

Article

Teaching Feature-Based 2D-to-3D Reconstruction for Metaverse-Oriented Content Generation: An OpenCV Structure-from-Motion Methodology for Computer Science Students

Oleksii Bahatskyi^{1,2}, Kyrylo Malakhov^{1,*}, Andrii Bondarchuk², Iryna Mashkina², Tetiana Nosenko²

¹ V.M. Glushkov Institute of Cybernetics of the National Academy of Sciences of Ukraine, Kyiv, Ukraine

² Borys Grinchenko Kyiv Metropolitan University, Kyiv, Ukraine

*Corresponding author: Kyrylo Malakhov, k.malakhov@incyb.kiev.ua

CITATION

Bahatskyi O, Malakhov K, Bondarchuk A, et al. Teaching Feature-Based 2D-to-3D Reconstruction for Metaverse-Oriented Content Generation: An OpenCV Structure-from-Motion Methodology for Computer Science Students. *Metaverse*. 2026; 7(4): m260011.
<https://doi.org/10.54517/m260011>

COPYRIGHT



Copyright © 2026 by author(s).
Metaverse is published by Asia Pacific Academy of Science Pte. Ltd. This work is licensed under the Creative Commons Attribution (CC BY) license.
<https://creativecommons.org/licenses/by/4.0/>

Abstract: Feature-based Structure from Motion (SfM) is frequently taught through isolated OpenCV calls, which can hide the relationships among acquisition geometry, correspondence quality, pose recovery, reconstruction error, and downstream asset readiness. This article presents a constructively aligned methodology for teaching those relationships to undergraduate computer science students. The instructional artifact combines an instrumented OpenCV pipeline, explicit mathematical models, quality gates, a five-block laboratory sequence, an assessment rubric, reproducibility packaging, and a metaverse-content handoff. Technical characterization combines a five-view classroom timing baseline with three controlled rendered reference scenes—a textured object, a repeated-pattern facade, and a curved-form artifact approximated by ten planar facets—processed at 12, 24, and 48 views. The evaluation reports the number of cameras added to the sequential pose chain, mean adjacent-pair pose-validated inlier ratio (RANSAC and cheirality), reprojection root-mean-square error (RMSE), filtered point count, pose error against known trajectories, surface consistency against known geometry, and stage timings. Every adjacent camera pair yielded a pose, so all cameras were added to the sequential chain. The 24-view condition produced the largest point clouds (13,133–16,154 points), whereas 48 closely spaced views reduced the mean adjacent-pair pose-validated inlier ratio to 6.7%–22.1%, demonstrating that image count cannot compensate for insufficient parallax. Reprojection RMSE remained 0.34–0.44 pixels, which also shows why low residuals must be interpreted together with correspondence and pose diagnostics. Ground-truth checks exposed the same degeneracy: median translation-direction error reached 78.3°–179.3° in the 48-view conditions, and only 0.0%–22.0% of points were surface-consistent at $\delta = 0.12$. The five-view timing baseline averaged 184.3 ms; feature processing and matching accounted for 46.4%. The sparse output is positioned as an inspectable upstream scaffold for dense reconstruction, meshing, texturing, semantic annotation, provenance, glTF packaging, and immersive-engine import. Because no student cohort was studied, the methodology specifies intended learning outcomes and a prospective evaluation protocol rather than claiming measured educational effectiveness.

Keywords: Metaverse; 3D Reconstruction; Computer Science Education; Structure from Motion; OpenCV; Feature Matching; Reproducible Computing; Semantic 3D Assets

1. Introduction

Persistent immersive environments require more than visually attractive objects: reusable content must have traceable acquisition conditions, consistent coordinates, manageable geometric complexity, machine-readable metadata, and an export path into interactive engines. Image-based 3D reconstruction is therefore a relevant

Computer Science topic at the intersection of computer vision, graphics, algorithms, data management, and metaverse-oriented content engineering.

Structure from Motion estimates camera motion and sparse scene structure from overlapping images [1–4]. Its educational value is not that the constituent algorithms are new, but that the pipeline exposes relationships among projective geometry, numerical conditioning, robust estimation, computational cost, and observable failure. A student can inspect why a correspondence is rejected, why translation is recoverable only up to scale, and why a low reprojection residual does not by itself establish a complete or usable model.

The study combines two complementary artifacts: a five-view OpenCV classroom baseline profiled over ten executions and a controlled, instrumented benchmark. The classroom baseline yielded 246 points but lacks retained calibration and configuration records; it is therefore used only for visual and timing comparison. A complete teaching methodology additionally requires reconstruction-quality validation, explicit configuration, measurable learning outcomes, assessment artifacts, and a defined handoff from sparse geometry to metaverse-ready content.

This design-and-characterization study addresses those technical and pedagogical requirements. It does not propose a new SfM algorithm. The contribution is an assessable instructional artifact that combines an instrumented OpenCV reference implementation, explicit mathematical models, stage-specific quality gates, a multi-scene sensitivity benchmark, a compact laboratory sequence, an assessment rubric, a reproducibility package, and an explicit path from sparse camera geometry to dense and semantically enriched content.

The study is deliberately bounded. Technical characterization establishes that the artifact makes reconstruction behavior and failure modes observable, including large translation-direction errors under weak parallax; it does not establish state-of-the-art geometric performance. No classroom intervention or human-participant study was conducted, so the stated competencies are intended learning outcomes rather than measured gains. A prospective educational evaluation protocol is supplied for subsequent deployment.

2. Related Work

2.1. Geometry-based reconstruction and the distinction from SLAM

Classical multi-view reconstruction combines calibrated or uncalibrated projection models, local-feature correspondences, epipolar constraints, triangulation, and nonlinear refinement [1–4]. COLMAP-style incremental SfM demonstrates how camera registration, feature tracks, triangulation, and bundle adjustment can be organized for large image collections [3]. Oriented FAST and Rotated BRIEF (ORB) offers a computationally economical binary descriptor suitable for transparent laboratory experiments [5], while comparative evaluation shows that local-feature choice affects matching behavior [6]. Random sample consensus (RANSAC) provides a robust mechanism for separating geometrically consistent matches from outliers [7].

Offline SfM and online simultaneous localization and mapping (SLAM) share feature tracking, pose estimation, and map optimization, but they solve different operational problems. SfM normally optimizes a finite image collection for scene reconstruction; SLAM estimates a trajectory and map under sequential, often real-time constraints. ORB-SLAM2 and ORB-SLAM3 are therefore cited here as related visual or visual-inertial SLAM systems, not as interchangeable implementations of the offline pipeline taught in this study [8,9].

2.2. Neural and feed-forward reconstruction

Learned depth estimation and multi-view stereo can produce denser geometric evidence than sparse SfM under suitable training and capture conditions [10,11]. Neural radiance fields (NeRFs) and Instant-NGP represent scenes for novel-view synthesis rather than automatically producing an explicit, engine-ready mesh [12,13]. Likewise, 3D Gaussian Splatting provides an efficient renderable representation, but its geometry, memory, and export properties differ from conventional point clouds and meshes [14].

