At the Limits of 2D-Only Methods: Studies in Extreme Viewpoint and Scale Variation

Aman Agarwal

Brown University, Department of Computer Science

Advised by Joel Salzman & James Tompkin

Reading & Research Report
Spring 2026

Introduction

Modern computer vision relies heavily on assumptions about how images relate to one another — that nearby views can be matched, that scale changes can be approximated by 2D blur, that learned priors generalise across viewpoints. These assumptions hold well in the settings the field has historically optimised for, but reach their limits at the extremes. Over the past year, working with Joel Salzman, I explored two such limits.

The first concerns extreme viewpoint variation: whether existing tools — 3D foundation models, learned feature matchers, and generative priors — can localise a street-level photograph against a satellite view, and where each succeeds and fails as the angular gap between cameras approaches 90°.

The second concerns extreme scale change on real 3D geometry: whether the Gaussian image pyramid — the workhorse of multi-scale vision, exemplified by SIFT — can faithfully emulate physical camera retreat in three-dimensional scenes, and where it breaks down. An isotropic, depth-independent 2D blur cannot reproduce the parallax, occlusion, and anisotropic shrinking that real camera motion produces. My ongoing work tests whether this disagreement is depth-dependent across a range of scene complexities, with longer-term interest in whether a learned, depth-aware operator could replace the Gaussian in scale-space construction.

Both share an underlying question: when does a 2D, depth-blind operation suffice as a proxy for a 3D geometric transformation, and when is the discarded depth information the very thing that matters? Both lines of work are early-stage and exploratory; this report describes the questions, the experiments I ran, where current methods succeed and fail, and what I plan to do next.

1 Extreme Viewpoint Variation

1.1 A documentation use case

In December 2024, The New York Times published a six-month visual investigation that identified senior commanders of Sudan's Rapid Support Forces by cross-referencing first-person "trophy" videos — often filmed by combatants on personal mobile phones — against commercial satellite

imagery, witness testimony, and contextual cues such as shadows, audio, and identifiable terrain features [1]. The Yale Humanitarian Research Lab, working independently, has produced over sixty reports on the same conflict using satellite imagery and open-source data, much of which has since been cited by United Nations fact-finding bodies [2]. Both efforts share a common methodological core: a human analyst manually establishes a spatial and temporal correspondence between a ground-level recording and an overhead view of the same scene. The labour involved is enormous and the available imagery far exceeds what skilled analysts can process.

1.2 Question and approach

The question this project investigates is whether contemporary computer vision tools can recover a spatial relationship between two views of the same scene captured under extreme viewpoint variation, and where each class of method succeeds or fails as the angular gap between cameras grows. Classical correspondence-based methods such as SIFT [3] are an immediate non-starter: they rely on dense pixel-to-pixel correspondence, which is sparse or non-existent under the angular and scale gap between a phone-held photograph and an overhead satellite tile.

The first method evaluated is MapAnything [4], a recent feed-forward 3D foundation model that maps one or more input views to a factored metric 3D output:

$$f_{\text{MapAnything}}\big(\hat{\mathcal{I}}, [\hat{\mathcal{R}}, \hat{\mathcal{Q}}, \hat{\mathcal{T}}, \hat{\mathcal{D}}]\big) = \big\{\, m,\ (R_i,\ \tilde{D}_i,\ \tilde{P}_i)_{i=1}^{N} \,\big\}, \tag{1}$$

where m is a global metric scaling factor and each view i contributes a predicted ray direction map $R_i \in \mathbb{R}^{3\times H\times W}$, an up-to-scale ray depth map $\tilde{D}_i \in \mathbb{R}^{1\times H\times W}$, and a pose $\tilde{P}_i \in \mathbb{R}^{4\times 4}$ expressed relative to the first input image (see [4] for the full derivation). Crucially, the model accepts a single image as input, which makes it possible to probe its monocular prior independently before introducing a correspondence task. Image pairs were drawn from the University-1652 benchmark [5, 6], which provides drone, satellite, and street-level views of the same buildings, supplemented with custom satellite/ground pairs rendered in Google Earth.

1.3 Where current methods succeed and fail

Given a single overhead view of a building with two towers, MapAnything predicted nearly identical depth for both towers — a plausible-looking but incorrect estimate that a non-expert viewer would not flag. Adding a drone view of the same building changed the predicted depth immediately and correctly, confirming that the second view was being used to refine the geometric estimate rather than processed independently (Figure 1). On University-1652 drone–satellite pairs, this multi-view refinement is reliable.

