LEARNING ADAPTIVE METADATA FOR GENERATIVE SPACE-TIME VIDEO SUPER-RESOLUTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Space-time video super-resolution (STVSR) reconstructs high-resolution, high-frame-rate video from sparse, degraded pixels, but a generative receiver cannot reliably recover source-specific structure that was never transmitted. We introduce MetaST, a sender–receiver framework that learns what source information to preserve under a shared transmission budget. Pixel anchors and structural metadata compete for the same bytes, valued by their reconstruction benefit to a fixed receiver system. A lightweight Metadata Value Predictor learns these values from offline receiver responses and guides allocation using source and encoder statistics, without sender-side reconstruction. On HC-Meta, a human-centric benchmark with aligned structural representations, reallocating rate to source structure improves Average PSNR by 3.0 dB over the strongest pixel-only baseline at the lossless-anchor reference budget. On the unpruned candidate grid, learned valuation recovers 79.4% of the oracle allocation gain and reduces regret by 33% relative to a rate-only predictor. H.264/H.266 and general-video experiments further show how the useful representation depends on the relative efficiency of pixel and metadata coding. MetaST provides a modular rate–value formulation for learning what to preserve for a generative receiver.

1 INTRODUCTION

Space-time video super-resolution reconstructs spatial detail and intermediate frames from degraded video. Progress has largely improved the receiver through joint reconstruction networks, continuous representations and generative visual priors (Xiang et al., 2020; Chen et al., 2022; 2023; Kim et al., 2025; Chen et al., 2025). The transmission interface usually remains fixed: the sender provides low-resolution, low-frame-rate pixels, and the receiver infers the missing information.

A strong generative receiver changes what is worth transmitting. Coarse observations may suffice for predictable appearance, while missing structural evidence can determine which reconstruction is correct. Two high-quality anchors can leave a fast-moving limb's trajectory ambiguous; preserving its intermediate position may then be more useful than refining those anchors. Human-centric video makes this tension visible: articulated motion and self-occlusion admit multiple plausible reconstructions, while edges, skeletons and body pose preserve different source evidence.

The central question is which information should be preserved for the receiver under a limited budget? The answer depends on content and the reconstruction model. Detailed edges may resolve ambiguous motion, compact skeletons may suffice elsewhere, and metadata may be redundant when anchors already constrain the output. Fixed representation priorities miss these differences. Evaluating every candidate with the receiver would reveal its value, but would move expensive reconstruction into the sender's decision loop.

MetaST addresses this problem through receiver-conditional representation selection (Figure 1). The sender jointly considers anchor qualities and metadata configurations, assigning each a transmission cost and reconstruction value. A Metadata Value Predictor (MVP) learns these values from offline responses. At deployment, source and encoder statistics suffice to predict candidate values and allocate the sequence budget. The framework supports different video and metadata codecs without changing the allocation objective. When a codec changes, we remeasure its transmission cost and, if the decoded inputs change, re-evaluate the resulting reconstruction quality.

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.

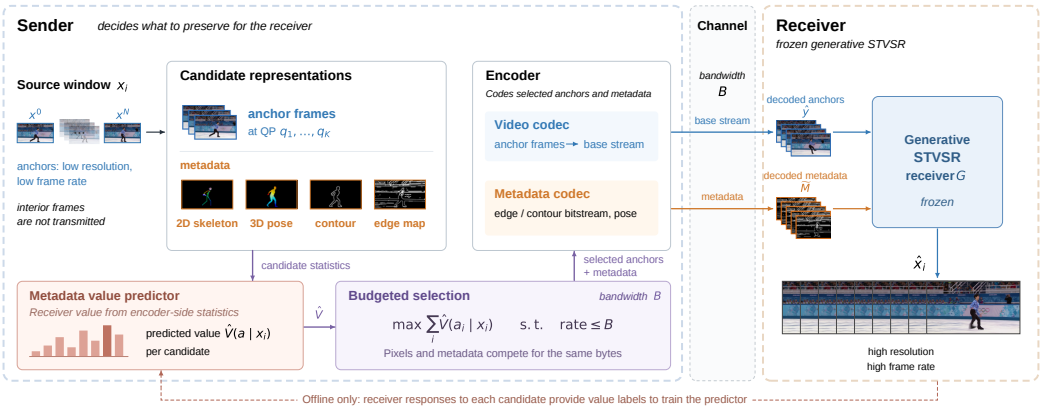


Figure 1: MetaST values candidate actions under a fixed receiver system and selects an allocation under a shared budget. Offline receiver responses supervise value prediction; deployment requires no sender-side reconstruction.

On HC-Meta, our evaluation benchmark of aligned structural representations, and on general-video datasets, we show that structural evidence can be more valuable than pixel fidelity. Useful representations vary with content and budget; evaluation on the full candidate grid shows that their value can be predicted without sender-side reconstruction.

Our contributions are:

- A formulation of generative STVSR transmission that allocates a shared budget to pixel anchors and heterogeneous metadata by their value to a fixed receiver system.
- A lightweight sender-side value predictor trained from offline receiver responses, coupled with sequence allocation that accounts for shared anchors, metadata and signalling.
- HC-Meta, with primary representation-selection and value-learning results on the complete unpruned 25-action Cartesian grid, complemented by codec comparisons, coded-stream validation and general-video experiments.

2 Related Work

STVSR and generative restoration. Joint STVSR networks reconstruct spatial detail and intermediate frames in a unified model (Xiang et al., 2020; Xu et al., 2021), while continuous formulations support reconstruction at arbitrary spatial and temporal coordinates (Chen et al., 2022; 2023; Kim et al., 2025). Generative restoration further uses learned priors to recover missing detail (Zhou et al., 2024; Wang et al., 2025a); DOVE, for example, adapts video diffusion to one-step restoration (Chen et al., 2025). These approaches motivate our receiver, while leaving open the sender's choice of information to preserve. We evaluate both joint methods and composed interpolation–super-resolution pipelines as pixel-only baselines. The Rich Detail Range perspective (Wang, 2026) motivates reconstruction-aware delivery for AI-enhanced displays, where spatial and temporal detail can be reconstructed instead of transmitted directly as pixels.

Structural guidance and generative compression. Structural conditions such as edges and human pose provide explicit spatial or geometric constraints that help diffusion models preserve source layout and motion (Zhang et al., 2023b; Mou et al., 2024). This form of conditioning has been particularly effective for human animation and keyframe interpolation, where appearance alone may not determine articulated motion or intermediate poses (Hu, 2024; Zhu et al., 2024; Guo et al., 2025). In parallel, generative compression uses strong decoder priors to reconstruct perceptually plausible content from learned latents or compact semantic and structural descriptions (Mentzer et al., 2020; Lei et al., 2023; Careil et al., 2024). MetaST connects these directions in the STVSR setting: structural guidance is treated as coded source evidence with an explicit byte cost, not as a freely available receiver input. The sender then chooses among heterogeneous metadata and pixel-anchor qualities according to their predicted reconstruction benefit for a fixed generative receiver, allowing the preferred representation to vary with the source content and available budget.

Rate allocation and information value. Classical rate–distortion optimization allocates coding decisions under a rate constraint (Shoham & Gersho, 1988; Ortega & Ramchandran, 1998; Sullivan & Wiegand, 1998). Value-of-information and adaptive acquisition methods similarly select observations by their expected benefit (Weiss & Taskar, 2013; Ma et al., 2019; Covert et al., 2023). We learn the reconstruction value of heterogeneous source representations for a fixed receiver system, guiding allocation without online receiver inference. Appendix N discusses the broader literature.

3 METHOD

3.1 RECEIVER-CONDITIONAL REPRESENTATION SELECTION

Let $X = (x_t)_{t=0}^{T-1}$ be a source video with frames in $\mathbb{R}^{H\times W\times 3}$. Divide it into reconstruction intervals $X_i = (x_{iL}, \dots, x_{iL+L})$, $i = 0, \dots, N-1$, where L is the temporal sampling factor, $N = \lfloor (T-1)/L \rfloor$, and neighbouring intervals share one boundary frame. The boundary frames are downsampled by a spatial factor s to form the pixel anchors. The goal is to reconstruct the source at its original spatial and temporal resolution.

An *action* $a = (q, f) \in \mathcal{A}$ pairs anchor quality q with a *metadata configuration* f, specifying representation families (e.g., skeleton, pose or edges), temporal sampling and fidelity; $f = \varnothing$ sends anchors only. Let $\hat{Y}_i^q$ denote decoded anchors and $\tilde{M}_i^{q,f}$ the completed metadata, which may depend on them. In the *fixed receiver system*, G_f denotes the frozen receiver configuration assigned to the families in f before valuation, yielding $\hat{X}_i^a = G_f(\hat{Y}_i^q, \tilde{M}_i^{q,f})$. For a higher-is-better quality measure $\mathcal{Q}$, define action value relative to a common reference a_0 as

$$V_i(a) = \mathcal{Q}(X_i, \hat{X}_i^a) - \mathcal{Q}(X_i, \hat{X}_i^{a_0}). \tag{1}$$

Quality is measured on the frames assigned to interval i. Anchors and metadata contribute to the same objective; the common reference leaves its rate-constrained maximizers unchanged.

An *allocation* $\mathbf{a} \in \mathcal{A}^N$ is a sequence of actions. Shared boundaries couple their costs: $\mathcal{R}(\mathbf{a}) = \sum_{i=0}^{N-1} r_i(a_{i-1}, a_i)$, where a_{-1} is a start state and each transition charges newly transmitted anchors, metadata and signalling. Given budget B, the exact optimum within this menu and sequence rate model is

$$\mathbf{a}^*(B) \in \arg\max_{\mathbf{a}\in\mathcal{A}^N} \sum_{i=0}^{N-1} V_i(a_i) \quad \text{subject to} \quad \mathcal{R}(\mathbf{a}) \le B. \tag{2}$$

Codec constraints, such as a common anchor quality within a predictive stream, restrict the feasible action sequences. Equation 2 depends on coding through its costs and on reconstruction through its values, providing a common interface for different codecs. Stream sizes and metadata amortization are specified in Appendix E.

3.2 METADATA AND THE RECONSTRUCTION INTERFACE

Metadata is extracted from the source before degradation and coded into a separate stream; we refer to this setting as *source-assisted transmission*. Geometric representations such as skeletons and body pose transmit compact parameters that are rendered at the receiver. Image-like representations such as edges and contours transmit coded curves, optionally at selected times, and the receiver completes the missing maps before reconstruction. Thus, representation family, temporal sampling and fidelity are explicit allocation decisions. Section 4.1 defines the experimental configurations; extraction and coding details are in Appendix C.

The receiver configurations G_f use a one-step video diffusion architecture (Figure 2) based on DOVE and CogVideoX (Chen et al., 2025; Yang et al., 2024b). Bilinear spatial upsampling and linear temporal blending of the decoded anchors form an initial video estimate. A shared VAE encodes this estimate and metadata renderings as separate latent views; joint attention updates the video latent in one step before decoding. The fixed system contains two preassigned frozen checkpoints: one for skeleton/pose views (also used for anchors-only actions) and one for edge/contour views. This mapping and all receiver parameters remain unchanged throughout valuation and allocation. Appendix D specifies the training protocol.

3.3 LEARNING THE RECEIVER'S INFORMATION DEMAND

Computing $V_i(a)$ for every action requires repeated receiver inference. We perform these evaluations offline and train MVP to predict values from source and encoder statistics. Its inputs $s_i(a)$ summarize coding cost, anchor fidelity, payload structure and source motion. A pooled representation $\bar{h}_i$ of the candidate set supplies context about the alternatives available for that source interval. With $\pi(a)$ the mean value of action a on the training set, MVP learns a content-dependent residual:

$$\hat{V}_i(a) = \pi(a) + g_\theta(s_i(a), \bar{h}_i) - g_\theta(s_i(a_0), \bar{h}_i). \tag{3}$$

Since $\pi(a_0) = 0$, this construction preserves $\hat{V}_i(a_0) = 0$. The prior captures stable differences among actions; the nonlinear residual learns when source content changes their usefulness.

