LONG-RANGE LATENT MODELING FOR QUANTA VIDEO RECONSTRUCTION AND TRANSMISSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Quanta image sensors (QIS) capture dynamic scenes through high-rate binary photon measurements, offering fine temporal resolution but extremely sparse photon evidence in each exposure. High-quality video reconstruction therefore requires jointly exploiting photon evidence over extended observation intervals. Existing QIS reconstruction methods typically use bounded temporal contexts to generate frames at predefined timestamps, fixing the reconstruction as a discrete image sequence before compression and transmission. We propose a temporal latent representation framework for long QIS sequences that models scene evolution throughout the observation interval as queryable latent trajectories. A hierarchical long-range encoder jointly aggregates sparse photon evidence and produces multi-scale temporal latent anchors, providing discrete support for latent trajectories at complementary resolutions. Given an arbitrary temporal coordinate within the encoded interval, a temporal query module constructs the corresponding latent state through interpolation between neighboring anchors and learned residual refinement, and a shared decoder maps this state to a video frame. Scene evolution is thus retained in a reusable latent representation, allowing receivers to select output timestamps after encoding and transmission without revisiting the raw measurements or rerunning the encoder. The same multi-scale latent state supports compact transmission through cross-scale predictive compression, unifying long-range reconstruction, state compression, and on-demand temporal querying within a shared representation. Experiments across five photon levels show that our method outperforms the evaluated QIS reconstruction baselines and effectively exploits photon evidence from longer observation intervals to improve reconstruction quality. For a 4096-frame binary QIS sequence, the compressed latent state occupies only 4.628 MiB, compared with 64.000 MiB for the bit-packed raw measurements, while retaining arbitrary temporal query capability after transmission.

1 INTRODUCTION

Quanta image sensors (QIS) and related single-photon cameras capture dynamic scenes at extremely high temporal resolution, enabling ultralow-light imaging, passive single-photon imaging, and high-speed vision (Fossum et al., 2016; Ma et al., 2022; Ingle et al., 2019; 2021; Zhang et al., 2025). During each short exposure, a one-bit QIS records whether each jot detects at least one photoelectron, yielding high-rate binary photon measurements (Yang et al., 2012; Chan et al., 2016; Chan, 2022). Recovering reliable scene intensity requires statistically aggregating many such binary Poisson observations (Yang et al., 2010; 2012; Chan & Lu, 2014; Remez et al., 2016; Fossum, 2013). At low photon levels, fine temporal sampling leaves sparse evidence per exposure: reconstruction must integrate observations over time while accounting for scene variation.

Existing dynamic-QIS methods typically aggregate observations around individual target frames. Classical approaches recover images or video frames by selecting a temporal neighborhood, estimating motion, and aligning or aggregating QIS measurements (Chi et al., 2020; Ma et al., 2020; Seets et al., 2021; Iwabuchi et al., 2019). QUIVEr and Bit2Bit improve dynamic reconstruction through recurrent fusion, motion reasoning, or self-supervised photon prediction (Chennuri et al., 2024; Liu et al., 2024), while generative and latent approaches provide stronger priors for photon-limited reconstruction (Purohit et al., 2024; Garg et al., 2026; Chennuri et al., 2025). Despite differences in architecture, supervision, and context, these methods typically treat frames reconstructed at predefined times as the

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.

final scene representation. This boundary is restrictive for high-rate photon sensing: long observation intervals provide complementary evidence, while downstream applications may require temporal sampling choices made after encoding.

This frame-centric interface creates a transmission trade-off. Transmitting the raw QIS bitstream preserves the receiver's choice of reconstruction method and output times, but its payload scales with acquisition length and all reconstruction occurs at the receiver (Zhang et al., 2025). Alternatively, standard or learned video codecs exploit spatiotemporal redundancy to reduce payload after the sender reconstructs a conventional video (Sullivan et al., 2012; Lu et al., 2019; Li et al., 2021; 2023; 2024). This route incurs sender-side reconstruction and fixes the temporal sampling grid before transmission. Implicit neural video representations, including NeRV variants, likewise represent an already reconstructed video compactly as a function of temporal coordinates (Sitzmann et al., 2020; Chen et al., 2021; 2023; Lee et al., 2023; Kwan et al., 2024; Zhang et al., 2024; Tang et al., 2025). They therefore do not eliminate QIS reconstruction. The key question is not only how to compress a reconstructed video, but whether QIS reconstruction must terminate in a fixed video.

Rather than materializing a fixed video, we move the representation boundary upstream: QIS measurements are encoded once into a reusable temporal latent state capturing scene evolution over the observation interval. Hierarchical anchors aggregate long-range photon evidence; a predictive codec enables compact transmission, and shared decoding supports receiver-selected queries without revisiting measurements or rerunning the encoder. Experiments across five photon levels demonstrate improved reconstruction, effective use of longer observations, and smaller payloads than raw QIS.

Our main contributions are:

- We introduce a measurement-to-latent representation: long QIS sequences are encoded into a reusable temporal state rather than fixed frames, unifying long-range photon aggregation, reconstruction, compression, and post-encoding querying.
- We introduce a hierarchical encoder that aggregates long-range photon evidence through multi-scale temporal latents and reconstructs frames with time-conditioned queries.
- We compress this state for one-time transmission and repeated receiver queries, reducing payload relative to raw QIS while retaining control over timestamps and sampling density.

2 Related Work

QIS and quanta-video reconstruction. QIS reconstruction has progressed from statistical estimation, sensor analysis, and threshold design for binary Poisson measurements (Yang et al., 2010; 2012; Chan & Lu, 2014; Chan et al., 2016; Fossum, 2013; Elgendy & Chan, 2018) to learned reconstruction and motion-aware temporal integration for low-light and dynamic scenes (Remez et al., 2016; Choi et al., 2018; Chi et al., 2020; Chan, 2022). QBP combines motion estimation, alignment, and denoising (Ma et al., 2020); adaptive methods select scene-dependent integration intervals (Seets et al., 2021; Iwabuchi et al., 2019); QUIVEr uses temporal fusion, while Bit2Bit learns reconstruction through self-supervision (Chennuri et al., 2024; Liu et al., 2024). Generative and latent methods strengthen reconstruction priors (Purohit et al., 2024; Garg et al., 2026; Chennuri et al., 2025). These methods remain frame-centric, converting temporal observations into explicit target frames. We instead encode the long QIS stream into a shared, queryable temporal state, unifying long-range photon aggregation, reconstruction, and transmission without fixing output timestamps during encoding.

Dynamic reconstruction from high-speed sparse vision sensors. Event cameras asynchronously record log-intensity changes, with reconstruction advancing from joint motion and intensity estimation (Kim et al., 2014) to learned temporal fusion (Rebecq et al., 2019; Weng et al., 2021; Ercan et al., 2024). Spike cameras encode accumulated brightness through threshold-triggered spikes (Zhu et al., 2019); reconstruction uses motion-based and biologically inspired models (Zhao et al., 2020; Zheng et al., 2021) or learned spatiotemporal representations (Zhao et al., 2021; Chen et al., 2022; Zhang et al., 2023; Zhao et al., 2024; Fan et al., 2024), with memory transfer extending context across clips (Dong et al., 2026). Despite their different sensing physics, these approaches share QIS reconstruction's need to aggregate sparse temporal evidence. Beyond reconstruction quality and temporal memory, our setting requires a compact, independently transmittable state supporting receiver-selected timestamps without retransmitting or re-encoding the original measurements.

Implicit video representations and neural video compression. Implicit neural representations model visual signals as coordinate-conditioned functions (Sitzmann et al., 2020). NeRV variants compactly represent videos and synthesize frames from temporal coordinates (Chen et al., 2021; 2023; Lee et al., 2023; Kwan et al., 2024; Zhang et al., 2024; Tang et al., 2025). HEVC and learned codecs such as DVC and DCVC exploit spatiotemporal redundancy among video frames (Sullivan et al., 2012; Lu et al., 2019; Li et al., 2021; 2023; 2024). These studies provide complementary perspectives on coordinate-based representation and compact video coding. Our focus is a different reconstruction interface: transforming sparse binary QIS measurements into a reusable temporal latent state that supports compression and post-encoding frame queries, rather than general-purpose coding of pre-existing video frames.

3 METHOD

We formulate QIS reconstruction as learning a latent representation of scene evolution over the encoded observation interval, rather than predicting a fixed frame sequence. Given T binary QIS bit-planes $\mathbf{Y} = \{Y_t\}_{t=0}^{T-1}$, the encoder produces a reusable latent state for compression and frame reconstruction through arbitrary-time coordinate queries. Figure 1 gives an overview; Appendices A–B detail the reconstruction network and latent codec.

3.1 QUERYABLE LONG-RANGE LATENT REPRESENTATION

For a spatial location $\mathbf{p}$, the one-bit QIS observation model is given by

$$Y_t(\mathbf{p}) \sim \text{Bernoulli}\left(1 - \exp[-\eta s_t(\mathbf{p})\Delta t]\right), \tag{1}$$

where $s_t(\mathbf{p})$ is incident photon flux, η quantum efficiency, and Δt exposure time. Sparse evidence per bit-plane motivates joint reconstruction over extended temporal ranges.

The long-range encoder $\mathcal{E}_\theta$ maps the entire QIS segment into a shared latent state $\mathcal{Z}_\mathbf{Y}$. A temporal query at coordinate τ then supplies latent features for image reconstruction:

$$\mathcal{Z}_\mathbf{Y} = \mathcal{E}_\theta(\mathbf{Y}), \qquad \hat{s}(\tau) = \mathcal{D}_\psi\left(\mathcal{Q}_\phi(\mathcal{Z}_\mathbf{Y}, \tau)\right). \tag{2}$$

Here, $\mathcal{Q}_\phi$ denotes the temporal query module and $\mathcal{D}_\psi$ the reconstruction decoder. Equation 2 replaces a direct mapping from measurements to fixed output frames with a measurement-to-state representation followed by temporal querying. Consequently, the output timestamps need not be specified before long-range encoding, and the same latent state can be queried repeatedly without reprocessing the original QIS measurements.

3.2 HIERARCHICAL LONG-RANGE STATE CONSTRUCTION

Directly modeling thousands of high-rate binary bit-planes is computationally expensive. We first apply temporal binning to aggregate local photon evidence and reduce the temporal input resolution. The sequence is partitioned into consecutive chunks, each divided into non-overlapping temporal bins. Bit-planes within each bin are averaged, and the resulting bin images form temporally ordered input channels. In the default setting, each 32-frame chunk produces four bins of eight bit-planes each. This representation preserves within-chunk ordering while reducing the number of temporal positions processed by the long-range encoder.

The long-range encoder $\mathcal{E}_\theta$ performs hierarchical spatiotemporal modeling over the complete input segment and produces multi-level features $\{F^\ell\}_{\ell=0}^{L-1}$. As the encoding depth increases, the features progressively aggregate photon evidence over larger temporal ranges, while shallower features retain richer local spatial structures. Features from different encoding levels are propagated through lateral pathways to the subsequent reconstruction stages and are mapped by scale-specific projections $\mathcal{P}_\ell$ to temporally ordered latent anchors, denoted by $A^\ell = \mathcal{P}_\ell(F^\ell) = \{a_i^\ell\}_{i=0}^{K_\ell-1}$. Here, a_i^ℓ is a latent feature map rather than a reconstructed frame, and scales are indexed from coarse to fine, starting at $\ell = 0$. The complete state $\mathcal{Z}_\mathbf{Y} = \{A^\ell\}_{\ell=0}^{L-1}$ provides discrete temporal supports for learned latent trajectories at complementary spatiotemporal resolutions. Temporal querying evaluates these trajectories between the stored anchors, while image formation takes place only in the reconstruction decoder.