Performance degrades as the viewpoint gap widens. Drone–street pairs are handled less reliably; the two captures often span different days and lighting conditions, and the model's prior is sensitive to appearance differences that have nothing to do with geometry. Pushed further to fully orthogonal pairs — a street-level photograph against an overhead satellite tile — the failure becomes more telling: MapAnything predicts the two camera poses as near-neighbours, as though the inputs were captured from adjacent positions rather than from orthogonal viewing directions of the same scene. The factored 3D output is internally consistent with this incorrect pose estimate but bears no relationship to the actual scene geometry.

The pattern is consistent: the model succeeds when the two inputs share similar appearance and fails when appearance diverges, even when the underlying geometry is unchanged.

A natural next step was to characterise the angular threshold at which 2D feature matching breaks down. MapAnything's failure mode — a collapse of the predicted pose space at large viewpoint

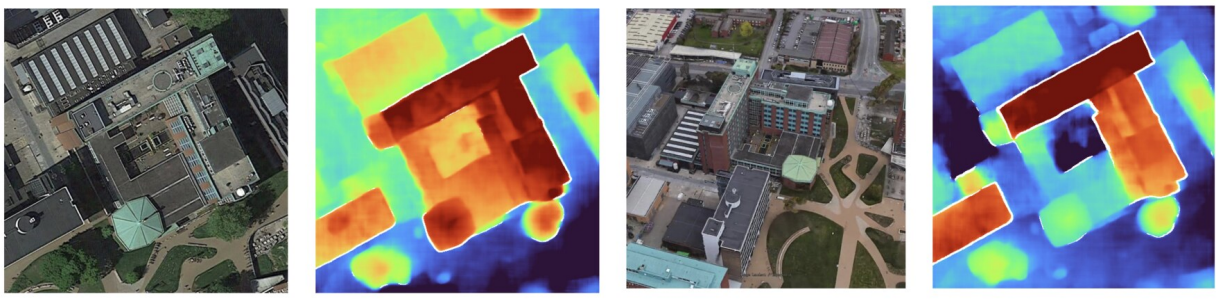


Figure 1: MapAnything depth estimates on a University-1652 building. Given the satellite view alone (column 1), the model predicts near-identical depth for the building's twin towers (column 2) — a plausible-looking but geometrically incorrect estimate. Adding a drone view of the same building (column 3) corrects the depth estimate and clearly differentiates the towers (column 4).

gaps — gives a binary signal but no information about *where* the transition occurs. A learned feature matcher offers a finer-grained probe: by counting recovered correspondences across a sweep of angular separations, the failure point can be located more precisely. Since the failure mode of MapAnything was tied to appearance variation across captures, I sought a matcher whose primitives were more stable under lighting and time-of-day differences. Line segments are a natural choice — a building edge remains a building edge under shadow, while corner-like keypoints can disappear. I used LightGlueStick [7], a recent matcher that jointly matches points and lines, combining the robustness of line features with the efficiency of LightGlue's attention-based architecture [8].

Rather than evaluating on University-1652 (where image pairs come at uncontrolled relative angles), I generated a custom set of renders using the Google Earth API, sweeping the camera around a fixed building while holding distance and altitude constant, in order to isolate angular separation as the single variable.

At 30° of angular separation, LightGlueStick recovered a dense set of correct point and line correspondences across the building's facades and roof edges (Figure 2). At 60°, matches degraded visibly: the count of inlier correspondences dropped and a growing fraction of them were spurious, though the most prominent structural lines were still recovered (Figure 3). At 90°, matching collapsed entirely: the recovered correspondences bore no structural relationship to the actual scene geometry (Figure 4).

Figure 2: LightGlueStick matches on a Google Earth render pair with 30° of angular separation. Inlier correspondences are dense and structurally correct across building facades and roof edges.

Figure 3: LightGlueStick matches at 60°. Inlier count drops and spurious matches appear, though prominent structural lines are still recovered.

Figure 4: LightGlueStick matches at 90°. The recovered correspondences bear no structural relationship to the actual scene geometry; matching has collapsed.

1.4 Bridging the gap with generative video models

MapAnything and LightGlueStick both rely, ultimately, on correspondence: pixel-level matches that establish where the same piece of scene appears in two views. As the angular gap widens, the shared visual content shrinks, and at 90° both methods collapse. A different class of method side-steps correspondence entirely. Rather than asking "where does this pixel in view A appear in view B?", a generative video model trained on first-frame to last-frame interpolation asks "what plausible sequence of intermediate views connects view A to view B?" If such a sequence can be generated coherently, the relationship between the two views is recovered implicitly — as the trajectory through which one transforms into the other.