The training objective combines value regression with the ranking needed for allocation. For a rate–value slope λ, define distributions over actions by

$$p_{i,\lambda} = \mathrm{softmax}((\mathbf{V}_i - \lambda \mathbf{c}_i)/\tau)\,, \qquad \hat{p}_{i,\lambda} = \mathrm{softmax}\Big((\hat{\mathbf{V}}_i - \lambda \mathbf{c}_i)/\tau\Big)\,, \tag{4}$$

where $\mathbf{c}_i$ contains the local action costs used to form the training frontiers and $\tau > 0$ is a temperature. These local scores provide a training surrogate; sequence allocation retains the transition costs in equation 2. The loss is

$$\mathcal{L}(\theta) = \mathbb{E}_{i,a}\Big[\ell_\delta(\hat{V}_i(a) - V_i(a))\Big] + \lambda_{\mathrm{dec}}\mathbb{E}_{i,\lambda}\left[-\sum_{a\in\mathcal{A}} p_{i,\lambda}(a)\log \hat{p}_{i,\lambda}(a)\right], \tag{5}$$

with Huber loss ℓ_δ (Huber, 1964) and slopes sampled from training frontiers. The second term encourages agreement about useful actions at different budgets (Cao et al., 2007); model selection likewise uses allocation regret rather than value error. Model dimensions, features and all training choices are in Appendix G.

3.4 ALLOCATION WITHOUT SENDER-SIDE RECONSTRUCTION

At deployment, predicted values replace measured values in equation 2. For a multiplier $\lambda \geq 0$, a chain dynamic program solves

$$\max_{\mathbf{a}} \sum_i \Big[\hat{V}_i(a_i) - \lambda r_i(a_{i-1}, a_i)\Big]\,. \tag{6}$$

Sweeping λ yields candidate allocations, whose billed rates are checked against the budget without sender-side reconstruction. For evaluation, the *menu-constrained oracle* is the exact measured-value optimum within the fixed candidate action space and stated sequence rate model, computed by a budget-indexed dynamic program. All uses of *oracle* refer to this constrained optimum. The finite Lagrangian sweep need not attain every frontier point; all predictor arms use the same solver.

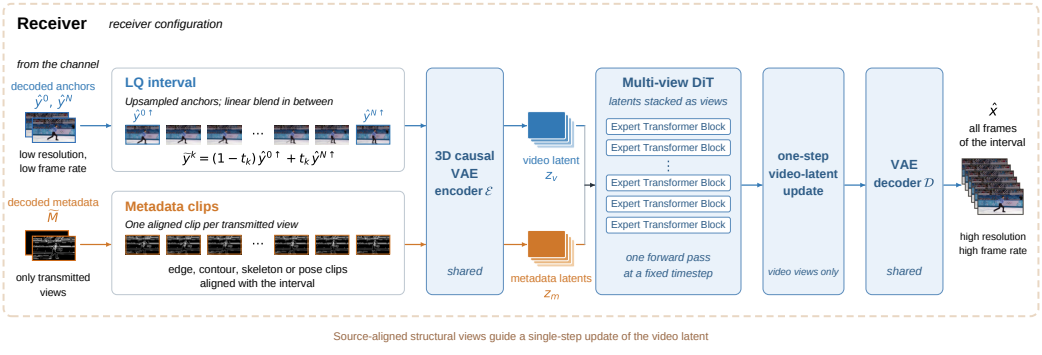


Figure 2: Receiver configuration in the fixed receiver system. A shared variational autoencoder (VAE) encodes the initial video estimate and metadata as separate latent views. Joint attention conditions a one-step video update on transmitted evidence. Details are in Appendix D.



Figure 3: HC-Meta source frames and aligned 2D skeletons, 3D poses, human contours and Canny edges. The representations expose different evidence about the same scene for studying what to transmit.

Value accuracy translates directly into allocation quality. If $|\hat{V}_i(a) - V_i(a)| \le \epsilon_i$ for every feasible action and a feasible allocation $\hat{\mathbf{a}}$ is within η of the maximum predicted total value under B, then

$$\sum_i V_i(a_i^*) - \sum_i V_i(\hat{a}_i) \le 2 \sum_i \epsilon_i + \eta. \tag{7}$$

Prediction and optimization errors enter additively; Appendix F gives the proof and dynamic-programming recurrences.

4 Experiments

4.1 Experimental setup

Data, task and metrics. HC-Meta contains 50 held-out human-centric clips with aligned structural metadata (Figure 3); for general-video tests on GoPro and Adobe240 (Nah et al., 2017; Su et al., 2017), we extract edge metadata from the source frames. Following BF-STVSR (Kim et al., 2025), $s = 4$ and $L = 8$: 1280×720 frames are reconstructed from 320×180 anchors eight frames apart. We report PSNR and SSIM on luma. *Average* pools interval positions 0–7, while *Center* averages positions 0 and 4 for methods that do not reconstruct every intermediate frame. Value learning uses Average PSNR as $\mathcal{Q}$. Dataset, split and metric details are in Appendices B, D and H.

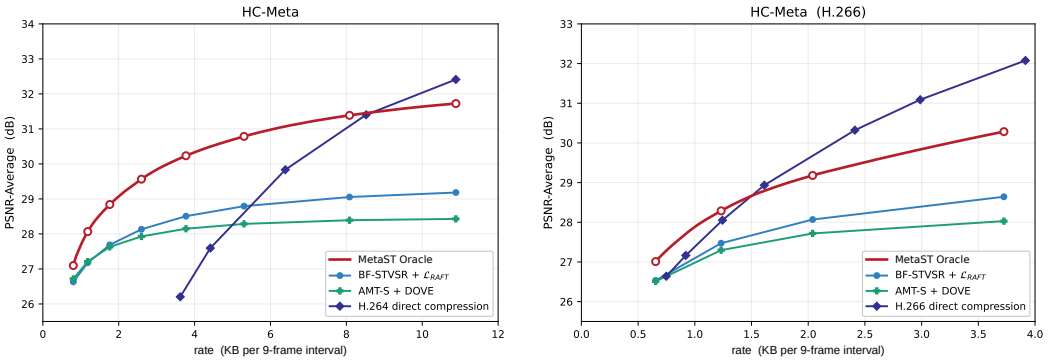
Metadata and action configurations. Edge density is the retained fraction of skeletonized Canny pixels; 100% denotes the full edge map. For temporally sparse edges, k counts transmitted interior maps: $k = 1$ sends position 4, and $k = 3$ sends positions 2, 4 and 6. Boundary maps at positions 0 and 8 come from decoded anchors at zero metadata rate, and TPS-Inbetween (Zhu et al., 2025) completes the missing times. The primary unpruned 5×5 grid crosses five anchor QPs with anchors only, skeleton, skeleton plus 3D pose, 75%-density sparse edges ($k = 3$), and dense edges at all nine positions. The value reference a_0 is anchors only at QP 37. Fixed-budget edge comparisons use full-density maps with $k \in \{1, 3\}$.

Baselines, budgets and menus. We compare joint STVSR models with video-frame-interpolation (VFI)–video-super-resolution (VSR) pipelines using the same decoded anchors; $\mathcal{L}_{\mathrm{RAFT}}$ denotes RAFT flow supervision during training. Lower codec QP gives higher anchor quality and rate. The reference budget B_0 is each clip's lossless anchor stream; MetaST chooses the finest anchor QP for which anchors, metadata and signalling fit B_0. Rate–distortion experiments use H.264 and H.266 anchors, and direct compression codes the full-resolution source. Primary allocation and value learning use the complete 25-action grid. Codec curves use a benchmark-derived 12-action subset whose oracle is within 0.05 dB of the full-grid oracle; it is not used for the primary predictor result. GoPro/Adobe240 use edge-based menus because pose metadata is unavailable. Rates are decimal KB per interval. Codec settings and all menu entries are in Appendix E.

Prediction and stream validation. Grouped five-fold cross-validation fits all components within training folds. Table 3 compares the item prior, rates-only ridge, ridge on MVP inputs and a permutation control. Regret is the oracle PSNR minus the PSNR of the selected allocation; gain recovered is normalized from the best uniform action to the oracle. Group-stratified paired bootstrap

Table 1: End-to-end transmission-system comparison at the lossless-anchor reference budget B_0 (PSNR / SSIM on Y; $\times 4$ space, $\times 8$ time). MetaST uses re-coded anchors and the full-density, three-map edge configuration ($k = 3$). Bold marks the best score per column and metric.

Method		HC-Meta		GoPro		Adobe240	
VFI	VSR	*Center*	*Average*	*Center*	*Average*	*Center*	*Average*
Super SloMo	Bicubic	29.12 / 0.8421	27.13 / 0.8142	27.04 / 0.7937	26.06 / 0.7720	26.09 / 0.7435	25.29 / 0.7279
	EDVR	30.37 / 0.8616	27.45 / 0.8242	28.24 / 0.8322	26.30 / 0.7960	27.25 / 0.7972	25.95 / 0.7682
	BasicVSR	30.14 / 0.8542	27.33 / 0.8201	28.23 / 0.8308	26.36 / 0.7977	27.28 / 0.7961	25.94 / 0.7679
	DOVE	28.56 / 0.8513	26.46 / 0.8183	26.87 / 0.8000	25.05 / 0.7560	26.27 / 0.7764	24.76 / 0.7358
AMT-S	Bicubic	29.53 / 0.8490	28.14 / 0.8341	27.90 / 0.8092	27.28 / 0.7942	26.79 / 0.7624	26.24 / 0.7473
	EDVR	30.76 / 0.8717	28.75 / 0.8526	29.11 / 0.8436	28.12 / 0.8250	28.10 / 0.8114	27.19 / 0.7909
	BasicVSR	30.78 / 0.8714	28.78 / 0.8534	29.09 / 0.8416	28.11 / 0.8245	28.14 / 0.8113	27.26 / 0.7933
	DOVE	29.40 / 0.8716	28.15 / 0.8590	28.08 / 0.8309	27.67 / 0.8232	27.46 / 0.8112	27.17 / 0.8058
DAIN	Bicubic	29.33 / 0.8481	27.77 / 0.8323	26.92 / 0.7911	26.11 / 0.7740	26.01 / 0.7461	25.40 / 0.7321
	EDVR	30.19 / 0.8643	27.86 / 0.8420	28.01 / 0.8239	26.37 / 0.7964	27.06 / 0.7895	26.01 / 0.7703
	BasicVSR	30.27 / 0.8647	27.90 / 0.8429	28.00 / 0.8227	26.46 / 0.7966	27.07 / 0.7890	26.23 / 0.7725
	DOVE	28.61 / 0.8475	27.08 / 0.8307	26.63 / 0.7915	25.35 / 0.7654	26.09 / 0.7712	25.14 / 0.7505
Zooming SlowMo		32.13 / 0.9037	– / –	30.69 / 0.8847	– / –	30.26 / 0.8821	– / –
TMNet		31.76 / 0.8963	29.36 / 0.8792	30.14 / 0.8692	28.83 / 0.8514	29.41 / 0.8524	28.30 / 0.8354
VideoINR		30.94 / 0.8935	29.10 / 0.8797	30.26 / 0.8792	29.41 / 0.8669	29.92 / 0.8746	29.27 / 0.8651
MoTIF		31.57 / 0.8967	29.44 / 0.8823	31.04 / 0.8877	30.04 / 0.8773	30.63 / 0.8839	29.82 / 0.8750
BF-STVSR + $\mathcal{L}_{\text{RAFT}}$		31.66 / 0.8989	29.63 / 0.8859	31.14 / 0.8893	30.20 / 0.8799	30.84 / 0.8877	30.14 / **0.8808**
BF-STVSR		31.63 / 0.8979	29.56 / 0.8845	31.17 / 0.8898	30.22 / 0.8802	30.83 / **0.8880**	30.12 / **0.8808**
MetaST (edge map, k=3)		**33.84 / 0.9061**	**32.65 / 0.8997**	**32.75 / 0.9129**	**32.34 / 0.9102**	**31.19** / 0.8749	**30.73** / 0.8715



(a) H.264 anchors.