Recent work illustrates the pace and diversity of this transition. Mip-Splatting addresses scale-dependent aliasing in Gaussian representations [15]. Sparse2DGS combines multi-view-stereo initialization with geometry-prioritized Gaussian optimization for sparse-view surface reconstruction [16]. MV-DUST3R+ performs feed-forward reconstruction from multiple sparse views and can attach Gaussian-splatting heads [17]. AnchorSplat uses explicit 3D geometric priors to guide a feed-forward Gaussian representation [18]. These methods should not be treated as a homogeneous category: they differ in outputs, supervision, computational demands, and dependence on camera geometry.

Relative to the classical SfM pipeline used in this module, the recent methods in References [15–18] differ in their inputs, outputs, and assumptions. Classical SfM accepts overlapping images with known or estimable calibration, assumes a predominantly rigid scene with repeatable local correspondences and sufficient parallax, and returns explicit camera poses and a sparse point model up to a similarity transformation for monocular capture [1–4]. Mip-Splatting assumes calibrated, posed views and an SfM-derived sparse initialization, performs per-scene optimization, and outputs an anti-aliased 3D Gaussian radiance representation [15]. Sparse2DGS assumes sparse posed views, uses learned multi-view-stereo initialization and geometry-oriented constraints, and produces an optimized surface-oriented Gaussian representation from which a mesh can be fused [16]. MV-DUST3R+ accepts sparse unposed RGB views and predicts aligned dense pointmaps and confidence in a feed-forward pass, with optional Gaussian heads [17], whereas AnchorSplat uses a pretrained multi-view predictor to estimate depth and camera parameters before decoding an anchor-aligned Gaussian representation [18]. References [15–18] therefore target dense, surface, or renderable outputs and depend on per-scene optimization or learned priors; the present framework retains an explicit sparse camera–point scaffold so that students can inspect and diagnose each geometric assumption.

2.3. Computer-vision pedagogy and instructional resources

Project-based computer-vision education is most informative when a motivating application is aligned with explicit outcomes, staged artifacts, and multiple forms of assessment. Geronimo et al. linked a real-world recognition project to course outcomes and student feedback [19]. A multi-semester drone-centered course combined intended learning outcomes, formative and summative assessment, surveys, and project evidence [20]. Staged computer-vision frameworks have also been validated through expert review and pilot implementation [21], while OpenCV programming resources and concept-exploration tools can support learning when their instructional role is clearly defined [22,23].

Existing SfM learning activities often emphasize capture and operation of packaged photogrammetry software [24]. That approach is useful for application-oriented fields, whereas the present computer science methodology exposes the matching, robust-estimation, pose-recovery, triangulation, and profiling stages

for inspection and modification. Model curricula for augmented reality similarly emphasize skills, learning outcomes, and lesson plans rather than technology lists [25]. The 2026 CSU101 dataset extends this logic through a unified progression from classification and detection to Gaussian-splatting reconstruction [26]. The gap addressed here is a compact, code-visible SfM module that joins geometry, quality diagnostics, reproducibility, and downstream semantic asset packaging.

2.4. Semantic and workspace support for reusable assets

For the module's metaverse-oriented handoff, image-derived geometry is accompanied by object identity, component relations, coordinate frames, provenance, intended interaction, and controlled domain terminology. Three cited works support distinct parts of that handoff: Palagin et al. [27] motivate integration of learned perceptual representations with explicit semantic knowledge; Petrenko et al. [28] support fine-grained concept representation for object and component meaning; and Palagin et al. [29] provide formal procedures for controlled vocabularies and relation constraints. These functions concern paradigm integration, semantic granularity, and ontology engineering rather than the SfM estimator.

Reproducibility introduces a separate workspace requirement. The Research and Development Workstation Environment (RDWE) model [30] informs the organization of datasets, calibration records, configurations, code, logs, assessment artifacts, and exports with traceable provenance across the laboratory lifecycle. The ontology-mediated dialog service in [31] is an applied architectural precedent for a downstream interface that consumes semantic descriptions for explanation and interaction; it is not evidence for SfM accuracy, clinical use, or an implemented e-rehabilitation function in this study. References [27–31] therefore support non-overlapping semantic-manifest, service-interface, and reproducibility requirements.

3. Educational Design

3.1. Learners, prerequisites, and intended outcomes

The module is designed for third- or fourth-year undergraduate students in computer science. Expected prerequisites are linear algebra, homogeneous coordinates, probability, analysis of algorithms, and intermediate Python or C++. The reference implementation serves as instructional scaffolding: students begin with a reproducible baseline, expose intermediate data, and then modify a controlled factor rather than assembling opaque function calls. Constructive alignment is used to connect each intended outcome to an observable artifact and assessment criterion [32].

After completing the module, a student should be able to:

- apply the calibrated perspective model and epipolar constraint to a multi-view image sequence;
- justify the descriptor distance, nearest-neighbor distance-ratio threshold, and geometric-filtering configuration;
- recover relative pose, explain monocular scale ambiguity, and diagnose weak-parallax configurations;
- triangulate and filter points using cheirality and reprojection diagnostics;
- evaluate reconstruction quality and runtime jointly under a controlled parameter change;
- package code, configuration, data provenance, logs, and outputs as a reproducible workspace;
- explain why sparse camera geometry is an upstream scaffold rather than a finished metaverse asset.

3.2. Five-block laboratory sequence

Block A—projection and calibration (2 h). Students inspect the intrinsic matrix, distortion model, coordinate conventions, and calibration residuals. The assessed output is a calibration record and a short numerical-conditioning explanation.

Block B—features and correspondences (3 h). Students visualize ORB keypoints, Hamming-distance neighbors, ratio-filtered matches, and RANSAC inliers. The assessed output includes raw and filtered match overlays plus an explanation of two rejected correspondences.

Block C—pose and triangulation (3 h). Students recover relative pose, inspect cheirality, triangulation angle, and reprojection residuals, and export a colored sparse cloud. The assessed output is a diagnostic report linked to the relevant equations.

Block D—sensitivity and profiling (3 h). Teams vary one factor—feature budget, ratio threshold, RANSAC threshold, image count, resolution, or keyframe spacing—while holding the remaining configuration fixed. The assessed output is a quality–runtime comparison and a prediction based on Amdahl’s law [33].

Block E—metaverse-content handoff (3 h). Students define the transition to dense reconstruction or neural rendering, mesh and texture generation, coordinate conversion, level of detail, semantic annotation, provenance, and glTF packaging [34]. The assessed output is an asset manifest and an import-readiness checklist.

3.3. Assignment, assessment, and reproducibility

Each team reproduces the baseline, diagnoses one deliberately degraded acquisition condition, changes one algorithmic factor, and submits a reproducibility bundle with source code, configuration, calibration or known camera parameters, random seed, machine-readable metrics, output cloud, and concise interpretation. A semantic extension requires a compact manifest that links perceptual outputs to explicit knowledge [27], distinguishes object and component meanings at an appropriate granularity [28], and applies controlled-vocabulary and relation constraints [29]. An RDWE-style directory structure preserves inputs, configurations, transformations, evidence, and versions [30], while an optional advanced task can expose selected manifest terms through an ontology-mediated explanation interface, for which [31] provides an architectural precedent.