I tested this with KlingAI by feeding two views of the same building as the first and last frames of a video generation prompt, and inspecting the interpolated intermediate frames for geometric consistency. Four representative trajectories are shown in Figures 5–8.

On a building view paired with a moderately-zoomed satellite view of the same structure, the model produced a clean 0°–to–90° trajectory: facades, roof, and surrounding context remained geometrically consistent across the intermediate frames (Figure 5). Pushed further to a much more zoomed-out satellite view, the model still tracked the correspondence; the building remained identifiable as the same structure through the pull-back and overhead rotation (Figure 6).

The failure cases share a different geometric character. With a street-level perspective view as the first frame and an overhead orthogonal satellite tile as the last frame, the trajectory is no longer

trivial even for a human observer to construct mentally. Figure 7 is the more forgiving of the two such pairs: the intermediate frames maintain enough structural plausibility that the trajectory reads as roughly correct, though I had difficulty verifying the precise correspondence between the two endpoints myself. Figure 8 fails more sharply: the model produced visually smooth intermediates that nevertheless contained no consistent intermediate geometry, with structure appearing and disappearing through the trajectory. The common factor in both harder cases is the mismatch between perspective and orthogonal projection at the two endpoints — a transition that has no continuous geometric interpolation, and which a human analyst would also find difficult to construct. In this sense the failures are honest: they occur on exactly the cases where the underlying geometric problem is genuinely hard.

This direction is not unique to this report. Concurrent work by Cai et al. [9] (CVPR 2025) shows systematically that generative video models can improve pose estimation on image pairs with little or no overlap — the regime in which DUSt3R [10] and similar correspondence-based methods struggle. Their approach (InterPose) uses a video model to hallucinate intermediate frames between two input images, then feeds the augmented sequence to a downstream pose estimator, with a self-consistency score to filter out implausible generations. The convergence between their findings and the qualitative results here is suggestive: video models trained on web-scale data appear to encode strong enough priors over scene geometry that they can supply the structural information that correspondence-based methods lack at large viewpoint gaps. For the documentation use case that opens this section, the practical implication is concrete — even when end-to-end automation remains out of reach, a tool that transforms a street-level photograph and a satellite tile into a continuous viewpoint trajectory is a candidate visual aid for the manual analyst workflow that the field currently relies on.

Figure 5: KlingAI-generated transition between a building view (left) and a moderately-zoomed satellite view (right) of the same structure. Intermediate frames maintain geometric consistency through the 0°–to–90° camera trajectory.

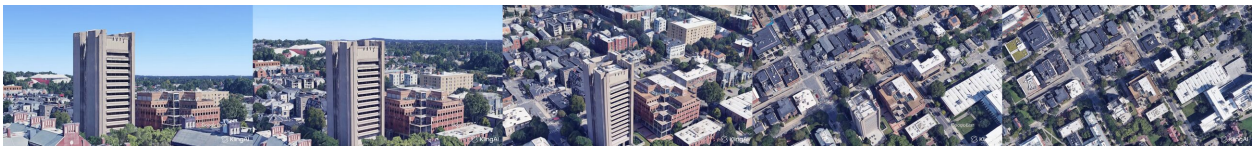


Figure 6: Same building paired with a much more zoomed-out satellite view. Despite the larger scale gap, the model maintains a consistent identity for the structure through the pull-back and overhead rotation.

2 Extreme Scale on Real 3D Geometry

2.1 A theoretical critique

The standard multi-scale construction in computer vision is the Gaussian image pyramid: repeatedly blurring an image with a Gaussian kernel and downsampling by a factor of two, used as the

Figure 7: A street-level perspective view paired with an overhead satellite tile. Intermediate frames retain enough structural plausibility that the trajectory reads as roughly correct, although verifying the precise correspondence between the two endpoints is itself difficult.

Figure 8: A second street–satellite pair where the perspective-to-orthogonal transition exceeds what the model can plausibly interpolate. Intermediate frames are smooth but contain inconsistent geometry: structure appears and disappears through the trajectory.

scale-space representation in SIFT and a long line of successor methods [3, 11]. The construction is well-motivated at the level of 2D signal analysis. Lindeberg derives the Gaussian as the unique kernel satisfying a small set of axioms on a two-dimensional signal — linearity, shift invariance, isotropy, causality, and the semi-group property [12, 13]. None of these axioms make any claim about three-dimensional camera geometry. In applied work, however, the Gaussian pyramid is often treated as an operational stand-in for "the camera moved further away" — a reading that does not appear as an explicit theoretical claim in the foundational scale-space literature but is embedded in how SIFT and its descendants build keypoint detectors across pyramid octaves.