(b) H.266 anchors.

Figure 4: HC-Meta codec curves. MetaST Oracle is the exact measured-value optimum within the fixed 12-action menu and stated sequence rate model. Primary learned allocation uses the unpruned 25-action grid (Table 3). Rates include anchors, metadata and signalling for MetaST, anchors for pixel-only STVSR, and the source stream for direct compression.

gives confidence intervals. Actual mixed-QP streams at ten budgets validate rate billing; Appendices G and E.1 give details.

4.2 Structural evidence improves reconstruction quality

At the matched budget B_0, reallocating anchor bytes to structure raises HC-Meta Average PSNR from 29.63 to 32.65 dB, a 3.0 dB gain over the strongest pixel-only baseline (Table 1). This comparison isolates the source-assisted interface; Sections 4.3 and 4.4 isolate adaptive selection and value prediction. Transmitted edges preserve limbs, silhouettes and printed detail that pixel-only methods blur or duplicate (Figure 7); additional qualitative results are provided in Appendix A.

On the H.264 curves, cleaner anchors eventually reach a pixel-only reconstruction ceiling because they do not resolve unobserved motion (Figure 4a). Oracle metadata allocation remains 1.2–2.9 dB

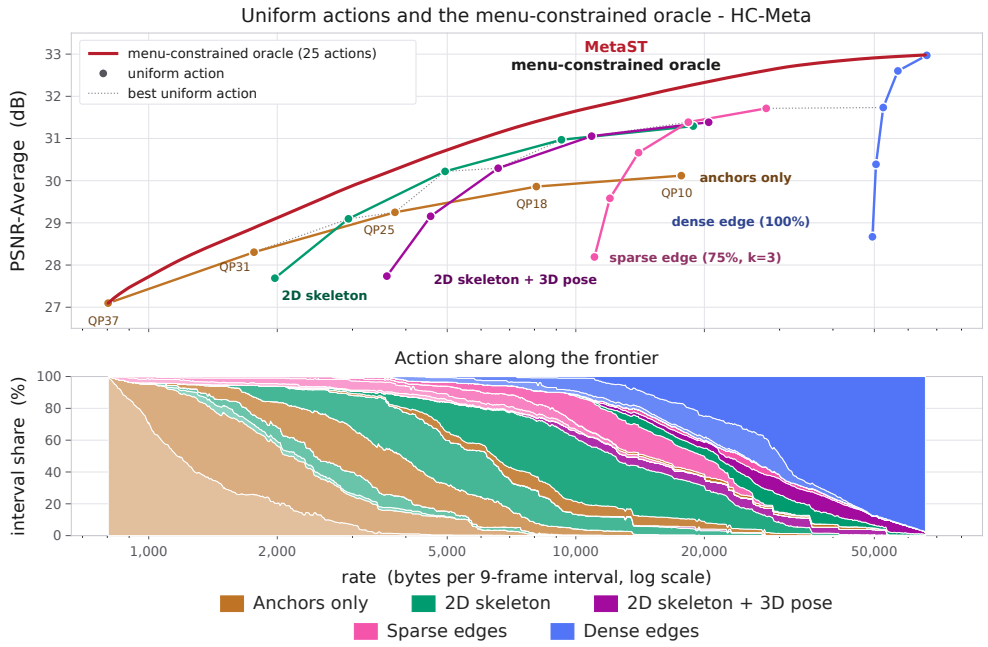


Figure 5: Adaptive allocation on the unpruned 25-action grid. Top: uniform actions, their envelope and the exact oracle. Bottom: action shares; colours denote metadata and shades denote anchor QP. Figure 12 gives the full legend.

Table 2: Metadata ablation at B_0 (PSNR / SSIM). Edge rows use full-density maps; k counts transmitted interior maps and missing times are interpolated. Anchor QP adapts to the budget; pose is evaluated on HC-Meta. Bold marks the best score per column and metric.

Configuration	HC-Meta		GoPro		Adobe240	
	Center	*Average*	*Center*	*Average*	*Center*	*Average*
Edges, $k = 1$	32.78 / 0.9027	30.59 / 0.8856	30.50 / 0.8836	29.24 / 0.8636	29.14 / 0.8512	28.02 / 0.8326
Edges, $k = 3$	**33.84** / 0.9061	**32.65** / 0.8997	**32.75** / **0.9129**	**32.34** / **0.9102**	**31.19** / **0.8749**	**30.73** / **0.8715**
2D skeleton	32.96 / 0.9070	31.60 / 0.8993	–	–	–	–
3D pose	32.71 / 0.9048	31.25 / 0.8964	–	–	–	–
2D skeleton + 3D pose	33.01 / **0.9074**	31.68 / **0.8999**	–	–	–	–

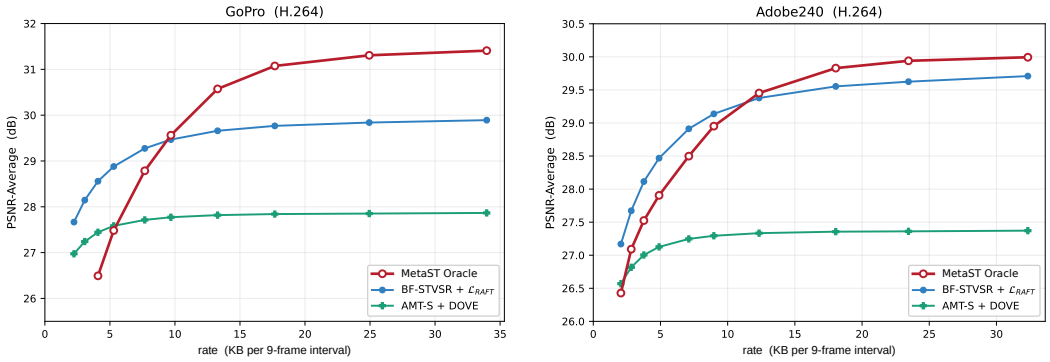
above BF-STVSR over 1.8–17.7 KB; at 5.3 KB, the scores are 30.79 versus 28.79 dB. Targeted structure is also competitive with direct full-resolution H.264 below the 8.7 KB crossover.

4.3 Content and budget determine the useful representation

Figure 5 isolates adaptive selection by comparing the oracle with uniform actions. At the rates of the QP-25 sparse-edge, dense-edge and anchors-only actions, allocation gains 1.32, 1.19 and 1.02 dB, respectively. The frontier uses 22 of 25 actions, including mixtures at a single budget.

The allocation progresses from anchors and compact skeletons at low rates to sparse and then dense edges as the budget grows. Selective transmission avoids paying for detailed maps where compact metadata suffices; at the highest rates, uniform dense edges nearly reach the oracle. Thus, useful evidence depends jointly on content, anchor quality and budget, and the decision structure is learned by the Metadata Value Predictor.

At B_0, compact skeletons provide substantial gains, while full-density $k = 3$ edges achieve the highest PSNR on all three datasets (Table 2). Adding body pose to skeletons helps little here, motivating receiver-value prediction rather than a fixed ranking by semantic richness.



(a) GoPro. (b) Adobe240.

Figure 6: General-video H.264 curves, discussed in Section 4.5. MetaST Oracle is the exact optimum within each fixed edge-based menu; rates include anchors, metadata and signalling. Appendix K specifies the menus.

Table 3: 25-action grid: mean and 90th-percentile (P90) regret (dB); gain is relative to the best uniform action.

Value model	Mean	P90	Gain
Item prior	0.252	0.441	65.0%
Rates-only ridge	0.220	0.412	69.3%
Ridge on MVP inputs	0.230	0.342	68.1%
MVP, permuted	0.349	0.489	51.4%
MVP	**0.148**	**0.216**	**79.4%**

Table 4: Mixed-QP H.264 validation: deviations from billing over ten budgets.

Quantity	Deviation
Anchor rate	-0.5% to $+2.3\%$
Sequence rate	-0.51% to $+1.92\%$
Average PSNR	< 0.07 dB (abs.)
Center PSNR	< 0.09 dB (abs.)

4.4 Learned values reduce allocation regret

On the unpruned 25-action grid, MVP reduces mean regret from 0.220 to 0.148 dB (33%) and recovers 79.4% of the oracle gain, versus 69.3% for rates-only ridge and 65.0% for the item prior (Table 3). The paired reduction is 0.072 dB, with a 95% interval of $[0.031, 0.122]$ from a group-stratified paired bootstrap; all arms use identical allocation and rate accounting.

With identical descriptors, nonlinear MVP improves mean regret from 0.230 to 0.148 dB; permuting its predictions raises regret to 0.349 dB. It also improves P90 regret (0.412 to 0.216 dB) and action-value rank correlation (0.932 to 0.951) over rates-only ridge (Appendix G).

On the supplementary 12-action subset, improvement peaks at 2–10 KB (0.337 to 0.186 dB; 45%). Above 30 KB, rates-only ridge and MVP approach the oracle (0.049 and 0.042 dB), while maximum regret falls from 0.53 to 0.28 dB (Appendix G).

4.5 Coded-stream realization and codec dependence

Realizing adaptive allocations. Actual mixed-QP H.264 streams for 50 clips at 10 budgets change sequence rate by -0.51 to $+1.92\%$ and dataset Average PSNR by less than 0.07 dB (Table 4). Cached sparse completion isolates predictive anchor coding; Appendix E.1 gives full results.

Stronger anchor coding. With one H.266 anchor QP per clip, oracle allocation remains above pixel-only STVSR across the plotted range (Figure 4b): at 1.23 KB it reaches 28.3 versus 27.47 dB for BF-STVSR. Direct H.266 crosses over near 1.4 KB, showing that codec efficiency determines the useful operating range.

General video. Without pose metadata, source assistance improves Average PSNR at B_0 by 2.1 dB on GoPro and 0.6 dB on Adobe240 (Table 1). The oracle surpasses BF-STVSR above 10 KB on GoPro and above 11 KB on Adobe240, gaining 1.5 dB near 34 KB on GoPro and 0.3 dB at Adobe240's highest rate (Figure 6).

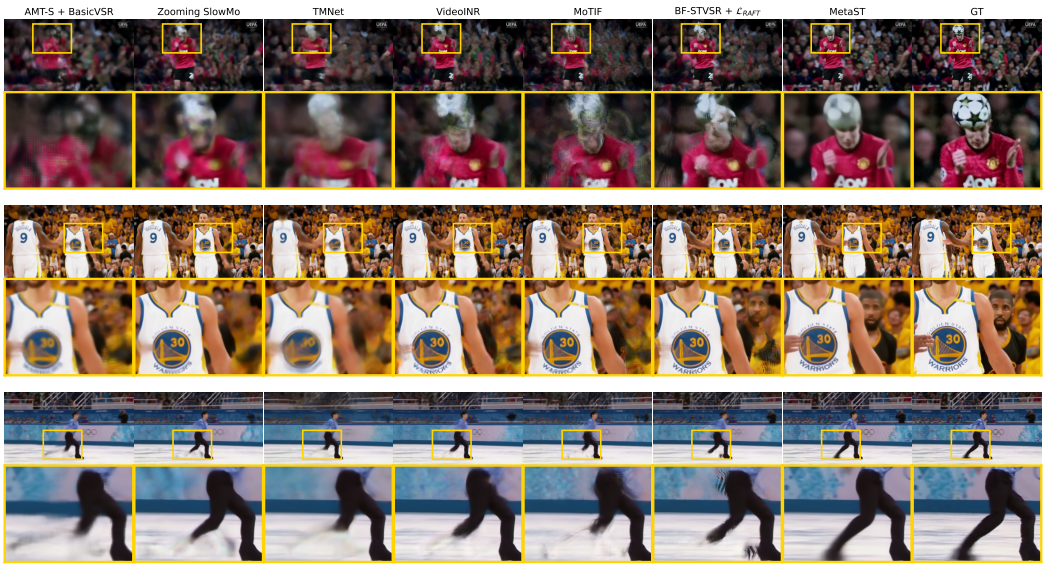


Figure 7: HC-Meta reconstructions at B_0, with full frames and crops. Columns are labelled above. The full-density $k = 3$ edge configuration preserves articulated motion and fine structure.