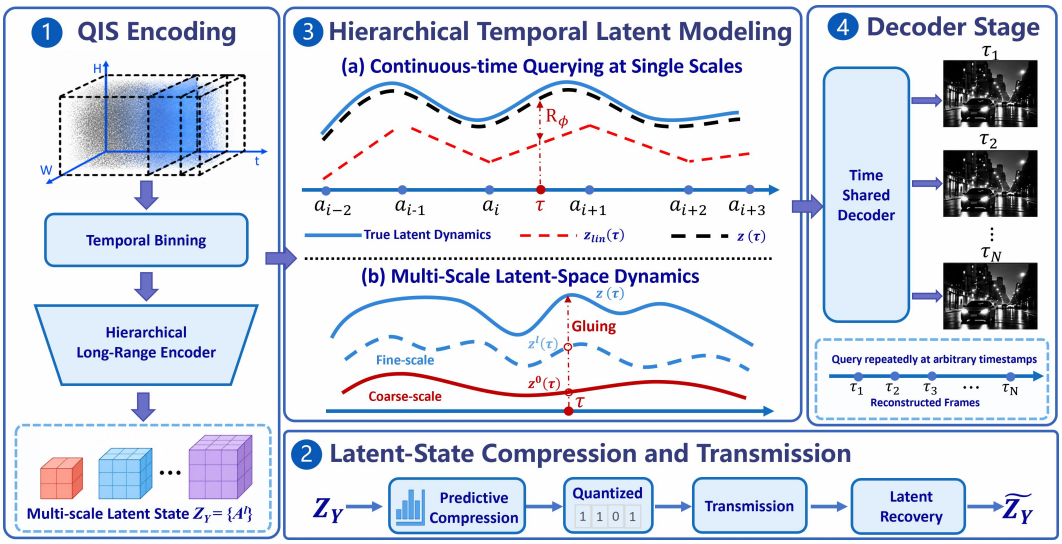

Figure 1: Measurement-to-latent framework. The encoder maps a long binary QIS segment to the multi-scale state $\mathcal{Z}_{\mathbf{Y}} = \{A^\ell\}$, which is quantized, transmitted, and recovered as $\widetilde{\mathcal{Z}}_{\mathbf{Y}}$. At a requested coordinate, $\mathcal{Q}_\phi$ interpolates and refines neighboring anchors: coarse latents capture long-range dynamics, fine latents add local detail, and together they determine the queried state. The shared decoder reconstructs receiver-selected frames, and the same path supports repeated queries without revisiting measurements or rerunning the encoder. Figure 8 expands the reconstruction network; Figures 9–10 expand its latent codec.

3.3 Time-Coordinate Querying

Given a normalized time $\tau \in [0, 1]$, we construct a piecewise-linear latent estimate from anchors at their associated temporal coordinates. Omitting the scale index, let $A = \{a_i\}_{i=0}^{K-1}$ have ordered supports $t_0 < \cdots < t_{K-1}$, with $K \geq 2$. For the selected neighboring pair, the relative temporal position and linear estimate are

$$\beta = \frac{\tau - t_i}{t_{i+1} - t_i}, \qquad z_{\text{lin}}(\tau) = (1 - \beta)a_i + \beta a_{i+1}. \tag{3}$$

Boundary queries use the nearest anchor pair for extrapolation (Appendix C.3). To capture nonlinear local evolution, we refine this estimate with a time-conditioned residual:

$$z(\tau) = z_{\text{lin}}(\tau) + \mathcal{R}_\phi \left(a_i, a_{i+1}, z_{\text{lin}}(\tau), \beta, e(\tau)\right), \tag{4}$$

where $e(\tau)$ denotes the Fourier temporal encoding and $\mathcal{R}_\phi$ is a lightweight residual refinement network conditioned on the neighboring anchors, the linear estimate, and the temporal coordinate. Thus, $z_{\text{lin}}(\tau)$ describes a piecewise-linear base trajectory, while $z(\tau)$ is the residual-refined latent query. The query module forms $\mathcal{Q}_\phi(\mathcal{Z}_{\mathbf{Y}}, \tau) = \{z^\ell(\tau)\}_{\ell=0}^{L-1}$. Coarse- and fine-scale features jointly determine the queried multi-scale state before shared decoding:

$$\hat{s}(\tau) = \mathcal{D}_\psi \left(\{z^\ell(\tau)\}_{\ell=0}^{L-1}\right), \tag{5}$$

where $z^\ell(\tau)$ denotes the queried latent feature at scale ℓ. The decoder shares its parameters across query times. Temporal querying therefore operates within the latent representation rather than between reconstructed video frames.

3.4 Cross-Scale Predictive Fixed-Rate Compression

The hierarchical latent state $\mathcal{Z}_{\mathbf{Y}} = \{A^\ell\}_{\ell=0}^{L-1}$ also serves as the transmitted representation. The scales are organized in a joint spatiotemporal coarse-to-fine hierarchy: as the scale becomes finer, the temporal anchor density and spatial resolution increase, while the channel dimension decreases. Since adjacent scales describe the same dynamic scene at different spatiotemporal resolutions, they

exhibit strong hierarchical dependence. We therefore adopt coarse-to-fine predictive coding. The coarsest state A^0 is first transformed by learned spatial downsampling, temporal downsampling, and channel compression to obtain the base transmitted representation. For each finer scale $\ell > 0$, the previously reconstructed state $\widetilde{A}^{\ell-1}$ is used to predict the current state, and only the prediction residual is compressed for transmission:

$$P^\ell = \mathcal{U}_\ell\left(\widetilde{A}^{\ell-1}\right), \qquad R^\ell = A^\ell - P^\ell, \qquad c^\ell = \mathcal{C}_\ell^{\mathrm{enc}}\left(R^\ell\right). \tag{6}$$

Here, $\mathcal{U}_\ell$ aligns the previous scale to the current spatiotemporal resolution through pointwise channel projection, spatial upsampling, temporal interpolation, and local temporal convolution, while $\mathcal{C}_\ell^{\mathrm{enc}}$ maps the prediction residual to a compact transmitted variable through learned spatiotemporal downsampling and channel compression. In this way, the coarse scale carries the dominant low-resolution spatiotemporal structure, while finer scales only encode the additional information that cannot be predicted using the reconstructed preceding scale.

At the receiver, the hierarchical state is reconstructed in the same coarse-to-fine order. The coarsest state is first recovered as $\widetilde{A}^0$. For each finer scale, the reconstructed previous state is used to regenerate the current prediction, and the decoded residual is added back:

$$\widetilde{A}^\ell = \mathcal{U}_\ell\left(\widetilde{A}^{\ell-1}\right) + \mathcal{C}_\ell^{\mathrm{dec}}\left(\widetilde{c}^\ell\right), \qquad \ell > 0, \tag{7}$$

where $\widetilde{c}^\ell$ denotes the recovered quantized code and $\mathcal{C}_\ell^{\mathrm{dec}}$ is the corresponding residual reconstruction transform. The multiscale states are progressively recovered to form $\widetilde{\mathcal{Z}}_{\mathbf{Y}} = \{\widetilde{A}^\ell\}_{\ell=0}^{L-1}$. Receiver-side reconstruction then evaluates $\mathcal{D}_\psi(\mathcal{Q}_\phi(\widetilde{\mathcal{Z}}_{\mathbf{Y}}, \tau))$, using recovered anchors in the same query equations. The cross-scale coding residual R^ℓ in equation 6 is distinct from the time-conditioned query correction $\mathcal{R}_\phi$ in equation 4. Transmitted coefficients are symmetrically quantized to signed eight-bit integers (INT8), with 16-bit floating-point (FP16) scaling factors. This fixed-rate codec uses no probability model or entropy coding; reported payloads are actual stored file sizes.

Appendix B details the codec architecture and its building blocks.

3.5 Query-Aware Training

Supervising only discrete anchor timestamps does not guarantee that latent states between neighboring anchors remain reliably decodable. We therefore sample both anchor-aligned queries and inter-anchor queries during training, using the same anchor selection, linear interpolation, residual refinement, and decoding procedure as at inference time. Anchor-aligned supervision constrains the reconstruction accuracy at discrete temporal supports, while inter-anchor supervision additionally regularizes the local transitions between neighboring states, encouraging the encoded representation to remain temporally queryable at the requested coordinates. The reconstruction objective is

$$\mathcal{L} = \mathcal{L}_1 + \lambda_\nabla \mathcal{L}_\nabla + \lambda_{\mathrm{ssim}} \mathcal{L}_{\mathrm{SSIM}} + \lambda_{\mathrm{perc}} \mathcal{L}_{\mathrm{VGG}}. \tag{8}$$

The four losses constrain reconstruction error, local gradient consistency, structural similarity, and perceptual quality, respectively.

4 Experiments

4.1 Experimental Setup

Datasets and Implementation Details. Following QUIVEr (Chennuri et al., 2024), we simulate QIS measurements from I2-2000FPS using 200 training and 13 validation sequences from disjoint source videos. We evaluate $\alpha \in \{0.05, 0.1, 0.2, 0.3, 0.5\}$ and real monochrome QIS recordings released with QBP (Ma et al., 2020), at estimated levels of 0.03, 0.14, and 0.66 photons/pixel (PPP). Each training sample contains 4096 consecutive binary frames with spatial crops up to 128×128. We train the main model for 100,000 iterations on an NVIDIA A100 using AdamW, batch size 3, weight decay 10^{-4}, and cosine learning-rate annealing from 2×10^{-4} to 2×10^{-5}. The L1, gradient, SSIM, and VGG loss weights are 1.0, 0.2, 0.2, and 0.1. Training proceeds from reconstruction without compression to latent-codec training and then joint optimization. Appendix C details simulation, sensor defects, photon-level scheduling, augmentation, and query supervision.

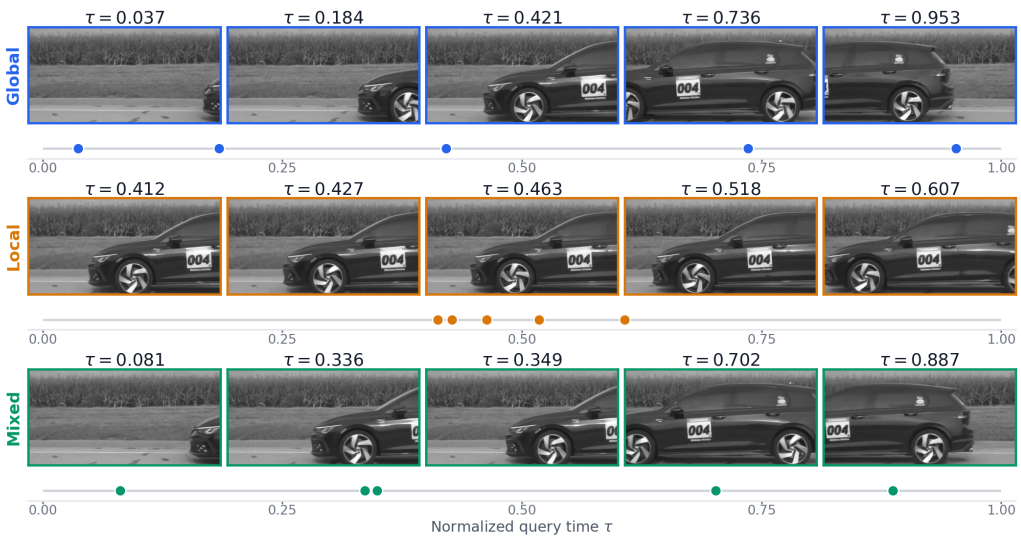

Figure 2: Arbitrary temporal queries on an automotive scene from a single encoded QIS sequence.

Comparison Methods and Evaluation Metrics. We compare with four representative QIS reconstruction methods: the model-based QBP (Ma et al., 2020), the supervised QUIVEr (Chennuri et al., 2024), the generative gQIR (Garg et al., 2026), and the self-supervised photon-prediction method Bit2Bit (Liu et al., 2024). For qualitative comparisons, QBP-E denotes QBP followed by BM3D denoising (Dabov et al., 2007). We evaluate reconstruction quality with PSNR and SSIM (Wang et al., 2004) and LPIPS (Zhang et al., 2018).

4.2 Comparison with Existing Methods

4.2.1 Qualitative Results

Figure 2 demonstrates arbitrary temporal querying on an automotive scene: a 24,576-frame binary QIS sequence is encoded once, and the same transmitted state reconstructs frames at times selected after transmission, without revisiting measurements or rerunning the encoder.