Physical camera retreat in a three-dimensional scene produces image changes that depend on scene depth: nearer surfaces shrink more than farther ones, occlusion boundaries shift relative to the surfaces they bound, and perspective foreshortening rotates surface patches non-uniformly. An isotropic, depth-independent 2D blur does not model any of these effects. The experiments that follow attempt to quantify the resulting gap in a controlled setting, and to characterise where it concentrates.

2.2 A controlled probe: zone plates

The experimental setup compares two pipelines side by side. The first is a Gaussian pyramid built from a base image: repeated application of $\sigma = 1.0$ Gaussian blur followed by bilinear $2\times$ downsampling. The second is a sequence of physical renders generated by moving a virtual camera backward through a scene in Mitsuba 3 [14], with each render centre-cropped to match the same world region as the corresponding pyramid level. Camera distance is the single variable; the rendered scene is held fixed. If the Gaussian pyramid is a faithful proxy for physical scale change, the two sequences should agree pixel-for-pixel.

For an initial probe, the rendered scene was deliberately trivial: a flat zone plate at $z = 0$, defined by $0.5 + 0.5\cos(\pi r^2/k)$, whose local spatial frequency grows linearly with radius ($f(r) = r/k$). This pattern encodes a wide range of spatial frequencies within a single image, which lets the disagreement between the two pipelines be examined as a function of frequency by computing RMSE inside radial annuli. A flat scene was chosen on purpose: with no depth variation, the depth-dependent effects of physical camera retreat do not arise, and the Gaussian pyramid should be an accurate proxy in this regime.

Across a sequence of variants of this flat-scene experiment — orthographic and perspective cameras, with and without post-processing, with and without intra-octave scale levels — the per-pixel RMSE between the Gaussian pyramid and the physical renders remained small, below 0.02 at every octave in each variant. This serves as a baseline: in the regime where the Gaussian pyramid is expected to be accurate, the comparison pipeline reports it as accurate. The finding does not, on its own, validate the Gaussian pyramid; it validates the measurement.

2.3 Introducing real 3D geometry

The same pipeline was then run on a scene with depth structure. The zone plate was used as the diffuse reflectance of a displaced heightfield mesh (256×256 vertices, smooth multi-scale value noise, bump amplitude 0.20 in world units — roughly 22% of the zone plate diameter). The camera setup and the Gaussian-pyramid baseline were unchanged from the flat-scene experiment, isolating the introduction of 3D geometry as the single variable.

The result is a sharp departure from the flat-scene baseline. The per-pixel RMSE between the Gaussian pyramid and the physical renders rises to between 0.24 and 0.42 across the inner three non-trivial octaves (Figure 9), an order of magnitude larger than the < 0.02 baseline. The error fills the zone plate disk rather than localising at a single boundary, suggesting that the disagreement is not produced by a specific feature of the scene (such as an occlusion edge) but by the depth-dependent character of physical camera retreat acting across the whole image.

Two patterns are visible in the error structure. The first is that RMSE decreases at coarser octaves: 0.4239 at octave 1, 0.3623 at octave 2, 0.2428 at octave 3. As the camera retreats, small bumps in the heightfield project to image regions below the local Nyquist limit, and the two pipelines converge toward a similar low-frequency signal — the bumps "blur out" relative to the viewing distance. The second is that the error map carries concentric structure that traces the underlying zone plate rings: the disagreement is not random but appears to depend systematically on radial position, consistent with parallax shifting ring positions in the physical render relative to where the Gaussian pyramid places them.

The flat-scene baseline and the heightfield experiment together establish that in this setup, the disagreement between the Gaussian pyramid and physical scale change is small when no depth variation is present and substantial when it is. The experiments are controlled tests on a specific scene class (a zone plate on a smooth multi-scale heightfield) and the numerical values reported are specific to the chosen parameters — bump amplitude, camera distances, render resolution. A general claim about scale-space behaviour on arbitrary 3D content would require a broader evaluation than these experiments provide. What can be claimed from them is narrower but still informative: the depth-independence assumption embedded in the Gaussian pyramid is observably violated on at least one well-controlled 3D scene, by a margin large enough to motivate investigating whether the violation matters downstream.