Figure 8: GoPro (top) and Adobe240 (bottom) reconstructions at B_0 using adaptive $k = 3$/dense edge actions. Columns follow Figure 7; Appendix E gives the codec configuration.

5 Discussion and Limitations

Primary representation selection and value learning use the unpruned 25-action grid; the 12-action subset serves only codec curves and supplementary analyses. Improving video or metadata coding can change the preferred action while preserving the common rate–value objective. Gains with widely deployed H.264 (Fraunhofer Heinrich Hertz Institute, n.d.) and the narrower advantage over direct H.266 motivate stronger metadata coding. New codecs update costs and receiver values; extending value prediction to new representation families and video domains remains open.

6 Conclusion

MetaST allocates transmission budgets by receiver-conditional reconstruction value. Source structure improves the system, while learned valuation reduces adaptive allocation regret. The broader principle is to preserve the source-specific evidence a generative receiver needs to resolve ambiguity, with representation choices shaped by content and budget.

Reproducibility statement

Appendix B specifies the HC-Meta data and frame accounting. Appendices C–D describe metadata processing and receiver training; Appendices E and F give coding, billing and allocation details; Appendix G gives predictor inputs, training and grouped validation. Appendix H specifies baseline models and metrics, and Appendix I collects numerical results. Code, the HC-Meta evaluation lists and metadata streams will be released.

AI use statement

We used generative AI tools to improve the readability of the manuscript, assist with modifications to scientific figures, and create or edit software code. The authors reviewed the AI-assisted text and figures and verified the relevant code before use. The authors take full responsibility for the final content of the paper, including its claims, figures, and code.

References

Eirikur Agustsson, Michael Tschannen, Fabian Mentzer, Radu Timofte, and Luc Van Gool. Generative adversarial networks for extreme learned image compression. In *ICCV*, 2019.

Johannes Ballé, Valero Laparra, and Eero P. Simoncelli. End-to-end optimized image compression. In *ICLR*, 2017.

Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational image compression with a scale hyperprior. In *ICLR*, 2018.

Wenbo Bao, Wei-Sheng Lai, Chao Ma, Xiaoyun Zhang, Zhiyong Gao, and Ming-Hsuan Yang. Depth-aware video frame interpolation. In *CVPR*, 2019.

Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, Varun Jampani, and Robin Rombach. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.

Yochai Blau and Tomer Michaeli. The perception-distortion tradeoff. In *CVPR*, 2018.

Yochai Blau and Tomer Michaeli. Rethinking lossy compression: The rate-distortion-perception tradeoff. In *ICML*, 2019.

Fred L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(6):567–585, 1989.

Eirina Bourtsoulatze, David Burth Kurka, and Deniz Gündüz. Deep joint source-channel coding for wireless image transmission. *IEEE Transactions on Cognitive Communications and Networking*, 5 (3):567–579, 2019.

John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: From pairwise approach to listwise approach. In *ICML*, 2007.

Marlene Careil, Matthew J. Muckley, Jakob Verbeek, and Stéphane Lathuilière. Towards image compression with perfect realism at ultra-low bitrates. In *ICLR*, 2024.

Kelvin C.K. Chan, Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. BasicVSR: The search for essential components in video super-resolution and beyond. In *CVPR*, 2021.

Kelvin C.K. Chan, Shangchen Zhou, Xiangyu Xu, and Chen Change Loy. BasicVSR++: Improving video super-resolution with enhanced propagation and alignment. In *CVPR*, 2022a.

Kelvin C.K. Chan, Shangchen Zhou, Xiangyu Xu, and Chen Change Loy. Investigating tradeoffs in real-world video super-resolution. In *CVPR*, 2022b.

Jiaben Chen and Huaizu Jiang. SportsSloMo: A new benchmark and baselines for human-centric video frame interpolation. In *CVPR*, 2024.

Yi-Hsin Chen, Si-Cun Chen, Yi-Hsin Chen, Yen-Yu Lin, and Wen-Hsiao Peng. MoTIF: Learning motion trajectories with local implicit neural functions for continuous space-time video super-resolution. In *ICCV*, 2023.

Zeyuan Chen, Yinbo Chen, Jingwen Liu, Xingqian Xu, Vidit Goel, Zhangyang Wang, Humphrey Shi, and Xiaolong Wang. VideoINR: Learning video implicit neural representation for continuous space-time super-resolution. In *CVPR*, 2022.

Zheng Chen, Zichen Zou, Kewei Zhang, Xiongfei Su, Xin Yuan, Yong Guo, and Yulun Zhang. DOVE: Efficient one-step diffusion model for real-world video super-resolution. In *NeurIPS*, 2025.

Myungsub Choi, Heewon Kim, Bohyung Han, Ning Xu, and Kyoung Mu Lee. Channel attention is all you need for video frame interpolation. In *AAAI*, 2020.

Ian Connick Covert, Wei Qiu, Mingyu Lu, Na Yoon Kim, Nathan J. White, and Su-In Lee. Learning to maximize mutual information for dynamic feature selection. In *ICML*, 2023.

Duolikun Danier, Fan Zhang, and David Bull. LDMVFI: Video frame interpolation with latent diffusion models. In *AAAI*, 2024.

David H. Douglas and Thomas K. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 10(2):112–122, 1973.

Ling-Yu Duan, Jiaying Liu, Wenhan Yang, Tiejun Huang, and Wen Gao. Video coding for machines: A paradigm of collaborative compression and intelligent analytics. *IEEE Transactions on Image Processing*, 29:8680–8695, 2020.

Fraunhofer Heinrich Hertz Institute. H.264 / AVC Technology. https://www.hhi.fraunhofer.de/en/departments/vca/technologies-and-solutions/h264-avc.html, n.d. Accessed September 24, 2026.

Soham Gadgil, Ian Covert, and Su-In Lee. Estimating conditional mutual information for dynamic feature selection. In *ICLR*, 2024.

Zhicheng Geng, Luming Liang, Tianyu Ding, and Ilya Zharkov. RSTT: Real-time spatial temporal transformer for space-time video super-resolution. In *CVPR*, 2022.

Zujin Guo, Size Wu, Zhongang Cai, Wei Li, and Chen Change Loy. Controllable human-centric keyframe interpolation with generative prior. In *NeurIPS*, 2025.

Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. Recurrent back-projection network for video super-resolution. In *CVPR*, 2019.

Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. Space-time-aware multi-resolution video enhancement. In *CVPR*, 2020.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J. Fleet. Video diffusion models. In *NeurIPS*, 2022.

Li Hu. Animate anyone: Consistent and controllable image-to-video synthesis for character animation. In *CVPR*, 2024.

Zhewei Huang, Tianyuan Zhang, Wen Heng, Boxin Shi, and Shuchang Zhou. Real-time intermediate flow estimation for video frame interpolation. In *ECCV*, 2022.

Zhewei Huang, Ailin Huang, Xiaotao Hu, Chen Hu, Jun Xu, and Shuchang Zhou. Scale-adaptive feature aggregation for efficient space-time video super-resolution. In *WACV*, 2024.

Peter J. Huber. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, 35 (1):73–101, 1964.

Siddhant Jain, Daniel Watson, Eric Tabellion, Aleksander Holynski, Ben Poole, and Janne Kontkanen. Video interpolation with diffusion models. In *CVPR*, 2024.

Huaizu Jiang, Deqing Sun, Varun Jampani, Ming-Hsuan Yang, Erik Learned-Miller, and Jan Kautz. Super SloMo: High quality estimation of multiple intermediate frames for video interpolation. In *CVPR*, 2018.

Younghyun Jo, Seoung Wug Oh, Jaeyeon Kang, and Seon Joo Kim. Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation. In *CVPR*, 2018.

Eunjin Kim, Hyeonjin Kim, Kyong Hwan Jin, and Jaejun Yoo. BF-STVSR: B-splines and fourier—best friends for high fidelity spatial-temporal video super-resolution. In *CVPR*, 2025.

Lingtong Kong, Boyuan Jiang, Donghao Luo, Wenqing Chu, Xiaoming Huang, Ying Tai, Chengjie Wang, and Jie Yang. IFRNet: Intermediate feature refine network for efficient frame interpolation. In *CVPR*, 2022.

Murat Kunt, Athanassios Ikonomopoulos, and Michel Kocher. Second-generation image-coding techniques. *Proceedings of the IEEE*, 73(4):549–574, 1985.

Hyeongmin Lee, Taeoh Kim, Tae-young Chung, Daehyun Pak, Yuseok Ban, and Sangyoun Lee. AdaCoF: Adaptive collaboration of flows for video frame interpolation. In *CVPR*, 2020.

Eric Lei, Yiğit Berkay Uslu, Hamed Hassani, and Shirin Saeedi Bidokhti. Text + sketch: Image compression at ultra low rates. In *ICML Workshop on Neural Compression*, 2023.

Jiahao Li, Bin Li, and Yan Lu. Deep contextual video compression. In *NeurIPS*, 2021.

Zhen Li, Zuo-Liang Zhu, Ling-Hao Han, Qibin Hou, Chun-Le Guo, and Ming-Ming Cheng. AMT: All-pairs multi-field transforms for efficient frame interpolation. In *CVPR*, 2023.

Jingyun Liang, Yuchen Fan, Xiaoyu Xiang, Rakesh Ranjan, Eddy Ilg, Simon Green, Jiezhang Cao, Kai Zhang, Radu Timofte, and Luc Van Gool. Recurrent video restoration transformer with guided deformable attention. In *NeurIPS*, 2022.

Jingyun Liang, Jiezhang Cao, Yuchen Fan, Kai Zhang, Rakesh Ranjan, Yawei Li, Radu Timofte, and Luc Van Gool. VRT: A video restoration transformer. *IEEE Transactions on Image Processing*, 33:2171–2182, 2024.

Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Transactions on Graphics*, 34(6):248:1–248:16, 2015.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

Guo Lu, Wanli Ouyang, Dong Xu, Xiaoyun Zhang, Chunlei Cai, and Zhiyong Gao. DVC: An end-to-end deep video compression framework. In *CVPR*, 2019.

Chao Ma, Sebastian Tschiatschek, Konstantina Palla, José Miguel Hernández-Lobato, Sebastian Nowozin, and Cheng Zhang. EDDI: Efficient dynamic discovery of high-value information with partial VAE. In *ICML*, 2019.

Cheng Ma, Yongming Rao, Yean Cheng, Ce Chen, Jiwen Lu, and Jie Zhou. Structure-preserving super resolution with gradient guidance. In *CVPR*, 2020.

Fabian Mentzer, George D. Toderici, Michael Tschannen, and Eirikur Agustsson. High-fidelity generative image compression. In *NeurIPS*, 2020.

Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, and Ying Shan. T2I-Adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *AAAI*, 2024.

Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *CVPR*, 2017.

Seungjun Nah, Sungyong Baik, Seokil Hong, Gyeongsik Moon, Sanghyun Son, Radu Timofte, and Kyoung Mu Lee. NTIRE 2019 challenge on video deblurring and super-resolution: Dataset and study. In *CVPR Workshops*, 2019.

Simon Niklaus, Long Mai, and Feng Liu. Video frame interpolation via adaptive separable convolution. In *ICCV*, 2017.

Antonio Ortega and Kannan Ramchandran. Rate-distortion methods for image and video compression. *IEEE Signal Processing Magazine*, 15(6):23–50, 1998.

