LEARNING FROM AUTOMATIC 3D GAUSSIAN SPLATTING EDITS FOR ONLINE BAYESIAN SCENE CHANGE DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Online scene change detection requires integrating incoming observations to determine which regions have changed relative to a reference scene. We update change beliefs as new images arrive using generalised Bayesian evidence fusion. However, the effective pixel prior and evidence models are unknown beforehand, and numerical Bayesian inference can be costly. We address these challenges by learning offline from automatic edits to a reference 3D Gaussian Splatting (3DGS) model. These edits generate labelled training views without manual annotations. Together with unchanged reference observations, they provide supervision that improves current-view change evidence and estimates the effective pixel prior and response statistics. The resulting models enable online prediction from current evidence and historical support, with closed-form local Bayes factors updating an accumulated 3D Gaussian field. These updates require neither online backpropagation nor past-image replay. Object-guided refinement produces the final change masks. On PASLCD, our method achieves state-of-the-art accuracy (67.08% mIoU, 78.37% F1) while also providing the fastest complete pipeline among the evaluated online methods at 16.6 FPS.

1 INTRODUCTION

Scene change detection identifies physical changes in an environment from images captured at different times. When captures follow different camera trajectories, viewpoint and visibility differences complicate direct image comparison (Lu et al., 2025; Wu et al., 2026). A reference 3D Gaussian Splatting (3DGS) model (Kerbl et al., 2023) enables rendering the earlier scene at query viewpoints, providing a common view for comparison. Recent methods, such as MV3DCD (Galappaththige et al., 2025) and O-SCD (Galappaththige et al., 2026b), integrate multi-view change cues by optimizing per-Gaussian change parameters, using batch and online processing, respectively.

In O-SCD, processing each new image requires iterative optimization using current and replayed observations, adding computational overhead during online inference. However, updating predictions as new evidence arrives does not inherently require repeated parameter optimization. This motivates updating change beliefs as new images arrive. Generalised Bayesian inference (Bissiri et al., 2016) provides a framework to combine a prior probability of change with incoming evidence.

Putting this idea into practice presents two challenges. The effective pixel prior and the relationship between image differences and change evidence are unknown beforehand. Bayesian evidence computation can also require numerical integration or sampling (Kass & Raftery, 1995), adding substantial cost when repeated for incoming frames. We address these challenges by learning offline from automatic reference 3DGS edits and unchanged observations. The edits provide labelled training views without manual annotations, supporting scene-specific learning of current-view evidence, the effective pixel prior, and response statistics. Using these learned models, we combine current evidence with historical support for prediction and update the accumulated field through closed-form local Bayes factors. These updates require neither online backpropagation nor past-image replay. Finally, object-guided refinement sharpens boundaries and completes missed object regions.

On PASLCD, our method achieves 67.08% mIoU and 78.37% F1, outperforming O-SCD Online by 17.84 mIoU points while running at 16.6 FPS, compared with 14.6 FPS for O-SCD Online.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Without refinement, the pixel pipeline maintains 60.13% mIoU at 25.1 FPS. Our complete method also exceeds GS-Diff's reported 64.40% mIoU (Galappaththige et al., 2026a) without requiring a post-change reconstruction. Given an existing reference reconstruction, offline preparation takes approximately 3.6 minutes per scene, with all learned components fixed before query processing.

2 Related Work

Scene change detection. Image-pair methods identify changes through correspondence and appearance comparison. GeSCF uses SAM features and mask matching without task-specific training (Kim & Kim, 2025), while SceneDiff compares objects across before–after captures using semantic and geometric information (Wu et al., 2026). A reference 3D representation additionally enables comparisons at matched viewpoints and integration across observations. 3DGS-CD associates image-level detections across views (Lu et al., 2025), and SCaR-3D combines change cues through multi-view voting and pruning (Zhou et al., 2026). MV3DCD optimizes Gaussian change channels from a collection of post-change images (Galappaththige et al., 2025); O-SCD extends this approach to streaming queries, updating change parameters using current and replayed observations (Galappaththige et al., 2026b). CL-Splats uses detected changes to guide local updates of a Gaussian scene representation (Ackermann et al., 2025). GS-Diff instead compares geometry and colour between independently reconstructed scene states (Galappaththige et al., 2026a). Our method accumulates evidence on the reference Gaussians without online optimization or past-image replay.

Learning from synthetic changes and 3D editing. Synthetic supervision reduces dependence on manually annotated changes. CYWS constructs training pairs from object segmentation data (Sachdeva & Zisserman, 2023a), while CYWS-3D generates paired observations from synthetic 3D scenes (Sachdeva & Zisserman, 2023b). Advances in 3DGS segmentation (Ye et al., 2024; Zhu et al., 2025), editing (Chen et al., 2024; Lee et al., 2025), and relighting (Liang et al., 2024; Gao et al., 2024) support controlled changes to scene content and appearance. These capabilities also support downstream learning: RoboSplat edits Gaussian scenes and varies their appearance to train robot policies (Yang et al., 2025), while Zanjani et al. (2025) insert Gaussian objects into driving scenes to train 3D detectors. We use reference-scene edits and lighting variations to generate labelled image pairs. Together with unchanged reference observations, these pairs support offline learning of the effective pixel prior and evidence models for online scene change detection.

Bayesian evidence accumulation. Bayesian updates provide a way to combine successive observations with existing beliefs. In scene mapping, POCD updates probabilistic object states using geometric and semantic information to track changes and maintain volumetric maps (Qian et al., 2022). Bissiri et al. (2016) provide a generalised Bayesian framework that connects observations to belief updates through a loss function. For online time-series changepoint detection, Altamirano et al. (2023) construct generalised Bayesian updates with closed-form posterior parameters using diffusion score matching. We apply generalised Bayesian prediction to scene change detection, using offline-learned response statistics to evaluate local Bayes factors in closed form for accumulation on reference Gaussians.

3 Preliminaries

Given a reference 3D Gaussian Splatting (3DGS) model $\mathcal{G}$ and its registered reference images $\mathcal{I}_{\mathrm{ref}}$, we detect changes in incoming query images I_t relative to the reference scene. Comparing each query with the reference requires rendering the earlier scene from the query viewpoint. Following O-SCD (Galappaththige et al., 2026b), we first match XFeat keypoints (Potje et al., 2024) between the query and registered reference images to obtain 2D–3D correspondences, then estimate the query pose using PnP-RANSAC.

At the estimated pose, the reference 3DGS renders image $\widehat{I}_t$ through front-to-back alpha compositing (Kerbl et al., 2023). Each Gaussian $g \in \mathcal{G}$ contributes a view-dependent colour $\mathbf{c}_{tg}$ with projected opacity $\alpha_{tug} \in [0,1]$ at pixel u, giving $\widehat{I}_t(u) = \sum_g a_{tug}\mathbf{c}_{tg}$. Here, $a_{tug} = \alpha_{tug}\prod_{h \prec_t g}(1-\alpha_{tuh})$ denotes its rendering contribution, with $h \prec_t g$ identifying Gaussians preceding it along the viewing ray. These contributions also connect pixels to reference Gaussians for evidence aggregation in Section 4. The reference reconstruction remains fixed during online inference.

Comparing I_t with $\widehat{I}_t$ produces the initial change cues. Following O-SCD, we compute a photometric difference from RGB L_1 and D-SSIM (Wang et al., 2004), and a semantic difference from the channel-averaged absolute difference between frozen SAM2 encoder features (Ravi et al., 2025). Per-image normalization and upsampling of the semantic difference produce pixel-aligned cues $\widetilde{D}_t^{\mathrm{rgb}}$ and $\widetilde{D}_t^{\mathrm{sem}}$. Their average gives the initial difference response $R_t^0 = \frac{1}{2}(\widetilde{D}_t^{\mathrm{rgb}} + \widetilde{D}_t^{\mathrm{sem}}) \in [0,1]$, which serves as the input to our evidence learning and online inference. Appendix A.2 provides the detailed cue construction and normalization.

4 METHOD

4.1 PROBLEM FORMULATION AND OVERVIEW

Image differences mix physical changes with rendering artifacts and illumination shifts. Our method uses generalised Bayesian inference to interpret these responses and accumulate change evidence. For query image I_t, current-view evidence C_t is constructed from the initial difference response R_t^0, while historical support H_t summarizes evidence from earlier observations in the query view. We combine them through generalised Bayesian inference (Bissiri et al., 2016) with an effective pixel prior π, fitted per scene to set the prediction operating point. The posterior change probability $Q_t^{\mathrm{pixel}}(u) \triangleq q_t(y=1 \mid u)$, where $y=1$ denotes change, satisfies

$$\underbrace{\frac{Q_t^{\mathrm{pixel}}(u)}{1-Q_t^{\mathrm{pixel}}(u)}}_{\text{posterior odds}} = \underbrace{\frac{\pi}{1-\pi}}_{\text{prior odds}} \cdot \underbrace{\frac{C_t(u)}{1-C_t(u)}}_{\text{current odds}} \cdot \underbrace{\frac{H_t(u)}{1-H_t(u)}}_{\text{history odds}}. \tag{1}$$

Neutral evidence, $C_t(u) = H_t(u) = 0.5$, leaves $Q_t^{\mathrm{pixel}}(u) = \pi$. Thresholding Q_t^{pixel} at 0.5 gives the pixel mask $\widehat{Y}_t^{\mathrm{pixel}}$.

However, the prior and evidence models are unknown beforehand: we must determine how image differences indicate change and how much evidence each observation contributes to history. Automatic reference 3DGS edits and unchanged reference observations provide supervision for offline learning. The learned response suppressor $\mathcal{T}$ and log-likelihood-ratio mapping ℓ_{img} construct current-view evidence. The unchanged-response model (μ_g, τ_g^2, c) supplies statistics for closed-form evidence accumulation. A contour filter ℓ_{contour} supports subsequent mask refinement.

All five models are fitted offline and remain fixed during online inference. Section 4.2 describes how they are used for prediction, history accumulation, and mask refinement (Figure 1), taking their fitted parameters as given. Section 4.3 explains how automatic reference edits and unchanged observations are used to fit them (Figure 2).

4.2 CAUSAL ONLINE CHANGE DETECTION

For each incoming query image I_t, we construct current-view evidence C_t, which supports both prediction and history accumulation. For prediction, C_t combines with historical support H_t from earlier frames; for accumulation, it provides the Gaussian responses used to compute local Bayes factors. The updated history is used only by subsequent queries.

Current-view evidence. To suppress view-synthesis artifacts in initial response R_t^0 (Section 3), two predictors give change log odds ℓ_{base} from response features and ℓ_{full} from response and reference rendering features. The response suppressor $\mathcal{T}$ uses their difference to obtain corrected response R_t:

$$R_t = \sigma\big(\mathrm{logit}\, R_t^0 + \mathcal{T}(\ell_{\mathrm{base}}, \ell_{\mathrm{full}})\big). \tag{2}$$

Here $\sigma(x) = (1+e^{-x})^{-1}$ is the sigmoid function and $\mathrm{logit}(p) = \log[p/(1-p)]$.

Although per-frame normalization unifies diverse cues, independent scaling leaves R_t contrast-dependent. We therefore map R_t to a shared evidence scale learned from reference data via the monotonic LLR mapping ℓ_{img}, giving probabilistic image evidence $P_t = \sigma(\ell_{\mathrm{img}}(R_t)) \in (0,1)$, where $P_t = 0.5$ represents neutral evidence. To suppress isolated pixel noise, we aggregate pixel evidence onto Gaussians and render it back to pixels using Gaussian blending weights a_{tug}. Specifically,

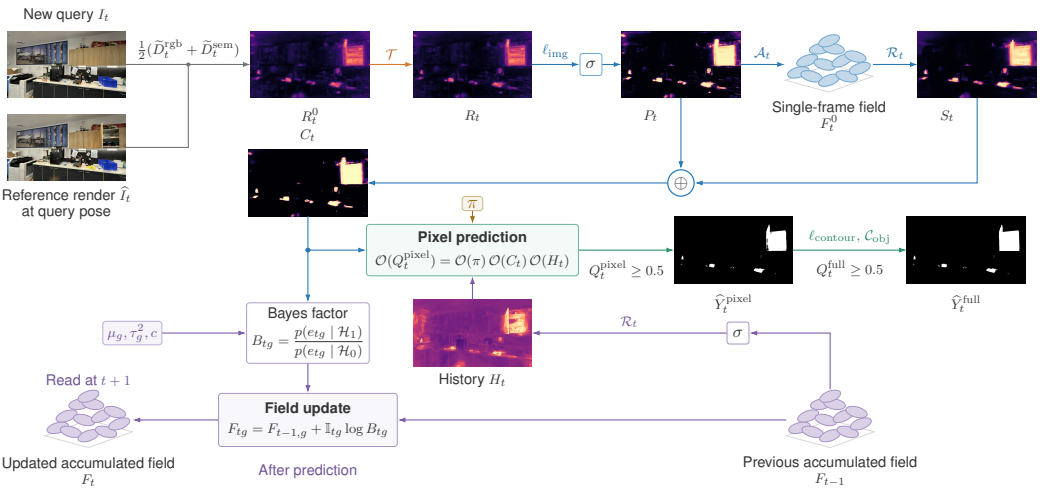

Figure 1: **Online evidence fusion pipeline.** Current evidence C_t combines mapped image evidence P_t and Gaussian-aggregated support S_t. Fusion with prior π and history H_t from F_{t-1} gives Q_t^{pixel}; refinement produces Q_t^{full}. Thresholding at 0.5 gives masks $\widehat{Y}_t^{\text{pixel}}$ and $\widehat{Y}_t^{\text{full}}$, respectively. Local Bayes factors B_{tg} update F_{t-1} to F_t for subsequent queries. $\oplus$ adds log odds.

aggregation $\mathcal{A}_t$ averages 2D pixel evidence X_u within each Gaussian's footprint, while rendering $\mathcal{R}_t$ blends 3D Gaussian support V_g back into pixels:

$$[\mathcal{A}_t(X)]_g = \frac{\sum_u a_{tug} X_u}{s_{tg}}, \qquad [\mathcal{R}_t(V)]_u = \frac{\sum_g a_{tug} V_g}{\chi_{tu}}, \tag{3}$$

where $s_{tg} = \sum_u a_{tug}$ and $\chi_{tu} = \sum_g a_{tug}$ normalize the weights, preserving neutral evidence (0.5). We apply these operators to obtain the single-frame response field $F_t^0 = \mathcal{A}_t(P_t)$ with entries in $[0, 1]$, and Gaussian-aggregated support $S_t = \mathcal{R}_t(F_t^0)$. Combining P_t with S_t retains pixel detail while incorporating evidence shared through Gaussian footprints, giving current-view evidence C_t:

$$C_t = \sigma(\text{logit}\, P_t + \text{logit}\, S_t)\,. \tag{4}$$

3D evidence accumulation. To determine how current-view evidence C_t should contribute to history, we evaluate Gaussian responses relative to their expected behaviour under unchanged conditions. The unchanged-response model provides each Gaussian's expected response μ_g, the uncertainty in the estimated unchanged-response mean τ_g^2, and the spatial co-response coefficient c. Each Gaussian's aggregated response $r_{tg} = [\mathcal{A}_t(C_t)]_g$ is standardized as $z_{tg} = (r_{tg} - \mu_g)/\sqrt{\tau_g^2 + v_{tg}}$, where $v_{tg} = [\mathcal{A}_t(C_t^2)]_g - r_{tg}^2$ measures response variance within its footprint. The observation indicator $\mathbb{I}_{tg} = \mathbb{1}\{s_{tg} \geq 1\}$ selects Gaussians with sufficient rendering contribution.