Figure 3 compares four simulated scenes at matched times and regions. QBP leaves noisy signs and statues; QBP-E suppresses noise but smooths details. QUIVEr restores major structures but blurs some lettering and statue details. gQIR restores relatively clear facial features and Bit2Bit major contours, though detail recovery varies across scenes. Ours balances noise suppression and structure preservation across lettering, statues, and vehicles. Both variants look similar, without obvious compression-induced detail degradation.

Figure 4 compares real QIS reconstructions at three photon levels. Under low photon levels, QBP leaves grainy backgrounds and surfaces; QBP-E suppresses grain but smooths faces and feet. QUIVEr, gQIR, and Bit2Bit improve visibility but also weaken these details to varying degrees. Ours balances denoising and detail recovery through clearer figurine contours and local structure against smooth backgrounds. Additional real QIS sequences are shown in Appendix D, Figures 11–14.

4.2.2 Quantitative Analysis

Figure 6 compares PSNR, SSIM, and LPIPS at five photon rates, averaging all reconstructed frames per video and then the 13 validation videos. Both variants outperform external methods at every rate; the compressed model exceeds the strongest external PSNR at each rate by 2.189–3.146 dB.

At $\alpha = 0.2$, the compressed model achieves 34.193 dB PSNR, 0.9259 SSIM, and 0.1128 LPIPS. PSNR gains over QBP, QUIVEr, gQIR, and Bit2Bit are 5.867, 2.979, 3.426, and 3.564 dB, respectively. Against Bit2Bit, the strongest external method in SSIM and LPIPS at this rate, SSIM improves by



Figure 3: Simulated QIS results. Ours-I: without compression; Ours-II: with compression.

0.0147 and LPIPS falls by 0.0181, extending accuracy gains to structural and perceptual quality (Appendix E, Table 1).

Transmission Payload and Processing Time. Figure 5(a)–(b) compares raw-QIS transmission with reconstruct–H.265 transmission. The queryable 4.628 MiB state reduces the raw 64.000 MiB payload by $13.8\times$; initialization plus one query takes 0.219 s, compared with 2.201, 1.594, and 1.947 s per frame for QBP, QUIVEr, and gQIR. In the H.265 workflow, Ours takes 0.382 s at the sender, while the baseline times are 17.802, 801.018, 978.015, and 36,073.064 s. Bit2Bit includes sample-specific optimization. Unlike the H.265 routes, our representation avoids materializing a fixed video at the sender and preserves receiver-selected timestamps after transmission. Because each method emits its native number of frames, these measurements compare end-to-end workflows rather than matched-output throughput.

4.3 Ablation Study

Component Ablation. Figure 7(a)–(c) compares the full model with two ablations. Compression reduces the state from 680 MiB to 4.628 MiB ($147\times$) while keeping absolute mean PSNR differences within 0.477 dB; the independently trained no-compression model has slightly lower LPIPS. Sender time increases by 0.020 s and receiver initialization by 0.052 s.

Removing residual correction lowers mean PSNR and SSIM at every photon rate. The full model gains 1.309–1.992 dB PSNR, peaking at $\alpha = 0.05$. The gap narrows at higher photon rates, indicating greater benefits under sparse photon evidence (Appendix G, Table 4).

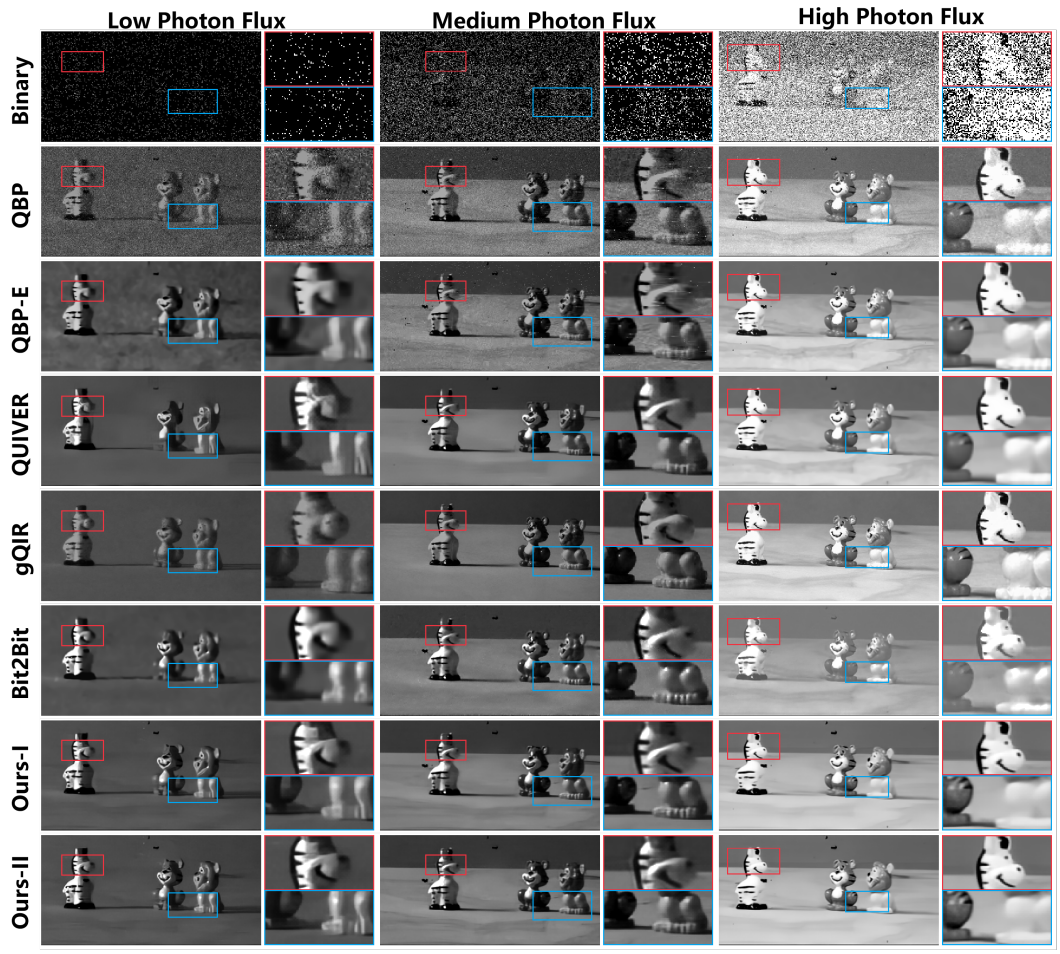

Figure 4: Real QIS reconstruction at estimated 0.03, 0.14, and 0.66 PPP. Ours-I/Ours-II query uncompressed/compressed states; both suppress noise and preserve figurine details.

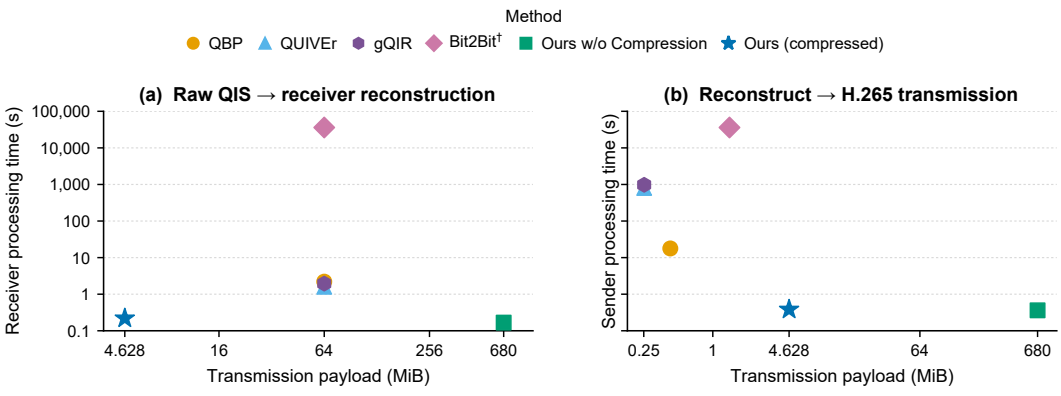

Figure 5: Payload versus receiver (a) and sender (b) processing time; see Tables 2 and 3.

5 Conclusion

Our measurement-to-latent formulation encodes long QIS sequences into a reusable state of scene evolution rather than fixed frames. Multi-scale anchors aggregate sparse photons; interpolation and residual refinement reconstruct receiver-selected timestamps.

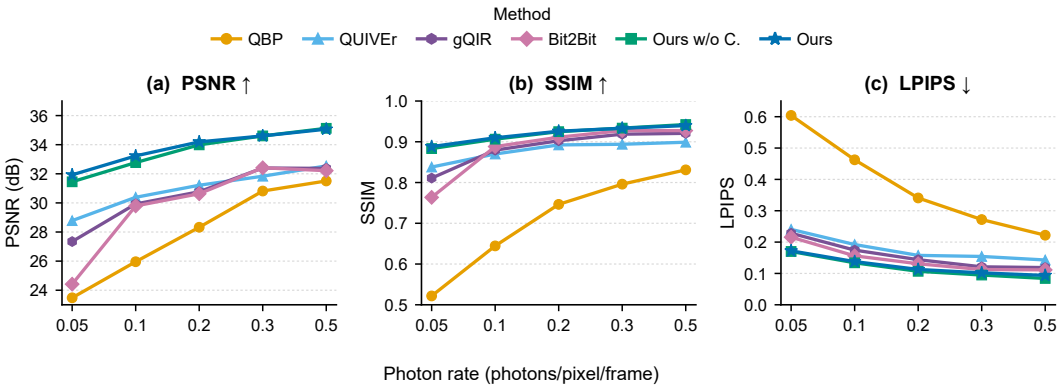

Figure 6: Video-level PSNR, SSIM, and LPIPS over five photon rates on 13 validation videos; full statistics appear in Supplementary Table 1.

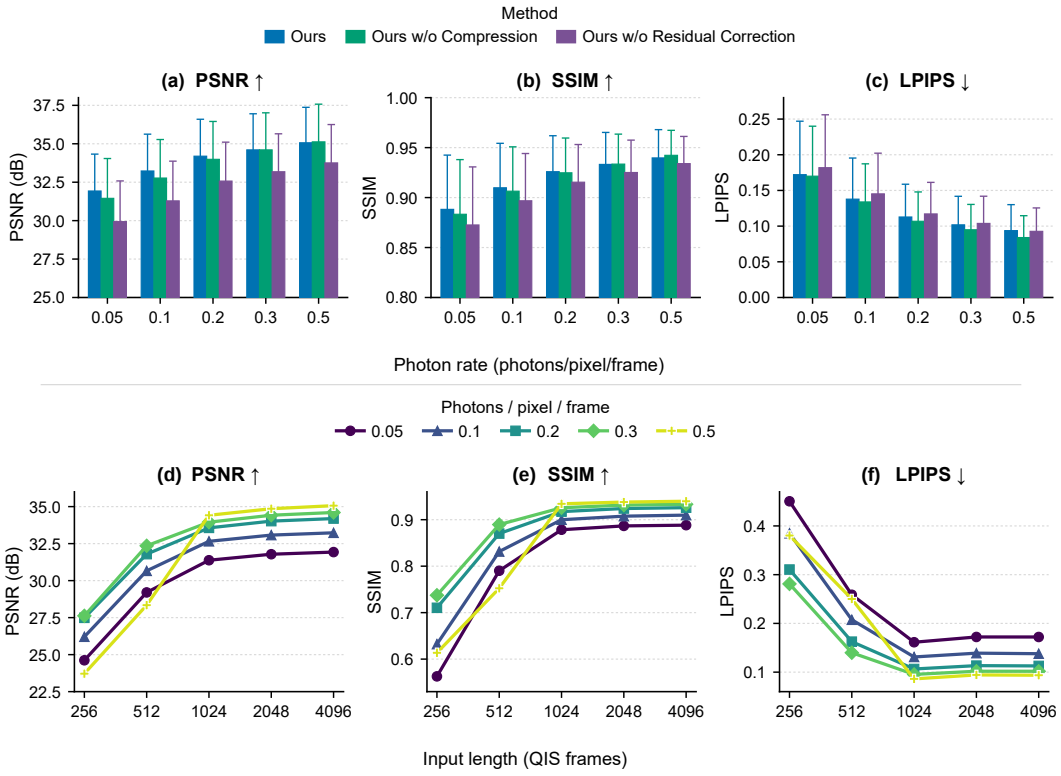

Figure 7: Model analysis on 13 validation videos: ablations (a)–(c) and input length (d)–(f). From 256 to 4096 frames, mean PSNR gains 7.874 dB and SSIM gains 0.2681.