Rémi Pautrat, Iago Suárez, Yifan Yu, Marc Pollefeys, and Viktor Larsson. GlueStick: Robust image matching by sticking points and lines together. In *ICCV*, 2023.

Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3D hands, face, and body from a single image. In *CVPR*, 2019.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. ZeRO: Memory optimizations toward training trillion parameter models. In *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis*, 2020.

Fitsum Reda, Janne Kontkanen, Eric Tabellion, Deqing Sun, Caroline Pantofaru, and Brian Curless. FILM: Frame interpolation for large motion. In *ECCV*, 2022.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

Guido M. Schuster, Gerry Melnikov, and Aggelos K. Katsaggelos. Operationally optimal vertex-based shape coding. *IEEE Signal Processing Magazine*, 15(6):91–108, 1998.

Heiko Schwarz, Detlev Marpe, and Thomas Wiegand. Overview of the scalable video coding extension of the H.264/AVC standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(9):1103–1120, 2007.

Shuwei Shi, Jinjin Gu, Liangbin Xie, Xintao Wang, Yujiu Yang, and Chao Dong. Rethinking alignment in video super-resolution transformers. In *NeurIPS*, 2022.

Hajin Shim, Sung Ju Hwang, and Eunho Yang. Joint active feature acquisition and classification with variable-size set encoding. In *NeurIPS*, 2018.

Yair Shoham and Allen Gersho. Efficient bit allocation for an arbitrary set of quantizers. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 36(9):1445–1453, 1988.

Hyeonjun Sim, Jihyong Oh, and Munchurl Kim. XVFI: eXtreme video frame interpolation. In *ICCV*, 2021.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *ICML*, 2023.

Shuochen Su, Mauricio Delbracio, Jue Wang, Guillermo Sapiro, Wolfgang Heidrich, and Oliver Wang. Deep video deblurring for hand-held cameras. In *CVPR*, 2017.

Gary J. Sullivan and Thomas Wiegand. Rate-distortion optimization for video compression. *IEEE Signal Processing Magazine*, 15(6):74–90, 1998.

Zachary Teed and Jia Deng. RAFT: Recurrent all-pairs field transforms for optical flow. In *ECCV*, 2020.

Yapeng Tian, Yulun Zhang, Yun Fu, and Chenliang Xu. TDAN: Temporally-deformable alignment network for video super-resolution. In *CVPR*, 2020.

Haohong Wang. Rich detail range (RDR): Redefining perceptual picture quality in the era of AI displays. In *2026 International Conference on Computing, Networking and Communications (ICNC)*, pp. 1–2. IEEE, February 2026. doi: 10.1109/ICNC68183.2026.11416982.

Jianyi Wang, Zongsheng Yue, Shangchen Zhou, Kelvin C.K. Chan, and Chen Change Loy. Exploiting diffusion prior for real-world image super-resolution. *International Journal of Computer Vision*, 132:5929–5949, 2024a.

Jianyi Wang, Zhijie Lin, Meng Wei, Yang Zhao, Ceyuan Yang, Chen Change Loy, and Lu Jiang. SeedVR: Seeding infinity in diffusion transformer towards generic video restoration. In *CVPR*, 2025a.

Xiaojuan Wang, Boyang Zhou, Brian Curless, Ira Kemelmacher-Shlizerman, Aleksander Holynski, and Steven M. Seitz. Generative inbetweening: Adapting image-to-video models for keyframe interpolation. In *ICLR*, 2025b.

Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *CVPR*, 2018.

Xintao Wang, Kelvin C.K. Chan, Ke Yu, Chao Dong, and Chen Change Loy. EDVR: Video restoration with enhanced deformable convolutional networks. In *CVPR Workshops*, 2019.

Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-ESRGAN: Training real-world blind super-resolution with pure synthetic data. In *ICCV Workshops*, 2021.

Yufei Wang, Wenhan Yang, Xinyuan Chen, Yaohui Wang, Lanqing Guo, Lap-Pui Chau, Ziwei Liu, Yu Qiao, Alex C. Kot, and Bihan Wen. SinSR: Diffusion-based image super-resolution in a single step. In *CVPR*, 2024b.

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.

David J. Weiss and Ben Taskar. Learning adaptive value of information for structured prediction. In *NeurIPS*, 2013.

Rongyuan Wu, Lingchen Sun, Zhiyuan Ma, and Lei Zhang. One-step effective diffusion network for real-world image super-resolution. In *NeurIPS*, 2024.

Xiaoyu Xiang, Yapeng Tian, Yulun Zhang, Yun Fu, Jan P. Allebach, and Chenliang Xu. Zooming slow-mo: Fast and accurate one-stage space-time video super-resolution. In *CVPR*, 2020.

Rui Xie, Yinhong Liu, Penghao Zhou, Chen Zhao, Jun Zhou, Kai Zhang, Zhenyu Zhang, Jian Yang, Zhenheng Yang, and Ying Tai. STAR: Spatial-temporal augmentation with text-to-video models for real-world video super-resolution. In *ICCV*, 2025.

Gang Xu, Jun Xu, Zhen Li, Liang Wang, Xing Sun, and Ming-Ming Cheng. Temporal modulation network for controllable space-time video super-resolution. In *CVPR*, 2021.

Xiangyu Xu, Li Siyao, Wenxiu Sun, Qian Yin, and Ming-Hsuan Yang. Quadratic video interpolation. In *NeurIPS*, 2019.

Zhongcong Xu, Jianfeng Zhang, Jun Hao Liew, Hanshu Yan, Jia-Wei Liu, Chenxu Zhang, Jiashi Feng, and Mike Zheng Shou. MagicAnimate: Temporally consistent human image animation using diffusion model. In *CVPR*, 2024.

Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T. Freeman. Video enhancement with task-oriented flow. *International Journal of Computer Vision*, 127(8):1106–1125, 2019.

Fuzhi Yang, Huan Yang, Jianlong Fu, Hongtao Lu, and Baining Guo. Learning texture transformer network for image super-resolution. In *CVPR*, 2020.

Ruoyu Yang, Yichi Zhang, Haohong Wang, and Fengqing Zhu. Decoder-guided lossy contour coding via anchor refinement. *arXiv preprint arXiv:2607.26426*, 2026.

Xi Yang, Chenhang He, Jianqi Ma, and Lei Zhang. Motion-guided latent diffusion for temporally consistent real-world video super-resolution. In *ECCV*, 2024a.

Zhendong Yang, Ailing Zeng, Chun Yuan, and Yu Li. Effective whole-body pose estimation with two-stages distillation. In *ICCV Workshops*, 2023.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, Da Yin, Xiaotao Gu, Yuxuan Zhang, Weihan Wang, Yean Cheng, Ting Liu, Bin Xu, Yuxiao Dong, and Jie Tang. CogVideoX: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024b.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Frédo Durand, William T. Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *CVPR*, 2024.

Zongsheng Yue, Jianyi Wang, and Chen Change Loy. ResShift: Efficient diffusion model for image super-resolution by residual shifting. In *NeurIPS*, 2023.

Guozhen Zhang, Yuhan Zhu, Haonan Wang, Youxin Chen, Gangshan Wu, and Limin Wang. Extracting motion and appearance via inter-frame attention for efficient video frame interpolation. In *CVPR*, 2023a.

Kai Zhang, Jingyun Liang, Luc Van Gool, and Radu Timofte. Designing a practical degradation model for deep blind image super-resolution. In *ICCV*, 2021.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, 2023b.

Zhifei Zhang, Zhaowen Wang, Zhe Lin, and Hairong Qi. Image super-resolution by neural texture transfer. In *CVPR*, 2019.

Shangchen Zhou, Peiqing Yang, Jianyi Wang, Yihang Luo, and Chen Change Loy. Upscale-a-video: Temporal-consistent diffusion model for real-world video super-resolution. In *CVPR*, 2024.

Shenhao Zhu, Junming Leo Chen, Zuozhuo Dai, Zilong Dong, Yinghui Xu, Xun Cao, Yao Yao, Hao Zhu, and Siyu Zhu. Champ: Controllable and consistent human image animation with 3D parametric guidance. In *ECCV*, 2024.

Tianyi Zhu, Wei Shang, and Dongwei Ren. Thin-plate spline-based interpolation for animation line inbetweening. In *AAAI*, 2025.

A Additional Qualitative Results on HC-Meta



Figure 9: Additional fixed-budget reconstructions on HC-Meta (I). For each window, from top to bottom: the transmitted anchor frame with the crop box, its crop, the centre frame with the same box, its crop. Columns as in Figure 7; same bytes for every method.

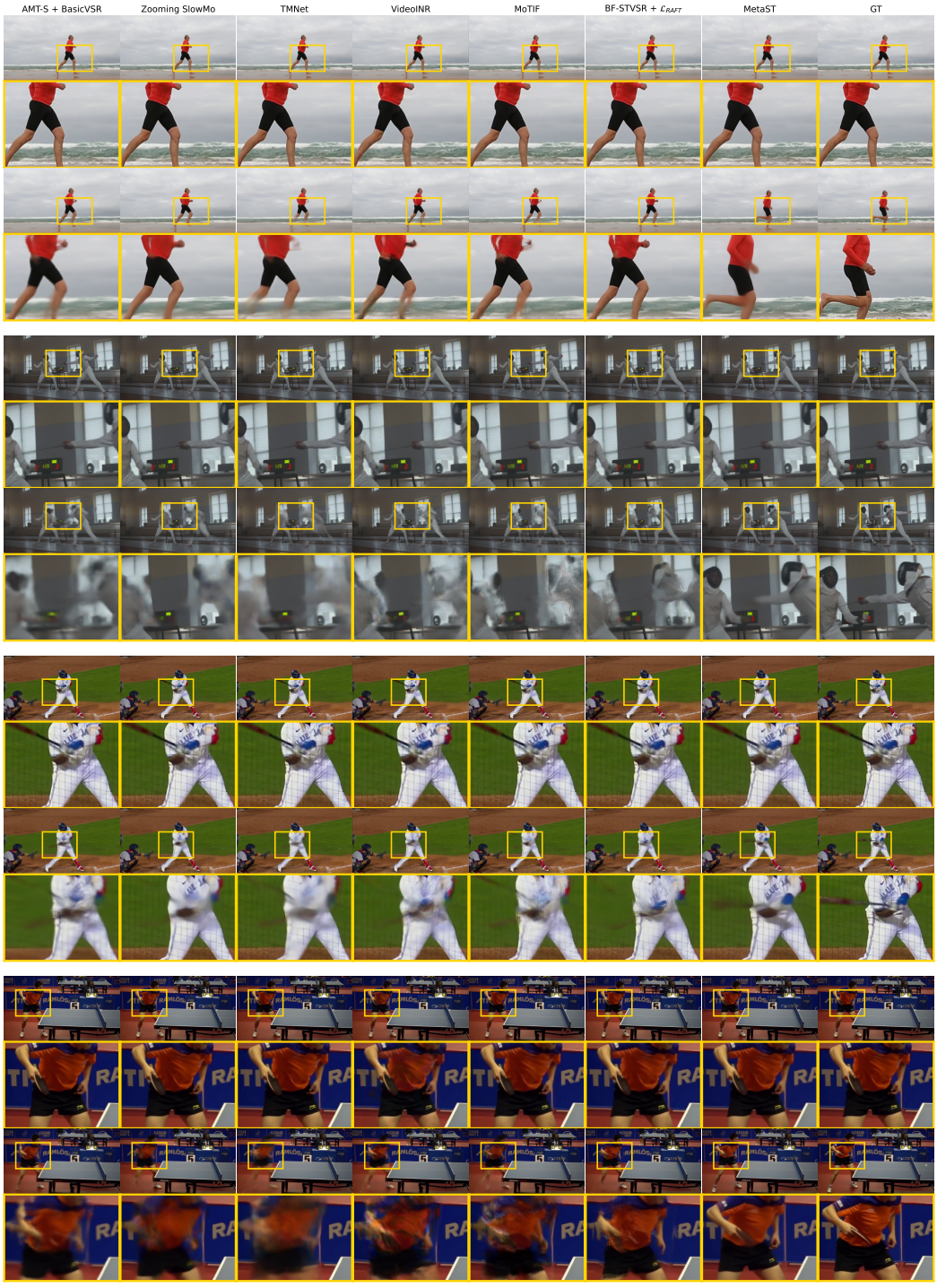


Figure 10: Additional fixed-budget reconstructions on HC-Meta (II); layout and columns as in Figure 9.