The 100-point analytic rubric allocates 25 points to geometric reasoning, 25 to implementation correctness, 20 to validation and failure diagnosis, 15 to reproducibility, 10 to metaverse-content integration, and 5 to technical communication. A reconstruction with many points cannot receive full validation credit unless correspondence quality, reprojection behavior, and capture geometry are interpreted coherently.

3.4. Educational evaluation boundary and prospective protocol

No student cohort was involved in the present study. Technical observability and reproducibility are demonstrated, but learning gains and causal educational effectiveness are not estimated. Accordingly, the module is described in terms of intended outcomes and assessable design rather than demonstrated competency development.

A subsequent consented pilot should combine a pretest–posttest concept inventory, an independently scored implementation artifact, and a transfer task in which students diagnose an unseen failure case. A perception survey may measure usability or self-efficacy but should not be treated as direct evidence of learning. A subset of artifacts should be double-scored, with inter-rater reliability reported. Changes in paired scores should be reported with a 95% confidence interval and

a standardized effect size; the inferential test should be selected according to the distributional properties of the difference scores.

4. Materials and Methods

4.1. OpenCV reference implementation

The reference implementation uses Python 3.12.13, OpenCV 4.11.0, and NumPy 2.5.1 on Linux [35]. The controlled benchmark was executed on a virtualized Intel Xeon Platinum 8370C environment with nine available CPU cores and 21 GB of RAM. The notebook used for the five-view timing baseline included a GPU, but no OpenCV CUDA functions were invoked; consequently, the GPU did not contribute to the reported timings. The five-view timing baseline provides a classroom-scale runtime characterization, whereas the controlled benchmark supplies the reproducible quality analysis.

ORB was configured with `nfeatures = 2600`, `scaleFactor = 1.2`, `nlevels = 8`, `edgeThreshold = 31`, `firstLevel = 0`, `WTA_K = 2`, `scoreType = cv2.ORB_HARRIS_SCORE`, `patchSize = 31`, and `fastThreshold = 20`. Binary descriptors were matched with `BFMatcher` and `NORM_HAMMING`; two nearest neighbors were retained per query descriptor and filtered with a distance-ratio threshold of 0.75.

For each adjacent pair, `cv2.findEssentialMat` estimated the essential matrix using RANSAC with a confidence level of 0.999 and a 1.5-pixel threshold. `cv2.recoverPose` selected the cheirality-consistent relative pose, and `cv2.triangulatePoints` generated homogeneous 3D points. A camera was counted as added to the sequential pose chain when its adjacent pair yielded a valid essential matrix and a cheirality-consistent pose. Points were rejected if non-finite, behind either camera, or associated with a two-view reprojection error of at least 3 pixels; the reported reprojection RMSE was computed over the retained two-view observations. A voxel-grid filter with a voxel size of 0.018 coordinate units removed near-duplicates. The fixed OpenCV random seed was 20260718.

4.2. Mathematical model and quality gates

For camera i observing point j , the calibrated perspective model maps the homogeneous world point $\tilde{X}_j$ to image point $\tilde{x}_{ij}$ through the intrinsic matrix K_i and extrinsic parameters R_i and t_i , where λ_{ij} is the nonzero projective depth-scale factor [1]:

$$\lambda_{ij} \tilde{x}_{ij} = K_i [R_i | t_i] \tilde{X}_j \quad (1)$$

For two calibrated views, let $\hat{\mathbf{x}}_{ij} = K_i^{-1} \tilde{x}_{ij}$ denote the normalized homogeneous image coordinate. A valid correspondence satisfies the essential-matrix constraint, where E is the essential matrix and $[t]_{\times}$ denotes the skew-symmetric cross-product matrix:

$$\hat{\mathbf{x}}_{2j}^T E \hat{\mathbf{x}}_{1j} = 0, \quad E = [t]_{\times} R \quad (2)$$

RANSAC and cheirality validation provide an observable quality gate. For each successfully processed adjacent image pair, the pose-validated numerator is the number of correspondences retained by `cv2.recoverPose` after essential-matrix RANSAC and cheirality validation, and the denominator is the number surviving the 0.75 descriptor-ratio filter. The pair-level percentage is 100 times the numerator divided by the denominator. The sequence-level summary is the unweighted arithmetic mean of these percentages over the $N - 1$ adjacent pairs in an N -view sequence; it is a local pairwise diagnostic, not a measure of global registration accuracy. Triangulated observations are assessed through the Euclidean reprojection residual and its aggregate RMSE. Here, $\mathcal{O}$ denotes the set of valid image observations

and $\pi(\bullet)$ denotes perspective division; in the reported benchmark, $\mathcal{O}$ contains only retained two-view observations with reprojection error below 3 pixels:

$$e_{ij} = \left\| \mathbf{x}_{ij} - \pi \left(\mathbf{K}_i \left(\mathbf{R}_i \mathbf{X}_j + \mathbf{t}_i \right) \right) \right\|_2, \quad \text{RMSE}_{\text{repr}} = \sqrt{\frac{1}{|\mathcal{O}|} \sum_{(i,j) \in \mathcal{O}} e_{ij}^2}. \quad (3)$$

A complete incremental system would refine cameras and points by minimizing a robust reprojection objective [1,3,36], for example with a Huber or Cauchy loss ρ :

$$\arg \min_{\{\mathbf{R}_i, \mathbf{t}_i, \mathbf{X}_j\}} \sum_{(i,j) \in \mathcal{O}} \rho(e_{ij}^2). \quad (4)$$

Bundle adjustment was not implemented in the compact demonstrator; Equation (4) is included as an extension target, not as a performed operation. Accordingly, the implementation is a compact adjacent-pair SfM demonstrator: it chains pairwise poses, triangulates independent two-view points, and voxel-deduplicates them; it does not construct persistent multi-view feature tracks, perform perspective-n-point resection, or optimize a global model. Monocular reconstruction is determined only up to a similarity transformation.

4.3. Scenes, image counts, and acquisition geometry

The five-view classroom baseline used five 640×480 USB-camera images at approximately -30° , -15° , 0° , $+15^\circ$, and $+30^\circ$ around a tabletop object and produced 246 sparse points. The camera model and calibration residual from that run were not retained, so the sequence is used only for visual and timing comparison, not as accuracy evidence.