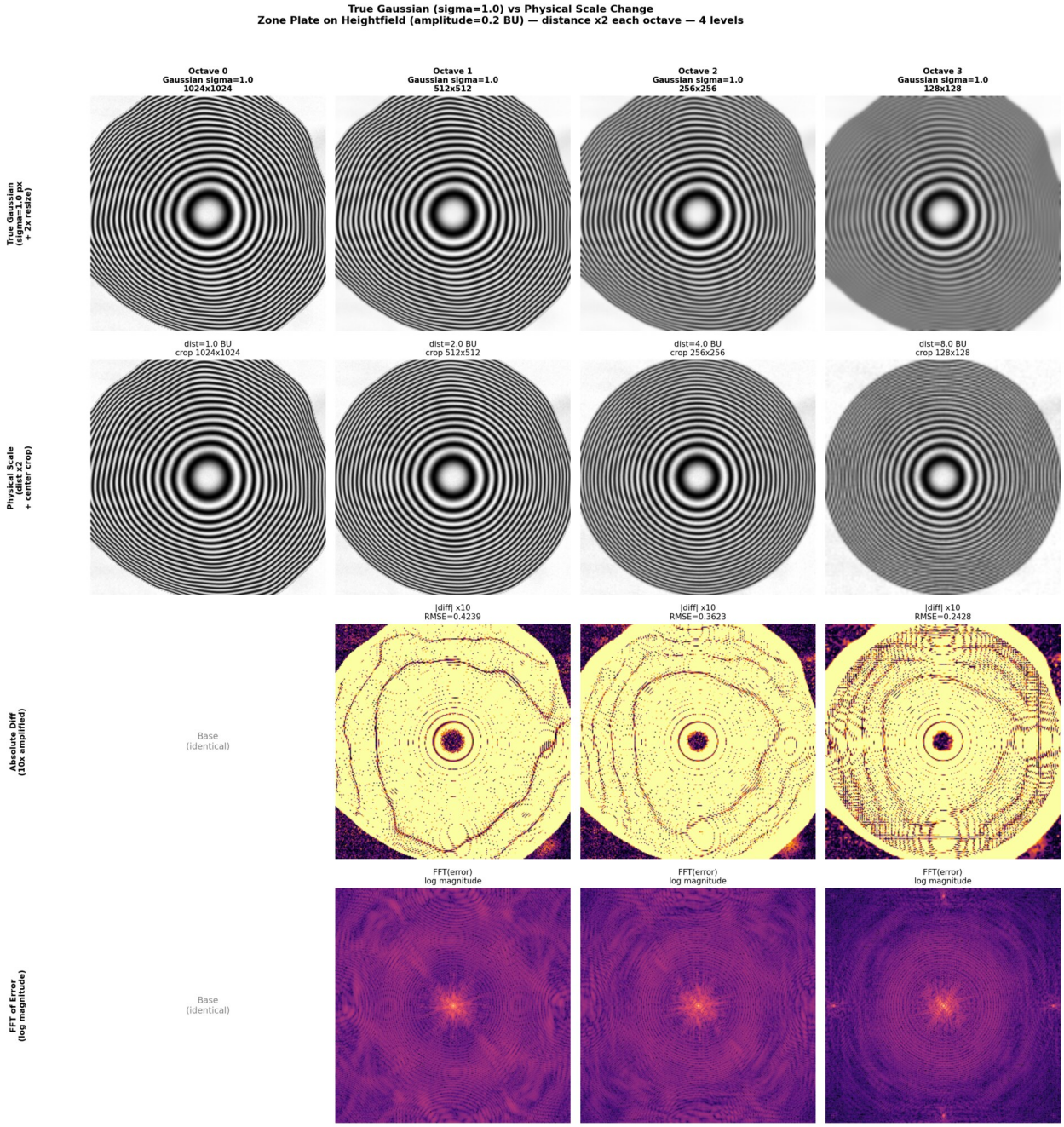


Figure 9: Comparison of a Gaussian pyramid (top row) and physical renders of a zone plate on a displaced heightfield mesh (second row), across four octaves. Camera distance doubles per octave from 1.0 to 8.0 BU; the physical renders are centre-cropped to match the apparent scale of the Gaussian pyramid. The third row shows the absolute difference (amplified 10× for visibility) with the per-pixel RMSE reported above each panel. The disagreement fills the zone plate disk and carries concentric structure tracing the underlying rings; it decreases at coarser octaves as small bumps fall below the local Nyquist limit. The bottom row shows the FFT of the error map (log magnitude).

2.4 Swapping pyramid levels for physical renders in SIFT

The previous subsections established that the Gaussian pyramid and physical scale change disagree substantially on scenes with real 3D geometry. A natural follow-up question is whether the disagreement matters for the downstream task that motivated scale-space in the first place. SIFT [3] builds its keypoint detector by finding extrema across octaves of a Gaussian pyramid: if the pyramid is a poor proxy for physical scale change on 3D content, the keypoints it extracts should differ in number and position from those that would be extracted from a pyramid whose levels are produced by physically moving the camera backward through the scene rather than by 2D blur.