B HC-META: CONSTRUCTION AND EVALUATION SPLITS

Source corpus. HC-Meta is the evaluation benchmark introduced here, built on the CHKI-Video source corpus (Guo et al., 2025), a human-centric video corpus with per-frame 2D keypoints and SMPL-X body-model annotations whose clips originate from the SportsSloMo benchmark (Chen & Jiang, 2024) and from Pexels. Receiver training and HC-Meta evaluation use disjoint CHKI-Video splits.

Evaluation set. The HC-Meta evaluation set consists of 50 clips in two provenance subsets: 34 originating from SportsSloMo and 16 from Pexels, at 1280×720, evaluated with the stride-8 reconstruction protocol of GoPro/Adobe240 (interval i covers frames $8i, \ldots, 8i{+}8$). Average scores use positions 0–7 of each complete interval; the last boundary frame and incomplete trailing intervals are excluded. The 7,448 source frames therefore yield 7,200 distinct scored frames. Table 5 gives the statistics; clips are between 21 and 519 frames long.

Table 5: HC-Meta evaluation set.

	SportsSloMo	Pexels	All
Clips	34	16	50
Frames	1,816	5,632	7,448
Reconstruction intervals (stride 8)	207	693	900
Average-scored frames	1,656	5,544	7,200
Cross-validation groups for MVP	16	16	32

Composition. The SportsSloMo subset was drawn with a fixed random seed from the corpus's test clips and covers tennis, basketball, soccer, American football, baseball, table tennis, gymnastics, figure skating and skiing. The Pexels subset consists of free-to-use stock videos contributed by the Pexels creator community, which CHKI-Video retrieved with keywords for everyday and athletic human movement (walking, kicking, throwing, catching, climbing) in varied indoor and outdoor scenes; the 16 clips are the longest Pexels test clips of the corpus. Content duplicates were removed.

Representation families. For every frame we assemble: a *2D skeleton*, the whole-body keypoints of the annotation rendered as a coloured limb-and-joint image in DWPose style (Yang et al., 2023); a *3D pose*, the SMPL-X body surface (Loper et al., 2015; Pavlakos et al., 2019) of the annotation rendered into the camera as a colour-coded surface map; a *contour*, the boundary of the human segmentation masks distributed with the corpus, coded with the contour codec of Appendix C; and a Canny *edge map* (Canny, 1986) of the source frame (thresholds 50/100, 5×5 Gaussian smoothing with $\sigma = 1.4$). Skeleton and 3D-pose renderings are indexed by frame position within the clip.

C METADATA CODING AND INTERPOLATION

Experimental configurations. Spatial and temporal factors are $s = 4$ and $L = 8$: nine $H \times W = 720 \times 1280$ frames per interval, with 180×320 anchors. Value learning uses Average PSNR as $\mathcal{Q}$ and anchors only at QP 37 as the reference. Metadata density levels are 100, 75, 50 and 25%. For sparse configurations, k counts transmitted interior maps: position 4 for $k = 1$, and positions 2, 4, 6 for $k = 3$. Both use the same coding and interpolation procedure, with endpoint maps derived from decoded anchors at zero metadata rate. The known positions for $k = 3$ are therefore $\{0, 2, 4, 6, 8\}$. Dense actions transmit source-derived maps at all nine positions (`--canny_source gt`), with each shared boundary encoded once per clip.

Sparsification. Density level 100% is the Canny map itself. For the lower levels the map is skeletonized, curves are merged at junctions by tangent continuity, ranked by the mean source gradient magnitude along the curve, and kept in rank order until the retained pixels reach the level's share of the skeleton pixels. Only positions that may be transmitted are sparsified.

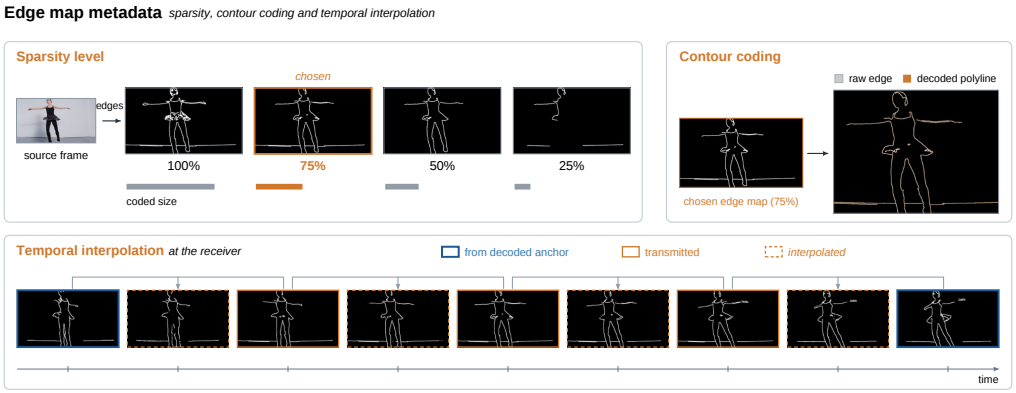


Figure 11: Edge-map metadata: density levels (left), polyline coding (right) and receiver-side interpolation between the decoded-anchor and transmitted maps (bottom).

Contour coding. A binary map is thinned to a one-pixel skeleton (8-connectivity preserved), endpoint gaps of up to two pixels are closed, one-pixel junction spurs are removed, components smaller than three pixels are discarded, and the skeleton graph is decomposed into open paths and closed curves of at least five pixels. Each path is simplified by Ramer–Douglas–Peucker (Douglas & Peucker, 1973) with tolerance $\epsilon = 1.5\,\text{px}$ and the resulting vertices are written as a Rice-coded keypoint bitstream by the anchor-refinement contour codec (Yang et al., 2026); the receiver reconstructs the map by drawing the polylines. The same coder with the same tolerance is used for edge maps of every density and for contours, so the bytes of every map are measured, not estimated. Sparse payload costs sum the coded sizes of the selected maps, plus 2 framing bytes per map in the fixed-budget protocol; dense costs use the clip-level amortization in Appendix E.

Temporal interpolation. Given a configuration's known maps, each temporal gap is filled by TPS-Inbetween (Zhu et al., 2025): a thin-plate-spline (Bookstein, 1989) motion model is fitted to point and line correspondences from GlueStick (Pautrat et al., 2023) between the two known maps, and the intermediate maps are synthesized. The matcher is frozen; the motion-fitting and synthesis stages were fine-tuned on Canny sequences of SportsSloMo training clips, starting from the released animation checkpoint. Endpoint maps derived from decoded anchors use bilinear $\times 4$ upsampling followed by Canny. For every sparse configuration, this endpoint conditioning makes the interpolated evidence depend on anchor QP; completed sequences are cached per (metadata configuration, anchor QP). The synthesis stage is not bit-deterministic across rebuilds (identical rebuilds differ in about 3% of interpolated edge pixels while the known positions are identical); every reported number uses the trees as built once and gated (nine maps per interval, binary, non-empty, transmitted positions identical across anchor QPs).

Skeleton and pose streams. The 2D skeleton stream contains per-frame keypoint coordinates and confidence scores; the 3D-pose stream contains SMPL-X rotation and translation parameters. Each clip's complete record stream is zlib-compressed once. For a clip with N complete intervals, covering $8N + 1$ frames, its compressed size S is amortized as S/N bytes per interval. Equivalently, a measured per-frame size b gives $b(8N+1)/N$ bytes per interval. Each selected pose action receives this clip-level amortized charge. The pooled payloads are 1,171 B per interval for skeletons and 2,807 B for skeleton plus 3D pose (Table 6).

D THE FIXED RECEIVER SYSTEM

Figure 2 summarizes the architecture shared by the receiver configurations.

Architecture and conditioning. The receiver is the CogVideoX-2B transformer (30 expert-transformer blocks, 30 attention heads of dimension 64, 16 latent channels, spatial patch size 2, 3D causal VAE with temporal compression 4) adapted to one-step restoration in the DOVE fashion

(Chen et al., 2025): the LQ interval is encoded, treated as the noisy latent at a fixed timestep, denoised in one forward pass, and decoded. The initial LQ interval is formed from the anchors: the two decoded anchors are upsampled to 1280×720 with bilinear interpolation and the seven interior frames are linear cross-fades. Every transmitted representation is rendered as a nine-frame image clip aligned with the interval, encoded by the same VAE, and its latent is appended to the video latent as an additional view; the transformer attends over all views, the one-step update is applied to the video view, and only the video view is decoded. Text conditioning uses a fixed empty prompt. Inference is deterministic. The VAE runs in fp16 with slicing and tiling.

Receiver configurations. The fixed receiver system maps representation families to two pre-specified frozen checkpoints before valuation: one for 2D skeleton and 3D-pose views, and one for edges and contours. The pose configuration also handles HC-Meta anchors-only actions. The family-to-checkpoint mapping and all receiver parameters remain fixed during valuation, allocation and evaluation, including codec and general-video comparisons.

Training recipe. Full fine-tuning of the transformer (1.69 B trainable parameters; VAE and text encoder frozen), AdamW (Loshchilov & Hutter, 2019) with $\beta = (0.9, 0.95)$, $\epsilon = 10^{-8}$, learning rate 2×10^{-5} constant after 100 linear warm-up steps, weight decay 10^{-4}, gradient clipping at 0.1, fp16 mixed precision with DeepSpeed ZeRO-2 (Rajbhandari et al., 2020) and gradient checkpointing, batch size 1 per GPU on four A100-80GB (effective batch 4), nine-frame 720×1280 training intervals with anchors at positions 0 and 8 and the LQ scale $\times 4$, pose-view dropout 0.3 so that any subset of the two views (including none) is in distribution, a pixel-space L_2 reconstruction objective, seed 42, and 16.4 s per iteration. The edge/contour checkpoint uses the same architecture and aligned-view interface with edge conditioning.

Cost. One interval costs 8.5 GPU-seconds on an A100-40GB when three inference jobs share a GPU (about 10 s alone); BF-STVSR costs 3.3 s and AMT-S + DOVE Stage-1 6.3 s per interval on the same hardware. Peak inference memory is 8.7 GB per job. Offline valuation costs about 26 GPU-hours for the twelve-action subset and a further 76 GPU-hours for the 13 complementary actions of the full grid.