The controlled benchmark used three 1280×720 rendered reference scenes with known geometry and camera trajectories: a textured tabletop object, a planar facade with repeated patterns and shallow relief, and a curved-form artifact approximated by

ten planar facets. The intrinsic matrix was $\mathbf{K} = \begin{bmatrix} 980 & 0 & 640 \\ 0 & 980 & 360 \\ 0 & 0 & 1 \end{bmatrix}$, and lens distortion

was zero. Cameras followed a -34° to $+34^\circ$ horizontal arc at a radius of 4.2 coordinate units. Each scene was rendered using 12, 24, or 48 views, corresponding to adjacent angular separations of approximately 6.18° , 2.96° , and 1.45° . This design isolates an important acquisition lesson: increasing image count while holding the arc fixed also reduces adjacent-view parallax. **Figure 1** shows representative viewpoints for each controlled scene.

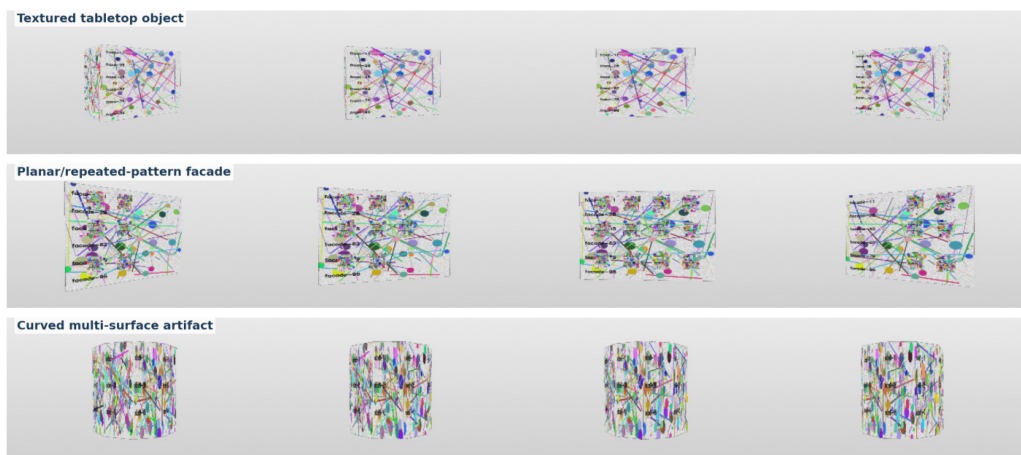


Figure 1. Controlled scene diversity and representative viewpoints. Each row shows four views from a 12-image sequence; the complete benchmark also evaluates 24 and 48 views.

4.4. Evaluation and statistical analysis

Technical outcomes were the number of cameras added to the sequential pose chain, mean adjacent-pair pose-validated inlier ratio (RANSAC and cheirality), reprojection RMSE, and voxel-unique pairwise triangulated point count. Point count is interpreted as density, not accuracy. Geometric accuracy was assessed against the known trajectory and geometry using median adjacent-pair rotation error e_R , median adjacent-pair translation-direction error e_t , and the proportion P_δ of reconstructed points P within $\delta = 0.12$ coordinate units of the densely sampled reference-surface set G . Subscripts gt and est denote ground-truth and estimated adjacent-pair relative transforms. This one-sided surface-proximity ratio measures precision relative to the reference geometry rather than surface coverage or completeness; δ corresponds to approximately 4.1%–6.8% of the scenes’ maximum lateral extent. The angular errors are expressed in degrees:

$$e_R = \frac{180^\circ}{\pi} \arccos \left(\frac{\text{tr}(\mathbf{R}_{\text{gt}}^T \mathbf{R}_{\text{est}}) - 1}{2} \right),$$

$$e_t = \frac{180^\circ}{\pi} \arccos \left(\frac{\mathbf{t}_{\text{gt}}^T \mathbf{t}_{\text{est}}}{\|\mathbf{t}_{\text{gt}}\|_2 \|\mathbf{t}_{\text{est}}\|_2} \right).$$
(5)

$$P_\delta = \frac{|\{\mathbf{p} \in \mathcal{P} : d(\mathbf{p}, G) < \delta\}|}{|\mathcal{P}|}, \quad \delta = 0.12,$$

$$d(\mathbf{p}, G) = \min_{\mathbf{g} \in G} \|\mathbf{p} - \mathbf{g}\|_2.$$
(6)

Adjacent-pair angular errors were computed directly from each estimated relative transform before translation scaling. For surface comparison, poses were chained in the first-camera reference frame and each estimated relative translation was multiplied by the known adjacent-camera baseline. No post hoc similarity alignment was applied; consequently, the one-sided surface-proximity ratio reflects accumulated pose error under this scaling convention. One deterministic evaluation run was executed for each scene–view configuration, so no inferential comparison across the nine configurations is claimed.

The five-view timing dataset comprises ten executions on one notebook. The timing labels follow the classroom baseline’s instrumentation: ORB feature extraction and descriptor construction, feature processing and FLANN-based matching, camera-pose estimation, and auxiliary operations. These values do not profile the controlled BFMatcher benchmark. Stage values were treated as primary, and total runtime for each execution was calculated as the sum of its four recorded stage timings. The summary reports the arithmetic mean, sample standard deviation, median, interquartile range, and stage share. These repetitions characterize execution variability on one system, not population-level performance.

4.5. Reproducibility protocol

The supplementary reproducibility package [35], publicly available on Hugging Face at <https://doi.org/10.57967/hf/9819>, contains the benchmark script, deterministic scene generator, fixed seed, machine-readable JSON results, representative images, figure source files, dependency specification, and run instructions. A student submission should additionally include an environment record, configuration file, commands, checksums, raw metrics, and an asset manifest. This converts reproducibility from an availability statement into an assessed learning artifact.

5. Results

5.1. Visual and geometric validation

As shown in **Figure 2**, the five-view baseline output is difficult to interpret because 246 uncolored points are displayed on a black background without scale or camera context. The controlled benchmark provides colored points, explicit filtering, and a denser reference-surface comparison. In the 24-view curved-artifact configuration, 13,883 points passed cheirality, reprojection, and voxel filters; 1,607 also lay within $\delta = 0.12$ of the known surface. The 22,000-point reference-surface cloud illustrates the density gap that remains before meshing or neural rendering. It is not presented as an OpenCV dense-reconstruction result.