A high response from one Gaussian may reflect local noise. Since object changes can affect several neighbouring Gaussians, we combine nearby responses using the neighbourhood mean $\bar{z}_{tg} = (\sum_j \eta_{gj} \mathbb{I}_{tj} z_{tj})/(\sum_j \eta_{gj} \mathbb{I}_{tj})$, where j indexes neighbours of Gaussian g on the reference 3D graph. Edge weights η_{gj} reflect spatial and colour similarity (Appendix, Eq. 23), and $\mathbb{I}_{tj}$ applies the same observation rule to neighbour j. Neighbours can also respond together without change. Under the unchanged hypothesis $\mathcal{H}_0$, we model the central response z_{tg} and neighbourhood mean $\bar{z}_{tg}$ as jointly Gaussian with zero means, unit central variance, and covariance approximated by c. This gives the conditional expectation $\mathbb{E}[\bar{z}_{tg} \mid z_{tg}, \mathcal{H}_0] \approx c z_{tg}$ (Appendix, Eq. 30). Subtracting it yields the residual $e_{tg} = \bar{z}_{tg} - c z_{tg}$, which measures neighbour response beyond that expected from the centre.

To quantify this residual as change evidence, we model $e_{tg} \mid \mathcal{H}_0 \sim \mathcal{N}(0, \nu_{tg})$. The residual variance $\nu_{tg} = \max(V_{tg} - c^2, \epsilon_\nu)$ uses the estimated variance V_{tg} of the neighbourhood mean (Appendix, Eq. 29); $\epsilon_\nu = 10^{-4}$ prevents degenerate variance. Under the change hypothesis $\mathcal{H}_1$, a shared

response increase $\delta \geq 0$ at the centre and its neighbours gives $e_{tg} \mid \delta, \mathcal{H}_1 \sim \mathcal{N}((1-c)\delta, \nu_{tg})$, with $\delta \sim \mathrm{HalfNormal}(1)$ and density $p_+(\delta) = 2\mathcal{N}(\delta; 0, 1)$ for $\delta \geq 0$. We integrate over δ to evaluate evidence for change. The Gaussian likelihood and half-normal prior give an exponent quadratic in δ, so completing the square evaluates this integral analytically. With $h_{tg} = (1-c)e_{tg}/\nu_{tg}$ and $J_{tg} = (1-c)^2/\nu_{tg} + 1$, the local Bayes factor B_{tg} is

$$B_{tg} = \frac{p(e_{tg} \mid \mathcal{H}_1)}{p(e_{tg} \mid \mathcal{H}_0)} = \frac{\displaystyle\int_0^\infty \mathcal{N}(e_{tg}; (1-c)\delta, \nu_{tg})\, p_+(\delta)\, d\delta}{\mathcal{N}(e_{tg}; 0, \nu_{tg})} = \frac{2}{\sqrt{J_{tg}}} \exp\left(\frac{h_{tg}^2}{2J_{tg}}\right) \Phi\left(\frac{h_{tg}}{\sqrt{J_{tg}}}\right), \quad (5)$$

where Φ is the standard normal cumulative distribution function. This expression requires neither numerical integration nor iterative optimization; its derivation and log form are given in Appendix C, Eqs. 32–33. Values $B_{tg} > 1$ support change and $B_{tg} < 1$ oppose it, with neutral evidence $B_{tg} = 1$ when no neighbours are observed. Adding these log Bayes factors to the previous accumulated field F_{t-1} gives the updated log-evidence field F_t for subsequent queries, with real-valued entries F_{tg} at Gaussians g:

$$F_{tg} = F_{t-1,g} + \mathbb{I}_{tg} \log B_{tg}. \quad (6)$$

Historical fusion and causal prediction. To predict pixel-level changes on query I_t, historical support H_t is rendered from the previous accumulated field F_{t-1} (with $F_0 = 0$):

$$H_t = \mathcal{R}_t(\sigma(F_{t-1})), \quad (7)$$

where σ maps accumulated log evidence into $(0, 1)$. Rendering accounts for view-dependent visibility and occlusion; unobserved Gaussians ($F = 0$) give neutral support 0.5 on covered pixels. Equation 1 gives the pixel score Q_t^{pixel}. Thresholding at 0.5 gives the pixel mask $\widehat{Y}_t^{\mathrm{pixel}}$.

Object-guided mask refinement. Gaussian overlap and alpha-blending can blur rendered change boundaries. We use connected components of $\widehat{Y}_t^{\mathrm{pixel}}$ to prompt the SAM2 decoder on both query and reference renderings. 3DGS-CD (Lu et al., 2025) fuses prompted segmentations into 3D object masks; our contour refinement retains the original scores of unprompted pixels. The contour model predicts log odds $\ell_{\mathrm{contour}}(u)$ from paired mask features. Their non-positive part penalizes pixel change scores, giving the contour-filtered score $Q_t^{\mathrm{contour}}(u)$:

$$Q_t^{\mathrm{contour}}(u) = \sigma\Big(\mathrm{logit}\, Q_t^{\mathrm{pixel}}(u) + \min(\ell_{\mathrm{contour}}(u),\, 0)\Big). \quad (8)$$

To recover missed object parts, surviving components prompt SAM2 on the query for candidate masks M. Their mean log-odds evidence is $\overline{L}_M = \mathrm{mean}_{u \in M} \mathrm{logit}\, Q_t^{\mathrm{pixel}}(u)$. We retain masks with $\overline{L}_M \geq 0$ and raise covered pixels to at least their mask score, giving the refined change score $Q_t^{\mathrm{full}}(u)$:

$$Q_t^{\mathrm{full}}(u) = \max\left(Q_t^{\mathrm{contour}}(u),\, \max_{M \ni u} \sigma(\overline{L}_M)\right), \quad (9)$$

where the inner maximum is over retained masks covering u and is set to 0 if none cover it. Thresholding at 0.5 gives the refined mask $\widehat{Y}_t^{\mathrm{full}}$ (Appendix F).

4.3 Learning from Automatic 3DGS Edits

Figure 2 illustrates our framework for learning from automatic 3DGS edits: we first synthesize object-level edits on the reference 3DGS to generate labelled training views, and then fit the scene-specific components in the order required by their inputs.

Automatic reference editing. We split reference cameras approximately 5:1 into an *editing subset* for learning and an *unchanged-only subset* for estimating unchanged-response statistics. A regular grid of automatic point prompts drives the SAM2 mask decoder (Ravi et al., 2025) on reference views, and masks are matched across views through their contributing Gaussians. Aggregating matched masks X via $[\mathcal{A}_t(X)]_g$ (Eq. 3) yields object membership $m_g \in [0, 1]$, measuring the fraction of each Gaussian's rendering footprint inside the object. We select objects based on their consistency across held-out views. Recolouring blends colours towards the palette colour furthest from the object's median colour with mixing coefficient $0.45 m_g$; removal scales opacity by $1 - m_g$; translation applies a shared 3D displacement to Gaussians with $m_g \geq 0.5$. Edits modify duplicate models, preserving

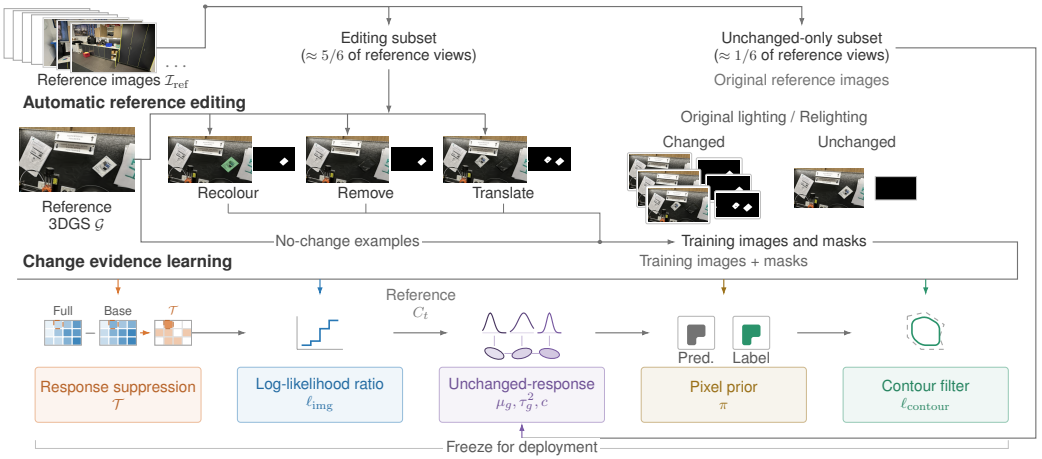

Figure 2: **Automatic reference editing and change evidence learning. Top**: Reference 3DGS edits (recolouring, removal, translation) and relighting synthesize labelled change pairs without manual annotation. **Bottom**: Five scene-specific components are fitted in the order required by their inputs (grey arrows): response suppressor $\mathcal{T}$, LLR mapping ℓ_{img}, unchanged-response model (μ_g, τ_g^2, c), pixel prior π, and contour filter ℓ_{contour}.

the original reference. For reference image $I_t^{\text{ref}} \in \mathcal{I}_{\text{ref}}$, original and edited models render $\widehat{I}_t$ and $\widehat{I}_t^{\text{edit}}$ at its registered pose; here t indexes reference views. Synthetic query $I_t^{\text{syn}} = I_t^{\text{ref}} + (\widehat{I}_t^{\text{edit}} - \widehat{I}_t)$ transfers changes to real reference images while preserving differences between reference images and their renderings. Regions covered by edit masks are labelled changed (1); valid regions outside are labelled unchanged (0), with ambiguous boundaries ignored. We introduce lighting variations using affine colour transforms, optionally augmented with normal-dependent directional shading, fitted to differences between unchanged reference images and their renderings. These transformations preserve the change labels (details and additional examples in Appendix A.1, Figure 4).

Response suppression. We fit the response suppressor $\mathcal{T}$ using two pixel-wise logistic classifiers on the editing subset. *Base* uses initial response R_t^0 and its local texture statistics (mean, standard deviation, gradient). *Full* adds pixel-aligned reference rendering features at the same pose: coverage, depth, Gaussian contribution statistics, and their gradients and interactions. The fitted classifiers predict change log odds ℓ_{base} and ℓ_{full}. We suppress responses when adding rendering features lowers the predicted change odds, using

$$\mathcal{T}(\ell_{\text{base}}, \ell_{\text{full}}) = \min(\ell_{\text{full}} - \ell_{\text{base}}, 0). \tag{10}$$

Fitting details and feature groups are provided in Appendix A.2 (Eq. 18).

Log-likelihood-ratio mapping. Under Bayesian hypothesis testing, the corrected response R_t is mapped to evidence via the log-likelihood ratio (LLR) between change ($\mathcal{H}_1$) and no-change ($\mathcal{H}_0$). To estimate this mapping, we pool corrected responses across the editing subset and use isotonic regression (Zadrozny & Elkan, 2002) to partition responses into monotonic blocks b. The unchanged distribution $p(b \mid \mathcal{H}_0)$ is an equal mixture of distributions estimated from regions outside edits ($\mathcal{H}_0^{\text{edit}}$) and unedited reference views ($\mathcal{H}_0^{\text{ref}}$). This gives the block LLR:

$$\ell_b = \log \frac{\widehat{p}(b \mid \mathcal{H}_1)}{\widehat{p}(b \mid \mathcal{H}_0)} = \log \frac{\widehat{p}(b \mid \mathcal{H}_1)}{\frac{1}{2}\widehat{p}(b \mid \mathcal{H}_0^{\text{edit}}) + \frac{1}{2}\widehat{p}(b \mid \mathcal{H}_0^{\text{ref}})}. \tag{11}$$

Linearly interpolating ℓ_b across block-mean responses yields the continuous monotonic mapping ℓ_{img} (details in Appendix A.2, Eq. 19).

Unchanged-response modelling. To fit the unchanged-response model (μ_g, τ_g^2, c) (Section 4.2), we estimate its response statistics over reference views t in the unchanged-only subset. We place a Beta prior on each Gaussian's latent mean response and treat bounded observations $r_{tg} \in [0, 1]$ as fractional counts. Taking the prior concentration to zero from above yields the mean μ_g and

uncertainty in the estimated unchanged-response mean τ_g^2. The reference graph is built from eight-nearest-neighbour candidates, then symmetrized and weighted by spatial and colour similarity η_{gj}. Co-observed neighbour pairs estimate the spatial co-response coefficient c:

$$\mu_g = \frac{\sum_t \mathbb{I}_{tg} r_{tg}}{\sum_t \mathbb{I}_{tg}}, \qquad \tau_g^2 = \frac{\mu_g(1-\mu_g)}{1+\sum_t \mathbb{I}_{tg}}, \qquad c = \frac{\sum_{t,(g,j)} \omega_{tgj}^{\mathrm{fit}} \widetilde{z}_{tg} \widetilde{z}_{tj}}{\sum_{t,(g,j)} \omega_{tgj}^{\mathrm{fit}}}, \tag{12}$$

where $\widetilde{z}_{tg}$ and $\widetilde{z}_{tj}$ denote responses of Gaussian g and neighbour j standardized across unchanged reference views, $\omega_{tgj}^{\mathrm{fit}} = \eta_{gj} \min(\mathbb{I}_{tg}, \mathbb{I}_{tj})$ weights co-observed neighbour pairs, and unobserved Gaussians ($\sum_t \mathbb{I}_{tg} = 0$) revert to scene-level priors (details in Appendix B, Eqs. 24–25).

Pixel prior fitting. To fit the pixel prior π (Eq. 1) robustly against synthetic edit artifacts (e.g., exposed backgrounds or spillover), we introduce Cauchy reliability weights $\rho_i \in (0, 1]$ to downweight views with abnormal responses in labelled unchanged regions. For each training image i, let Y_i denote its binary change-label mask and $\widehat{Y}_i^{\mathrm{pixel}}(\pi)$ the pixel prediction obtained with prior π. We determine the optimal prior $\pi^\star$ by maximizing their reliability-weighted foreground IoU over the editing subset:

$$\pi^\star \in \arg\max_{\pi \in (0,1)} \frac{\sum_i \rho_i \, \mathrm{IoU}\Big(\widehat{Y}_i^{\mathrm{pixel}}(\pi),\, Y_i\Big)}{\sum_i \rho_i}. \tag{13}$$

We solve this one-dimensional search via branch-and-bound (Appendix E, Eqs. 42–43).