Across five photon rates, both compressed and uncompressed variants outperform the evaluated QIS baselines, showing that the hierarchical representation remains effective after latent compression. Increasing the observation window from 256 to 4096 QIS frames improves mean PSNR by 7.874 dB and SSIM by 0.2681, demonstrating that the model can exploit photon evidence distributed over longer intervals. For a 4096-frame sequence, the compressed 4.628 MiB state reduces the bit-packed raw-QIS payload by $13.8\times$ while retaining receiver-selected timestamps and repeated queries without rerunning the encoder. By unifying long-range photon aggregation, reconstruction, state compression, and temporal access, our framework shifts QIS reconstruction from fixed frame outputs toward reusable measurement-derived latent representations.

AI USAGE STATEMENT

We used ChatGPT to assist with translation and language polishing. The authors reviewed and revised all AI-assisted content adopted in the manuscript and take full responsibility for its final content, including technical claims, experimental results, and figures.

REFERENCES

Stanley H. Chan. What does a one-bit quanta image sensor offer? *IEEE Transactions on Computational Imaging*, 8:770–783, 2022. doi: 10.1109/TCI.2022.3202012.

Stanley H. Chan and Yue M. Lu. Efficient image reconstruction for gigapixel quantum image sensors. In *GlobalSIP*, pp. 312–316, 2014. doi: 10.1109/GlobalSIP.2014.7032129.

Stanley H. Chan, Omar A. Elgendy, and Xiran Wang. Images from bits: Non-iterative image reconstruction for quanta image sensors. *Sensors*, 16(11):1961, 2016. doi: 10.3390/s16111961.

Hao Chen, Bo He, Hanyu Wang, Yixuan Ren, Ser-Nam Lim, and Abhinav Shrivastava. Nerv: Neural representations for videos. In *Advances in Neural Information Processing Systems*, volume 34, pp. 21557–21568, 2021.

Hao Chen, Matthew Gwilliam, Ser-Nam Lim, and Abhinav Shrivastava. Hnerv: A hybrid neural representation for videos. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10270–10279, 2023.

Shiyan Chen, Chaoteng Duan, Zhaofei Yu, Ruiqin Xiong, and Tiejun Huang. Self-supervised mutual learning for dynamic scene reconstruction of spiking camera. In *IJCAI*, pp. 2859–2866, 2022. doi: 10.24963/IJCAI.2022/396.

Prateek Chennuri, Yiheng Chi, Enze Jiang, G. M. Dilshan Godaliyadda, Abhiram Gnanasambandam, Hamid R. Sheikh, Istvan Gyongy, and Stanley H. Chan. Quanta video restoration. In *ECCV*, pp. 152–171, 2024. doi: 10.1007/978-3-031-73661-2_9.

Prateek Chennuri, Dongdong Fu, and Stanley H. Chan. Quanta diffusion. In *ICIP*, pp. 229–234, 2025. doi: 10.1109/ICIP55913.2025.11084414.

Yiheng Chi, Abhiram Gnanasambandam, Vladlen Koltun, and Stanley H. Chan. Dynamic low-light imaging with quanta image sensors. In *ECCV*, pp. 122–138, 2020. doi: 10.1007/978-3-030-58589-1_8.

Joon Hee Choi, Omar A. Elgendy, and Stanley H. Chan. Image reconstruction for quanta image sensors using deep neural networks. In *ICASSP*, pp. 6543–6547, 2018. doi: 10.1109/ICASSP.2018.8461685.

Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on Image Processing*, 16 (8):2080–2095, 2007. doi: 10.1109/TIP.2007.901238.

Yanchen Dong, Ruiqin Xiong, Rui Zhao, Xinfeng Zhang, and Tiejun Huang. Spike stream memory transfer for dynamic scene reconstruction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(5):3650–3658, 2026. doi: 10.1609/AAAI.V40I5.37364.

Omar A. Elgendy and Stanley H. Chan. Optimal threshold design for quanta image sensor. *IEEE Transactions on Computational Imaging*, 4(1):99–111, 2018. doi: 10.1109/TCI.2017.2781185.

Burak Ercan, Onur Eker, Canberk Saglam, Aykut Erdem, and Erkut Erdem. Hypere2vid: Improving event-based video reconstruction via hypernetworks. *IEEE Transactions on Image Processing*, 33: 1826–1837, 2024. doi: 10.1109/TIP.2024.3372460.

Bin Fan, Jiaoyang Yin, Yuchao Dai, Chao Xu, Tiejun Huang, and Boxin Shi. Spatio-temporal interactive learning for efficient image reconstruction of spiking cameras. In *NeurIPS*, volume 37, pp. 21401–21427, 2024. doi: 10.52202/079017-0675.

Eric R. Fossum. Modeling the performance of single-bit and multi-bit quanta image sensors. *IEEE Journal of the Electron Devices Society*, 1(9):166–174, 2013.

Eric R. Fossum, Jiaju Ma, Saleh Masoodian, Leo Anzagira, and Rachel Zizza. The quanta image sensor (QIS): Every photon counts. *Sensors*, 16(8):1260, 2016. doi: 10.3390/s16081260.

Aryan Garg, Sizhuo Ma, and Mohit Gupta. gqir: Generative quanta image reconstruction. *arXiv preprint arXiv:2602.20417*, 2026.

Atul Ingle, Andreas Velten, and Mohit Gupta. High flux passive imaging with single photon sensors. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Atul Ingle, Trevor Seets, M. Buttafava, Shantanu Gupta, Alberto Tosi, Andreas Velten, and Mohit Gupta. Passive inter-photon imaging. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Kiyotaka Iwabuchi, Tomohiro Yamazaki, and Takayuki Hamamoto. Iterative image reconstruction for quanta image sensor by using variance-based motion estimation. In *International Image Sensor Workshop (IISW)*, 2019.

Hanme Kim, Ankur Handa, Ryad Benosman, Sio-Hoi Ieng, and Andrew J. Davison. Simultaneous mosaicing and tracking with an event camera. In *BMVC*, pp. 26.1–26.12, 2014. doi: 10.5244/C.28.26.

Ho Man Kwan, Ge Gao, Fan Zhang, Andrew Gower, and David Bull. Hinerv: Video compression with hierarchical encoding-based neural representation. In *Advances in Neural Information Processing Systems*, volume 36, 2024.

Joo Chan Lee, Daniel Rho, Jong Hwan Ko, and Eunbyung Park. Ffnerv: Flow-guided frame-wise neural representations for videos. In *ACM International Conference on Multimedia*, pp. 7859–7870, 2023.

Jiahao Li, Bin Li, and Yan Lu. Deep contextual video compression. In *Advances in Neural Information Processing Systems*, volume 34, pp. 18114–18125, 2021.

Jiahao Li, Bin Li, and Yan Lu. Neural video compression with diverse contexts. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22616–22626, 2023.

Jiahao Li, Bin Li, and Yan Lu. Neural video compression with feature modulation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26099–26108, 2024.

Yehe Liu, Alexander Krull, Hector Basevi, Aleš Leonardis, and Michael W. Jenkins. bit2bit: 1-bit quanta video reconstruction via self-supervised photon prediction. In *NeurIPS*, volume 37, pp. 88443–88485, 2024. doi: 10.52202/079017-2806.

Guo Lu, Wanli Ouyang, Dong Xu, Xiaoyun Zhang, Chunlei Cai, and Zhiyong Gao. DVC: An end-to-end deep video compression framework. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11006–11015, 2019.

Jia Ma, Stanley Chan, and Eric R. Fossum. Review of quanta image sensors for ultralow-light imaging. *IEEE Transactions on Electron Devices*, 69(6):2824–2839, 2022. doi: 10.1109/TED.2022.3166716.

Sizhuo Ma, Shantanu Gupta, Arin C. Ulku, Claudio Bruschini, Edoardo Charbon, and Mohit Gupta. Quanta burst photography. *ACM Transactions on Graphics*, 39(4):133:1–133:16, 2020. doi: 10.1145/3386569.3392470.

Vishal Purohit, Junjie Luo, Yiheng Chi, Qi Guo, Stanley H. Chan, and Qiang Qiu. Generative quanta color imaging. In *CVPR*, pp. 25138–25148, 2024. doi: 10.1109/CVPR52733.2024.02375.

Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. Events-to-video: Bringing modern computer vision to event cameras. In *CVPR*, pp. 3852–3861, 2019. doi: 10.1109/CVPR.2019.00398.

Tal Remez, Or Litany, and Alex Bronstein. A picture is worth a billion bits: Real-time image reconstruction from dense binary threshold pixels. In *IEEE International Conference on Computational Photography (ICCP)*, 2016. doi: 10.1109/ICCPHOT.2016.7492874.

Trevor Seets, Atul Ingle, Martin Laurenzis, and Andreas Velten. Motion adaptive deblurring with single-photon cameras. In *WACV*, pp. 1944–1954, 2021.

Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. In *Advances in Neural Information Processing Systems*, volume 33, pp. 7462–7473, 2020.

Gary J. Sullivan, Jens-Rainer Ohm, Woo-Jin Han, and Thomas Wiegand. Overview of the high efficiency video coding (HEVC) standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 22(12):1649–1668, 2012.

Lv Tang, Jun Zhu, Xinfeng Zhang, Li Zhang, Siwei Ma, and Qingming Huang. Canerv: Content adaptive neural representation for video compression. *arXiv preprint arXiv:2502.06181*, 2025.

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.

Wenming Weng, Yueyi Zhang, and Zhiwei Xiong. Event-based video reconstruction using transformer. In *ICCV*, pp. 2543–2552, 2021. doi: 10.1109/ICCV48922.2021.00256.

Feng Yang, Yue M. Lu, Luciano Sbaiz, and Martin Vetterli. An optimal algorithm for reconstructing images from binary measurements. In *Computational Imaging VIII*, volume 7533, pp. 75330K, 2010. doi: 10.1117/12.850887.

Feng Yang, Yue M. Lu, Luciano Sbaiz, and Martin Vetterli. Bits from photons: Oversampled image acquisition using binary poisson statistics. *IEEE Transactions on Image Processing*, 21(4): 1421–1436, 2012. doi: 10.1109/TIP.2011.2179306.

Jiyuan Zhang, Shanshan Jia, Zhaofei Yu, and Tiejun Huang. Learning temporal-ordered representation for spike streams based on discrete wavelet transforms. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(1):137–147, 2023. doi: 10.1609/AAAI.V37I1.25085.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, pp. 586–595, 2018. doi: 10.1109/CVPR.2018.00068.

Tianyi Zhang, Matthew Dutson, Vivek Boominathan, Mohit Gupta, and Ashok Veeraraghavan. Streaming quanta sensors for online, high-performance imaging and vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(3):1564–1577, 2025.

Xinjie Zhang, Ren Yang, Dailan He, Xingtong Ge, Tongda Xu, Yan Wang, Hongwei Qin, and Jun Zhang. Boosting neural representations for videos with a conditional decoder. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2556–2566, 2024.

Jing Zhao, Ruiqin Xiong, and Tiejun Huang. High-speed motion scene reconstruction for spike camera via motion aligned filtering. In *ISCAS*, pp. 1–5, 2020. doi: 10.1109/ISCAS45731.2020.9181055.

Jing Zhao, Ruiqin Xiong, Hangfan Liu, Jian Zhang, and Tiejun Huang. Spk2imgnet: Learning to reconstruct dynamic scene from continuous spike stream. In *CVPR*, pp. 11991–12000, 2021. doi: 10.1109/CVPR46437.2021.01182.