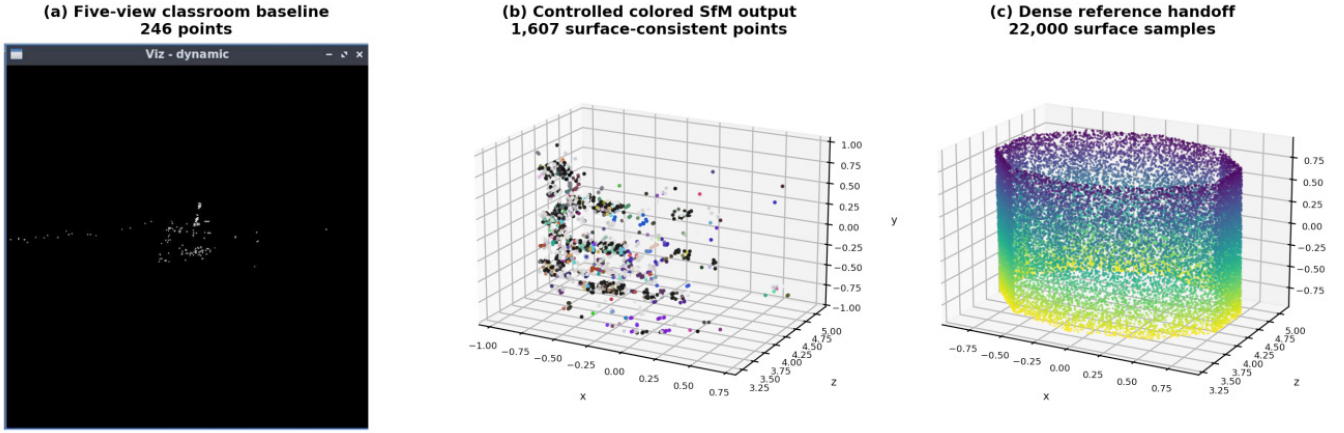


Figure 2. Density comparison: (a) five-view classroom baseline (246 points); (b) surface-consistent subset (distance to the reference surface < 0.12 coordinate units) of the 13,883-point controlled artifact reconstruction (1,607 points); and (c) dense reference-surface handoff (22,000 points).

5.2. Sensitivity to scene type and view count

Table 1 summarizes reconstruction quality across scene types and view counts. All cameras met the operational criterion for addition to the sequential pose chain, but density did not increase monotonically with image count. The 24-view condition produced the largest cloud for every scene. With 48 views on the same 68° arc, adjacent separation fell to 1.45° and the mean pose-validated inlier ratio declined to 6.7%–22.1%; the lower inlier ratios were accompanied by lower filtered point counts in all three scenes relative to the 24-view conditions. The repeated facade retained the largest 48-view point cloud but remained vulnerable to ambiguous correspondences.

Table 1. Reconstruction-quality indicators across scenes and view counts. The pose-validated inlier ratio is the unweighted mean over the adjacent image-pair ratios defined in Section 4.2; it is not a global registration-accuracy measure.

Scene	Views, n	Cameras in pose chain	Pose-validated inlier ratio, %	Reprojection RMSE, px	Filtered points, n
Textured object	12	12/12	62.3	0.43	8,024
Textured object	24	24/24	46.6	0.38	13,133
Textured object	48	48/48	6.7	0.37	3,842
Repeated facade	12	12/12	84.5	0.43	10,531

Continuation Table:

Scene	Views, n	Cameras in pose chain	Pose-validated inlier ratio, %	Reprojection RMSE, px	Filtered points, n
Repeated facade	24	24/24	51.3	0.42	16,154
Repeated facade	48	48/48	22.1	0.36	14,104
Curved artifact	12	12/12	89.3	0.44	8,930
Curved artifact	24	24/24	56.6	0.40	13,883
Curved artifact	48	48/48	12.5	0.34	7,822

Reprojection RMSE varied only from 0.34 to 0.44 pixels and was lowest in the 48-view conditions with the weakest adjacent-view parallax. This apparent improvement is not evidence of superior global geometry: short-baseline configurations can fit local image measurements while remaining poorly conditioned for depth. The result supports a central assessment principle of the module—students must interpret residuals together with pose-validated inlier ratio, parallax, pose behavior, and filtered-point retention.

As shown in **Figure 3**, median adjacent-pair rotation error ranged from 1.45° to 6.85°, while adjacent-pair translation-direction error spanned 2.55°–179.28°. At 48 views, the median translation-direction error was 78.33° for the facade, 90.29° for the curved artifact, and 179.28° for the textured object. Surface consistency was 0.0%–22.0%; the 24-view artifact in **Figure 2** achieved 11.6%. Addition to the pose chain and low reprojection residual did not imply accurate global geometry; the benchmark demonstrates that the workflow exposes failure modes rather than asset-grade accuracy.

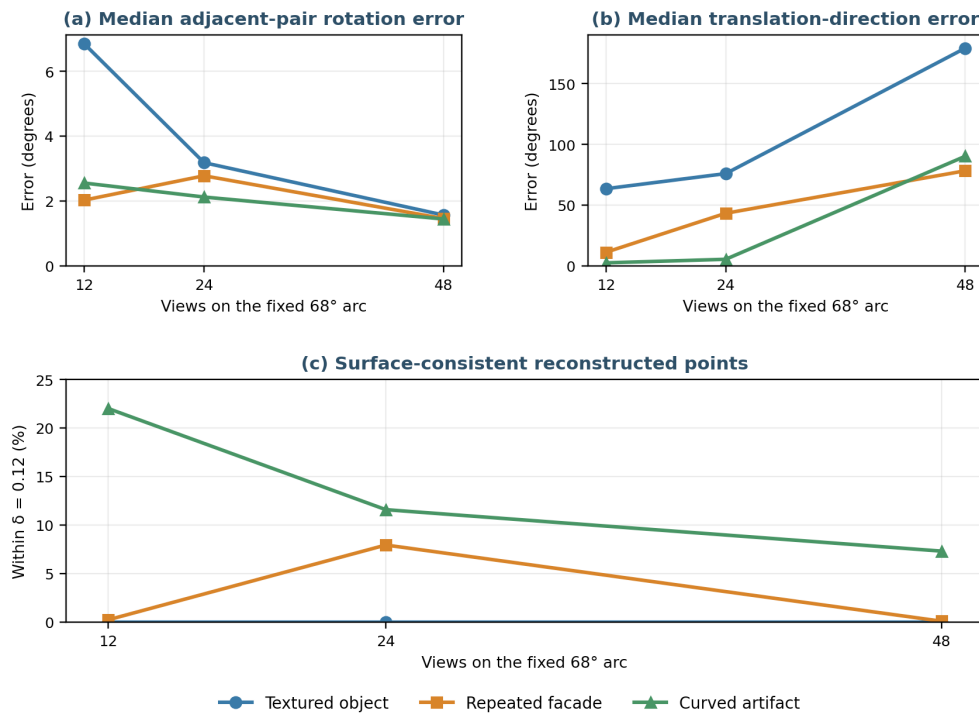


Figure 3. Ground-truth accuracy across controlled scenes: (a) median adjacent-pair rotation error; (b) median adjacent-pair translation-direction error; and (c) percentage of reconstructed points within $\delta = 0.12$ of the known surface. Lower values are better in panels (a) and (b); higher values are better in panel (c).

5.3. Timing analysis

Table 2 summarizes the stage-level timing statistics for the ten executions of the five-view baseline.

Table 2. Summary statistics for the ten-run five-view timing baseline. SD = sample standard deviation; IQR = interquartile range.