To test this, I built two SIFT pipelines on the same view pair. *Classical SIFT* uses PythonSIFT [15] to construct a Gaussian pyramid from the rendered baseline RGB and then runs its standard DoG-extrema detector, descriptor extraction, and Lowe-ratio matching (0.75). The *multi-scale physical* variant replaces only the pyramid construction: each scale level is a separate render in which the camera origin is translated along the view ray to target $+ 2^{n/3} \cdot (\text{origin} - \text{target})$ and the resulting 1024×1024 image is centre-cropped to the matching world region. The same DoG detector, descriptor, and matcher then operate on this physical pyramid in place of the Gaussian one. Matches are scored against ground truth recovered from the renderer's world-position AOV: a match is counted as *correct* if its predicted target pixel lies within 5 pixels of the ground-truth projection of the source keypoint's 3D position into the target view.

Figure 10 reports view pair $0 \leftrightarrow 5$ (separated by $56.25°$ in azimuth on a 32-view orbit) for two PolyHaven assets. Several observations.

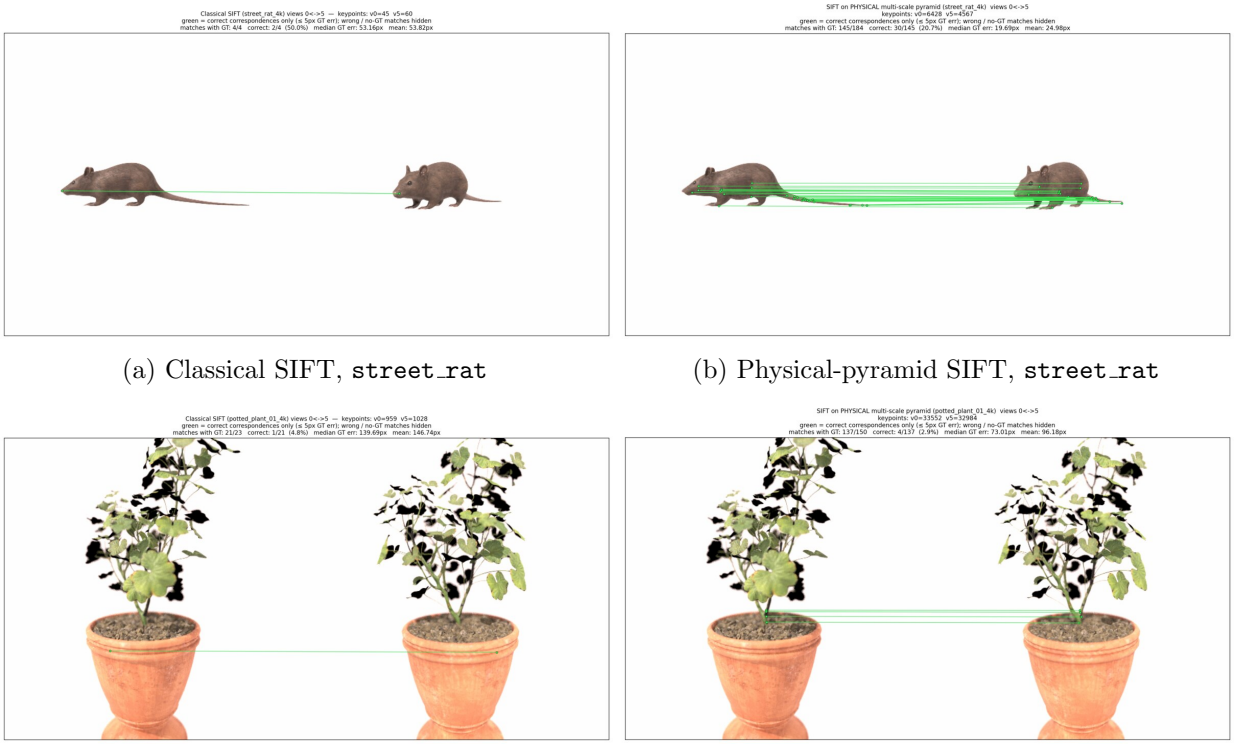
Keypoint count. On `street_rat_4k`, the classical pipeline detects 45 keypoints in view 0 and 60 in view 5. The physical pipeline detects 6,428 and 4,567 respectively — two orders of magnitude more. On `potted_plant_01_4k`, the classical pipeline reports about 10^3 keypoints per view; the physical pipeline reports about 3.3×10^4, again roughly $30\times$ more. The simplest reading is that the Gaussian pyramid is suppressing scale-space structure that exists in the physical signal: either through over-blurring of high-frequency content that survives physical scale change at distance (the heightfield experiment is consistent with this), or because true cross-octave keypoints whose existence depends on parallax-induced changes in image structure are unavailable to a pipeline that has no parallax.