Rui Zhao, Ruiqin Xiong, Jian Zhang, Zhaofei Yu, Shuyuan Zhu, Lei Ma, and Tiejun Huang. Spike camera image reconstruction using deep spiking neural networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 34(6):5207–5212, 2024. doi: 10.1109/TCSVT.2023.3326294.

Yajing Zheng, Lingxiao Zheng, Zhaofei Yu, Boxin Shi, Yonghong Tian, and Tiejun Huang. High-speed image reconstruction through short-term plasticity for spiking cameras. In *CVPR*, pp. 6354–6363, 2021. doi: 10.1109/CVPR46437.2021.00629.

Lin Zhu, Siwei Dong, Tiejun Huang, and Yonghong Tian. A retina-inspired sampling method for visual texture reconstruction. In *ICME*, pp. 1432–1437, 2019. doi: 10.1109/ICME.2019.00248.

A Detailed Reconstruction Architecture

Figure 8 expands the Hierarchical Long-Range Encoder $\mathcal{E}_\theta$ and Shared Reconstruction Decoder $\mathcal{D}_\psi$ in Figure 1 and Sections 3.2–3.3. The implementation instantiates the general L-scale hierarchy with $L = 4$, indexed from coarse to fine: $A^3 = A^{L-1}$ has the highest spatial resolution, before the first spatial downsampling operation, whereas A^2, A^1, and A^0 follow one, two, and three such operations. The branches denote anchor sequences $A^\ell = \mathcal{P}_\ell(F^\ell) = \{a_i^\ell\}$; scale-specific projections are omitted for clarity. Input Preparation groups the Temporal Binning shown separately in Figure 1 with input projection (3×3 convolution, GroupNorm, and GELU).

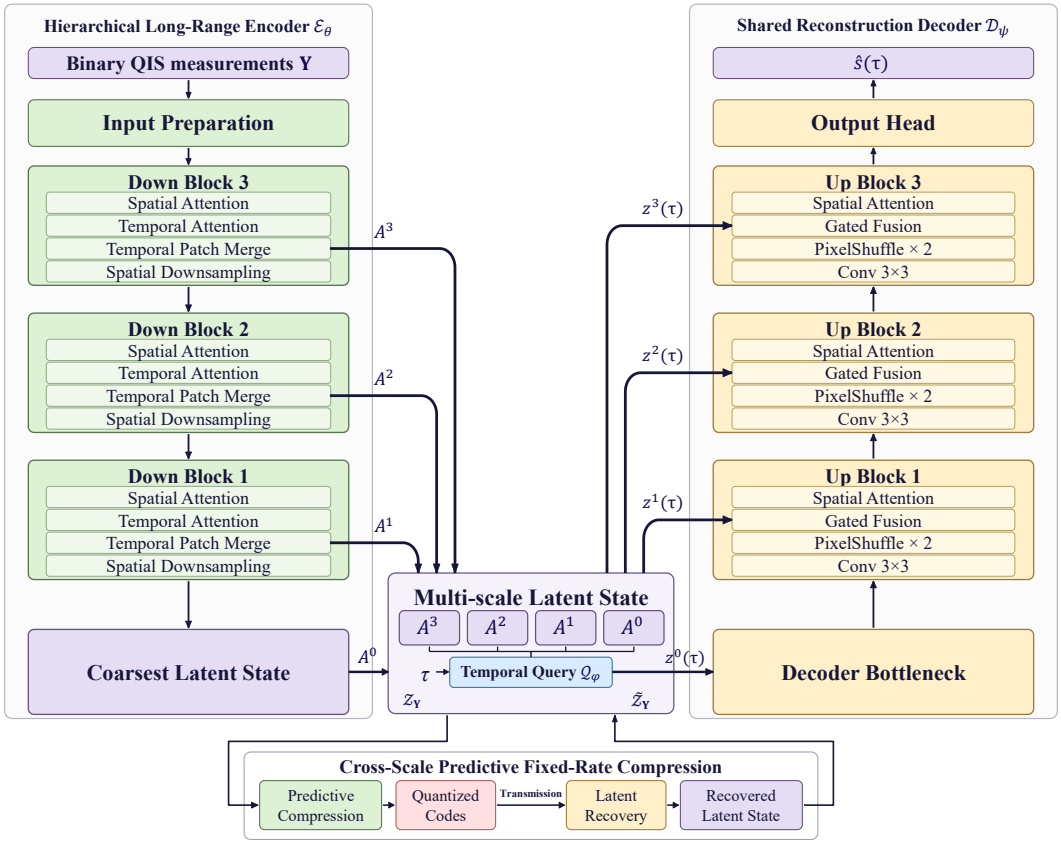

Figure 8: Reconstruction network underlying Figure 1. Green denotes encoding, yellow decoding, purple latent states, blue Temporal Query $\mathcal{Q}_\phi$, and pink quantized codes. The query module interpolates anchors and applies residual refinement $\mathcal{R}_\phi$, yielding $z^\ell(\tau)$. The coarsest query enters Decoder Bottleneck; three Up Blocks then fuse the finer queries at matching spatial resolutions. Figure 9 expands the bottom Predictive Compression and Latent Recovery pathway.

The complete state is $\mathcal{Z}_\mathbf{Y} = \{A^\ell\}_{\ell=0}^3$. Predictive Compression produces Quantized Codes (INT8 coefficients with FP16 scaling factors); Latent Recovery reconstructs $\widetilde{\mathcal{Z}}_\mathbf{Y} = \{\widetilde{A}^\ell\}_{\ell=0}^3$ in coarse-to-fine order. Temporal Query accepts either the original state or this recovered state and returns queried features $\{z^\ell(\tau)\}$. Thus, A^ℓ denotes an anchor sequence, a_i^ℓ one stored anchor, and $z^\ell(\tau)$ the feature evaluated at a requested time. Receiver-side reconstruction evaluates $\mathcal{D}_\psi(\mathcal{Q}_\phi(\widetilde{\mathcal{Z}}_\mathbf{Y}, \tau))$ without rerunning the encoder.

B Detailed Latent Compression Architecture

Figure 9 expands the Predictive Compression and Latent Recovery pathway in Figures 1 and 8, following Section 3.4. It receives the same four-scale state $\mathcal{Z}_{\mathbf{Y}} = \{A^\ell\}_{\ell=0}^{3}$. Moving from A^0 to A^3 increases temporal anchor density and spatial resolution while reducing channels. The codec transforms these states into compact variables c^ℓ and recovers $\widetilde{A}^\ell$ from dequantized codes $\widetilde{c}^\ell$. Its latent decoders $\mathcal{C}_\ell^{\mathrm{dec}}$ restore features; the separate reconstruction decoder $\mathcal{D}_\psi$ forms images only after Temporal Query evaluates the recovered hierarchy.

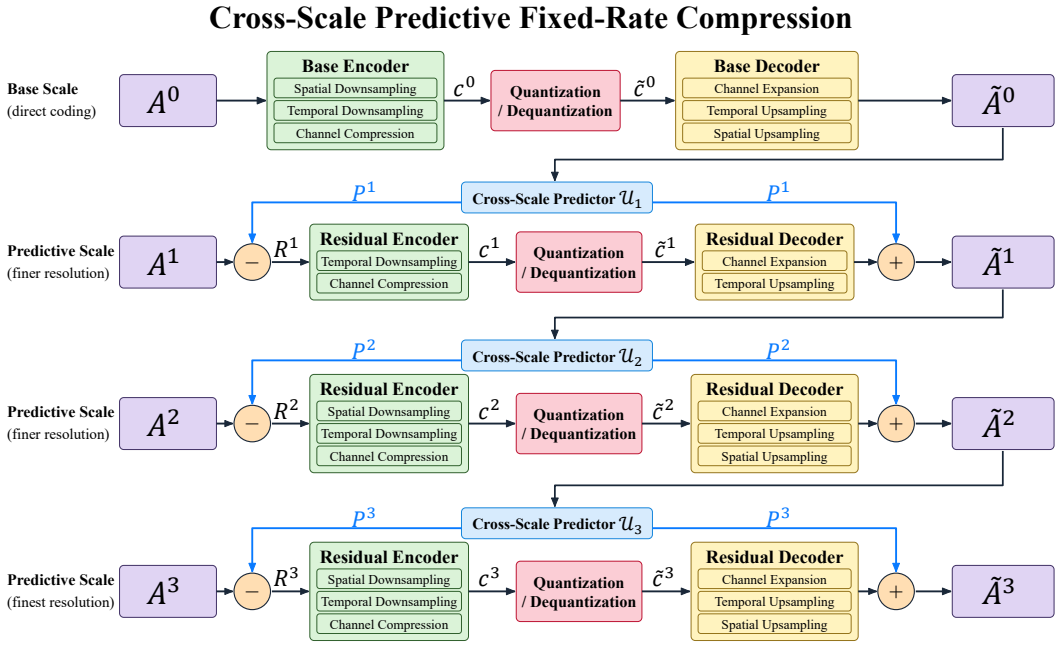

Figure 9: Four-scale expansion of the latent codec in Figure 8. Green/yellow blocks encode/decode latent features; blue paths carry cross-scale predictions. Subtraction and addition isolate and restore fine-scale residuals. Pink blocks summarize quantization, transmission, and dequantization. Figure 10 details the encoder, decoder, and predictor blocks.

Predictive encoding and recovery. The Base Encoder directly transforms A^0 into c^0, and the Base Decoder recovers $\widetilde{A}^0$ from its dequantized code. For each finer scale, the blue predictor computes $P^\ell = \mathcal{U}_\ell(\widetilde{A}^{\ell-1})$. The subtraction node forms $R^\ell = A^\ell - P^\ell$, and the Residual Encoder $\mathcal{C}_\ell^{\mathrm{enc}}$ compresses this difference rather than the complete state. At recovery, the Residual Decoder $\mathcal{C}_\ell^{\mathrm{dec}}$ reconstructs the difference, which the addition node combines with P^ℓ to obtain $\widetilde{A}^\ell$. Both sides predict from the recovered preceding scale, so the receiver can reproduce the prediction without access to the original anchors. The black return paths carry these recovered states to the next predictor; the two blue branches carry the same prediction to subtraction and addition.

Quantization and transmission. Each pink block summarizes three steps: sender-side quantization, transmission, and receiver-side dequantization. For each temporal position and channel of c^ℓ, a symmetric quantization scale is determined from the maximum absolute coefficient over spatial positions. Coefficients are divided by this scale, rounded, and clipped to the signed range $[-127, 127]$. The payload stores the resulting INT8 coefficients and FP16 scales. Multiplying received coefficients by their scales yields $\widetilde{c}^\ell$, the input to the corresponding latent decoder. No entropy model or entropy coding is used. After coarse-to-fine recovery, the receiver reuses $\widetilde{\mathcal{Z}}_{\mathbf{Y}}$ for arbitrary-time queries without accessing the raw QIS measurements or rerunning the long-range encoder.

Resolution-changing modules. Figure 10 details the blocks in the main flow. Spatial Downsampling first applies PixelUnshuffle, which rearranges each local spatial neighborhood into channels

without itself discarding values. A pointwise convolution then mixes and reduces these channels, followed by GroupNorm, GELU, and a Spatial Residual Block. Spatial Upsampling applies a Spatial Residual Block, expands channels by pointwise convolution, rearranges channels back into spatial positions with PixelShuffle, and refines the result with another Spatial Residual Block. Temporal Downsampling uses a temporally strided convolution to reduce the number of latent time positions while preserving spatial resolution, followed by GroupNorm, GELU, and a Factorized Spatio-Temporal Residual Block. Temporal Upsampling interpolates the temporal dimension, applies that residual block, and finishes with a temporal convolution. These learned recovery operations restore feature dimensions, not an exact inverse of lossy compression. Scale A^1 omits spatial downsampling and upsampling; the other scales include them. Each stage label summarizes its function rather than a fixed repetition count.