Contour filter fitting. To learn the boundary penalty in Eq. 8, we first fit a linear classifier with score $s_{\mathrm{contour}}(u) = \mathbf{w}^\top \mathbf{f}_u + b$ over standardized features $\mathbf{f}_u$ that are unchanged when the query and reference images are swapped. These view-symmetric features combine SAM2 mask logits, predicted IoUs, and their products using $(\max, \min)$ across the two views (Appendix F). Across training image pairs i and candidate pixels u, the feature vectors are $\mathbf{f}_{iu}$ and edit-mask labels are $y_{iu} = Y_i(u) \in \{0, 1\}$. We fit $\mathbf{w}$ and b by minimizing regularized binary cross-entropy:

$$\min_{\mathbf{w},b} \sum_{i,u} w_{iu} \mathcal{L}_{\mathrm{CE}}\big(\sigma(\mathbf{w}^\top \mathbf{f}_{iu} + b),\, y_{iu}\big) + \frac{\lambda}{2}\|\mathbf{w}\|_2^2, \tag{14}$$

where λ controls L_2 regularization. Excluding ignored pixels, we construct w_{iu} by balancing classes within each image, multiplying by its reliability ρ_i, and normalizing each class to total weight 0.5 within the fitting split. Platt scaling (Platt, 2000) fits a slope a_{cal} and intercept b_{cal} on held-out objects using the same weighting procedure, giving contour log odds $\ell_{\mathrm{contour}}(u) = a_{\mathrm{cal}} s_{\mathrm{contour}}(u) + b_{\mathrm{cal}}$ (Appendix F).

5 Experiments

Protocol. We use real-world PASLCD (Galappaththige et al., 2025) as our primary benchmark for complete scene change detection (20 scene-instances, $4\times$ downsampling following convention), and four synthetic CCS3D scenes (Zhou et al., 2026) to assess performance under different scene and rendering conditions. Published CCS3D comparisons use post-state masks. We evaluate O-SCD and our method using the union of pre- and post-state masks to include both vacated and newly occupied regions (Table 2). The published results are shown for context; direct comparisons use our evaluations of O-SCD under the same union-mask protocol.

Tables 1–2 compare online and offline methods, using official-code evaluations for O-SCD and published results for other baselines. We report foreground mIoU and F1 for **Ours (pixel)** ($\widehat{Y}_t^{\mathrm{pixel}}$) and **Ours (full)** ($\widehat{Y}_t^{\mathrm{full}}$), using a score threshold of 0.5 (Figure 1). Our models are fitted offline from reference data without test-time tuning.

Complete evaluation. Our method achieves state-of-the-art accuracy on PASLCD with 67.08% mIoU and 78.37% F1 (Table 1), exceeding O-SCD(O) by 17.84 mIoU points and offline GS-Diff (Galappaththige et al., 2026a) by 2.68 points without a post-change reconstruction. Ours (pixel) also exceeds both O-SCD variants. On CCS3D, our method exceeds O-SCD(O) by 23.83 mIoU points under the same union-mask evaluation (Table 2). Figure 3 shows more complete object masks in *Porch*, fewer boundary false positives in *Pots*, and fewer missed changes in *Playground* than O-SCD(R). Appendix G provides per-instance results.

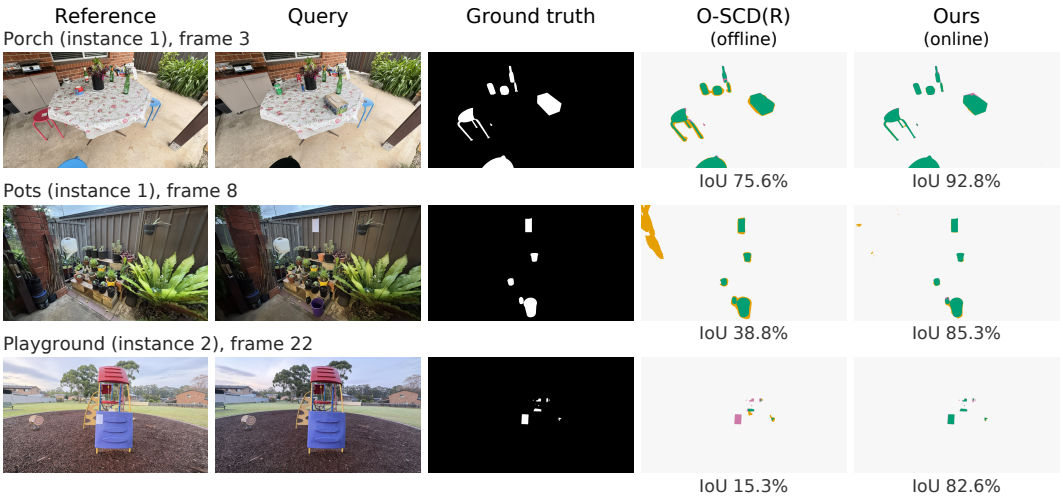

Figure 3: **Visual results on representative scenes.** Columns show reference renderings, queries, ground truth, O-SCD(R) (offline), and our complete method (online). ■ correct detections; ■ false positives; ■ missed changes. Numbers give per-frame foreground IoU. Additional visual results, including remaining errors, are shown in Appendix H.

Table 1: PASLCD comparison over 20 scene-instances (%). O-SCD is locally re-evaluated; other baselines are paper-reported.

Method	Venue	Mode	mIoU ↑	F1 ↑
Paper-reported				
SCaR-3D	3DV 2026	offline	19.10	28.90
3DGS-CD	RA-L 2025	offline	20.90	33.90
MV3DCD	CVPR 2025	offline	46.10	61.20
GS-Diff	arXiv 2026	offline	64.40	75.80
Our evaluation				
O-SCD(O)	CVPR 2026	online	49.24	64.60
O-SCD(R)	CVPR 2026	offline	55.99	70.27
Ours (pixel)	—	online	60.13	73.52
Ours (full)	—	online	**67.08**	**78.37**

Table 2: CCS3D comparison over four scenes (%). Paper-reported baselines use post-state masks; our local evaluation uses union masks.

Method	Venue	Mode	mIoU ↑	F1 ↑
Paper-reported				
MV3DCD	CVPR 2025	offline	30.40	44.10
CL-Splats	ICCV 2025	offline	39.90	53.80
3DGS-CD	RA-L 2025	offline	41.60	51.20
SCaR-3D	3DV 2026	offline	65.00	72.40
Our evaluation				
O-SCD(O)	CVPR 2026	online	49.94	64.61
O-SCD(R)	CVPR 2026	offline	43.59	57.82
Ours (pixel)	—	online	66.67	79.31
Ours (full)	—	online	**73.77**	**84.43**

Cross-instance reference reuse. We reuse each reference reconstruction and all its fitted parameters to process the other instance of the same PASLCD scene, without refitting. Across 20 directed pairs and 500 frames, pixel and full-output mIoU remain close to the original pairing: 59.94% versus 60.13%, and 67.12% versus 67.08%, respectively (Appendix, Table 6). These results support reference reuse between closely related instances of the same scene. Appendix G.1 provides the protocol and per-pair results.

Independent component ablations. Removing any evaluated component reduces mean mIoU (Table 3), with the pixel prior π having the largest effect in the pixel pipeline: its removal causes a 16.47-point drop. This highlights the importance of selecting the decision operating point from reference data. Appendix G.2 reports the fitted priors and performance by change area. Gaussian-aggregated support contributes 1.96 points, supporting the use of spatially shared evidence. Response suppression and likelihood-ratio mapping provide additional gains. Historical support connects each prediction to a shared 3D field that accumulates change evidence from earlier views. Removing it reduces mean pixel mIoU by 2.68 points. To evaluate how this shared evidence should be accumulated, we replace our Bayes-factor updates with a temporal average of each Gaussian's observed responses, reducing pixel mIoU by 4.32 points (Appendix, Table 10). Contour filtering suppresses excess responses, while object completion recovers missed object regions. Removing either reduces mean mIoU by more than four points, whereas contour image weights have only a marginal effect.

Table 3: Independent component ablations on PASLCD (%). Each component is removed separately from the corresponding complete pipeline with the decision threshold fixed at 0.5.

Pixel pipeline			*Output refinement*		
Configuration	mIoU ↑	F1 ↑	Configuration	mIoU ↑	F1 ↑
Full pixel pipeline	**60.13**	**73.52**	**Full method**	**67.08**	**78.37**
w/o response suppression	58.47	71.99	w/o contour filtering	62.83	75.45
w/o likelihood-ratio mapping	59.06	72.88	w/o object completion	62.12	74.89
w/o Gaussian-aggregated support	58.17	71.95	w/o contour image weights	67.02	78.30
w/o historical support	57.45	71.26			
w/o pixel prior	43.66	57.95			

Table 4: Runtime on PASLCD (RTX 5090; 20 instances). Online rows average frames 4–25, with a 32.67 ms shared frontend. O-SCD(R) includes offline refinement amortized over full sequences; its FPS is an equivalent rate, not online throughput.

Method	Mode	Post-cue (ms) ↓	End-to-end (ms) ↓	FPS ↑
O-SCD(R)	offline	168.57	201.24	5.0
O-SCD(O)	online	35.78	68.45	14.6
Ours (pixel)	online	**7.11**	**39.78**	**25.1**
Ours (full)	online	27.57	60.24	16.6

Relighting strength. Removing the relighting perturbation during data generation reduces full-output mIoU by 0.71 points, while doubling its strength gives similar aggregate results (Appendix, Table 9).

Runtime. We measure runtime on an NVIDIA RTX 5090 across all 20 PASLCD instances using synchronized GPU wall-clock timing (Table 4). Including the separately measured shared frontend, our complete method reaches 16.6 FPS, compared with 14.6 FPS for O-SCD(O), while improving mIoU by 17.84 points. Closed-form local Bayes factors update accumulated evidence without online backpropagation or past-image replay. Without refinement, our pixel pipeline reaches 25.1 FPS at 60.13% mIoU, still exceeding the offline refined O-SCD(R) result of 55.99%. Given an existing reference reconstruction, offline preparation averages 215.82 s (∼3.6 minutes) per scene, including automatic editing and model fitting. Appendix G.5 provides the timing breakdown.

6 Limitations and Conclusion

Our data generation prioritizes short offline preparation, using direct object edits and simple relighting. These edits may not cover real changes or unusual lighting conditions. More advanced 3DGS segmentation, editing, and relighting methods could improve object masks and broaden the generated changes, but may require additional optimization. Fixed reference geometry also limits how accurately newly introduced objects and occlusion boundaries can be represented in the accumulated field. The response uncertainty and spatial covariance models are working approximations. Although the local Bayes factor is evaluated analytically under the residual model, the fused scores combine dependent evidence and are not guaranteed to be calibrated probabilities of real-world changes. We presented an online scene change detection method that learns from automatic reference 3DGS edits and combines incoming evidence through generalised Bayesian updates. Offline learning provides scene-specific evidence models and statistics for closed-form local Bayes factors, enabling historical accumulation without online backpropagation or past-image replay. Experiments on PASLCD and CCS3D demonstrate improved detection accuracy over O-SCD; on PASLCD, the complete method achieves 67.08% mIoU at 16.6 FPS. Future work will investigate whether richer data generation improves detection accuracy enough to justify additional offline preparation time.

AI Use Statement

In this work, generative AI tools were used to assist with initial methodology discussions and exploratory brainstorming, algorithm and code implementation, preliminary manuscript drafting, and

textual refinement for grammar, phrasing, and readability. Generative AI tools were not used for formulating theoretical claims, deriving mathematical formulations, collecting benchmark observations, or interpreting empirical results. All AI-assisted contributions were rigorously reviewed, tested, and vetted by the authors: all algorithmic implementations and evaluation pipelines were independently executed and verified, all mathematical derivations were checked manually, and all experimental results and claims were thoroughly audited. The authors assume full responsibility for the entirety of this paper and its claims.

Ethics Statement

This work uses existing scene-change benchmarks and automatically generated training examples derived from reference scenes. Incorrect predictions can lead to missed changes or unnecessary follow-up inspections. Deployment in private or occupied environments should respect privacy and data-use permissions, with human review when errors could have consequential effects.

Reproducibility Statement

We submit our implementation as supplementary material. Appendices A–F detail reference camera assignment, automatic training-data generation, feature construction, model fitting, and causal online inference, including the mathematical derivations and numerical settings. The experimental protocol and evaluation metrics are specified in Section 5. All scene-specific parameters are fitted offline using reference data and remain fixed during target evaluation, without test-time tuning.

References

Jan Ackermann, Jonas Kulhanek, Shengqu Cai, Haofei Xu, Marc Pollefeys, Gordon Wetzstein, Leonidas J. Guibas, and Songyou Peng. CL-Splats: Continual learning of gaussian splatting with local optimization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7808–7817, October 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Ackermann_CL-Splats_Continual_Learning_of_Gaussian_Splatting_with_Local_Optimization_ICCV_2025_paper.html.

Matias Altamirano, Francois-Xavier Briol, and Jeremias Knoblauch. Robust and scalable Bayesian online changepoint detection. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 642–663. PMLR, 2023. URL https://proceedings.mlr.press/v202/altamirano23a.html.

Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B (Methodological)*, 57(1):289–300, 1995. doi: 10.1111/j.2517-6161.1995.tb02031.x.

P. G. Bissiri, C. C. Holmes, and S. G. Walker. A general framework for updating belief distributions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 78(5):1103–1130, 2016. doi: 10.1111/rssb.12158. URL https://arxiv.org/abs/1306.6430.

Yiwen Chen, Zilong Chen, Chi Zhang, Feng Wang, Xiaofeng Yang, Yikai Wang, Zhongang Cai, Lei Yang, Huaping Liu, and Guosheng Lin. GaussianEditor: Swift and controllable 3D editing with gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21476–21485, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Chen_GaussianEditor_Swift_and_Controllable_3D_Editing_with_Gaussian_Splatting_CVPR_2024_paper.html.

Chamuditha Jayanga Galappaththige, Jason Lai, Lloyd Windrim, Donald Dansereau, Niko Sünderhauf, and Dimity Miller. Multi-view pose-agnostic change localization with zero labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11600–11610, June 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Galappaththige_Multi-View_Pose-Agnostic_Change_Localization_with_Zero_Labels_CVPR_2025_paper.html.

Chamuditha Jayanga Galappaththige, Jason Lai, Timothy Patten, Donald Dansereau, Niko Sünderhauf, and Dimity Miller. From pixels to primitives: Scene change detection in 3D gaussian splatting. *arXiv preprint arXiv:2605.07203*, 2026a. URL https://arxiv.org/abs/2605.07203.

Chamuditha Jayanga Galappaththige, Jason Lai, Lloyd Windrim, Donald Dansereau, Niko Sünderhauf, and Dimity Miller. Changes in real time: Online scene change detection with multi-view fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 32246–32256, June 2026b. URL https://openaccess.thecvf.com/content/CVPR2026/html/Galappaththige_Changes_in_Real_Time_Online_Scene_Change_Detection_with_Multi-View_CVPR_2026_paper.html.

Jian Gao, Chun Gu, Youtian Lin, Zhihao Li, Hao Zhu, Xun Cao, Li Zhang, and Yao Yao. Relightable 3D gaussians: Realistic point cloud relighting with BRDF decomposition and ray tracing. In *European Conference on Computer Vision (ECCV)*, 2024. URL https://www.ecva.net/papers/eccv_2024/papers_ECCV/html/6121_ECCV_2024_paper.php.

Andrew Gelman, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, and Donald B. Rubin. *Bayesian Data Analysis*. Chapman and Hall/CRC, 3 edition, 2013. ISBN 9781439840955. URL https://www.routledge.com/Bayesian-Data-Analysis/Gelman-Carlin-Stern-Dunson-Vehtari-Rubin/p/book/9781439840955.