Correct correspondences. The number of matches that survive both the ratio test and the 5 px ground-truth tolerance is 2 (classical) vs. 30 (physical) on the rat, and 1 vs. 4 on the plant. The physical pyramid produces more usable correspondences in absolute terms in both scenes.

Precision and error magnitudes. The picture is more nuanced than "physical wins." The fraction of GT-eligible matches that are within 5 pixels is lower for the physical pyramid in both scenes (20.7% vs. 50.0% on the rat; 2.9% vs. 4.8% on the plant). The comparison inverts when run on median GT pixel error: the physical pipeline cuts median error roughly in half (19.69 px vs. 53.16 px on the rat; 73.01 px vs. 139.69 px on the plant). The picture is consistent with the physical pyramid producing many more candidate keypoints — a fraction of which are correct, more of which fall narrowly outside the 5 px threshold, and the rest of which are spread over distractor regions that the larger candidate set inevitably reaches. A larger evaluation across more view pairs and scenes, rather than two illustrative pairs, would be needed to say whether the precision-vs-recall picture trades cleanly between pipelines or whether the physical pyramid dominates on a particular operating point. The current pipeline supports this evaluation (leakage-free leave-one-out test pairs are constructed from a 32-view orbit with a $\leq 45°$ angular gap and rendered ground-truth correspondences computed from the position AOV); running it at scale is left for future work.

The parallax is visible. Figure 11 shows the 12 physically-rendered scale levels for `potted_plant_01` laid out left to right. As the camera retreats, the plant moves upward and toward the pot relative

to image coordinates, while the pot itself shrinks more slowly — the parallax characteristic of depth-dependent scale change. A Gaussian pyramid of the same input would not exhibit this relative motion; it would shrink the entire image uniformly. The difference between these two stacks is what the SIFT detectors above are operating on.



(a) Classical SIFT, `street_rat`

(b) Physical-pyramid SIFT, `street_rat`

(c) Classical SIFT, `potted_plant_01`

(d) Physical-pyramid SIFT, `potted_plant_01`

Figure 10: Classical SIFT (left) vs. SIFT on a physically-rendered multi-scale pyramid (right) for view pair $0 \leftrightarrow 5$ on two PolyHaven scenes. Green lines are correspondences within 5 px of the ground-truth projection from view 0's world-position AOV; matches that fail the tolerance or have no GT signal are hidden. Keypoint counts and match statistics are reported in the title of each panel. The physical pyramid produces 10^2–$10^3\times$ more keypoints and more correct correspondences in absolute terms, at lower percentage precision but substantially lower median GT error.

Figure 11: The 12 physically-rendered pyramid levels for `potted_plant_01`, view 0, laid out left to right. Camera distance increases by $2^{1/3}$ between adjacent levels. The plant rises in the image relative to the pot rim and recedes faster than the pot does — the depth-dependent differential scaling that an isotropic Gaussian blur cannot produce. This is the parallax cue the SIFT comparison in Figure 10 is implicitly exploiting.

2.5 Future work

The SIFT comparison above shows that the change in detector behaviour when the Gaussian pyramid is replaced by a physically-rendered one is not isotropic — the parallax in Figure 11 makes this visible directly — and appears to be depth-dependent. Confirming this empirically is ongoing work: the next experiments run the same SIFT-swap comparison on a graded series of scenes spanning a range of depth variation, to test whether the gap between the two pipelines tracks that variation.

Conclusion

This report has examined two settings in which methods that treat the image as a two-dimensional signal — without an explicit account of the underlying three-dimensional scene — run into observable limits. The first was cross-view localisation across extreme viewpoint variation. MapAnything recovers reasonable geometry and pose when the two inputs share similar appearance, but degrades when appearance diverges — most starkly in the street-to-satellite regime where perspective and orthogonal projection meet; LightGlueStick, a recent sparse-feature matcher, fails to produce reliable correspondences past roughly 60°; and a generative video model can occasionally synthesise plausible intermediate viewpoints but also produces confidently hallucinated geometry in cases where correspondence-based methods would simply fail. The second was the Gaussian image pyramid as a stand-in for physical scale change. On flat scenes, the Gaussian construction agrees with physical camera retreat to within RMSE 0.02; on a controlled scene with real 3D geometry, the disagreement rises to RMSE 0.24–0.42, and replacing the Gaussian pyramid with a physically-rendered one in SIFT changes the detector's behaviour in ways consistent with that gap mattering downstream.