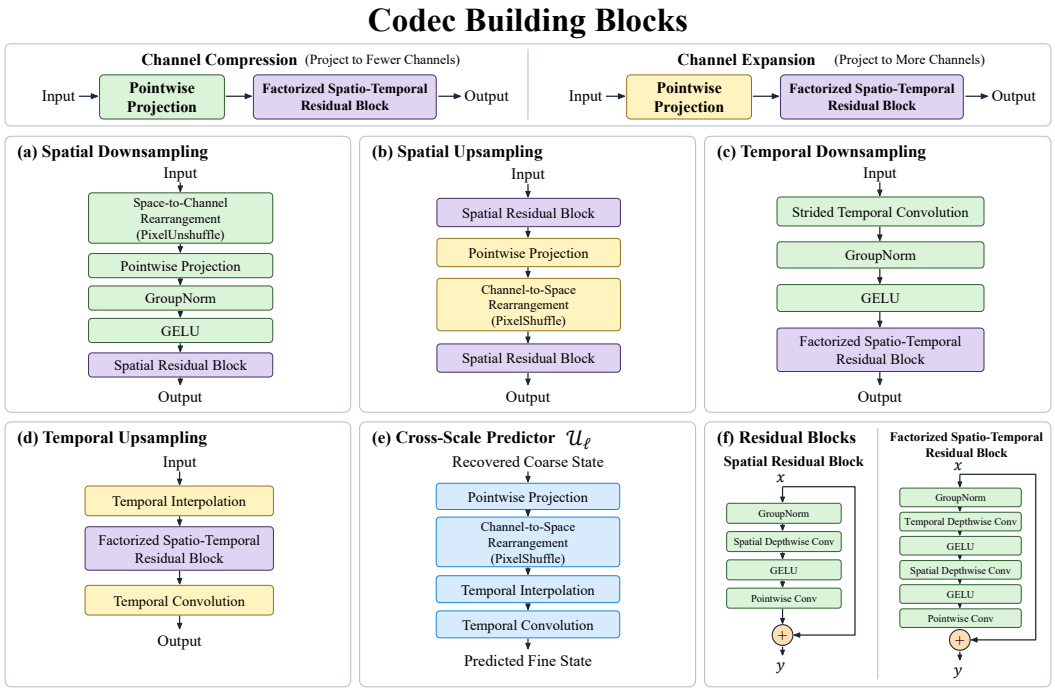

Figure 10: Codec building blocks: channel transforms (top), spatial and temporal resolution changes (a–d), cross-scale prediction (e), and residual blocks with additive bypasses (f). Arrows indicate operation order. Green, yellow, lavender, and blue distinguish encoding operations, decoding operations, residual blocks, and prediction, respectively.

Channel transforms, prediction, and residual blocks. The top-row Channel Compression and Channel Expansion modules combine Pointwise Projection with a Factorized Spatio-Temporal Residual Block. In panels (a), (b), and (e), Pointwise Projection denotes only the channel-mixing convolution. The predictor $\mathcal{U}_\ell$ follows projection with PixelShuffle, temporal interpolation, and temporal convolution to match A^ℓ. The Spatial Residual Block applies GroupNorm, spatial depthwise convolution, GELU, and pointwise convolution. The factorized block applies GroupNorm, temporal depthwise convolution, GELU, spatial depthwise convolution, GELU, and pointwise convolution. Depthwise convolutions filter channels separately; pointwise convolutions mix them. Both blocks use the additive bypass in panel (f), distinct from coding residual R^ℓ and query correction $\mathcal{R}_\phi$.

C Measurement Simulation and Training Details

This section supplements the settings in Section 4. Input length, crop probabilities, and query composition stay fixed during main-model training. Only the photon-level sampling distribution follows a curriculum, separate from staged network optimization.

C.1 Simulated Binary Measurements

We read clean reference videos as grayscale eight-bit images and normalize them as $x_t = I_t/255 \in [0, 1]$. A single photon level α is sampled for each training example and held constant throughout its 4096 frames. The ideal signal rate is $\lambda_t^{\mathrm{sig}} = \alpha x_t$. To model sensor defects, let M_{h} and M_{d} be non-overlapping binary masks for hot and dead pixels. With spatial indices omitted, binary measurements are generated as

$$\widetilde{Y}_t \sim \mathrm{Bernoulli}\left(1 - \exp[-(\alpha x_t + M_{\mathrm{h}}\lambda^{\mathrm{hot}})]\right), \qquad Y_t = (1 - M_{\mathrm{d}})\widetilde{Y}_t. \tag{9}$$

Each mask occupies 0.5% of the spatial positions. Defect locations are fixed over time within a sample and independently resampled for different samples. Dead pixels are stuck off: their measurements are forced to zero after binary sampling. At hot pixels, dark-count probabilities p_{hot} are drawn from a four-component mixture with the probability ranges and selection weights below. Their equivalent additional photon rates are $\lambda^{\mathrm{hot}} = -\log(1 - p_{\mathrm{hot}})$.

Dark-count probability range	$[0.15, 0.21]$	$[0.21, 0.38]$	$[0.38, 0.67]$	$[0.67, 0.90]$
Component weight	0.2	0.3	0.3	0.2

Ordered temporal binning. After binary sampling, every 32 consecutive bit-planes form one temporal chunk. Each chunk is divided into four consecutive eight-frame bins, and the measurements within each bin are averaged. The encoder receives four-channel bin images rather than 32 concatenated bit-planes, yielding 128 temporal chunks for a 4096-frame input. Averaging aggregates local photon evidence while channel order preserves coarse within-chunk timing. This implements Temporal Binning in Figure 1, grouped into Input Preparation in Figure 8.

C.2 Photon-Level Curriculum and Augmentation

The photon-level curriculum progressively expands the training range toward sparser measurements. At main-model training step s, define

$$p(s) = \mathrm{clip}\Big(\frac{s}{30000}, 0, 1\Big), \qquad \alpha_{\mathrm{floor}}(s) = 0.5\left(\frac{0.01}{0.5}\right)^{p(s)}. \tag{10}$$

With probability 0.8, α is sampled log-uniformly from $[\alpha_{\mathrm{floor}}(s), 0.5]$; with probability 0.2, it is sampled log-uniformly from the full range $[0.01, 0.5]$. The progressive branch starts at $\alpha = 0.5$, while the full-range branch exposes the model to low-photon measurements from the beginning. After 30,000 steps, both branches sample the full range.

Spatial augmentation uses 64×128 crops with probability 0.8 and 128×128 crops with probability 0.2, together with horizontal and vertical flips and 90° rotations. Temporal reversal is not used. The crop distribution, 4096-frame input length, and supervised query composition remain fixed throughout training; only photon-level sampling follows a curriculum.

C.3 Temporal Coordinates, Queries, and Supervision

Time-aligned latent anchors. We express anchors, queries, and supervision on the same normalized time axis. For a T-frame sequence indexed from 0 to $T-1$, an anchor associated with a consecutive d_ℓ-frame interval is located at its center:

$$t_i^\ell = \frac{i d_\ell + (d_\ell - 1)/2}{T - 1}. \tag{11}$$

The 16 coarsest-scale anchors correspond to 256-frame intervals, and the 32 finer-scale anchors used for supervised anchor queries correspond to 128-frame intervals. Equation 3 uses these interval centers to align training and inference in time.

Fixed query composition. Each training sample uses 80 queries: the 16 coarsest-scale anchor centers, the same 32 finer-scale anchor centers, and 32 stratified random queries. For the random component, $[0, 1]$ is divided into 32 equal subintervals and one time is sampled in each. If a rounded random query duplicates an already selected anchor supervision frame, the sampler attempts to select another frame. This fixed mixture supervises discrete supports and samples the intervals between them.

Reference-frame supervision. For a query τ, the clean target is selected as

$$j(\tau) = \mathrm{round}[\tau(T-1)], \qquad x^\star(\tau) = x_{j(\tau)}. \tag{12}$$

Supervision frames are selected directly from pre-generated high-frame-rate reference videos, without online interpolation, optical-flow warping, or averaging of adjacent frames during training. Thus, continuous-time queries are supervised by the nearest available reference frames, as defined in Equation 12. The reconstruction loss and weights are given in Sections 3.5 and 4.

Boundary queries. Inside the anchor range, the query selects the pair bracketing τ at each scale. Before the first anchor it uses the first two anchors; after the last anchor it uses the last two. The same Equation 3 is evaluated with an unclipped β at these boundaries, giving linear extrapolation followed by the same learned residual refinement. This extrapolation covers only the boundary portions of the observed interval; all requested times remain within $[0, 1]$.

C.4 Staged Optimization

Training begins with the reconstruction model alone to establish a queryable latent representation. The latent codec is then introduced and trained to compress and recover this state, followed by joint optimization of reconstruction and compression. The corresponding architectures are detailed in Appendices A and B. These optimization stages concern the network components being trained, whereas the photon-level curriculum controls the distribution of input measurements.

D ADDITIONAL REAL QIS RESULTS

Figures 11–14 show the full model, including latent compression and recovery, applied directly to real QIS recordings without real-data fine-tuning. The network is trained on simulated measurements. These sequences complement the main-text comparisons with rotating textured objects, camera motion, and changing target orientation. Each figure places raw binary measurements above the reconstructed sequence.

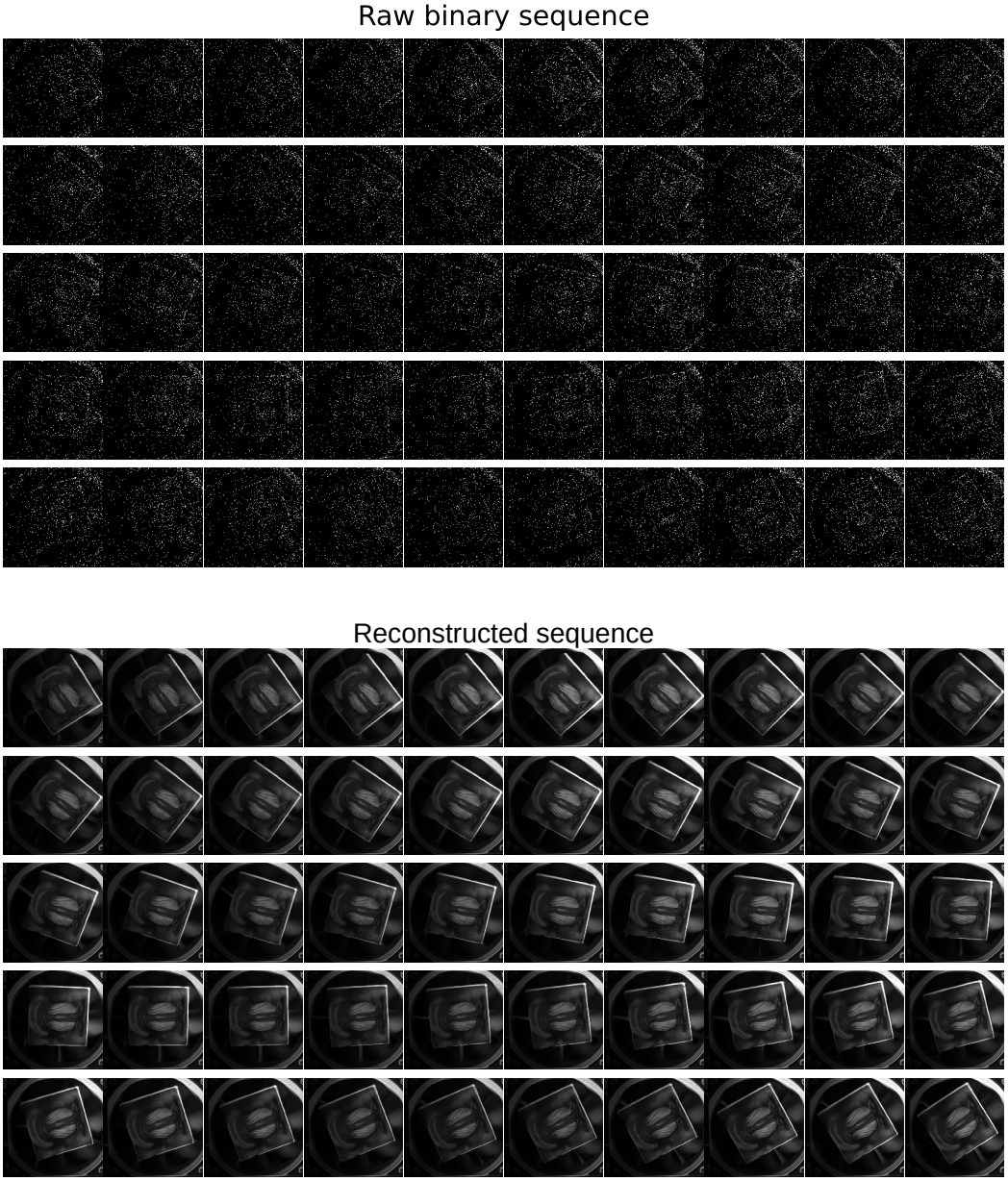

Figure 11: Real QIS reconstruction of an image rotating on a CPU fan. Top: raw binary sequence. Bottom: full-model reconstruction. The recovered image content and its boundary remain visible as their orientation changes; the raw bit-planes contain only sparse evidence of this structure.