Stage	Mean, ms	SD, ms	Median, ms	IQR, ms	Share, %
ORB feature extraction and descriptor construction	49.8	1.87	50.0	0.75	27.0
Feature processing and matching	85.5	3.72	87.0	2.00	46.4
Camera-pose estimation	41.1	3.96	42.0	2.50	22.3
Auxiliary operations	7.9	1.85	8.0	0.75	4.3
Total	184.3	10.45	187.5	5.25	100.0

Feature processing and matching account for 46.4% of the mean total runtime, followed by ORB feature extraction and descriptor construction at 27.0% and camera-pose estimation at 22.3%. This timing profile serves as a pedagogical example rather than an algorithmic novelty claim. It supports an exercise based on Amdahl's law [33], in which p is the measured share of the optimized stage and s is its local speed-up:

$$S = \frac{1}{(1-p) + \frac{p}{s}}. \quad (7)$$

For $p = 0.464$, a twofold acceleration of feature processing and matching predicts an overall speed-up of approximately 1.30, while even an infinite speed-up of that stage limits the overall speed-up to approximately 1.87. Students must then test whether an optimization preserves correspondence quality and reconstruction stability.

5.4. Instructional implementation framework

Figure 4 represents the instructional artifact as a C4 Dynamic diagram at component scope [37]. Numbered interactions show how image and camera inputs move through feature correspondence, robust geometry, quality profiling, and evidence export. The dashed external system denotes prospective downstream metaverse-content processing (densification or neural rendering, meshing and texturing where applicable, level-of-detail and coordinate control, semantic and provenance packaging, glTF/GLB validation, and engine import), which lies outside the OpenCV reference implementation.

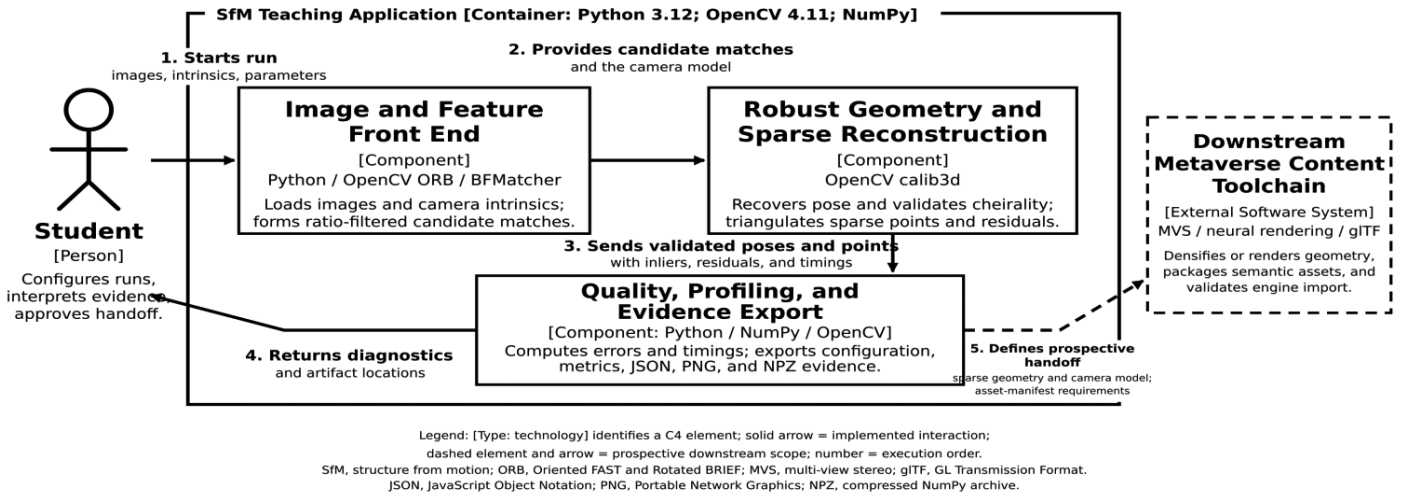


Figure 4. C4 Dynamic diagram at component scope for the instrumented OpenCV SfM teaching workbench. Numbers show execution order; the dashed system and relationship denote prospective downstream activities outside the reference implementation.

6. Discussion

6.1. Nature of the contribution

The contribution is pedagogical and methodological rather than algorithmic. ORB, Hamming matching, RANSAC, essential-matrix recovery, and triangulation are established components [1,3,5,7]. The methodological contribution lies in converting their dependencies and failure modes into a coherent instructional instrument with explicit outcomes, quality gates, controlled sensitivity conditions, an assessment rubric, and a reproducible handoff to semantic 3D content. Accordingly, the assembly of OpenCV calls is not presented as a technical innovation.

The results demonstrate that more images do not necessarily produce a better model. When the capture arc is fixed, additional views reduce adjacent parallax; local reprojection residuals can remain low while depth conditioning deteriorates. This provides a concrete diagnostic problem in which students must distinguish density, local residual fit, and global geometric reliability.

6.2. Position in the metaverse-content lifecycle

The OpenCV output is not a finished metaverse asset. Sparse camera geometry and pairwise triangulated points form an inspectable acquisition scaffold for multi-view stereo, neural rendering, Gaussian splatting, meshing, texturing, level-of-detail generation, semantic annotation, and engine import. This position is pedagogically useful because students can trace how calibration or correspondence errors propagate into downstream representations.

The compact demonstrator is therefore not a substitute for a complete reconstruction system. Because it omits persistent multi-view feature tracks, perspective-n-point camera resection, and bundle adjustment, instructors who require geometrically reliable reconstructions should pair the module with a complete, globally refined pipeline such as COLMAP [3]. A useful extension is to compare the demonstrator's local diagnostics with COLMAP's registered cameras and optimized geometry before any downstream asset-generation step.

An asset-readiness check should cover metric scale or declared scale ambiguity, coordinate-system conversion, orientation, normals, mesh integrity, polygon or

Gaussian-primitive budget, texture resolution, license, provenance, and successful glTF/GLB validation [34]. The semantic layer can link visual entities to concepts and relations [27–29], while an ontology-mediated service can use those descriptions for domain-specific explanation or interaction [31]. This establishes a substantive metaverse orientation without claiming that sparse points alone are deployable content.

6.3. Pedagogical implications

Teaching time is allocated by conceptual dependency and assessment demand, not by runtime share. Calibration and capture may be inexpensive computationally but are prerequisites for meaningful geometry. Matching receives extended laboratory attention because it is both computationally prominent and causally connected to pose and triangulation. Profiling is therefore embedded within a quality-constrained optimization task rather than presented as an isolated speed contest.

The module can support face-to-face, hybrid, or remote laboratories when installation, sample data, expected outputs, and failure cases are packaged in advance. It can also support advanced assignments that compare SfM with learned sparse-view reconstruction methods [16–18] or use the semester-scale educational dataset [26]. These extensions require different computational resources and should not be treated as equivalent substitutes for the transparent baseline.

6.4. Threats to validity

Construct validity is limited because point count measures density rather than accuracy, reprojection RMSE measures image-space consistency rather than complete geometry, and timing does not measure learning. Known-trajectory pose errors and surface consistency provide external checks for the controlled scenes, but they remain dependent on the single baseline-scaling convention and simplified reference geometry. Internal validity is affected by RANSAC stochasticity, caching, thread scheduling, and the use of a virtualized CPU; a fixed seed and explicit environment reduce but do not eliminate these effects.