Neither study settles the question it raises. In both cases the report identifies a regime where current methods are observably constrained, characterises the constraint empirically on a controlled setting, and leaves the broader empirical sweep — across more viewpoints, more scenes, and more downstream tasks — to ongoing work.

Acknowledgments

I am grateful to Joel Salzman for teaching me the first principles of conducting research, and for being incredibly patient with me and my experiments over the past year. He has pushed me to run experiments carefully and taught me to set up simple classical baselines before reaching for complicated state-of-the-art methods. He has been an extraordinary mentor and has shown me what it takes to be a PhD student.

I would also like to thank James Tompkin for being incredibly supportive and for guiding me since my first day at Brown as a Masters student. He has been patient in discussing, guiding, and critiquing the half-baked ideas I have brought to him, and has taught me how to reason through problems.

References

[1] Sanjana Varghese, Natalie Reneau, Christoph Koettl, Aaron Byrd, and Declan Walsh. How 'Trophy' Videos Link Paramilitary Commanders to War Crimes in Sudan. The New York Times, December 2024. https://www.nytimes.com/video/world/africa/100000009893408/sudan-war-crimes-rsf.html.

[2] Nathaniel A. Raymond, Caitlin Howarth, et al. HUMAN SECURITY EMERGENCY: El-Fasher Falls to RSF: Evidence of Mass Killing. Humanitarian Research Lab at Yale School of Public Health, October 2025. https://files-profile.medicine.yale.edu/documents/876b4afc-e1da-495b-ac32-b5098699a371.

[3] David G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[4] Nikhil Keetha, Norman Müller, Johannes Schönberger, Lorenzo Porzi, Yuchen Zhang, Tobias Fischer, Arno Knapitsch, Duncan Zauss, Ethan Weber, Nelson Antunes, Jonathon Luiten, Manuel Lopez-Antequera, Samuel Rota Bulò, Christian Richardt, Deva Ramanan, Sebastian Scherer, and Peter Kontschieder. MapAnything: Universal Feed-Forward Metric 3D Reconstruction. In *International Conference on 3D Vision (3DV)*, 2026. https://arxiv.org/abs/2509.13414.

[5] Zhedong Zheng, Yunchao Wei, and Yi Yang. University-1652: A Multi-view Multi-source Benchmark for Drone-based Geo-localization. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1395–1403, 2020.

[6] Zhedong Zheng, Yujiao Shi, Tingyu Wang, Jun Liu, Jianwu Fang, Yunchao Wei, and Tat-Seng Chua. UAVM '23: 2023 Workshop on UAVs in Multimedia: Capturing the World from a New Perspective. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 9715–9717, 2023.

[7] Aidyn Ubingazhibov, Rémi Pautrat, Iago Suárez, Shaohui Liu, Marc Pollefeys, and Viktor Larsson. LightGlueStick: A Fast and Robust Glue for Joint Point-Line Matching. arXiv preprint arXiv:2510.16438, 2025. Accepted at ICCVW 2025. https://arxiv.org/abs/2510.16438.

[8] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. LightGlue: Local Feature Matching at Light Speed. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 17627–17638, October 2023.

[9] Ruojin Cai, Jason Y. Zhang, Philipp Henzler, Zhengqi Li, Noah Snavely, and Ricardo Martin-Brualla. Can Generative Video Models Help Pose Estimation? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16764–16773, June 2025. https://arxiv.org/abs/2412.16155.

[10] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. DUSt3R: Geometric 3D Vision Made Easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20697–20709, June 2024.

[11] Peter J. Burt and Edward H. Adelson. The Laplacian Pyramid as a Compact Image Code. *IEEE Transactions on Communications*, 31(4):532–540, 1983.

[12] Tony Lindeberg. *Scale-Space Theory in Computer Vision*. Springer, 1994.

[13] Jan J. Koenderink. The Structure of Images. *Biological Cybernetics*, 50(5):363–370, 1984.

[14] Wenzel Jakob, Sébastien Speierer, Nicolas Roussel, Merlin Nimier-David, Delio Vicini, Tizian Zeltner, Baptiste Nicolet, Miguel Crespo, Vincent Leroy, and Ziyi Zhang. Mitsuba 3 Renderer. Software, 2022. Version 3.0.0. https://mitsuba-renderer.org.

[15] rmislam. PythonSIFT: A Clean and Concise Python Implementation of SIFT. GitHub repository, 2015. https://github.com/rmislam/PythonSIFT.