Robert E. Kass and Adrian E. Raftery. Bayes factors. *Journal of the American Statistical Association*, 90(430):773–795, 1995. URL https://sites.stat.washington.edu/courses/stat527/s14/readings/KassRaftery_jasa1995.pdf.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):139:1–139:14, July 2023. doi: 10.1145/3592433. URL https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/.

Jae-Woo Kim and Ue-Hwan Kim. Towards generalizable scene change detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24463–24473, June 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Kim_Towards_Generalizable_Scene_Change_Detection_CVPR_2025_paper.html.

Dong In Lee, Hyeongcheol Park, Jiyoung Seo, Eunbyung Park, Hyunje Park, Ha Dam Baek, Sangheon Shin, Sangmin Kim, and Sangpil Kim. EditSplat: Multi-view fusion and attention-guided optimization for view-consistent 3D scene editing with 3D gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11135–11145, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Lee_EditSplat_Multi-View_Fusion_and_Attention-Guided_Optimization_for_View-Consistent_3D_Scene_CVPR_2025_paper.html.

Zhihao Liang, Qi Zhang, Ying Feng, Ying Shan, and Kui Jia. GS-IR: 3D gaussian splatting for inverse rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21644–21653, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Liang_GS-IR_3D_Gaussian_Splatting_for_Inverse_Rendering_CVPR_2024_paper.html.

Ziqi Lu, Jianbo Ye, and John J. Leonard. 3DGS-CD: 3D gaussian splatting-based change detection for physical object rearrangement. *IEEE Robotics and Automation Letters*, 10(3):2662–2669, 2025. doi: 10.1109/LRA.2025.3533457. URL https://doi.org/10.1109/LRA.2025.3533457.

John C. Platt. Probabilities for SV machines. In Alexander J. Smola, Peter Bartlett, Bernhard Schölkopf, and Dale Schuurmans (eds.), *Advances in Large-Margin Classifiers*, pp. 61–74. The MIT Press, 2000. doi: 10.7551/mitpress/1113.003.0008. URL https://doi.org/10.7551/mitpress/1113.003.0008.

Guilherme Potje, Felipe Cadar, André Araujo, Renato Martins, and Erickson R. Nascimento. XFeat: Accelerated features for lightweight image matching. In *Proceedings of the IEEE/CVF*

Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2682–2691, June 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Potje_XFeat_Accelerated_Features_for_Lightweight_Image_Matching_CVPR_2024_paper.html.

Jingxing Qian, Veronica Chatrath, Jun Yang, James Servos, Angela P. Schoellig, and Steven L. Waslander. POCD: Probabilistic object-level change detection and volumetric mapping in semi-static scenes. In *Proceedings of Robotics: Science and Systems*, New York City, NY, USA, June 2022. doi: 10.15607/RSS.2022.XVIII.013. URL https://www.roboticsproceedings.org/rss18/p013.html.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. SAM 2: Segment anything in images and videos. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/45c1f6a8cbf2da59ebf2c802b4f742cd-Abstract-Conference.html.

Ragav Sachdeva and Andrew Zisserman. The change you want to see. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 3993–4002, January 2023a. URL https://openaccess.thecvf.com/content/WACV2023/html/Sachdeva_The_Change_You_Want_To_See_WACV_2023_paper.html.

Ragav Sachdeva and Andrew Zisserman. The change you want to see (now in 3D). In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pp. 2060–2069, October 2023b. URL https://openaccess.thecvf.com/content/ICCV2023W/OpenSUN3D/html/Sachdeva_The_Change_You_Want_to_See_Now_in_3D_ICCVW_2023_paper.html.

Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004. doi: 10.1109/TIP.2003.819861. URL https://doi.org/10.1109/TIP.2003.819861.

Yuqun Wu, Chih-hao Lin, Henry Che, Aditi Tiwari, Chuhang Zou, Shenlong Wang, and Derek Hoiem. SceneDiff: A benchmark and method for multiview object change detection. In *European Conference on Computer Vision (ECCV)*, 2026. URL https://arxiv.org/abs/2512.16908.

Sizhe Yang, Wenye Yu, Jia Zeng, Jun Lv, Kerui Ren, Cewu Lu, Dahua Lin, and Jiangmiao Pang. Novel demonstration generation with gaussian splatting enables robust one-shot manipulation. In *Proceedings of Robotics: Science and Systems (RSS)*, 2025. doi: 10.15607/RSS.2025.XXI.146. URL https://www.roboticsproceedings.org/rss21/p146.html.

Mingqiao Ye, Martin Danelljan, Fisher Yu, and Lei Ke. Gaussian grouping: Segment and edit anything in 3D scenes. In *European Conference on Computer Vision (ECCV)*, 2024. doi: 10.1007/978-3-031-73397-0_10. URL https://www.ecva.net/papers/eccv_2024/papers_ECCV/html/4195_ECCV_2024_paper.php.

Bianca Zadrozny and Charles Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2002. doi: 10.1145/775047.775151. URL https://www.cs.columbia.edu/~djhsu/ML/handouts/zadrozny2002kdd.pdf.

Farhad G. Zanjani, Davide Abati, Auke Wiggers, Dimitris Kalatzis, Jens Petersen, Hong Cai, and Amirhossein Habibian. Gaussian splatting is an effective data generator for 3D object detection. *arXiv preprint arXiv:2504.16740*, 2025. URL https://arxiv.org/abs/2504.16740.

Zirui Zhou, Junfeng Ni, Shujie Zhang, Yixin Chen, and Siyuan Huang. 3D scene change modeling with consistent multi-view aggregation. In *International Conference on 3D Vision (3DV)*, 2026. URL https://openreview.net/forum?id=sV7lwfLo6H.

Runsong Zhu, Shi Qiu, Zhengzhe Liu, Ka-Hei Hui, Qianyi Wu, Pheng-Ann Heng, and Chi-Wing Fu. Rethinking end-to-end 2D to 3D scene segmentation in gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3656–3665, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Zhu_Rethinking_End-to-End_2D_to_3D_Scene_Segmentation_in_Gaussian_Splatting_CVPR_2025_paper.html.

A Image Evidence Processing and Gaussian Aggregation

A.1 Automatic reference editing and data generation

The inputs are the fixed reference 3DGS and its registered images. Object selection, edit parameters, labels, and filtering are determined automatically from these inputs. We use SAM2.1 Hiera-Tiny automatic masks, images with a maximum side length of 1008 pixels, and random seed 0. The following settings are shared by all 20 scene instances.

Reference camera assignment. We implement the approximately 5:1 camera split described in Section 4.3. The unchanged-only subset contains $\lceil N/6 \rceil$ cameras for unchanged-response estimation, where N is the total number of reference cameras. We select these cameras to represent the scene's visible surfaces with a limited number of observations. For camera t, let $\mathcal{V}_t$ denote its observable reference Gaussians. We measure the distance between cameras t and l by the Jaccard distance $1 - |\mathcal{V}_t \cap \mathcal{V}_l|/|\mathcal{V}_t \cup \mathcal{V}_l|$, so cameras observing similar surfaces are close. Greedy facility-location selection adds cameras one at a time to minimise the total distance from all reference cameras to their nearest selected camera. We then apply medoid swaps, replacing a selected camera with an unselected one whenever this reduces the total distance. Both steps require the editing subset to retain at least one available view of every synthetic object/change event. The editing subset provides labelled examples generated by automatic 3DGS edits for learning response suppression, the likelihood-ratio mapping, the effective pixel prior, and the contour filter. Examples are included only where the corresponding edit is available. With the image-evidence pipeline fixed, the unchanged-only subset provides unchanged reference images for estimating the Gaussian unchanged-response model and spatial co-response. The two subsets contain different cameras but may observe overlapping surfaces.

Object proposals and cross-view association. We first generate candidate regions in each reference image using SAM2: automatically placed point prompts produce masks, followed by filtering of low-quality and duplicate results. To determine whether masks in different views describe the same object, we associate each mask with the reference Gaussians that mainly contribute to it.

For reference view t, let a_{tug} be the rendering contribution defined in Section 3. At each pixel, we retain only the two largest contributions and set the others to zero; denote these retained weights by $\widetilde{a}_{tug}$. For a mask M, Gaussian g's contributions inside and outside it are

$$A_{tg}^{\text{in}} = \sum_{u \in M} \widetilde{a}_{tug}, \qquad A_{tg}^{\text{out}} = \sum_{u \in \Omega_t \setminus M} \widetilde{a}_{tug}, \tag{15}$$

where Ω_t is the reference image domain. We assign g to the mask when it contributes more to the inside than the outside, allowing ties:

$$\mathcal{S}_t(M) = \left\{ g : A_{tg}^{\text{in}} > 0, \quad A_{tg}^{\text{in}} \geq A_{tg}^{\text{out}} \right\}. \tag{16}$$

This *support set* therefore contains Gaussians with at least half of their retained contribution inside the mask. Masks with empty support sets are discarded.

To exploit view continuity in sequentially captured reference images, we match each view with the next K views in the reference sequence. This reduces pairwise comparisons while allowing matches to be linked through intermediate views into longer tracks. We use $K = 2$. We match masks across views by comparing the Gaussians assigned to them. For each candidate pair, we use a hypergeometric test to determine whether the two masks share more Gaussians than expected by chance. The test accounts for both support-set sizes and uses all Gaussians visible in either view, $\mathcal{V}_t \cup \mathcal{V}_l$, as the population. We apply the Benjamini–Hochberg procedure (Benjamini & Hochberg, 1995) at level 0.01 to filter chance overlaps across the candidate pairs. Among the surviving pairs, we rank matches by Jaccard overlap (intersection size divided by union size), breaking ties by the

number of shared Gaussians. We retain a pair only if each mask is the other's highest-ranked match in the corresponding view. Finally, we link these matches into cross-view tracks, where each track is a collection of masks associated with the same candidate object across reference views, with at most one mask per view.

Membership and edit-view validation. We estimate how strongly each editable Gaussian belongs to the candidate object using the masks in its representative track. We use all available views of this track, adding views from strongly linked tracks if fewer than two are available. For each editable Gaussian g, we sum its rendering contributions inside these masks and divide by its total contribution across the selected views. This ratio defines its soft membership $m_g \in [0, 1]$: larger values indicate that more of its contribution falls inside the object masks. Gaussians unobserved in these views receive $m_g = 0$.

Rendering these membership values produces a 2D object-support map in each selected view. We convert the maps into binary masks using a shared threshold chosen to maximise mean IoU with the corresponding SAM2 masks. To limit search cost, we divide the quantile range $[0, 1]$ into 256 equal intervals for support values above 10^{-7}. The boundary quantiles serve as candidate thresholds; we select the one maximising mean IoU with the SAM2 masks, breaking ties in favour of larger thresholds.

To check consistency across views, we repeat this procedure while leaving out one view at a time: both membership and the threshold are fitted using the remaining views, then used to predict the object mask in the omitted view. We retain candidates with mean IoU at least 0.65 and minimum IoU at least 0.5 against the omitted SAM2 masks. For accepted candidates, we refit membership and the threshold using all selected views. **All validation uses automatically generated reference masks and requires no manually annotated change masks. Gaussian recolouring.** We compute a channelwise median RGB colour in the first selected view over object Gaussians with membership weight at least 0.5, falling back to all selected object Gaussians if none meet this criterion. The target maximizes Euclidean RGB distance from this median over the palette $(0.90, 0.15, 0.18)$, $(0.12, 0.72, 0.24)$, $(0.12, 0.35, 0.92)$, $(0.92, 0.72, 0.10)$, $(0.82, 0.16, 0.82)$, and $(0.08, 0.76, 0.82)$. For each selected Gaussian, we interpolate its view-dependent RGB colour towards this shared target with coefficient 0.45 times its object membership weight. The target is fixed across the object's rendered views.

Gaussian removal and translation. Removal replaces Gaussian g's opacity parameter o_g by $(1 - m_g)o_g$ for every selected Gaussian. Translation moves Gaussians with $m_g \geq 0.5$ by a common displacement, leaving their other attributes fixed. Its magnitude is relative to object extent. The search uses core Gaussians supported by at least half of the anchor-track masks, intersected with the moved set; it falls back to the moved set if that intersection is empty. The centre is their coordinatewise median, and directional extents are measured between the 1st and 99th percentiles, using extrema when fewer than eight points are available.

We select up to four evenly spaced seed views. In each seed view, eight equally spaced directions, with a seeded random angular offset, lie in the plane perpendicular to the centre's viewing direction. The initial translation magnitude is 1.5 times the object's 3D extent along the candidate direction. Up to six scaling steps increase it to obtain a projected centre displacement of 1.5 projected object extents in the seed view, corresponding to an extent-relative clearance parameter of 0.5. A view passes the geometric check when both projected centres and at least two core points before and after translation are valid, some moved core points remain in frame, the old–new projected interval overlap is at most 0.01, and the centre displacement is at least 1.5×0.995 original projected extents. Interval overlap is divided by the shorter interval length.

Within each seed, valid directions are ranked by retained in-frame fraction, image-border margin, smaller overlap, and shorter translation, in that order. Each seed's selected displacement is then checked across all supporting views. The final choice maximizes the number of valid views, followed by the same ranking criteria using the worst valid view. At least one valid view is required. A final rendering check requires a nonempty moved mask and intersection with the original mask below 1% of the smaller mask area. Only passing views produce translation examples; an object without a valid translation may still provide recolouring and removal examples.

Changed, unchanged, and ignored regions. Let M denote the candidate changed region: the reference SAM2 object mask for recolouring and removal, or its union with the thresholded moved Gaussian-support map for translation. Let D be the channel-mean absolute RGB difference between

edited and original renders. Sorting D in descending order defines an effect footprint E containing at least 99% of its summed mass, including ties at the cutoff. This footprint identifies edit effects extending beyond the object mask.

Removal also checks the newly exposed background. For each Gaussian, background reliability is the fraction of its observed contribution outside the object's masks across the selected reference views. Rendering these reliabilities after removal and dividing by remaining coverage gives a background-provenance map. Pixels with remaining coverage below 0.05 or provenance below 0.5 are invalid. For removal and translation, let U contain such pixels inside the original object mask; for recolouring, U is empty. The changed region is $M \setminus U$, and the ignored region is $U \cup (E \setminus M)$. All other pixels are labelled unchanged. Changed and ignored regions are disjoint.

We also store soft edited-Gaussian footprint maps. For recolouring and removal this is the rendered membership; for translation it is $1 - (1 - f_{\text{old}})(1 - f_{\text{new}})$, where the two f maps render membership of the moved Gaussian set at its original and translated positions. The hard masks, ignore masks, and soft maps remain associated with the same object, camera, and edit.

Filtering individual edits. At the generated image resolution, an original-lighting edit is retained only if all conditions below hold. A five-pixel boundary neighbourhood is formed by dilating the change mask with an 11×11 elliptical kernel; ignored pixels outside this dilation are called distant ignored pixels.

Quantity	Acceptance condition
Changed pixel count	> 0
Ignored area / image area	≤ 0.03
Ignored area / changed area	≤ 20
Distant ignored area / changed area	≤ 1
RGB edit-effect mass outside M / total effect mass	≤ 0.20
Invalid background area / original object-mask area (removal)	≤ 0.50

The effect-mass check uses M before invalid interior pixels are excluded. Translation requires at least one retained view. The filters operate on reference-derived masks and render differences.