E RATE ACCOUNTING, MENUS AND ALLOCATION

Anchor codecs. Anchors are the 320×180 frames at interval positions 0 and 8, obtained by bicubic $\times \frac{1}{4}$ downscaling of the source (the pre-generated LR trees of the BF-STVSR protocol). *Fixed budget*: the baselines receive the anchors coded losslessly with x264 in RGB 4:4:4 IPPP mode (QP 0) so that their decoded input is bit-identical to the LR frames, and the bytes of that per-clip stream define B_0 (HC-Meta: 76,470.7 B per interval pooled; GoPro 87,542; Adobe240 82,786). MetaST re-codes the same anchors with the same coder at the finest QP for which anchors plus metadata (plus 2 framing bytes per map and the selection side information) fit in B_0 for that clip. All fixed-budget results satisfy the reference budget for each clip. *Rate–distortion*: anchors are coded with x264 (one I picture followed by P pictures, SEI and filler NAL units stripped) as

```
ffmpeg -c:v libx264 -pix_fmt yuv420p -preset medium -qp Q \
-x264-params bframes=0:scenecut=0:keyint=infinite:ref=4
```

the HC-Meta plot ladder is QP $\{0, 10, 15, 18, 22, 25, 28, 31, 34, 37\}$ and the menu uses $\{10, 18, 25, 31, 37\}$, whose anchor rates are 17,659, 8,084, 3,772, 1,764 and 803 B per interval. The GoPro/Adobe240 plot ladder is QP $\{0, 8, 11, 15, 18, 20, 23, 25, 27, 29\}$ and the menu uses $\{8, 15, 20, 25, 39\}$. QP 39 is an additional low-rate menu rung and is not part of the plotted anchor ladder. *Direct compression* codes the full-resolution source sequence with the same x264 settings at QP 22–51 (H.264) or with vvenc low-delay P at QP 22–63 (H.266), strips SEI/filler, decodes, and scores the same intervals; the plotted curve is the upper convex hull of the measured points. *H.266*: one chain per clip coded as

```
vvencapp --preset medium --fps 3.75 --qp Q -rt none \
--additional "PicReordering=0 GOPSize=8 NumRefPics=4"
```

running over its anchor pictures, decoded through vvdec to 10-bit YUV and converted to RGB; ladder QP $\{17, 22, \ldots, 57\}$, menu QPs $\{17, 27, 37, 47, 57\}$ mapped from the H.264 menu by ladder index. H.266 uses a common anchor QP within each clip, while metadata configurations can vary across intervals.

Byte model of a sequence. For allocation $\mathbf{a} = (a_0, \ldots, a_{N-1})$, the rate includes coded anchors, metadata and selection signalling. Anchor costs are measured from the constant-QP streams, including overhead. A shared picture is charged once if adjacent actions agree on QP, and once per distinct QP otherwise. Section E.1 evaluates this model using actual mixed-QP streams. Skeleton and pose costs follow the clip-level amortization in Appendix C. For dense edges, every source-frame map is encoded once over the clip's $8N + 1$ frames; the sum of these map sizes is divided by N to obtain the amortized interval cost, so shared boundaries are counted once. The `dense_k: 7` field in the plotting metadata counts interior positions only; the rate includes source-derived boundary maps as well. All sparse configurations use the same billing rule: charge for the coded source-derived interior maps and their framing; endpoint maps derived from decoded anchors incur no metadata payload cost. Selection signalling uses $\lceil \log_2 |\mathcal{A}| \rceil$ bits per interval, rounded to whole bytes per sequence (4 bits for 12 or 15 actions; 5 for 25). Table 6 lists the uniform actions.

Table 6: HC-Meta 25-action grid: bytes per interval of each action applied uniformly (anchors, transmitted metadata, total) and its Average PSNR. This full grid defines the predictor study in Table 3; the twelve-action subset of Figure 4 is listed below.

Action	Anchor	Metadata	Total	PSNR (dB)
anchors only, QP 37	803	0	803	27.10
anchors only, QP 31	1,764	0	1,764	28.30
anchors only, QP 25	3,772	0	3,772	29.25
anchors only, QP 18	8,084	0	8,084	29.86
anchors only, QP 10	17,659	0	17,659	30.12
2D skeleton, QP 37	803	1,171	1,974	27.69
2D skeleton, QP 31	1,764	1,171	2,935	29.10
2D skeleton, QP 25	3,772	1,171	4,943	30.22
2D skeleton, QP 18	8,084	1,171	9,254	30.97
2D skeleton, QP 10	17,659	1,171	18,829	31.29
2D skeleton + 3D pose, QP 37	803	2,807	3,611	27.74
2D skeleton + 3D pose, QP 31	1,764	2,807	4,571	29.16
2D skeleton + 3D pose, QP 25	3,772	2,807	6,579	30.29
2D skeleton + 3D pose, QP 18	8,084	2,807	10,891	31.06
2D skeleton + 3D pose, QP 10	17,659	2,807	20,466	31.38
sparse edge (75%, k=3), QP 37	803	10,253	11,057	28.19
sparse edge (75%, k=3), QP 31	1,764	10,253	12,017	29.58
sparse edge (75%, k=3), QP 25	3,772	10,253	14,025	30.66
sparse edge (75%, k=3), QP 18	8,084	10,253	18,337	31.39
sparse edge (75%, k=3), QP 10	17,659	10,253	27,912	31.71
dense edge (100%), QP 37	803	48,683	49,486	28.67
dense edge (100%), QP 31	1,764	48,683	50,447	30.39
dense edge (100%), QP 25	3,772	48,683	52,455	31.73
dense edge (100%), QP 18	8,084	48,683	56,767	32.60
dense edge (100%), QP 10	17,659	48,683	66,342	32.97

Menu subset. The twelve-action subset used for the codec curves and supplementary predictor analysis keeps the following combinations of metadata configuration and QP (dense edges at QP 10/18/25; skeleton+pose at QP 10; skeleton at QP 18/25/31; anchors only at QP 25/31/37; sparse edges at QP 18/25). The complete 5×5 grid (25 actions, 5-bit index) was allocated as well (Figure 5, Table 6): its oracle frontier lies above the twelve-action frontier by at most 0.05 dB (0.03 dB on average over the log-rate range), the 13 complementary actions carry at most 24% of the intervals at any budget (12% on average), and three of them are never selected (dense edges at QP 37, skeleton + pose at QP 31 and 37). Primary representation-selection and value-learning results use the complete unpruned 25-action Cartesian grid; the twelve-action subset is used only for codec curves and supplementary analyses. The GoPro/Adobe240 menu uses sparse-edge configurations at 25% density with $k \in \{1, 3\}$ and at 75% density with $k = 3$ on anchor QPs $\{8, 15, 20, 25, 39\}$ with

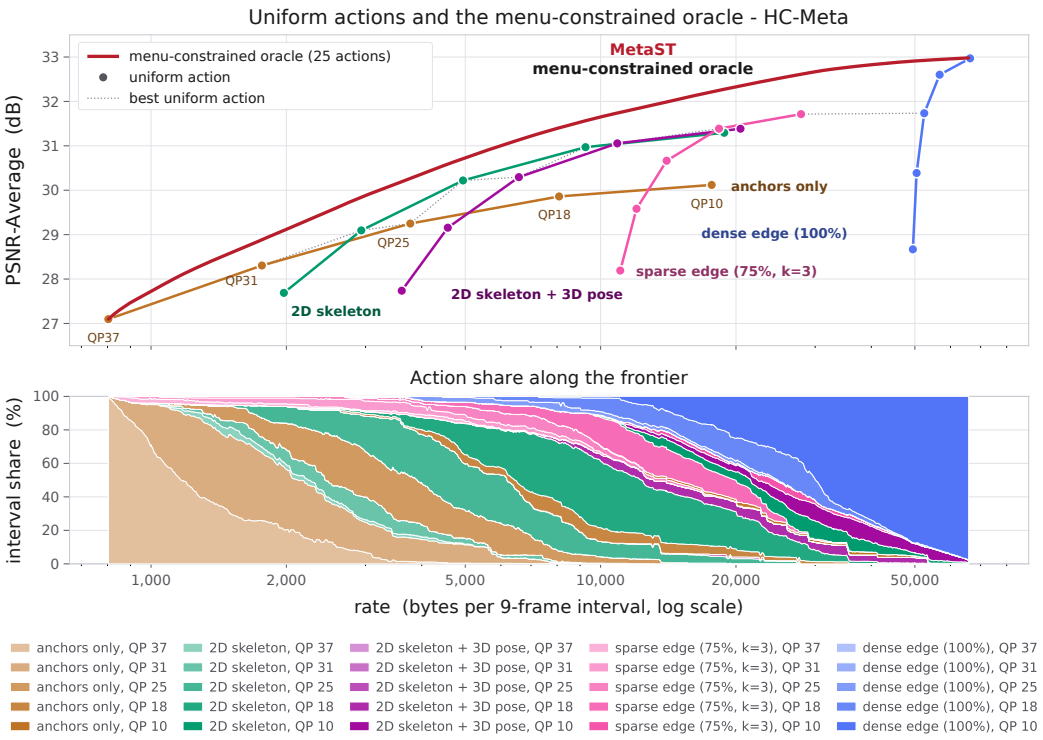


Figure 12: Complete action-level legend for the allocation analysis of Figure 5. The plotted data are identical; this version retains every action label.

per-configuration QP bands (on Adobe240 extended by anchors-only actions at QP 20, 23, 25, 27, 29 and the one-map 25% action at QP 25 and 39); the known positions of each sparse configuration follow Appendix C.

Menu construction and evaluation scope. The 12-action subset was chosen using oracle responses on the same 50 HC-Meta evaluation clips: a 61-action H.266 all-intra exact-DP study at 15 frontier points identified active metadata configurations and their contiguous QP bands. This selection precedes and is outside the predictor's cross-validation folds. The supplementary twelve-action predictor results are conditional on this menu. The primary study in Table 3 uses the complete Cartesian grid, with no action pruning; both studies fit and select predictor models within training folds only.

Exact solver. For each sequence the chain DP runs over intervals with the previous action as state at 1-byte budget resolution, with a lower (ceil) and upper (floor) rounding pass; sequences are merged with a budget ladder (resolutions 4, 16 and 64 bytes above 6, 20 and 66 KB per interval), allocations are recovered by backtracking, every recovered allocation is re-billed exactly, and the certified gap between the lower staircase and its upper bound is reported (0 at every point of the HC-Meta frontier; 658 recovered points on the twelve-action menu between 804 and 65,864 B per interval). For learned allocation and for every predictor ablation the same billing is used inside a Lagrangian chain DP (Viterbi with $|\mathcal{A}|$ states: 25 for the primary study and 12 for the supplementary study) swept over 4,001 log-spaced multipliers in $[10^{-8}, 10^{-1}]$ dB/B; at a budget the feasible allocation with the largest exact bytes is taken, and its true quality is looked up. On the twelve-action menu, exact values passed through this solver are about 0.01 dB below the exact staircase on average (maximum 0.09 dB); each study measures all arms against its own exact oracle, with the menu-specific signalling cost.

Fixed-budget configurations on GoPro and Adobe240. The tables use a fixed sparse-edge configuration for each row ($k \in \{1, 3\}$ at source-map density 100%), with anchor QP selected per sequence to meet B_0, including map framing and signalling. Figure 8 uses an adaptive $k = 3$/dense

Table 7: Mixed-QP anchor-stream evaluation over all 50 HC-Meta clips at each budget. Rate ratios are actual/billed for anchors, and actual anchors plus unchanged metadata and signalling charges/billed total for the sequence. PSNR columns are dataset means in dB; "decoded" uses the actual anchor streams with cached sparse-edge completion. The final column counts intervals whose Average PSNR changes by more than 0.5 dB.

Budget (B)	Rate ratio		Average PSNR		Center PSNR		$\lvert\Delta\rvert > 0.5$
	Anchor	Sequence	Model	Decoded	Model	Decoded	(of 900)
1,000	0.995	0.9949	27.676	27.648	28.203	28.174	28
1,500	1.011	1.0097	28.539	28.471	29.123	29.037	54
2,500	1.023	1.0192	29.487	29.459	30.153	30.111	43
4,000	1.012	1.0093	30.333	30.316	31.029	31.002	15
6,000	1.004	1.0025	30.978	30.965	31.743	31.726	8
8,999	0.999	0.9993	31.517	31.520	32.254	32.254	5
13,999	0.998	0.9991	31.968	31.967	32.666	32.664	1
20,998	0.995	0.9977	32.357	32.358	33.011	33.012	0
32,997	0.996	0.9980	32.717	32.718	33.354	33.355	0
49,998	0.997	0.9989	32.908	32.908	33.511	33.511	0

variant with an anchor ladder that also includes VVC all-intra QP 0, 17 and 37 at the same B_0 (GoPro 32.45 dB; Adobe240 30.79 dB). Table 1 reports uniform $k = 3$ (32.34 and 30.73 dB).

