accuracy/completeness, with processed coverage. For valid reference depths D^* , a scale-preserving depth metric is

$$\text{AbsRel} = \frac{1}{|\Omega|} \sum_{u \in \Omega} \frac{|\hat{D}(u) - D^*(u)|}{D^*(u)}. \quad (5)$$

Depth RMSE, error tails, and invalid-output accounting should accompany it. Confidence thresholds should produce error-coverage curves; rejecting difficult pixels must not silently improve the headline score.

Trajectory tests should retain rotation, translation, relative drift, and revisit behavior. Metric-scale performance is the primary spatial-use test. A once-per-sequence similarity-aligned diagnostic can be useful but must be labeled separately because it removes global scale error. Independent per-frame rescaling would conceal inconsistency that the system is intended to resolve.

Surface evaluation should use matched reference geometry and declared sampling. Prediction-to-reference distance measures accuracy; reference-to-prediction distance measures completeness. Report both and F-scores at physical tolerances chosen before final testing. Observed-surface reconstruction and hidden-surface completion must not be combined into an unsupported completeness claim.

5.2 Use comparisons that answer the research question

An unadapted, version-pinned model versus the adapted model tests data utility when architecture, preprocessing, and optimization budget are controlled. A few-scene versus diverse-scene comparison tests environment diversity when frames seen and updates are matched. Changing both architecture and supervision at once would not isolate either contribution.

Table 3: Proposed development milestones and decision evidence. All milestones below require additional work. Quantitative acceptance thresholds, schedules, and budgets are not established by the supplied record; thresholds should be fixed on development data before final testing.

Milestone	Deliverable	Decision evidence
Model-data readiness	Versioned training manifest, decoded-input checks, model bundle, declared geometric and causal contracts.	Examples consumed are traceable; units, cameras, validity, and split isolation pass checks.
Full-training readiness	Small-shard qualification, distributed execution study, restorable checkpoints, effective training settings.	Training is reproducible; the eight-H200 workload completes within its declared resource plan.
Unseen-scene reconstruction	Frozen checkpoints, matched baselines, raw predictions, paired reference geometry, scene-level results.	Depth, trajectory, surface accuracy, and coverage meet predeclared criteria on disjoint environments.
Long-sequence operation	Full 3,000-frame run and longer held-out streams; history-policy comparisons and whole-system telemetry.	Completion, memory, tail latency, drift, and retained coverage satisfy the chosen operating envelope.
Application integration	A defined input/output interface and candidate downstream workflow, evaluated separately.	Geometry utility is measured; any appearance refinement and its cost remain separately reported.

Offline full-sequence access and online prefix-only access are different information budgets. A bidirectional VGGT result can be an offline reference, not an equal-information streaming baseline. Prefix recomputation must include its repeated work. Related streaming models provide comparison settings, with code, weights, preprocessing, chunk policy, and information access pinned [3, 4, 5].

History ablations should compare the pilot-style host history with implemented alternatives at matched input and quality settings. Chunk partitions of one, two, and four frames can expose within-chunk look-ahead, partition sensitivity, collection delay, and execution overhead. Oracle camera or depth inputs may diagnose bottlenecks, but belong outside the RGB-only headline comparison.

An optional Splatfiction interface should keep pre-refinement geometry and post-optimization appearance separate [8]. Include the refinement time and any externally supplied calibration. That experiment would assess a potential downstream connection; no integrated benefit is part of the present results.

5.3 Qualify real deployment boundaries

Long-sequence testing must account for GPU peaks, host RSS, descriptor bytes, transfers, decoding, geometry conversion, visualization, and queueing. A synchronized model timer is valuable but does not replace observation-to-publication latency.

For a live-camera test, retain acquisition timestamps and backlog under a declared arrival rate. A four-frame group introduces collection delay even when forward computation is fast. The required operating point should specify allowed

latency and backlog as well as average throughput; a growing queue would restrict a sustained interactive claim.

Causality verification should hold the observed prefix fixed while changing later images, including changes within the current chunk. Compare emitted camera, depth, and confidence tensors, not only final screenshots. Preserve published outputs or their hashes to test non-revision. A separate revisit test asks whether later geometry remains consistent without editing previous predictions.

Repeated training seeds and repeated runtime trials address different sources of variation. Aggregate accuracy by scene, retaining the relationship between frames, trajectories, and runs. The pilot’s 54 chunks are correlated observations from one run, not 54 independent accuracy or repeatability trials.

5.4 Readiness and remaining evidence requirements

The model-specific review package should include version-matched training and inference code, base and learned checkpoint identities, trainable-tensor selection, effective objectives, history representation, causal masks, and logger boundaries. The supplied hashes identify reported artifacts; independently checking them requires the corresponding objects.

Commercial-use readiness also depends on permitted uses of source scenes, data, and model components, and on application-specific quality requirements. The supplied materials do not establish redistribution rights, a public dataset license, or general availability of an integrated product. These are readiness questions to resolve alongside the technical evaluation, not assumptions to derive from a successful run.