Reference-fitted lighting variations. We fit a colour transform to explain differences between unchanged reference images and their renders. Its affine component maps the view-dependent Gaussian colour $\mathbf{c}_{tg}$ to $\boldsymbol{\gamma} \odot \mathbf{c}_{tg} + \boldsymbol{\beta}$, with three channel gains $\boldsymbol{\gamma}$ and offsets $\boldsymbol{\beta}$; $\odot$ denotes componentwise multiplication. An optional directional component adds $(\mathbf{d}^\top \mathbf{n}_g)\mathbf{c}_{tg}$, where $\mathbf{d} \in \mathbb{R}^3$ is fitted and $\mathbf{n}_g$ is the Gaussian's smallest-axis direction, oriented towards the mean reference camera position. Transformed colours are restricted to $[0, 1]$ before rendering.

We select whether to include the directional component using six reference views, sampled at evenly spaced indices in the reference camera list. For each view, both models are fitted at a maximum side length of 320 pixels on even-parity checkerboard pixels with coverage above 0.05, excluding reliably tracked objects. Fewer than 64 fitting pixels triggers a fallback to all covered even-parity pixels. Four iteratively reweighted least-squares steps use per-pixel RGB residual norms r, scale $s = \max(\text{median}\, r, 10^{-4})/0.6745$, and weights $\min(1, 1.345s/r)$. A ridge term towards the identity transform has strength $\epsilon_{\text{fp32}} \max(\text{tr}(X^\top W X), 1)$, where X is the rendered colour-basis design matrix, W contains fitting weights, and $\epsilon_{\text{fp32}} = 2^{-23}$ is the machine epsilon for float32. For model selection, predictions are evaluated on odd-parity covered object pixels at the original resolution, falling back to non-object pixels when fewer than 16 object pixels are available. The comparison averages squared RGB errors capped at their per-view 90th percentile. We enable the directional component when its mean error across the six views does not exceed the affine model's. This is a fixed-size model-selection check; the selected views are not chosen to optimize spatial coverage. It enables the directional component for 7 of the 20 evaluated scene instances.

After model selection, coefficients are fitted separately for every reference camera represented in the generated event list (43–108 cameras per instance in our data). Thus, the six-view check does not limit the views used for coefficient fitting or event generation. We retain three representative coefficient vectors: the first is closest to the componentwise median, followed by farthest-point selection after scaling by median absolute deviation, with each scale bounded below by 10^{-4}. For each camera, a fixed seeded choice selects one of the two representatives farthest from its own fitted

coefficients. That transform is applied to both edited and unedited Gaussians, producing a relighting condition alongside the identity condition. Relighting preserves the parent change masks and soft targets; additional edit effects outside the changed regions are included in the ignore mask. All lighting fitting and selection use reference data before deployment.

Paired-condition and object-level filtering. Relighting is applied only to edits that pass the individual-edit filters. Its change masks and soft targets are inherited; the ignore mask also includes any new edit-effect footprint outside the changed regions. We measure this footprint from the relit edited–unedited rendering pair, so lighting itself does not define changed pixels. A relit sample requires nonempty changed regions and at most 20% of edit-effect mass outside the union of changed and ignored regions. We then retain an object–camera–edit tuple only when both identity and relighting records are present exactly once, and both have distant ignored area no larger than changed area. A failed condition removes the pair.

Among objects with surviving pairs, more than half of the original object–camera–edit records must have passed the individual-edit filters, with at least two passing edit types. This fraction is computed before relighting expansion. Finally, two candidates are linked as duplicates if their anchor Gaussian sets have intersection over the smaller set at least 0.9, or Jaccard overlap at least 0.6. We retain one candidate per connected duplicate group, ranked by passing fraction, passing-record count, and the number of Gaussians with membership at least 0.5, with the object identifier breaking any remaining tie. Only then are retained records assigned to the reference camera subsets described above.

Synthetic training observations. We transfer rendered interventions to the corresponding real reference image. For reference image I_t^{ref}, original rendering $\widehat{I}_t$, and edited rendering $\widehat{I}_t^{\mathrm{edit}}$, the synthetic observation is $I_t^{\mathrm{syn}} = I_t^{\mathrm{ref}} + \widehat{I}_t^{\mathrm{edit}} - \widehat{I}_t$, restricted to $[0, 1]$. This retains the reference image's image–render discrepancies while adding the rendered edit. An unchanged control uses the unedited rendering under the same lighting condition in place of $\widehat{I}_t^{\mathrm{edit}}$. The image-evidence pipeline compares each resulting observation with the original reference rendering. Applying the camera assignment above gives 11,384 edited training records across the 20 instances: 5,692 identity and 5,692 relighting records. These are object–view–edit observations, not distinct camera views.

A.2 Image evidence processing and aggregation

Query relocalisation. We follow O-SCD (Galappaththige et al., 2026b) to construct a database of registered reference keyframes. XFeat keypoints (Potje et al., 2024) are matched across reference images and triangulated with their known poses to obtain 3D points. For each query, reference keyframes are selected by descriptor-match count, and fundamental-matrix RANSAC rejects inconsistent matches. Query matches to reference keypoints with known 3D positions establish 2D–3D correspondences. PnP-RANSAC estimates the query pose, followed by reprojection-error refinement with camera intrinsics and reference points held fixed. The fixed reference 3DGS is then rendered at the estimated pose.

Input response and renderer-conditioned response suppression. The initial response uses the photometric and feature cues of O-SCD (Galappaththige et al., 2026b), rescaling their sum to $[0, 1]$:

$$R_t^0(u) = \tfrac{1}{2}\big[\widetilde{D}_t^{\mathrm{rgb}}(u) + \widetilde{D}_t^{\mathrm{sem}}(u)\big]. \tag{17}$$

The photometric cue is $D_t^{\mathrm{rgb}}(u) = (1 - \lambda_{\mathrm{rgb}})L_{1,t}(u) + \lambda_{\mathrm{rgb}} L_{\text{D-SSIM},t}(u)$, where $L_{1,t}$ is the mean absolute RGB difference, $L_{\text{D-SSIM},t} = 1 - \mathrm{SSIM}_t$, and $\lambda_{\mathrm{rgb}} = 0.2$. For encoder ϕ with K channels, the semantic discrepancy at feature-grid location ξ is $D_t^{\mathrm{sem}}(\xi) = K^{-1}\|\phi(I_t)_\xi - \phi(\widehat{I}_t)_\xi\|_1$. The SAM2 encoder is frozen, and both images are resized to 1024×1024 before feature extraction, following the O-SCD input construction. RGB values lie in $[0, 1]$. SSIM is channel-averaged over an 11×11 window (Wang et al., 2004), and $1/K$ averages the feature-channel absolute differences. The tildes in Eq. 17 denote min–max normalization for RGB, and min–max normalization followed by bilinear upsampling for semantic differences. The semantic cue is then rescaled with the RGB extrema and restricted to $[0, 1]$.

Renderer-conditioned response suppression uses two logistic models with standardized inputs. Their outputs $\ell_{\mathrm{base}}(x)$ and $\ell_{\mathrm{full}}(x, z)$ are predicted change log odds. The vector x contains the fused response, its 21×21 mean, 7×7 standard deviation, and gradient magnitude. The full model additionally uses renderer features z: coverage, gradients, ray-composition statistics, depth dispersion,

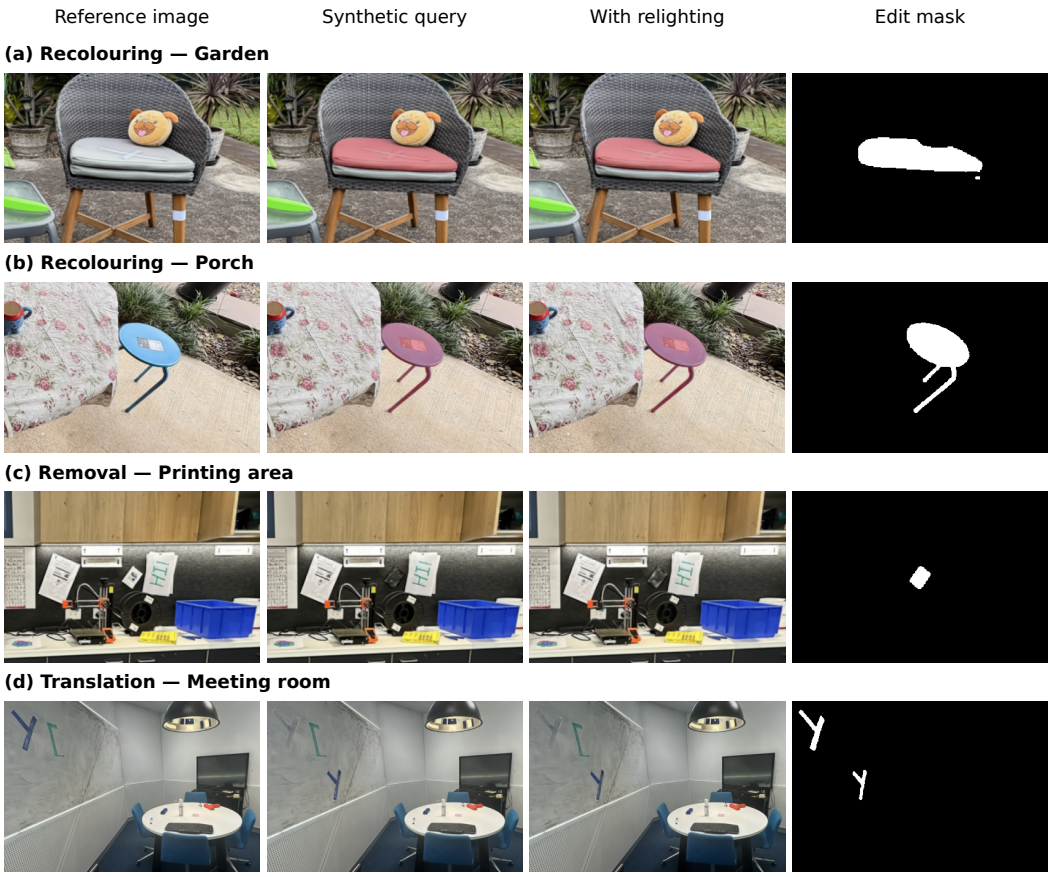

Figure 4: Selected examples from the editing subset, showing recolouring, removal, and translation in four scenes. Each row uses the same crop across the real reference image, synthetic query, relit synthetic query, and edit mask. White marks regions labelled changed; translation includes both original and displaced locations. Relighting preserves these labels. Condition-specific ignore masks are omitted.

and response–renderer interactions. The correction rule $\mathcal{T}(\ell_{\text{base}}, \ell_{\text{full}}) = \min\{\ell_{\text{full}} - \ell_{\text{base}}, 0\}$ acts on response log odds:

$$R_t = \sigma\big(\text{logit}\, R_t^0 + \mathcal{T}\big(\ell_{\text{base}}(x_t), \ell_{\text{full}}(x_t, z_t)\big)\big)\,. \tag{18}$$

Both regressions use L2 regularization. The correction is lower-bounded by the 1st percentile of training logit differences and upper-bounded by zero; it is disabled where rendered coverage is at most 10^{-3}. Fitting and deployment use this same corrected R_t.

Response-to-evidence mapping. The mapping ℓ_{img} is a monotone log likelihood ratio. Following Section 4.3, we separately normalize response histograms for changed regions of edited images $\mathcal{H}_1$, unchanged regions of those images $\mathcal{H}_0^{\text{edit}}$, and unedited reference controls $\mathcal{H}_0^{\text{ref}}$. Changed regions from all edit types contribute to $\mathcal{H}_1$. The unchanged likelihood is their fixed mixture $p_0 = \frac{1}{2}p_0^{\text{edit}} + \frac{1}{2}p_0^{\text{ref}}$. These coefficients specify the relative contributions of unchanged regions in edited images and unedited controls.

Isotonic regression (Zadrozny & Elkan, 2002) of the class-balanced changed fraction, weighted by pooled histogram mass, determines contiguous response blocks. For source $s \in \{1, \text{edit}, \text{ref}\}$ and block $\mathcal{B}_b$, let M_{sb} be its normalized histogram mass and N_s its effective training-record count. Smoothed block probabilities and the corresponding log likelihood ratio are

$$\widehat{p}_{sb} = \frac{N_s M_{sb} + \frac{1}{2}}{N_s + \frac{1}{2}K_{\mathcal{B}}}, \qquad \ell_b = \log \frac{\widehat{p}_{1b}}{\frac{1}{2}\widehat{p}_{\text{edit},b} + \frac{1}{2}\widehat{p}_{\text{ref},b}}, \tag{19}$$

where $K_{\mathcal{B}}$ is the number of current blocks. Effective counts respect training-record grouping: object–camera units for edited-image classes and camera units for unchanged controls. Violations of monotonicity after smoothing are merged and re-estimated. We interpolate the block values onto a 4,096-entry lookup table and keep the tails constant. For a response block b, define $\widehat{p}_{0b} = \frac{1}{2}\widehat{p}_{\mathrm{edit},b} + \frac{1}{2}\widehat{p}_{\mathrm{ref},b}$. With equal prior weights for changed and unchanged regions, Bayes' rule gives

$$P_{\mathrm{emp},b} = \frac{\widehat{p}_{1b}}{\widehat{p}_{1b} + \widehat{p}_{0b}} = \sigma(\ell_b). \tag{20}$$

Applying the interpolated mapping gives $P_t = \sigma(\ell_{\mathrm{img}}(R_t))$. These equal prior weights define the image-evidence scale. The scene-specific effective pixel prior π is fitted separately using the causal prediction and field update in Eqs. 1 and 6.

Gaussian response moments. Both aggregation and support rendering use the rasterizer contributions defined in Section 3. Zero denominators are guarded numerically. Pixels with insufficient reference coverage receive zero change probability but remain included in evaluation. For the second aggregation, define

$$r_{tg} = [\mathcal{A}_t(C_t)]_g, \qquad v_{tg} = [\mathcal{A}_t(C_t^2)]_g - r_{tg}^2. \tag{21}$$

Thus v_{tg} measures dispersion across the current image footprint. It is not a variance of the estimated mean across independent images. The first aggregation applies the same operator to the current image evidence P_t.

Observation indicators. Neighbourhood evidence and history accumulation use binary support indicators,

$$\mathbb{I}_{tg} = \mathbb{1}\{s_{tg} \geq 1\}, \tag{22}$$

where s_{tg} is the rendering contribution summed over the image (Section 4.2). The same rule is used in synthetic sequence replay for effective-prior estimation and in online inference. Every sufficiently supported Gaussian contributes one observation.

Unchanged-response fitting applies Eq. 22 to reference images in the unchanged-only subset. Thus each sufficiently supported observation receives equal fitting weight; observations below the support threshold are excluded. The resulting unchanged-response model and spatial co-response coefficient c are frozen before effective-prior estimation and online inference.