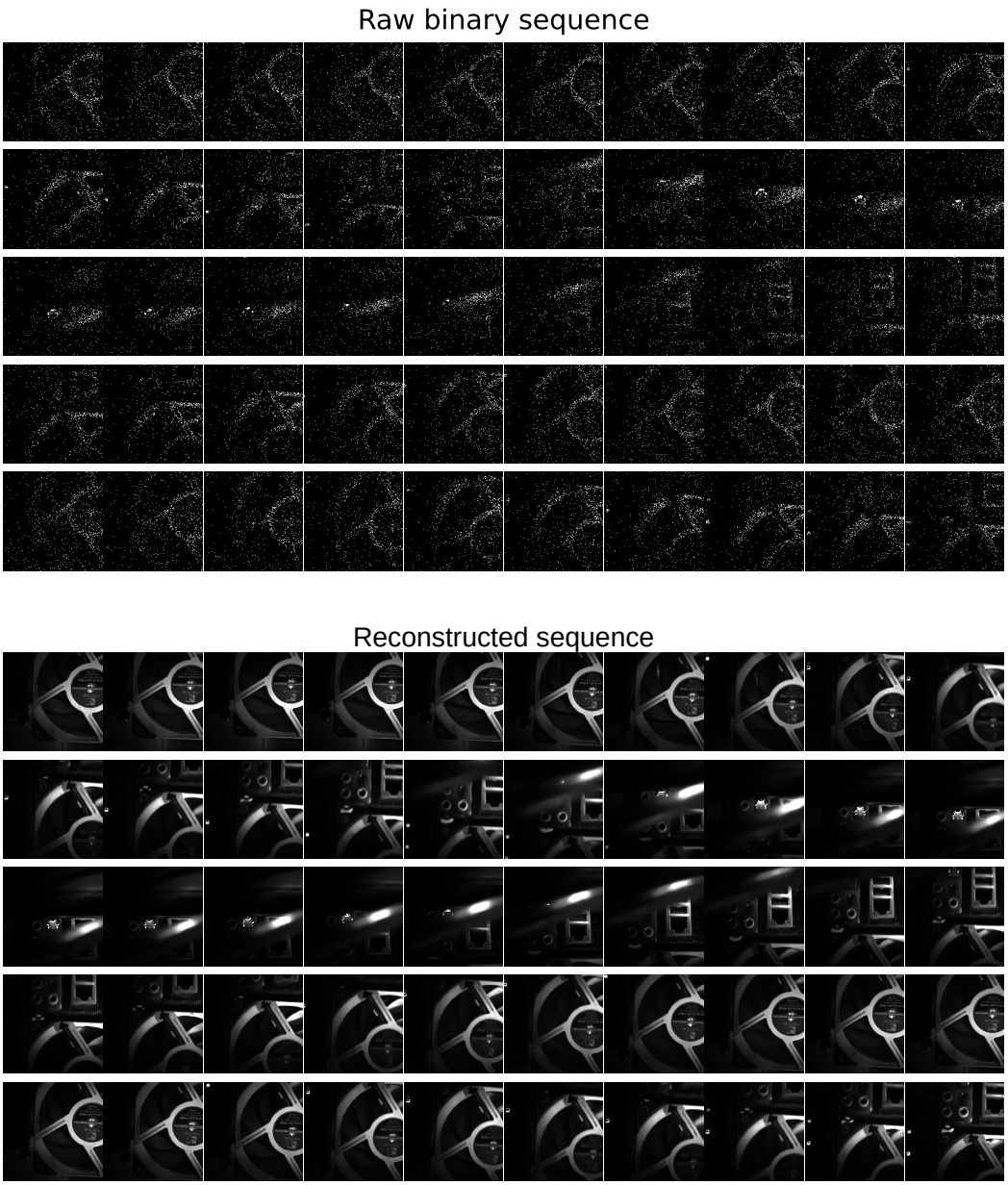

Figure 12: Real QIS reconstruction of a CPU fan with camera motion. The view shifts between the fan and surrounding hardware. Reconstruction reveals the fan rim, central hub, and background structures as they move through the field of view. Raw measurements are above the reconstructed sequence, as in Figure 11.

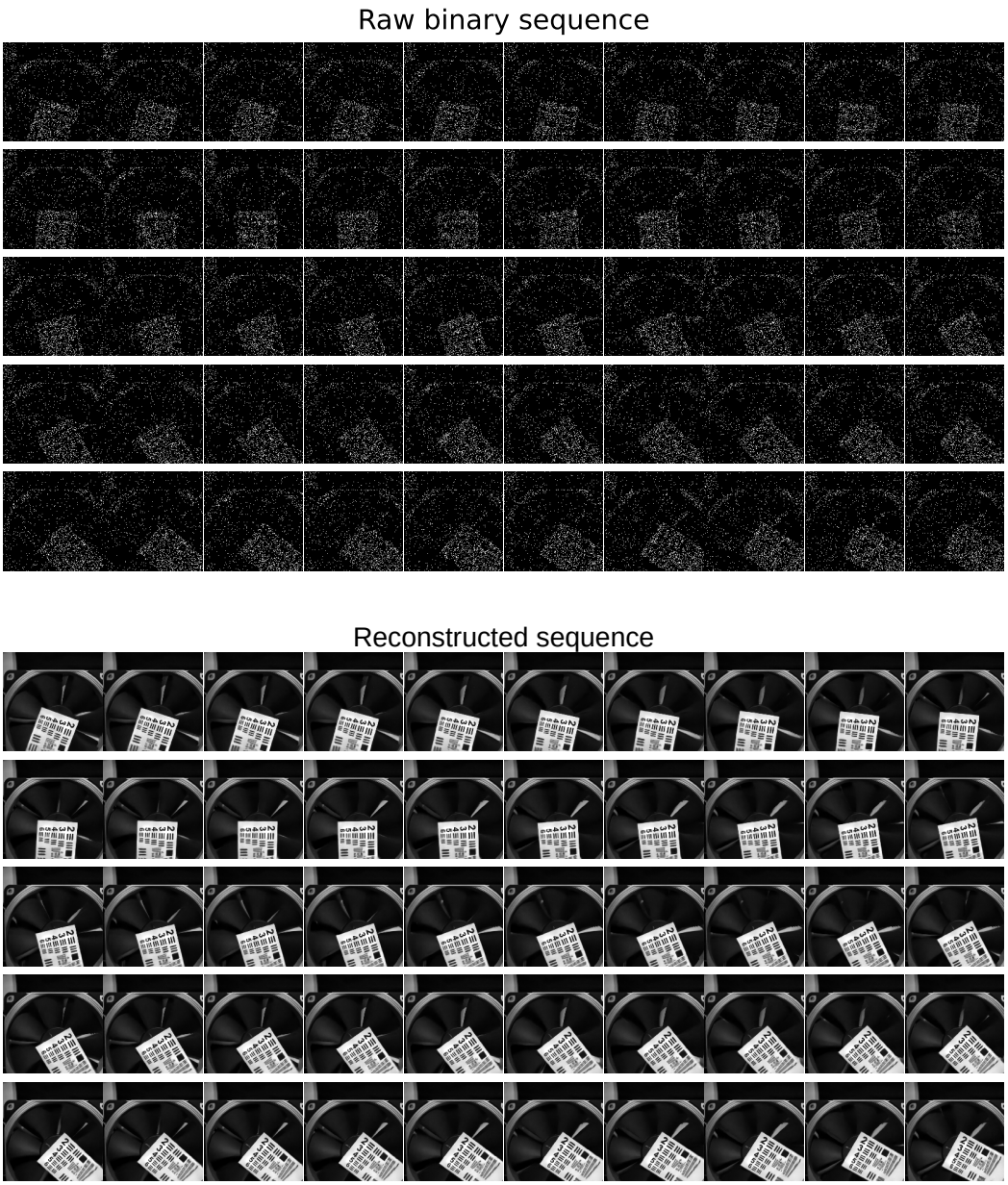

Figure 13: Real QIS reconstruction of a resolution target rotating on a CPU fan. The target's outer boundary and internal high-contrast patterns remain distinguishable at different orientations, allowing inspection of spatial detail during rotation. The layout follows Figure 11.

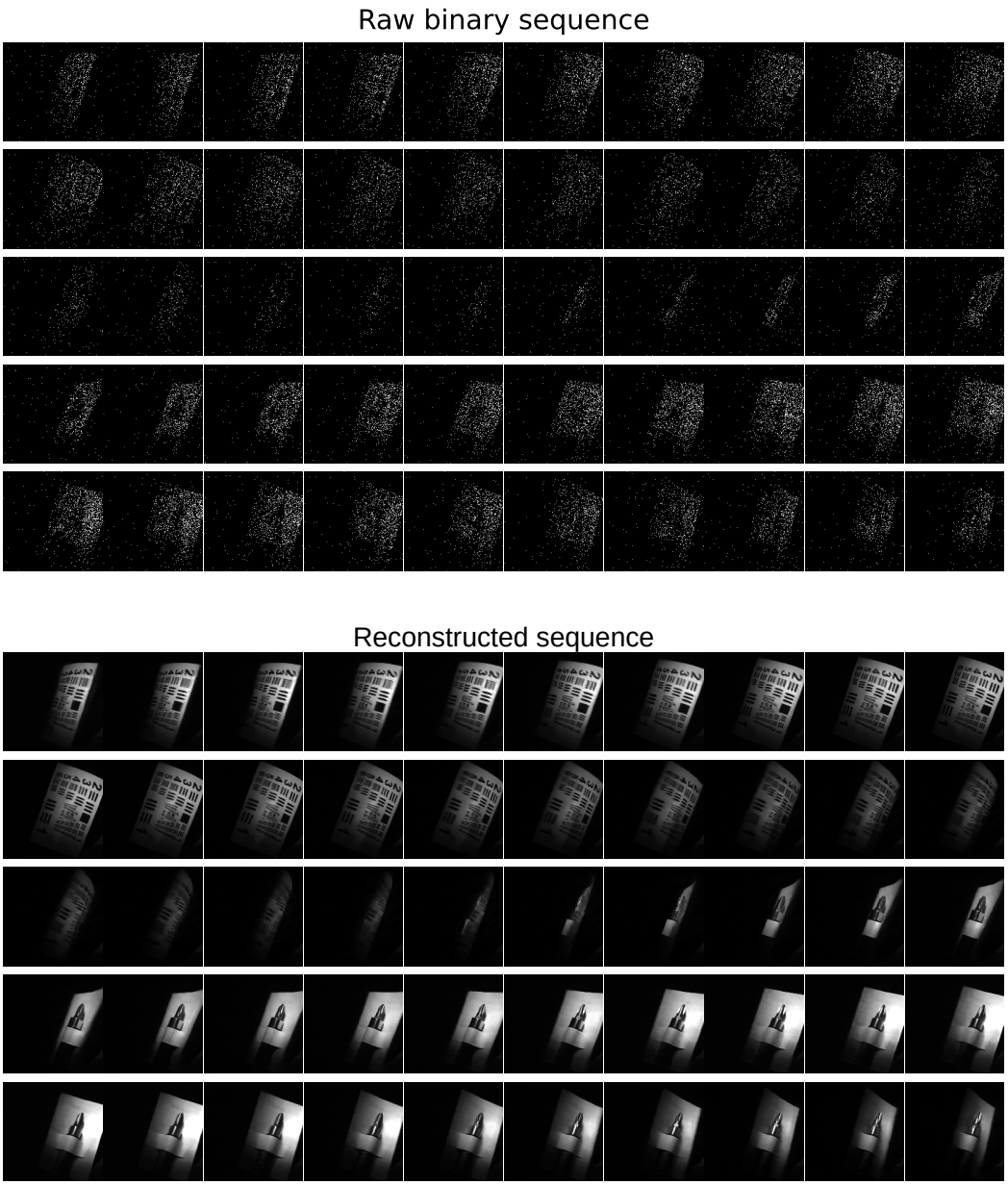

Figure 14: Real QIS reconstruction of a resolution target rotating on a drill bit. The target turns from a broad visible face to a narrow edge and exposes its opposite side. The reconstructed sequence makes these orientation and visibility changes apparent. The layout follows Figure 11.