External validity is restricted by three controlled scene classes, one intrinsic camera model, one capture arc, and one software configuration. The rendered benchmark permits known camera parameters but cannot reproduce all effects arising from real optics, exposure variation, motion blur, non-Lambertian surfaces, or background clutter. The five-view USB-camera sequence lacks retained calibration evidence and is excluded from accuracy claims.

Pedagogical validity remains prospective because the study included no learners, instructors, comparison condition, or authentic classroom deployment. Prior programming knowledge, instructor support, and institutional computational access may influence outcomes. Ecological validity is also bounded: a sparse or reference point cloud does not establish that a textured, optimized, semantically described asset will perform acceptably in an immersive application.

7. Conclusion

This study presents a reproducible instructional methodology based on an OpenCV SfM demonstration for metaverse-oriented content generation. The artifact links calibrated projection, ORB correspondence, robust pose estimation, triangulation, quality diagnostics, computational profiling, reproducibility, and semantic asset packaging through assessable outputs.

Technical characterization across three scene types and three image-count

conditions shows that 24 views produced the largest filtered point clouds, while 48 closely spaced views reduced pose-validated inlier ratios despite low reprojection residuals. Ground-truth checks additionally found adjacent-pair translation-direction errors of 78.33° – 179.28° in those 48-view conditions and surface consistency of only 0.0%–22.0% across all conditions; these results indicate that the method should be treated as an inspectable teaching demonstrator rather than an accurate asset generator. In the five-view timing baseline, feature processing and matching was the largest measured stage; its educational value is to support quality-constrained optimization and analysis based on Amdahl's law rather than an algorithmic novelty claim.

The scope of the technical evidence remains limited to three controlled rendered scene classes, one intrinsic camera model, one 68° capture arc, one fixed OpenCV configuration, and one deterministic evaluation run per scene–view condition. The benchmark does not reproduce real-camera effects such as lens distortion, exposure variation, motion blur, non-Lambertian appearance, or background clutter; validation with real image collections, broader scene classes, and more varied capture trajectories is therefore required before the reported behavior can be generalized.

The methodology explicitly stops short of claiming classroom effectiveness or a finished metaverse asset. A subsequent study should deploy the module with a student cohort recruited with informed consent, evaluate conceptual change and transfer performance, and validate the complete transition from calibrated sparse geometry to an optimized, semantically annotated, engine-imported asset.

Author contributions: Conceptualization, OB and KM; methodology, OB and KM; software, OB; validation, OB, AB and TN; formal analysis, OB and KM; investigation, IM; data curation, OB; writing—original draft preparation, OB and KM; writing—review and editing, OB, KM, AB, IM and TN; visualization, OB; supervision, OB and KM; project administration, OB. All authors have read and agreed to the final version of the manuscript.

Funding: This research received no external funding.

Declaration on Generative AI: The authors used an OpenAI Codex agent running GPT-5.6 Sol (model identifier: gpt-5.6-sol) in Ultra mode to assist with literature-oriented drafting, code generation, document organization, and the production and analysis of the controlled synthetic reference-scene benchmark and technical figures. The benchmark was synthetically generated and does not represent human-participant observations or a classroom intervention. The locally deployed gpt-oss-120b model was additionally used for grammar and spelling support. Independent author verification comprised direct inspection of the generated source code; execution of the released benchmark from a clean package directory with the documented dependencies and fixed seed; comparison of the implemented intrinsic matrix, camera trajectories, scene geometry, and sampled reference surfaces with the ground-truth specification in Section 4.3; recalculation of selected geometric metrics and summary statistics from the retained machine-readable data; and cross-checking of the reported table and figure values against the released JSON outputs. The authors independently verified the equations and interpretation and take full responsibility for the final content.

Data Availability Statement: The controlled benchmark generator, machine-readable results, representative images, and figure source files are publicly available in the supplementary reproducibility package on Hugging Face at <https://doi.org/10.57967/>

hf/9819. The five-view baseline sequence and retained outputs are available from the corresponding author upon reasonable request; because its calibration record was not preserved, the baseline is not used for accuracy claims.

Code Availability Statement: The executable OpenCV benchmark script, dependency specification, fixed seed, and run instructions are publicly available in the same supplementary reproducibility package at <https://doi.org/10.57967/hf/9819>.

Ethics Statement: This study did not involve human participants, animals, clinical data, or personally identifiable information; therefore, ethics committee approval was not required.

Acknowledgments: The authors thank Oleksandr V. Palagin for scientific mentorship on cognitive systems, ontology engineering, and semantic support for reproducible research and educational information systems.

Conflicts of interest: The authors declare no conflicts of interest.

References

1. Hartley R, Zisserman A. Multiple View Geometry in Computer Vision, 2nd ed. Cambridge, UK: Cambridge University Press; 2004. <http://doi.org/10.1017/CBO9780511811685>
2. Faugeras O. Three-Dimensional Computer Vision: A Geometric Viewpoint. Cambridge, MA, USA: MIT Press; 1993. ISBN: 9780262061582.
3. Schönberger JL, Frahm JM. Structure-from-Motion Revisited. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2016; Las Vegas, NV, USA. p. 4104–4113. <http://doi.org/10.1109/CVPR.2016.445>
4. Snavely N, Seitz SM, Szeliski R. Photo Tourism: Exploring Photo Collections in 3D. ACM Transactions on Graphics. 2006,25(3):835–846. <http://doi.org/10.1145/1141911.1141964>
5. Rublee E, Rabaud V, Konolige K, et al. ORB: An Efficient Alternative to SIFT or SURF. Proceedings of the IEEE International Conference on Computer Vision (ICCV); 2011; Barcelona, Spain. p. 2564–2571. <http://doi.org/10.1109/ICCV.2011.6126544>
6. Schönberger JL, Hardmeier H, Sattler T, et al. Comparative Evaluation of Hand-Crafted and Learned Local Features. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017; Honolulu, HI, USA. p. 6959–6968. <http://doi.org/10.1109/CVPR.2017.736>
7. Fischler MA, Bolles RC. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Communications of the ACM. 1981,24(6):381–395. <http://doi.org/10.1145/358669.358692>
8. Mur-Artal R, Tardós JD. ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras. IEEE Transactions on Robotics. 2017,33(5):1255–1262. <http://doi.org/10.1109/TRO.2017.2705103>
9. Campos C, Elvira R, Gómez Rodríguez JJ, et al. ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual-Inertial, and Multimap SLAM. IEEE Transactions on Robotics. 2021,37(6):1874–1890. <http://doi.org/10.1109/TRO.2021.3075644>
10. Eigen D, Puhrsch C, Fergus R. Depth Map Prediction from a Single Image Using a Multi-Scale Deep Network. Advances in Neural Information Processing Systems. 2014,27:2366–2374. Available at: https://proceedings.neurips.cc/paper_files/paper/2014/hash/91c56ce4a249fae5419b90cba831e303-Abstract.html
11. Yao Y, Luo Z, Li S, et al. MVSNet: Depth Inference for Unstructured Multi-view Stereo. Computer Vision – ECCV 2018. Lecture Notes in Computer Science. 2018,11212:785–801. http://doi.org/10.1007/978-3-030-01237-3_47
12. Mildenhall B, Srinivasan PP, Tancik M, et al. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. Computer Vision – ECCV 2020. Lecture Notes in Computer Science. 2020,12346:405–421. http://doi.org/10.1007/978-3-030-58452-8_24
13. Müller T, Evans A, Schied C, et al. Instant Neural Graphics Primitives with a Multiresolution Hash Encoding. ACM Transactions on Graphics. 2022,41(4):102:1–102:15. <http://doi.org/10.1145/3528223.3530127>
14. Kerbl B, Kopanas G, Leimkühler T, et al. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. ACM Transactions on Graphics. 2023,42(4):139:1–139:14. <http://doi.org/10.1145/3592433>