Reference neighbourhoods. We first retrieve eight nearest Gaussian centres, make the candidate edges undirected, and remove duplicates and self-edges. Geometry and colour affinities then filter and weight these connections. The final undirected degree can exceed eight. For centre displacement Δx_{gj}, spatial covariances Σ_g, Σ_j, and reference RGB colours $\mathbf{k}_g, \mathbf{k}_j$ from the zeroth-order spherical harmonic coefficients, the affinity is

$$\eta_{gj} = \exp\left[-\frac{1}{2}\Delta x_{gj}^{\mathsf{T}}(\Sigma_g + \Sigma_j)^{-1}\Delta x_{gj} - \frac{\|\mathbf{k}_g - \mathbf{k}_j\|^2}{2s_{\mathrm{colour}}^2}\right]. \tag{23}$$

The colour scale s_{colour} is the median nonzero colour distance among geometrically overlapping candidates, floored at 0.001. The covariance solve is regularized numerically, and the geometric squared distance is capped at 50. Edges with combined affinity no greater than 10^{-4} are removed. The spatial covariance matrices describe Gaussian shape. The resulting connections define each Gaussian's local reference neighbourhood.

B Unchanged-Scene Response Statistics

The frozen image-evidence pipeline is applied to the smaller camera group's unchanged images. For image t, aggregating current evidence C_t gives the Gaussian response $r_{tg} = [\mathcal{A}_t(C_t)]_g$, with observation indicator $\mathbb{I}_{tg}$ from Eq. 22 in Appendix A. All subsequent unchanged-response statistics refer to this normalized response coordinate, including current-view Gaussian-aggregated support.

Unchanged-response mean and uncertainty. All sums below use reference observations from the unchanged-only subset. The scene mean is $\mu_{\mathrm{scene}} = (\sum_{t,g}\mathbb{I}_{tg}r_{tg})/(\sum_{t,g}\mathbb{I}_{tg})$, clipped to $[10^{-3}, 1-10^{-3}]$. To derive the response moments, start with a $\mathrm{Beta}(\kappa\mu_{\mathrm{scene}}, \kappa(1-\mu_{\mathrm{scene}}))$ prior for response level θ_g, with $\kappa > 0$. Updating it with fractional response counts gives

$$\alpha_g = \kappa\mu_{\mathrm{scene}} + \sum_t \mathbb{I}_{tg}r_{tg}, \qquad \beta_g = \kappa(1-\mu_{\mathrm{scene}}) + \sum_t \mathbb{I}_{tg}(1-r_{tg}). \tag{24}$$

This follows from multiplying the prior density by $\prod_t [\theta_g^{r_{tg}}(1-\theta_g)^{1-r_{tg}}]^{\mathbb{I}_{tg}}$ using Beta conjugacy (Gelman et al., 2013); continuous responses make this a soft-count approximation. We use the limit $\kappa \to 0^+$ of the mean and variance, $\alpha_g/(\alpha_g+\beta_g)$ and $\alpha_g\beta_g/[(\alpha_g+\beta_g)^2(\alpha_g+\beta_g+1)]$:

$$\mu_g = \begin{cases} \dfrac{\sum_t \mathbb{I}_{tg} r_{tg}}{\sum_t \mathbb{I}_{tg}}, & \sum_t \mathbb{I}_{tg} > 0, \\ \mu_{\text{scene}}, & \sum_t \mathbb{I}_{tg} = 0, \end{cases} \qquad \tau_g^2 = \frac{\mu_g(1-\mu_g)}{1+\sum_t \mathbb{I}_{tg}}. \tag{25}$$

Thus observed Gaussians use their own mean without shrinkage, while unobserved Gaussians retain the limiting scene-based moments. The implementation evaluates these moments directly, including zero limiting variance when all observed responses are zero or one. The variance represents uncertainty in this soft-count construction, not the empirical predictive variance of a new image. At inference, we use the approximate standardized coordinate

$$z_{tg} = \frac{r_{tg} - \mu_g}{\sqrt{\tau_g^2 + v_{tg}}}. \tag{26}$$

The denominator combines uncertainty in the fitted unchanged-response level with dispersion within the current image footprint to define a working standardization scale.

Spatial co-response in unchanged scenes. To estimate spatial correlation, first standardize reference responses by each Gaussian's empirical weighted mean and across-view standard deviation, giving $\widetilde{z}_{tg}$. These empirical statistics are used for estimating c, and are distinct from Eq. 26. For neighbourhood affinity η_{gj} and edge weight $\omega^{\text{fit}}_{tgj} = \eta_{gj}\min(\mathbb{I}_{tg}, \mathbb{I}_{tj})$, summing over the unchanged-only subset gives

$$c = \frac{\sum_{t,(g,j)} \omega^{\text{fit}}_{tgj} \widetilde{z}_{tg}\widetilde{z}_{tj}}{\sum_{t,(g,j)} \omega^{\text{fit}}_{tgj}}. \tag{27}$$

C Conditional Neighbourhood Bayes Factor

Weighted neighbourhood statistics. For neighbours j of Gaussian g, let $\omega_{tgj} = \eta_{gj}\mathbb{I}_{tj}$ and $W_{tg} = \sum_j \omega_{tgj}$. For $W_{tg} > 0$,

$$\bar{z}_{tg} = \frac{\sum_j \omega_{tgj} z_{tj}}{W_{tg}}, \qquad d_{tg} = \frac{\sum_j \omega_{tgj}(z_{tj} - \bar{z}_{tg})^2}{W_{tg}}, \qquad n_{tg} = \frac{W_{tg}^2}{\sum_j \omega_{tgj}^2}. \tag{28}$$

Here $\bar{z}_{tg}$ is the weighted mean, d_{tg} the weighted dispersion, and n_{tg} the effective neighbour count. The working variance V_{tg} of that mean accounts for dispersion and correlation; ν_{tg} is the conditional residual variance used by the Bayes factor:

$$V_{tg} = \frac{\max(1, d_{tg})}{\max(1, n_{tg})}\left[1 + c(n_{tg}-1)_+\right], \qquad \nu_{tg} = \max(V_{tg} - c^2, \epsilon_\nu). \tag{29}$$

Here $(x)_+ = \max(x, 0)$, and we use variance floor $\epsilon_\nu = 10^{-4}$.

Why subtract the anchor response? Suppressing the frame and anchor indices, write $\bar{z} = \bar{z}_{tg}$, $z_g = z_{tg}$, $V = V_{tg}$, and $\nu = \nu_{tg}$. We use a zero-mean Gaussian working model under unchanged conditions. Setting $\text{Var}(z_g \mid \mathcal{H}_0) = 1$ and approximating $\text{Cov}(\bar{z}, z_g \mid \mathcal{H}_0)$ by c gives the conditional mean

$$\mathbb{E}[\bar{z} \mid z_g, \mathcal{H}_0] = \frac{\text{Cov}(\bar{z}, z_g \mid \mathcal{H}_0)}{\text{Var}(z_g \mid \mathcal{H}_0)} z_g \approx c z_g. \tag{30}$$

The corresponding conditional variance is $V - c^2$. Unit anchor variance is a fixed modelling scale: Eq. 26 does not guarantee that the empirical variance of z_g equals one. Using c transfers the spatial co-response estimated in Eq. 27 to this working model as a covariance approximation. The residual variance remains neighbourhood-dependent through Eq. 29. A shared increase δ in the responses of the anchor and neighbours leaves residual mean $(1-c)\delta$. This motivates $e = \bar{z} - c z_g$ and loading $a = 1 - c$.

Within this residual model, $\mathcal{H}_0$ denotes zero residual mean and $\mathcal{H}_1$ a shared increase in response, with half-normal prior scale τ:

$$\mathcal{H}_0 :\ e \sim \mathcal{N}(0, \nu), \qquad \mathcal{H}_1 :\ e \mid \delta \sim \mathcal{N}(a\delta, \nu), \qquad \delta \sim \text{HalfNormal}(\tau), \quad \tau = 1. \tag{31}$$

The variance floor in Eq. 29 also regularizes cases where the plug-in covariance approximation is not positive definite.

Integrating over the response increase. We evaluate the marginal likelihood of the residual e with a and $\nu > 0$ held fixed. For $\delta \geq 0$, the half-normal density $p_+(\delta)$ is

$$p_+(\delta) = 2\mathcal{N}(\delta; 0, \tau^2) = \frac{2}{\sqrt{2\pi}\tau} \exp\left(-\frac{\delta^2}{2\tau^2}\right).$$

Here $\tau > 0$ is the enhancement prior scale, fixed to 1 in our method; it is distinct from the fitted uncertainty τ_g^2. Since the enhancement δ is unknown, the marginal likelihood $p(e \mid \mathcal{H}_1)$ integrates over this prior:

$$p(e \mid \mathcal{H}_1) = \int_0^\infty \mathcal{N}(e; a\delta, \nu) p_+(\delta)\, d\delta.$$

The local Bayes factor B therefore becomes

$$B = \frac{p(e \mid \mathcal{H}_1)}{p(e \mid \mathcal{H}_0)} = \int_0^\infty \frac{\mathcal{N}(e; a\delta, \nu)}{\mathcal{N}(e; 0, \nu)} p_+(\delta)\, d\delta.$$

Both likelihoods have variance ν, so their normalization factors cancel. Expanding the squared terms gives

$$\begin{aligned}
\frac{\mathcal{N}(e; a\delta, \nu)}{\mathcal{N}(e; 0, \nu)} &= \exp\left(-\frac{(e - a\delta)^2}{2\nu} + \frac{e^2}{2\nu}\right) \\
&= \exp\left(-\frac{e^2 - 2ae\delta + a^2\delta^2 - e^2}{2\nu}\right) \\
&= \exp\left(\frac{ae}{\nu}\delta - \frac{a^2}{2\nu}\delta^2\right).
\end{aligned}$$

Multiplying by the half-normal prior yields

$$B = \frac{2}{\sqrt{2\pi}\tau} \int_0^\infty \exp\left[\frac{ae}{\nu}\delta - \frac{1}{2}\left(\frac{a^2}{\nu} + \tau^{-2}\right)\delta^2\right] d\delta.$$

Define the coefficients $h = ae/\nu$ and $J = a^2/\nu + \tau^{-2} > 0$. Completing the square gives

$$\begin{aligned}
h\delta - \frac{1}{2}J\delta^2 &= -\frac{J}{2}\left(\delta^2 - \frac{2h}{J}\delta\right) \\
&= -\frac{J}{2}\left[\left(\delta - \frac{h}{J}\right)^2 - \frac{h^2}{J^2}\right] \\
&= -\frac{J}{2}\left(\delta - \frac{h}{J}\right)^2 + \frac{h^2}{2J}.
\end{aligned}$$

The last term does not depend on δ, so

$$B = \frac{2}{\sqrt{2\pi}\tau} \exp\left(\frac{h^2}{2J}\right) \int_0^\infty \exp\left[-\frac{J}{2}\left(\delta - \frac{h}{J}\right)^2\right] d\delta.$$

To evaluate the remaining integral, substitute $x = \sqrt{J}(\delta - h/J)$, giving $d\delta = dx/\sqrt{J}$. The lower limit $\delta = 0$ becomes $x = -h/\sqrt{J}$, while the upper limit remains infinity. Thus

$$B = \frac{2}{\tau\sqrt{J}} \exp\left(\frac{h^2}{2J}\right) \left[\frac{1}{\sqrt{2\pi}} \int_{-h/\sqrt{J}}^\infty e^{-x^2/2}\, dx\right].$$

Writing Φ for the standard normal cumulative distribution function, the bracketed term equals

$$1 - \Phi\left(-\frac{h}{\sqrt{J}}\right) = \Phi\left(\frac{h}{\sqrt{J}}\right),$$

by symmetry of the standard normal distribution. The resulting Bayes factor B is

$$B = \frac{2}{\tau\sqrt{J}} \exp\left(\frac{h^2}{2J}\right) \Phi\left(\frac{h}{\sqrt{J}}\right). \tag{32}$$

Taking logarithms gives the log evidence $\log B$:

$$\log B = \log 2 - \log \tau - \tfrac{1}{2} \log J + \frac{h^2}{2J} + \log \Phi\left(\frac{h}{\sqrt{J}}\right). \tag{33}$$

With $\tau = 1$, restoring the indices gives $h_{tg} = (1-c)e_{tg}/\nu_{tg}$ and $J_{tg} = (1-c)^2/\nu_{tg} + 1$, recovering Eq. 5. The Gaussian likelihood and half-normal prior make the exponent quadratic in δ, reducing marginalization to a standard normal CDF evaluation. The integral is therefore evaluated analytically under the specified residual model, without numerical integration or iterative optimization. If no neighbour has positive observation weight, we use neutral evidence $B = 1$, so the log evidence is zero.

Implementation bounds. We evaluate $\log \Phi(h/\sqrt{J})$ directly using the numerically stable `log_ndtr` function, avoiding an explicit CDF floor. The implementation floors $\tau_g^2 + v_{tg}$ at 10^{-6} to avoid division by zero, while $\epsilon_\nu = 10^{-4}$ prevents a degenerate residual variance. Neither variance floor is activated for observations used in history updates on the 500 PASLCD query frames.

We clip $\log B$ to $[-12, 12]$. For the 16-bit probability maps, the smallest nonzero value $\Delta = 1/65535$ corresponds to $|\operatorname{logit} \Delta| \approx 11.09$; rounding this magnitude up gives a log-evidence limit of 12. Standardized responses use the same numerical limit. These caps limit extreme evidence contributions. The dispersion floor and correlation adjustment in Eq. 29 are part of the working model.

D Accumulated Support and Causal Decoding

Generalized pixel posterior. Let $y = 1$ denote pixel change and $y = 0$ no change. For a support value $p \in (0, 1)$, define the binary cross-entropy loss and the sum used for fusion:

$$\begin{aligned} \mathrm{BCE}(y, p) &= -y \log p - (1-y) \log(1-p), \\ \mathcal{L}_t(y, u) &= \mathrm{BCE}(y, C_t(u)) + \mathrm{BCE}(y, H_t(u)). \end{aligned} \tag{34}$$

The two terms assign unit weight to agreement with current evidence and historical support. Let $q_t(y \mid u)$ denote the generalised posterior distribution over the two labels at pixel u, formed from $C_t(u)$, $H_t(u)$, and prior π. A loss-based Bernoulli update (Bissiri et al., 2016) gives

$$q_t(y \mid u) = \frac{\pi^y (1-\pi)^{1-y} \exp[-\mathcal{L}_t(y, u)]}{\sum_{v \in \{0,1\}} \pi^v (1-\pi)^{1-v} \exp[-\mathcal{L}_t(v, u)]}. \tag{35}$$

Taking the ratio of the two label probabilities gives

$$\frac{q_t(1 \mid u)}{q_t(0 \mid u)} = \frac{\pi}{1-\pi} \frac{C_t(u)}{1 - C_t(u)} \frac{H_t(u)}{1 - H_t(u)}. \tag{36}$$

The scalar map $Q_t^{\mathrm{pixel}}(u) = q_t(y = 1 \mid u)$ contains the generalised posterior probability of change; $q_t(y = 0 \mid u) = 1 - Q_t^{\mathrm{pixel}}(u)$. This gives the odds update in Eq. 1, equivalently $Q_t^{\mathrm{pixel}} = \sigma(\operatorname{logit} \pi + \operatorname{logit} C_t + \operatorname{logit} H_t)$. The implementation evaluates this closed form; the loss representation adds no training step. It specifies a composite evidence update rather than a factorization into independent observations. Selecting π by synthetic training IoU is a separate optimization objective from $\mathcal{L}_t$; neither that selection nor normalization guarantees exact probabilities on the target distribution.