E VIDEO RECONSTRUCTION QUALITY: FULL NUMERICAL RESULTS

Table 1 lists the values underlying Figure 6. Each photon-rate group compares four external baselines with the compressed and uncompressed variants of our method. Higher PSNR and SSIM and lower LPIPS indicate better reconstruction. Metrics are averaged within each video before aggregation across the 13 validation videos, so the reported standard deviation describes variation between videos rather than between repeated training runs. Both variants outperform the external baselines, while their relative ranking varies with the metric and photon level. The comparison therefore separates reconstruction quality from the payload reduction achieved by compression.

Table 1: Average video reconstruction quality on the 13 validation videos. Metrics are first averaged over all reconstructed frames in each video; each entry reports the mean $\pm$ population standard deviation of the resulting video-level scores.

Photon rate	Method	PSNR (dB) $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
0.05	QBP	23.480 ± 1.481	0.5217 ± 0.0661	0.6041 ± 0.0820
	QUIVEr	28.782 ± 1.951	0.8380 ± 0.0430	0.2405 ± 0.0778
	gQIR	27.344 ± 1.723	0.8109 ± 0.0387	0.2283 ± 0.0693
	Bit2Bit	24.417 ± 1.414	0.7636 ± 0.0325	0.2156 ± 0.0631
	Ours w/o Compression	31.451 ± 2.590	0.8833 ± 0.0547	0.1700 ± 0.0700
	Ours (compressed)	31.928 ± 2.399	0.8882 ± 0.0543	0.1721 ± 0.0749
0.1	QBP	25.958 ± 1.502	0.6445 ± 0.0559	0.4625 ± 0.0714
	QUIVEr	30.385 ± 1.922	0.8700 ± 0.0352	0.1922 ± 0.0669
	gQIR	29.929 ± 1.653	0.8795 ± 0.0334	0.1744 ± 0.0592
	Bit2Bit	29.793 ± 1.271	0.8883 ± 0.0311	0.1561 ± 0.0520
	Ours w/o Compression	32.771 ± 2.506	0.9064 ± 0.0444	0.1339 ± 0.0534
	Ours (compressed)	33.230 ± 2.395	0.9098 ± 0.0445	0.1378 ± 0.0576
0.2	QBP	28.326 ± 1.740	0.7464 ± 0.0432	0.3408 ± 0.0635
	QUIVEr	31.214 ± 1.709	0.8923 ± 0.0288	0.1576 ± 0.0574
	gQIR	30.767 ± 1.574	0.9025 ± 0.0297	0.1439 ± 0.0503
	Bit2Bit	30.629 ± 1.324	0.9112 ± 0.0292	0.1309 ± 0.0431
	Ours w/o Compression	33.989 ± 2.463	0.9248 ± 0.0348	0.1068 ± 0.0411
	Ours (compressed)	34.193 ± 2.402	0.9259 ± 0.0360	0.1128 ± 0.0459
0.3	QBP	30.816 ± 1.973	0.7960 ± 0.0363	0.2719 ± 0.0562
	QUIVEr	31.834 ± 1.986	0.8939 ± 0.0235	0.1542 ± 0.0572
	gQIR	32.399 ± 1.684	0.9186 ± 0.0273	0.1211 ± 0.0442
	Bit2Bit	32.414 ± 1.535	0.9274 ± 0.0276	0.1124 ± 0.0365
	Ours w/o Compression	34.603 ± 2.411	0.9335 ± 0.0300	0.0949 ± 0.0355
	Ours (compressed)	34.603 ± 2.349	0.9332 ± 0.0320	0.1018 ± 0.0400
0.5	QBP	31.508 ± 2.328	0.8311 ± 0.0283	0.2219 ± 0.0483
	QUIVEr	32.541 ± 2.181	0.8989 ± 0.0475	0.1429 ± 0.0302
	gQIR	32.371 ± 2.417	0.9207 ± 0.0342	0.1187 ± 0.0313
	Bit2Bit	32.214 ± 2.752	0.9274 ± 0.0260	0.1113 ± 0.0311
	Ours w/o Compression	35.125 ± 2.451	0.9423 ± 0.0250	0.0840 ± 0.0306
	Ours (compressed)	35.065 ± 2.307	0.9397 ± 0.0283	0.0937 ± 0.0364

F Detailed System Timing and Payload

Tables 2 and 3 separate one-time sender processing, one-time receiver initialization for our method, and repeated per-frame queries. In Figure 5(a), our receiver cost includes initialization plus one query: $0.189 + 0.03006 = 0.21906$ s for the compressed model and $0.137 + 0.03006 = 0.16706$ s without compression. After initialization, each additional query costs 30.06 ms; this measurement excludes reading or writing state files (I/O). For N queries, processing time is $T_{\text{init}} + NT_{\text{query}}$, excluding network transfer and storage. Figure 5(b) instead plots sender processing time.

The two tables represent different transmission workflows. Table 2 sends raw measurements for receiver-side reconstruction, whereas Table 3 first materializes a video and transmits its H.265 encoding. Their native output counts differ, so these measurements compare workflow costs rather than equal-output throughput. The smaller H.265 files encode a fixed set of frames; our payload retains a reusable state for receiver-selected times. Bit2Bit's sample-specific optimization is a reconstruction setup cost, not feed-forward per-frame inference. Payloads are transmitted object sizes, not model parameter sizes.

Table 2: Transmitting the raw QIS bitstream for receiver-side reconstruction versus our latent-state route, evaluated on one matched 4096-frame, 256×512, 1-bit sequence. Sender denotes sender-side processing; the raw route only packages the bitstream, so its cost is negligible (—). Receiver reports per-frame reconstruction for the baselines and one-time state initialization for Ours. Query reports reconstruction after initialization, excluding state I/O; Figure 5(a) plots initialization plus one query. A separate query cost does not apply to the raw-route baselines (—). †Bit2Bit includes non-amortized sample-specific optimization.

Method	Transmitted object	Sender	Receiver	Payload	Query
QBP	raw bitstream	—	2.201 s	64.000 MiB	—
QUIVEr	raw bitstream	—	1.594 s	64.000 MiB	—
gQIR	raw bitstream	—	1.947 s	64.000 MiB	—
Bit2Bit†	raw bitstream	—	36,070.163 s	64.000 MiB	—
Ours w/o Compression	latent state	0.362 s	0.137 s	680 MiB	30.06 ms
Ours (compressed)	latent state	**0.382 s**	**0.189 s**	**4.628 MiB**	30.06 ms

Table 3: Reconstructing and H.265-encoding a fixed video before transmission versus our latent-state route, evaluated on the same 4096-frame, 256×512, 1-bit sequence. Sender includes reconstruction and H.265 encoding for the baselines and state generation for Ours; Figure 5(b) plots this cost. Receiver reports one-time state initialization for Ours; a separate H.265 initialization measurement is unavailable (—). Query reports H.265 frame access or reconstruction from our initialized state, excluding I/O. †Bit2Bit includes sample-specific optimization. H.265 uses CRF 10.3, and each transmitted video's native frame count is shown.

Method	Transmitted object	Sender	Receiver	Payload	Query
QBP	H.265 video (8 frames)	17.802 s	—	0.426 MiB	11.47 ms
QUIVEr	H.265 video (502 frames)	801.018 s	—	0.252 MiB	15.87 ms
gQIR	H.265 video (502 frames)	978.015 s	—	0.252 MiB	15.87 ms
Bit2Bit†	H.265 video (4096 frames)	36,073.064 s	—	1.398 MiB	15.94 ms
Ours w/o Compression	latent state	0.362 s	0.137 s	680 MiB	30.06 ms
Ours (compressed)	latent state	**0.382 s**	**0.189 s**	**4.628 MiB**	30.06 ms

G Ablation Results

Table 4 complements Figure 7(a)–(c) by listing PSNR, SSIM, and LPIPS for the full model and two ablations. *Ours w/o Compression* queries the uncompressed latent hierarchy, whereas *Ours w/o Residual Correction* removes the learned nonlinear correction from temporal querying. Here, residual correction refers to $\mathcal{R}_\phi$ in Section 3.3, not the cross-scale coding residual R^ℓ in the latent codec. The full and no-compression variants are independently trained, as noted in the main text; their difference therefore compares the complete variants rather than an isolated quantization perturbation. Values summarize the 13 validation videos using the same mean and population-standard-deviation convention as Table 1.

Table 4: Dense temporal-query quality for the full model and two ablations over the 13 validation videos. Values are mean $\pm$ population standard deviation across videos. The *Ours (compressed)* and *Ours w/o Compression* rows correspond to the same runs reported in Table 1.

Photon rate	Method	PSNR (dB)↑	SSIM↑	LPIPS↓
0.05	Ours (compressed)	31.928 $\pm$ 2.399	0.8882 $\pm$ 0.0543	0.1721 $\pm$ 0.0749
	Ours w/o Compression	31.451 $\pm$ 2.590	0.8833 $\pm$ 0.0547	0.1700 $\pm$ 0.0700
	Ours w/o Residual Correction	29.936 $\pm$ 2.647	0.8726 $\pm$ 0.0581	0.1819 $\pm$ 0.0741
0.10	Ours (compressed)	33.230 $\pm$ 2.395	0.9098 $\pm$ 0.0445	0.1378 $\pm$ 0.0576
	Ours w/o Compression	32.771 $\pm$ 2.506	0.9064 $\pm$ 0.0444	0.1339 $\pm$ 0.0534
	Ours w/o Residual Correction	31.286 $\pm$ 2.583	0.8968 $\pm$ 0.0473	0.1452 $\pm$ 0.0570
0.20	Ours (compressed)	34.193 $\pm$ 2.402	0.9259 $\pm$ 0.0360	0.1128 $\pm$ 0.0459
	Ours w/o Compression	33.989 $\pm$ 2.463	0.9248 $\pm$ 0.0348	0.1068 $\pm$ 0.0411
	Ours w/o Residual Correction	32.574 $\pm$ 2.531	0.9154 $\pm$ 0.0377	0.1171 $\pm$ 0.0442
0.30	Ours (compressed)	34.603 $\pm$ 2.349	0.9332 $\pm$ 0.0320	0.1018 $\pm$ 0.0400
	Ours w/o Compression	34.603 $\pm$ 2.411	0.9335 $\pm$ 0.0300	0.0949 $\pm$ 0.0355
	Ours w/o Residual Correction	33.181 $\pm$ 2.468	0.9251 $\pm$ 0.0324	0.1038 $\pm$ 0.0381
0.50	Ours (compressed)	35.065 $\pm$ 2.307	0.9397 $\pm$ 0.0283	0.0937 $\pm$ 0.0364
	Ours w/o Compression	35.125 $\pm$ 2.451	0.9423 $\pm$ 0.0250	0.0840 $\pm$ 0.0306
	Ours w/o Residual Correction	33.756 $\pm$ 2.495	0.9340 $\pm$ 0.0272	0.0927 $\pm$ 0.0328