15. Yu Z, Chen A, Huang B, et al. Mip-Splatting: Alias-Free 3D Gaussian Splatting. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2024:19447–19456. <http://doi.org/10.1109/CVPR52733.2024.01839>
16. Wu J, Li R, Zhu Y, et al. Sparse2DGS: Geometry-Prioritized Gaussian Splatting for Surface Reconstruction from Sparse Views. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2025:11307–11316. <http://doi.org/10.1109/CVPR52734.2025.01056>
17. Tang Z, Fan Y, Wang D, et al. MV-DUST3R+: Single-Stage Scene Reconstruction from Sparse Views in 2 Seconds. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2025:5283–5293. <http://doi.org/10.1109/CVPR52734.2025.00498>
18. Zhang X, Zheng X, Yin Y, et al. AnchorSplat: Feed-Forward 3D Gaussian Splatting With 3D Geometric Priors. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2026:18924–18933. Available at: https://openaccess.thecvf.com/content/CVPR2026/html/Zhang_AnchorSplat_Feed-Forward_3D_Gaussian_Splatting_With_3D_Geometric_Priors_CVPR_2026_paper.html
19. Geronimo D, Serrat J, Lopez AM, et al. Traffic Sign Recognition for Computer Vision Project-Based Learning. IEEE Transactions on Education. 2013,56(3):364–371. <http://doi.org/10.1109/TE.2013.2239997>
20. Abichandani P, Iaboni C, Lobo D, et al. Artificial Intelligence and Computer Vision Education: Codifying Student Learning Gains and Attitudes. Computers and Education: Artificial Intelligence. 2023,5:100159. <http://doi.org/10.1016/j.caeai.2023.100159>
21. Jeon IS, Kang SJ, Kang SJ. A Staged Framework for Computer Vision Education: Integrating AI, Data Science, and Computational Thinking. Applied Sciences. 2024,14(21):9792. <http://doi.org/10.3390/app14219792>
22. Gevorgyan M, Mamikonyan A, Beyeler M. OpenCV 4 with Python Blueprints: Build Creative Computer Vision Projects with the Latest Version of OpenCV 4 and Python 3, 2nd ed. Birmingham, UK: Packt Publishing; 2020. ISBN: 9781789801811.
23. Sigut J, Castro M, Arnay R, et al. OpenCV Basics: A Mobile Application to Support the Teaching of Computer Vision Concepts. IEEE Transactions on Education. 2020,63(4):328–335. <http://doi.org/10.1109/TE.2020.2993013>
24. Pratt-Sitaula B. Getting Started with Structure from Motion (SfM) Photogrammetry. Cutting Edge. Science Education Resource Center at Carleton College; 2020. <http://doi.org/10.67165/tb4tmgze>
25. Fominykh M, Wild F, Klamma R, et al. Model Augmented Reality Curriculum. Proceedings of the Working Group Reports on Innovation and Technology in Computer Science Education. 2020:131–149. <http://doi.org/10.1145/3437800.3439205>
26. Seefried E, Jung C, Venkatesha V, et al. CSU101: An Educational Dataset for Introductory Computer Vision. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). 2026:10864–10873. Available at: https://openaccess.thecvf.com/content/CVPR2026W/CV4Edu/html/Seefried_CSU101_An_Educational_Dataset_for_Introductory_Computer_Vision_CVPRW_2026_paper.html
27. Palagin O, Kaverinskiy V, Malakhov K, et al. Fundamentals of the Integrated Use of Neural Network and Ontolinguistic Paradigms: A Comprehensive Approach. Cybernetics and Systems Analysis. 2024,60(1):111–123. <http://doi.org/10.1007/s10559-024-00652-z>
28. Petrenko M, Cohn E, Shchurov O, et al. Ontology-Driven Computer Systems: Elementary Senses in Domain Knowledge Processing. South African Computer Journal. 2023,35(2):127–144. <http://doi.org/10.18489/sacj.v35i2.17445>
29. Palagin OV, Petrenko MG, Velychko VYu, et al. Development of Formal Models, Algorithms, Procedures, Engineering and Functioning of the Software System “Instrumental Complex for Ontological Engineering Purpose.” CEUR Workshop Proceedings. 2017,1843:221–232. Available at: <http://ceur-ws.org/Vol-1843/221-232.pdf>
30. Palagin OV, Velychko VYu, Malakhov KS, et al. Research and Development Workstation Environment: The New Class of Current Research Information Systems. CEUR Workshop Proceedings. 2018,2139:255–269. Available at: <http://ceur-ws.org/Vol-2139/255-269.pdf>
31. Palagin O, Kaverinsky V, Petrenko M, et al. Digital Health Systems: Ontology-Based Universal Dialog Service for Hybrid E-Rehabilitation Activities Support. 2023 IEEE 12th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS). 2023,1:84–89. <http://doi.org/10.1109/IDAACS58523.2023.10348639>
32. Biggs J, Tang C. Teaching for Quality Learning at University, 4th ed. Maidenhead, UK: Open University Press; 2011. ISBN: 9780335242757.

33. Amdahl GM. Validity of the Single Processor Approach to Achieving Large Scale Computing Capabilities. *AFIPS Conference Proceedings*. 1967,30:483–485. <http://doi.org/10.1145/1465482.1465560>
34. Khronos Group. glTF 2.0 Specification. Available at: <https://registry.khronos.org/glTF/specs/2.0/glTF-2.0.html> (accessed 18 July 2026).
35. Malakhov K, Bahatskyi O. teaching-sfm-a01260048-supplementary [dataset]. Hugging Face; 2026. <http://doi.org/10.57967/hf/9819>
36. Weber S, Demmel N, Chan TC, et al. Power Bundle Adjustment for Large-Scale 3D Reconstruction. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; 2023; Vancouver, BC, Canada. p. 281–289. <http://doi.org/10.1109/CVPR52729.2023.00035>
37. Brown S. The C4 Model for Visualising Software Architecture. Available at: <https://c4model.com/> (accessed 23 July 2026).