Accumulated neighbourhood evidence. After pixel prediction, each sufficiently supported observation adds its log Bayes factor to the previous accumulated field F_{t-1}, giving the updated accumulated field F_t:

$$F_{tg} = F_{t-1,g} + \mathbb{I}_{tg} \log B_{tg}. \tag{37}$$

The accumulator starts at zero; the decoded support depends directly on F_{tg}. Neither initial image responses nor final pixel posteriors are written into this state. An observed Gaussian without usable neighbours contributes zero log evidence and leaves the accumulator unchanged.

Rendering the previous support. The previous accumulated field F_{t-1} is decoded per Gaussian before rendering:

$$H_t(u) = \frac{\sum_g a_{tug} \sigma(F_{t-1,g})}{\chi_{tu}}. \tag{38}$$

The sigmoid is applied to each Gaussian before spatial averaging; only then do we take $\operatorname{logit} H_t$ for the pixel update. The normalized contributions a_{tug}/χ_{tu} mix support from reference Gaussians associated with the pixel. With no previous observations, $F = 0$, giving $\sigma(F) = 0.5$ and neutral history on covered pixels. The effective prior π is applied after this rendering and leaves the Gaussian evidence accumulator unchanged.

Effective-prior search parameterization and storage. Let $Z = \operatorname{logit} C + \operatorname{logit} H$. The generalised posterior is $\sigma(Z + \operatorname{logit} \pi) = \sigma(Z - b_\pi)$, where $b_\pi = -\operatorname{logit} \pi$. For threshold search the implementation uses the equivalent coordinate $s = Z/2$, $\beta = b_\pi/2$, so $\sigma(Z - b_\pi) = \sigma(2(s - \beta))$ and the mask is determined by $s \geq \beta$. The scale factor is retained when evaluating scores. The fitted prior is stored directly in log-odds form.

Probabilities are clamped to $[10^{-5}, 1 - 10^{-5}]$ before taking their logits to avoid infinities at 0 and 1. The guard 10^{-5} is just below the 16-bit quantization step $1/65535 \approx 1.53 \times 10^{-5}$ and gives a maximum logit magnitude of approximately 11.51, within the limit of 12. The synthetic replay matches the deployment's coverage mask and 16-bit history-map storage. **Execution order.** For each query image, the implementation:

1. computes the image–render response, applies the learned response suppression, and applies the monotone evidence mapping;
2. aggregates this score onto the Gaussians, renders Gaussian-aggregated support, and combines the two to obtain C_t;
3. aggregates C_t once, standardizes against the unchanged-response model, and computes conditional neighbourhood evidence;
4. renders history from the previous accumulated field F_{t-1} and combines it with C_t using the fitted effective prior to obtain the pixel posterior;
5. adds the current log Bayes factor to the evidence accumulator for each sufficiently supported Gaussian exactly once.

Object-guided mask refinement acts on the image prediction without modifying history. A new target sequence starts with empty state.

Insufficient reference coverage. The observation indicator $\mathbb{I}_{tg}$ prevents history updates when Gaussian g's total rendering contribution is below one; without usable neighbours, $\log B_{tg} = 0$. Pixels with reference coverage at most 10^{-3} are assigned zero change probability and excluded from object completion, but remain in the evaluation domain. These rules avoid unsupported updates but cannot recover missing geometry, so changes in uncovered regions may be missed. The indicator measures rendering contribution, not geometric accuracy.

E Reliability-Weighted Effective-Prior Estimation

The effective pixel prior is fitted for the synthetic training objective in Eq. 13; it specifies an operating point, not an estimate of deployment change frequency. We implement this optimization through the equivalent threshold coordinate $b_\pi = -\operatorname{logit} \pi$, using $s = Z/2$ and $\beta = b_\pi/2$ for the deployment-matched search below. This one-to-one reparameterization preserves the objective and its optimality gap. For image i, foreground IoU is $\mathrm{IoU}(\widehat{Y}_i^{\mathrm{pixel}}(\pi), Y_i) = \mathrm{TP}_i(\pi)/(N_i^+ + \mathrm{FP}_i(\pi))$, where N_i^+ counts labelled changed pixels, and TP_i and FP_i count true and false positives before refinement.

Comparing unchanged-response tails across events. For synthetic image i, let $H_{0,i}$ be its labelled unchanged pixels and $s_i(x)$ the decision score at pixel x. For a score cutoff v, the unchanged survival curve is the fraction of these pixels exceeding it:

$$T_i(v) = \frac{|\{x \in H_{0,i} : s_i(x) \geq v\}|}{|H_{0,i}|}. \tag{39}$$

An edit event is an object/edit-type combination. First average T_i within each event e to obtain $\overline{T}_e$. For an image from event e, its reference curve is the pointwise median of $\overline{T}_{e'}$ over other events $e' \neq e$, denoted T_{-e}^{ref}. This forms an unchanged-response reference across events, limiting the influence of repeated images of any one event.

Let $e(i)$ be image i's event and $[v_{\min}, v_{\max}]$ the range of training-score cutoffs used to compare tails. Excess high response is

$$d_i = \frac{1}{v_{\max} - v_{\min}} \int_{v_{\min}}^{v_{\max}} \left[T_i(v) - T^{\mathrm{ref}}_{-e(i)}(v) \right]_+ dv. \tag{40}$$

The implementation evaluates this integral by the trapezoidal rule on the fixed training-score cutoffs. The subsequent decision-threshold search operates over the full score range.

Let $D_e = \mathrm{median}_{i:e(i)=e}\, d_i$. For event e, compute the centre m_{-e} and scale a_{-e} from the other event medians:

$$\begin{aligned} m_{-e} &= \mathrm{median}_{e' \neq e}\, D_{e'}, \qquad a_{-e} = 1.4826\ \mathrm{median}_{e' \neq e}\, |D_{e'} - m_{-e}|, \\ z_i^{\mathrm{tail}} &= \left[\frac{d_i - m_{-e(i)}}{a_{-e(i)}} \right]_+, \qquad \rho_i = \frac{1}{1 + (z_i^{\mathrm{tail}})^2}. \end{aligned} \tag{41}$$

Images with $z_i^{\mathrm{tail}} = 0$ receive weight one; standardized excesses of one and two receive weights $1/2$ and $1/5$. The factor 1.4826 scales the median absolute deviation (MAD) to match the standard deviation under a Gaussian model. The deployed fit requires positive across-event MAD. We retain the original pixel counts and IoU definitions; ρ_i weights each image's contribution to the mean IoU according to its unchanged-response reliability.

Training sequences and fixed weights. Sequences of images from the editing subset use the same image-evidence pipeline, unchanged-response model, history update order, and pixel prediction rule as online inference. The weights in Eq. 41 are normalized as $\widetilde{\rho}_i = \rho_i / \sum_j \rho_j$, computed once before searching for the threshold and remain fixed throughout optimization. Synthetic labels supply the unchanged regions used for reliability weighting and the changed regions used for IoU.

A valid branch-and-bound upper bound. Let $J(v)$ be the reliability-weighted IoU at decision threshold v in the s coordinate. Increasing v can only decrease both TP and FP. Consequently, for any interval $[l, r]$ and any v within it,

$$J(v) \leq U(l, r) = \sum_i \widetilde{\rho}_i \, \frac{\mathrm{TP}_i(l)}{N_i^+ + \mathrm{FP}_i(r)}. \tag{42}$$

The numerator at l bounds candidate numerators from above, and the denominator at r bounds denominators from below. Nonnegative weights preserve the bound under summation, even if no threshold attains it.

The search starts from a range enclosing the training scores, including all-positive and all-negative decisions. Sorted changed and unchanged scores support exact count queries by binary search. The optimizer maintains the best evaluated objective J_{best}, splits the remaining interval with the highest bound, evaluates new candidates, and prunes intervals whose bound cannot improve the best objective value found so far. It stops when

$$\max_{\text{remaining intervals}} U(l, r) - J_{\mathrm{best}} \leq 10^{-4}. \tag{43}$$

This is 0.01 percentage points in the synthetic training IoU objective. Midpoint subdivision does not require IoU to be monotone or unimodal. The certificate applies to the recorded, weighted training objective with the deployment pixel prediction rule.

F Object-Guided Mask Refinement

Object-guided mask refinement uses shared SAM2 image embeddings to suppress unsupported detections and complete retained object regions.

Contour learning from pixel proposals. We process sequences of images from the editing subset with the fixed pixel models, effective prior, and causal history. Thresholding the source pixel posterior at 0.5 produces connected components whose boxes prompt SAM2 on the synthetic query and reference rendering. Predictions supply prompts only: change masks from automatic reference edits (Appendix A.1) supply pixel labels, and ignored pixels are excluded.

The classifier uses 18 query/reference-symmetric features. We rank SAM2's three candidates independently in each view by predicted IoU and pair them by rank, forming paired mask logits, quality scores, and their per-side products. Each pair is replaced by its maximum and minimum, using the same transformation during fitting and inference. This retains support from either image regardless of which side contains the object. Within each source image, existing class-balanced pixel weights are multiplied once by its fixed prior-fitting reliability ρ_i. Global balancing then gives each class total weight 0.5 within its fitting or validation split. These weights are used for feature standardization, L2-regularized logistic regression, and Platt sigmoid scaling (Platt, 2000). At inference, standardization and both fitted affine transformations are folded into a single affine score ℓ_{contour} on the unstandardized symmetric features. The contour fitting/validation split is by edited object; upstream pixel models still use the larger camera group. The pixel parameters and prior are unchanged.

Online contour filtering. We decode masks on the query image and reference rendering using bounding-box prompts from detected connected components. The fitted classifier converts their mask-comparison features into $\ell_{\mathrm{contour}}(u)$. Its non-positive part gives the contour-filtered score $Q_t^{\mathrm{contour}}(u)$ through Eq. 8.

Object completion. Point prompts at response peaks of the surviving components produce SAM2 candidates in the query image. For each prompt, we select the candidate with highest predicted mask quality. For a candidate mask M over covered pixels, its mean log-odds evidence is $\overline{L}_M = |M|^{-1} \sum_{u \in M} \mathrm{logit}\, Q_t^{\mathrm{pixel}}(u)$, where $|M|$ is the number of covered pixels in the mask. We retain masks with $\overline{L}_M \geq 0$ and combine their confidence $\sigma(\overline{L}_M)$ with Q_t^{contour} to obtain the refined change score Q_t^{full} using Eq. 9. The component filter skips regions smaller than the SAM feature cell, and empty or uncovered masks are excluded. The pixel prior is already included in Q_t^{pixel}; refinement modifies only the final output.

G Additional Quantitative Results

Table 5 reports the full 20-instance comparison for the pixel and full outputs in Table 1.

G.1 Cross-instance Reference Reuse

Each PASLCD scene has two instances. We retain the reference reconstruction and all fitted parameters from one instance and evaluate queries from the other, in both directions. This gives 20 directed pairs, each with 25 frames. The reused parameters include response suppression, the LLR mapping, Gaussian-indexed unchanged-response statistics and spatial co-response, the pixel prior, and the contour filter. No parameters are refitted or selected using target data, and each sequence starts with empty history.

Queries are processed on the reference image canvas using reference-derived poses and initial difference cues. Query RGB is resized to this canvas for refinement, and the fixed reference model is rendered at the query pose. Predictions are evaluated against the target instance's original masks using the same resizing and thresholding convention as the main experiments. Table 7 reports all pairs. The original and cross-instance aggregate comparisons each contain every target instance once. Because the paired query sets are highly similar, this experiment tests within-scene reference reuse rather than cross-scene generalization. Runtime was not measured.

G.2 Fitted Priors and Change Area

Figure 5 shows the effective pixel priors $\pi^\star$ fitted from reference data for the 20 PASLCD instances. They range from 0.00578 to 0.30477, with median 0.02369 and interquartile range $[0.01476, 0.05471]$. Each prior selects an instance-specific decision operating point and remains fixed during online inference; it does not estimate the fraction of changed pixels in a query image.

To examine performance across change sizes, we group the 500 evaluated frames by the fraction of ground-truth changed pixels, keeping all fitted parameters and predictions unchanged (Table 8). All metrics are averaged over frames within each group, with false-positive rate $\mathrm{FPR} = \mathrm{FP}/(\mathrm{FP} + \mathrm{TN})$ and false-negative rate $\mathrm{FNR} = \mathrm{FN}/(\mathrm{TP} + \mathrm{FN})$. Frames with less than 1% change have lower mIoU and higher FNR than the 1–5% group; refinement improves their mIoU while slightly increasing

Table 5: PASLCD comparison using our evaluator over 20 scene-instances (%, higher is better). O-SCD(O) is online; O-SCD(R) includes offline refinement. Ours (pixel) is the pixel output; Ours (full) adds object-guided mask refinement. Total summarizes all 20 instances. Cell backgrounds mark first, second, and third place for each row and metric; these values are bold, and ties at the displayed precision share a rank.

Scene	Inst.	O-SCD(O) CVPR 2026		O-SCD(R) CVPR 2026		Ours pixel		Ours full	
		mIoU	F1	mIoU	F1	mIoU	F1	mIoU	F1
Cantina	1	50.24	65.86	**62.05**	**76.05**	**57.30**	**72.23**	**62.33**	**75.78**
	2	49.38	65.16	**54.75**	**70.19**	**54.88**	**70.12**	**58.19**	**72.59**
Garden	1	49.58	65.52	**52.28**	**67.07**	**68.64**	**80.64**	**75.22**	**85.30**
	2	51.54	67.78	**56.62**	**72.27**	**67.47**	**80.40**	**73.42**	**84.55**
Lounge	1	54.74	70.38	**60.04**	**74.91**	**69.47**	**81.72**	**75.95**	**86.18**
	2	**47.75**	**64.12**	**47.96**	**63.78**	45.15	60.64	**55.20**	**69.12**
Lunch room	1	38.85	53.95	**49.19**	**64.36**	**48.61**	**62.81**	**48.20**	**62.15**
	2	38.18	53.48	**49.15**	**64.60**	**48.02**	**62.83**	**50.85**	**64.12**
Meeting room	1	49.17	65.39	**55.44**	**70.83**	**56.92**	**71.75**	**62.64**	**75.89**
	2	45.53	61.83	**51.66**	**67.50**	**52.61**	**68.04**	**63.18**	**75.99**
Playground	1	**38.74**	**54.18**	33.89	49.68	**52.35**	**67.19**	**56.67**	**70.36**
	2	**36.22**	**51.64**	31.19	46.13	**51.09**	**65.77**	**57.04**	**69.94**
Porch	1	57.02	71.82	**71.48**	**83.14**	**73.75**	**84.41**	**81.88**	**89.38**
	2	53.03	68.58	**63.60**	**77.46**	**63.53**	**76.93**	**70.14**	**81.32**
Pots	1	60.83	75.37	**65.20**	**78.49**	**73.64**	**84.72**	**87.54**	**93.30**
	2	**57.67**	**72.63**	**63.04**	**76.89**	55.86	71.31	**75.21**	**85.55**
Printing area	1	60.69	74.60	**69.68**	**81.82**	**76.05**	**86.17**	**78.10**	**87.46**
	2	52.11	67.94	**57.86**	**73.03**	**60.13**	**74.47**	**63.83**	**77.33**
Zen	1	47.42	60.72	**62.23**	**72.33**	**67.43**	**76.16**	**75.27**	**81.03**
	2	46.17	60.97	**62.54**	**74.88**	**59.72**	**72.17**	**70.80**	**80.16**
Total		49.24	64.60	**55.99**	**70.27**	**60.13**	**73.52**	**67.08**	**78.37**

Table 6: Same-scene cross-instance reference reuse on PASLCD (%). Original pairing uses each instance's own reference; cross-instance pairing uses the other instance's reference and fitted parameters. Both average 20 sequences.