6 Conclusion

Reaigen-VGGT has progressed from a proposed workflow to a recorded RGB-to-geometry execution pilot. On the supplied RTX 4070 Ti configuration, one trace processes 214 frames at 6.598 frames/s with 5.010 GiB peak reserved GPU memory. The same trace identifies a specific next-stage challenge: host history grows and per-frame full-chunk time increases by 1.78× over the observed prefix [2].

The technology thesis is the combination of calibrated data, causal reconstruction, and resource-aware state. The next milestone is to establish useful unseen-scene geometry and a measured operating range through full training, matched evaluation, and complete-sequence testing. Full eight-H200 training, full prototype-dataset evaluation, and geometric quality acceptance remain open. The program couples an inspectable prototype with defined development milestones. Broader spatial-intelligence capability depends on the next validation results.

A Pilot Detail and Evidence Notes

A.1 Runtime and checkpoint scope

The original report records Windows 11, approximately 128 GiB host RAM, Python 3.13.7, PyTorch 2.8.0 with CUDA 12.6, bfloat16 aggregator autocasting, and the efficient attention backend. FlashAttention is marked unavailable, TF32 disabled, and historical H200 values not reused [2].

Its 866 learned tensors contain 604,739,584 parameters. This count does not establish total network size or a complete trainable-module specification. The raw report retains the exact same-capture experiment identifier, and explicitly sets `quality_evaluated=false` and `quality_accepted=false`.

A.2 Streaming timing and memory detail

Streaming forward work is reported as 31.01338 seconds; the rounded chunk entries sum to 31.0135 seconds, consistent with rounding.

Most four-frame chunks add 30,539,776 history bytes, or 7,634,944 per added frame. At frame 202 the increment is 138,477,568 bytes. These are log properties, not an identified cache algorithm. The endpoint-average descriptor estimate and 25.56-GiB full-target projection are distinct from that usual incremental rate.

A.3 Provenance and review package

The report is the sole quantitative source. Figure 1 contains unchanged crops of the 00:00.5 video frame; later reset/progress states are excluded. Crop coordinates, hashes, the original report, all 54 chunk records, plots, and analysis scripts are

retained in the companion package. They reproduce arithmetic and figures, not model execution.

Frame Factory’s separate snapshot reports 142,010 unique approved landed frames and zero sealed datasets [6]; neither is the model-training release. Its archive documents data production and handoff, not the VGGT implementation. Model code, weights, raw paired predictions, and the training manifest remain necessary for independent model-level review. The full video, scene assets, model weights, and corporate source archive are not redistributed in the companion package.

References

- [1] Reaigen s.r.o. 2026. *VGGT Prototype Viewer Recording*. Author-supplied screen recording, 21 September 2026 at 21:57:58 (filename time). Duration 51.15 s, 1974 × 1006 pixels. Figure 1 uses crops of the 00:00.5 frame.
- [2] Reaigen s.r.o. 2026. *VGGT Live-Viewer Run Report*. Author-supplied JSON, 21 September 2026, 19:56:14 UTC. File `run-images-20260921-215339.json`; checkpoint `generation-u000001976`. One 214-frame trace. Full hashes are retained in the companion provenance files.
- [3] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. 2025. VGGT: Visual Geometry Grounded Transformer. In *CVPR*. <https://arxiv.org/abs/2503.11651>.
- [4] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A. Efros, and Angjoo Kanazawa. 2025. Continuous 3D Perception Model with Persistent State. *arXiv:2501.12387*. <https://arxiv.org/abs/2501.12387>.
- [5] Dong Zhuo, Wenzhao Zheng, Jiahe Guo, Yuqi Wu, Jie Zhou, and Jiwen Lu. 2025. Streaming 4D Visual Geometry Transformer. *arXiv:2507.11539*. <https://arxiv.org/abs/2507.11539>.
- [6] Tomáš Sikora. 2026. *Frame Factory: Market-Aware Cloud Rendering for Large-Scale EXR Datasets*. Reaigen s.r.o. Author-supplied technical paper, September 2026, 11 pages. `Frame_Factory_Cloud_Rendering_Paper-2.pdf`.
- [7] Reaigen s.r.o. 2026. *Spatial Dataset Creation: Control-Plane Source and Handoff Documentation*. Author-supplied archive `Reaigen-spatial-dataset-creation-main-1.zip`. Snapshot `de47a255f935cf711d839494d5cb14e82a30ddce`. Dataset production and handoff; the VGGT model implementation is not included in this snapshot.
- [8] Tomáš Sikora. 2026. *Splatfition: High-Fidelity 3D Gaussian Splatting Training in WebGPU*. Reaigen s.r.o. Author-supplied technical paper, September 2026, 8 pages. `Splatfition_Rebuilt_Paper_v2(1).pdf`.