	Original pairing		Cross-instance	
Output	mIoU	F1	mIoU	F1
Ours (pixel)	60.13	73.52	59.94	73.40
Ours (full)	67.08	78.37	67.12	78.39

FNR. The observed change fractions span 0.17–20.13%, with only one frame above 20% and none at 30% or higher. Scene and viewpoint composition differ across groups, so this analysis describes fixed-prior performance rather than isolating the effect of change area or establishing robustness to prior drift.

G.3 Relighting Strength

We vary the strength of the reference-fitted colour perturbation while retaining the object edits, labels, camera subsets, and query order. For each selected coefficient vector θ, we use $\theta_K = \theta_{\mathrm{id}} + K(\theta - \theta_{\mathrm{id}})$, where θ_{id} gives the identity colour transform, and render the transformed Gaussian colours with the same clipping rule. The default is $K = 1$; $K = 0$ removes the relighting perturbation, and $K = 2$ doubles it. At $K = 0$, the relighting sample slots are retained with identity coefficients, so all settings contain the same 11,384 training records. For $K = 0$ and $K = 2$, we refit suppression, the likelihood-ratio mapping, unchanged-response statistics, image weights, the pixel prior, and the contour filter using only reference data and the same fitting settings. The $K = 2$ rendered images are reused from a verified generation run; all fitted models are recomputed for the current method. Table 9 reports pixel and full outputs on the same 20 PASLCD instances and 500 query frames, with

Table 7: Cross-instance results for all 20 directed PASLCD pairs (%). Direction denotes reference instance → query instance. Pixel and full outputs use the same fixed reference reconstruction and fitted parameters.

		Pixel output		Full output	
Scene	Direction	mIoU	F1	mIoU	F1
Cantina	1 → 2	57.82	72.61	62.82	76.14
Cantina	2 → 1	54.89	70.08	58.16	72.41
Garden	1 → 2	67.24	79.68	74.07	84.67
Garden	2 → 1	67.14	80.14	74.62	85.38
Lounge	1 → 2	68.94	81.48	74.89	85.48
Lounge	2 → 1	48.16	64.12	58.57	72.68
Lunch room	1 → 2	47.53	62.08	48.50	62.13
Lunch room	2 → 1	46.31	60.98	49.63	62.92
Meeting room	1 → 2	56.78	71.68	64.26	77.50
Meeting room	2 → 1	52.98	68.34	63.62	76.19
Playground	1 → 2	52.41	67.13	57.25	70.78
Playground	2 → 1	50.52	65.21	56.37	69.29
Porch	1 → 2	74.08	84.64	81.33	89.09
Porch	2 → 1	63.12	76.28	69.63	80.45
Pots	1 → 2	73.11	84.38	87.35	93.19
Pots	2 → 1	55.72	71.16	74.60	85.13
Printing area	1 → 2	75.11	85.55	77.60	87.15
Printing area	2 → 1	59.79	74.29	64.02	77.29
Zen	1 → 2	67.62	76.30	75.37	81.07
Zen	2 → 1	59.54	71.81	69.73	78.95
Mean		59.94	73.40	67.12	78.39

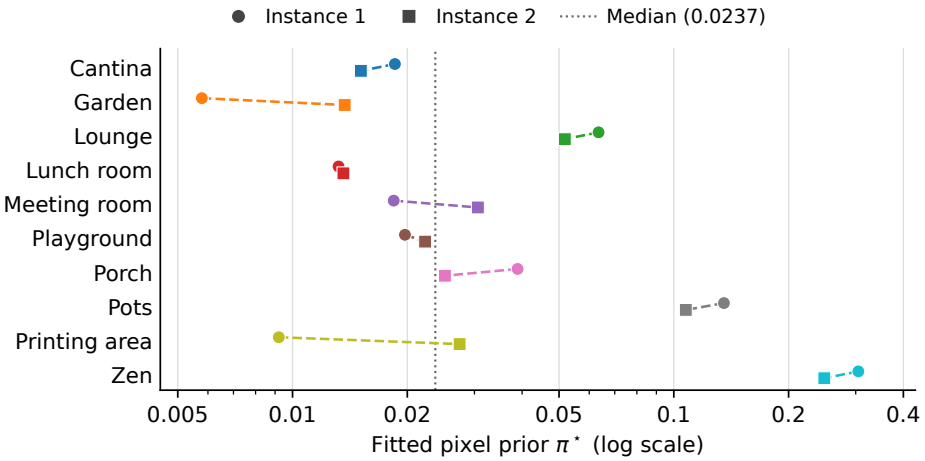

Figure 5: Fitted pixel priors on PASLCD. Each scene has a distinct colour; circles and squares denote its two instances, joined by a dashed line. The vertical dotted line marks the median over all 20 instances. The horizontal axis uses a logarithmic scale.

$K = 1$ reusing the main results. The new fits use the stable log-CDF evaluation in Appendix C; an independent fixed-parameter comparison changed full mIoU by only 0.0015 points relative to the earlier CDF-floor evaluation used for the retained $K = 1$ results.

Default relighting improves full-output mIoU by 0.71 points over $K = 0$, with gains in 13 of 20 instances. Doubling the strength changes pixel and full mIoU by $+0.03$ and -0.09 points, respectively, relative to the default. These results support a modest benefit from the adopted augmentation, without an additional aggregate full-output mIoU gain from stronger perturbations.

Table 8: PASLCD performance by ground-truth change area. Metrics are percentages averaged over frames. The 20–30% group contains only one frame; no evaluated frame has $\geq 30\%$ change area.

			Pixel output			Refined output		
Change area	Frames	Instances	mIoU↑	FPR↓	FNR↓	mIoU↑	FPR↓	FNR↓
$< 1\%$	88	12	50.23	0.68	35.42	55.03	0.59	37.72
1–5%	298	18	60.83	1.00	18.36	69.47	0.64	17.10
5–10%	84	12	65.65	1.11	25.33	70.49	0.80	22.69
10–20%	29	4	66.97	1.77	25.26	69.74	1.58	23.12
20–30%	1	1	62.66	5.49	23.68	54.83	10.98	21.27

Table 9: Relighting strength on PASLCD, averaged over 20 instances (%). All settings retain the same training records; K scales the lighting perturbation about the identity transform.

	Pixel output		Full output	
Strength	mIoU ↑	F1 ↑	mIoU ↑	F1 ↑
$K = 0$	59.76	73.23	66.37	77.92
$K = 1$ (default)	60.13	73.52	**67.08**	**78.37**
$K = 2$	**60.16**	**73.54**	67.00	78.31

G.4 Construction of Historical Support

We compare two ways of constructing historical support: our Bayes-factor accumulation and temporal averaging, using pixel outputs on the same 20 PASLCD instances and 500 frames. Temporal averaging takes the mean of each Gaussian's valid responses $r_{tg} = [\mathcal{A}_t(C_t)]_g$ across earlier frames, selected by $\mathbb{I}_{tg}$, and assigns support 0.5 to unobserved Gaussians. These values are rendered to obtain H_t, with prediction preceding the current update as in our method. We retain the image-evidence pipeline and query order, and refit the pixel prior on the editing subset using the same fitting procedure, without target labels. Table 10 reuses the no-history and full-pixel results from Table 3. Our construction improves mean mIoU by 4.32 points over temporal averaging and performs better in 16 of 20 instances.

G.5 Offline Preparation and Component Timing Breakdown

Table 11 details the offline preparation and component fitting time across all 20 PASLCD scene-instances, measured on an NVIDIA RTX 5090 using synchronized GPU timing. The offline workflow comprises two main phases: (1) *synthetic edit generation*, which extracts multi-view object proposals with SAM2, performs 3D Gaussian edits (recolouring, removal, translation), renders lighting-varied views, and forms training observation pairs; and (2) *scene component fitting*, which executes reference preprocessing, fits the pixel pipeline components (response suppression, log-likelihood ratio mapping, unchanged response moments, and 1D branch-and-bound prior search $\pi^\star$), and trains the 18-dimensional symmetric contour filter with Platt scaling. Across all instances, the complete offline workflow averages 215.82 s ($\sim$3.6 minutes; median 197.13 s, ranging from 142.15 s to 440.69 s), given an existing reference reconstruction.

Table 10: Historical support on PASLCD, averaged over 20 instances (%, higher is better). Ours uses Bayes-factor accumulation. All results use pixel outputs without refinement.

Historical support	mIoU ↑	F1 ↑
None	57.45	71.26
Temporal averaging	55.81	70.13
Ours	**60.13**	**73.52**

Table 11: Complete offline preparation and fitting runtime breakdown across 20 PASLCD scene-instances (seconds, measured on an NVIDIA RTX 5090). Generation includes automatic object proposals, 3DGS edits, and synthetic paired observation rendering. Fitting covers reference preprocessing, pixel models, prior search, and contour filter fitting.

Scene	Inst.	Retained objects	Training rows	Generation (s) ↓	Preprocess (s) ↓	Pixel (s) ↓	Contour (s) ↓	Total (s) ↓
Cantina	1	22	366	106.95	8.01	43.83	16.89	175.68
	2	28	602	112.19	7.74	53.64	26.46	200.02
Garden	1	37	1080	119.42	7.15	78.43	35.70	240.69
	2	35	752	121.05	7.64	71.95	31.82	232.46
Lounge	1	18	384	112.07	7.57	43.14	19.08	181.86
	2	18	340	79.13	6.94	37.24	18.84	142.15
Lunch_room	1	56	1744	226.44	10.42	110.15	57.15	404.16
	2	64	2034	231.67	11.28	131.89	65.85	440.69
Meeting_room	1	17	410	90.87	7.17	50.84	21.64	170.52
	2	21	484	85.33	7.19	58.74	25.81	177.07
Playground	1	31	718	114.45	7.82	79.18	37.88	239.34
	2	33	642	117.92	8.44	92.42	45.03	263.81
Porch	1	44	842	102.77	8.35	63.87	36.39	211.38
	2	43	788	122.44	8.95	63.88	37.36	232.63
Pots	1	14	280	74.38	7.74	39.81	20.62	142.55
	2	20	400	75.64	7.37	45.73	24.28	153.02
Printing_area	1	23	756	107.85	8.08	54.08	29.76	199.77
	2	23	416	85.96	6.76	39.95	20.66	153.33
Zen	1	24	564	91.22	8.07	60.91	34.29	194.49
	2	18	378	77.34	7.07	49.33	26.94	160.68
Mean	—	29.45	699.0	112.75	7.99	63.45	31.62	**215.82**
Median	—	23.50	583.0	107.40	7.74	56.41	28.35	**197.13**

H Additional Visual Results

Figure 6 presents additional results across six scenes. Our method reduces excess responses and recovers changed regions, but some errors remain: thin structures are partly missed in Garden, false positives persist near the ceiling in Lounge, and parts of changed objects are missed in Lunch room. These examples illustrate the limits of transferring reference-fitted evidence models to real observations and correcting residual errors through refinement.

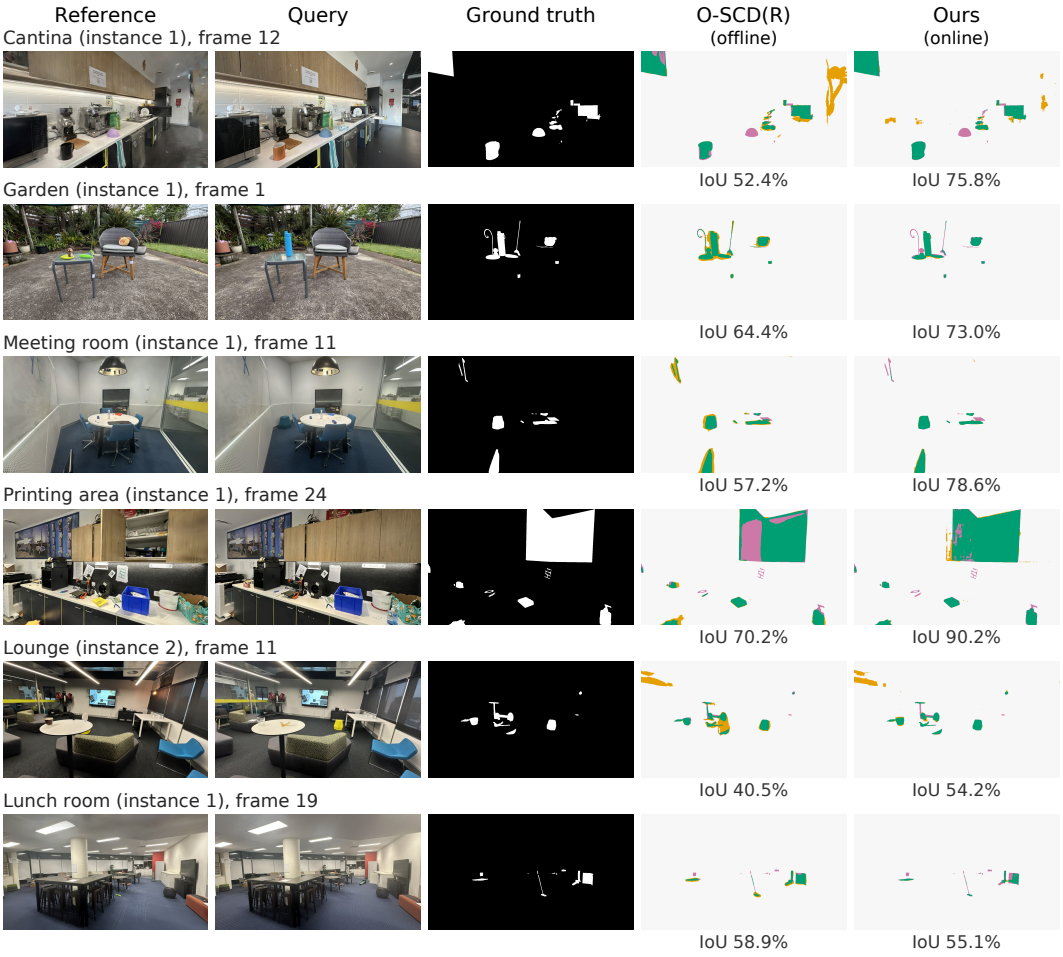

Figure 6: **Additional visual results on PASLCD.** Columns show reference renderings, queries, ground truth, O-SCD(R) (offline), and our complete method (online). ■ correct detections; ■ false positives; ■ missed changes. Numbers report per-frame foreground IoU. The final two rows highlight remaining errors.