E.1 MIXED-QP ANCHOR-STREAM REALIZATION

We realize ten allocations on the reported H.264 frontier as predictive anchor streams, with one stream per clip. At a boundary shared by actions with different QPs, the stream encodes the picture once at the finer QP; subsequent P pictures reference the decoded history. The billing model instead charges that picture once per distinct QP using the constant-QP streams. Across the ten allocations, 1–32 shared pictures have differing adjacent QPs. The H.266 protocol fixes anchor QP per clip and has no such switches.

Encoding and calibration. We use x264 0.164 with the same IPPP settings as the rate–distortion experiment (preset medium, no B pictures or scenecut, infinite keyframe interval and four references). Per-picture QPs are supplied through a QP file in CRF mode with `--aq-mode 0 --no-mbtree`; the initial I picture uses the assigned P-picture QP minus three, matching x264's constant-QP convention. On constant-QP sequences, this recipe reproduces the original streams within 0.1% in bytes and 0.01 dB in anchor PSNR. We then encode the allocated QP sequences, allowing references to cross QP switches.

Receiver evaluation. For each budget, we decode the actual anchor streams and reconstruct all 900 intervals with their allocated actions. To isolate anchor-stream effects, sparse-edge completion retains the action's cached single-QP maps, including endpoint maps; the other metadata inputs and charges remain unchanged. The sequence-rate ratio in Table 7 therefore substitutes actual anchor bytes into the existing sequence bill. Quality compares these reconstructions with the action values measured from constant-QP anchors. Pooling the original action values reproduces each reported frontier marker within 10^{-3} dB.

Rate and quality agreement. Actual anchor bytes differ from their billed values by -0.5 to $+2.3\%$. With metadata charges unchanged, the sequence-rate difference is -0.51 to $+1.92\%$. The dataset Average PSNR changes by less than 0.07 dB at every budget, and Center PSNR by less than 0.09 dB. The larger changes occur at low rates, where switches are more frequent and finer-QP pictures can reference coarser reconstructions. Up to 54 of 900 intervals change by more than 0.5 dB at a low-rate budget; none do at the three highest budgets. Thus the aggregate rate–quality agreement coexists with localized sensitivity to predictive coding context.

F ALLOCATION OBJECTIVE AND DYNAMIC PROGRAMS

Common-reference invariance. Let $Q_i(a) = \mathcal{Q}(X_i, \hat{X}_i^a)$. For any feasible action sequence,

$$\sum_i V_i(a_i) = \sum_i Q_i(a_i) - \sum_i Q_i(a_0). \tag{8}$$

The reference term does not depend on the chosen actions. Maximizing value is therefore equivalent to maximizing summed reconstruction quality under the same rate constraint. The fixed receiver system, scored frames and rate model remain unchanged across candidate allocations. We use Average PSNR as $\mathcal{Q}$, a mean of frame-level PSNR scores; each interval contributes the same number of scored frames, so maximizing the sum of interval qualities also maximizes this dataset mean.

Constrained dynamic program. The menu-constrained oracle is the exact measured-value optimum within the fixed candidate action space and stated sequence rate model of equation 2. For integer costs, let $F_i(b, a)$ be the largest cumulative value through interval i, with billed rate at most b and final action a. The recurrence is

$$F_i(b, a) = V_i(a) + \max_{a' \in \mathcal{A}:\, r_i(a', a) \le b} F_{i-1}\big(b - r_i(a', a), a'\big). \tag{9}$$

The start state has zero value, infeasible states have value $-\infty$, and the menu-constrained optimum is $\max_a F_{N-1}(B, a)$. Backtracking recovers the actions. A codec constraint is enforced by allowing only legal transitions or by conditioning the chain on a shared codec setting. Multiple clips form independent chains whose rate–value frontiers can be combined under an aggregate budget. The fixed-budget protocol instead enforces a separate reference budget for each clip.

When billed costs are fractional, floor and ceiling passes provide bounds for the discretized search. The recovered allocations are checked against the original billing model, including sequence-level rounding of signalling. The certified gap is zero at every reported HC-Meta menu-constrained oracle point.

Lagrangian dynamic program. With predicted values and fixed λ, the budget dimension is removed:

$$H_i(a; \lambda) = \hat{V}_i(a) + \max_{a' \in \mathcal{A}} \left[H_{i-1}(a'; \lambda) - \lambda r_i(a', a) \right]. \tag{10}$$

This is exact for the penalized problem at that multiplier, under the allowed transitions. The sweep produces a set of candidate sequences; for the reported predictor experiments, the feasible candidate with the largest billed rate is selected at each requested budget, following the original evaluation protocol. A finite sweep need not recover every supported point, and even an exhaustive multiplier search need not recover unsupported points of a discrete constrained frontier. On the twelve-action subset, passing measured values through the same sweep gives mean regret of about 0.01 dB and maximum regret of 0.09 dB relative to the budget-indexed oracle. Regret to each menu's exact oracle includes the sweep's optimization error as well as prediction error.

Local learning scores and global allocation. The action costs $c_i(a)$ used in Equation 4 are the local costs used to construct the training rate–value frontiers. They provide a listwise preference target over the candidates of one interval. At deployment, Equations 2 and 10 use the transition costs $r_i(a', a)$, which also account for shared boundaries. Thus, the listwise term is a surrogate for allocation preferences, while the dynamic program enforces the sequence-level rate model.

Prediction error and decision quality. To prove equation 7, add and subtract the predicted sums at $\mathbf{a}^*$ and $\hat{\mathbf{a}}$. Each prediction-error sum is bounded by $\sum_i \epsilon_i$, and the predicted-value difference is at most η. The latter follows from the assumed optimization gap because $\mathbf{a}^*$ is feasible under the same budget. This is a deterministic statement about a fixed candidate set and rate model. The experiments directly measure allocation regret rather than estimating a uniform prediction-error bound.

G METADATA VALUE PREDICTOR: INPUTS, MODEL, TRAINING AND VALIDATION

Labels. For every interval and action, $V_i(a)$ is the Average PSNR difference between its receiver reconstruction and the anchors-only reference at QP 37, from the same exhaustive runs that define the menu-constrained oracle. The predictor inputs contain no receiver reconstruction.

Inputs. Per (interval, action), 22 continuous features and two one-hot codes (metadata configuration, 5; QP, 5), z-scored on the training fold: log anchor bytes at the action's QP; $\log(1 + \text{metadata bytes})$; log of the action's rate relative to the reference; QP$/51$; seven metadata-configuration descriptors (sparse density, sparse k, dense flag, anchors-only flag, pose-receiver flag, skeleton flag, 3D-pose flag); log reference rate of the interval; log ratio of the cheapest anchor to the reference; Y-PSNR of the decoded anchor at the action's QP against the clean LR anchor; payload density (fraction of edge pixels in the transmitted maps) and payload overlap (fraction of payload pixels where the source differs from naive linear interpolation by more than 10/255); and six interval scalars: motion energy, PSNR of linear interpolation for frames 1–7, source Canny density, mean gradient magnitude, and bicubic-upsampling PSNR of the two anchors. Features read only the source frames, the clean LR frames, the decoded anchors at the five QPs, the dense edge payload and the pose renderings; extraction takes 75 s for the 900 intervals on 16 CPU workers.

Model and loss. g_θ: Linear($32 \to 64$), SiLU, Dropout(0.1), concatenation with the mean of the hidden vector over the interval's candidate actions (25 in the primary study), Linear($128 \to 64$), SiLU, Linear($64 \to 1$) initialized to zero, so training starts from the item prior. Prediction follows equation 3. Loss equation 5 with smooth-L_1 ($\delta = 0.1$), $\lambda_{\text{dec}} = 0.2$ and $\tau = 0.1$ (a listwise softmax cross-entropy in the sense of Cao et al. (2007)), and Λ the 10th, 30th, 50th, 70th and 90th percentiles of the positive slopes of the per-interval rate–value convex hulls of the training intervals. AdamW, learning rate 10^{-3}, weight decay 10^{-4}, gradient clipping 1.0, batches of 8 intervals (all candidate actions per interval), up to 200 epochs with checkpoints every 5 epochs; the epoch is chosen by inner four-fold early stopping on the inner out-of-fold allocation regret (three-point smoothed, earliest minimum), after which the model is retrained on the whole training fold. Training and evaluation use CPUs.

Validation. Intervals from one clip remain together. Groups are consecutive-clip-id blocks (gap > 10) for SportsSloMo and video identifiers for Pexels: 16 groups per source subset, covering 207 and 693 intervals. Five repeats of stratified grouped five-fold cross-validation (147–245 intervals per fold; seeds 20260920–20260924) produce out-of-fold predictions. The primary study uses the unpruned 5×5 grid. The prior, normalization, training slopes, early stopping and ridge penalties are fitted within training folds. Regret is the oracle PSNR minus the true PSNR of the allocation selected from predictions, averaged over the evaluated budgets of the corresponding frontier. Both candidate sets use identical folds, model-specific features, optimization protocol and seeds. The supplementary twelve-action study evaluates 658 budgets and retains its original results.

Comparators. The *item prior* uses training-fold mean values per action. *Rates-only ridge* uses 36 rate features, with its penalty selected by inner out-of-fold allocation regret. *Ridge on MVP inputs* uses the same 22 continuous descriptors and metadata-configuration/QP codes as MVP. The *permutation control* reassigns MVP predictions across source intervals within each fold. All arms use the same Lagrangian sweep and exact rate recomputation. No predictor arm calls the receiver at deployment.

Full-grid significance and decision quality. We use 2,000 paired bootstrap resamples, stratified over the 32 clip groups, re-solving allocations for each resample. On the complete 25-action grid, the mean-regret reduction from rates-only ridge to MVP is 0.072 dB, with a paired 95% interval of $[0.031, 0.122]$ dB. All resamples favour MVP. This satisfies the pre-specified criterion: a positive lower confidence bound and a point improvement of at least 0.05 dB. The 90th-percentile regret falls from 0.412 to 0.216 dB, and the per-interval rank correlation rises from 0.932 to 0.951. Permuting predictions raises mean regret to 0.3488 dB, supporting the role of source-dependent valuation. The full-grid predictor study takes 1,049 s on CPUs.

Table 8: Complete predictor results on the unpruned 25-action grid. Regret and BD-PSNR are in dB; BD-PSNR is relative to the exact oracle within this grid and the stated sequence rate model. Rank correlation measures action-value ordering within an interval.

Value model	Mean regret	90th pct.	Gain recovered	BD-PSNR	Rank corr.
Item prior	0.2515	0.441	65.0%	−0.250	0.907
Rates-only ridge	0.2200	0.412	69.3%	−0.219	0.932
Ridge on MVP inputs	0.2295	0.342	68.1%	−0.228	0.921
MVP, permuted	0.3488	0.489	51.4%	−0.347	0.876
MVP	**0.1480**	**0.216**	**79.4%**	**−0.147**	**0.951**

Table 9: Supplementary predictor results on the benchmark-derived twelve-action subset. Regret is in dB; gain recovered is relative to the best uniform action; BD-PSNR is relative to the exact oracle within this subset and the stated sequence rate model.

Value model	Mean regret	Gain recovered	BD-PSNR
Item prior	0.2245	67.7%	−0.223
Rates-only ridge	0.2193	68.4%	−0.218
Ridge on MVP inputs	0.2303	66.8%	−0.229
MVP	**0.1368**	**80.3%**	**−0.136**

Table 10: Supplementary regret by rate band on the twelve-action HC-Meta subset (dB; lower is better). Parentheses give the number of evaluated budgets. Learning brings its largest improvement over rates-only ridge in the 2–10 KB band.

Value model	$<$ 2 KB (137)	2–10 KB (243)	10–30 KB (165)	$>$ 30 KB (113)
Item prior	0.236	0.307	0.216	0.045
Rates-only ridge	0.187	0.337	0.190	0.049
MVP	**0.112**	**0.186**	**0.151**	**0.042**

Complementary twelve-action analysis. The subset uses the same features, repeated grouped cross-validation and optimization protocol. Table 9 reports 0.1368 dB regret and 80.3% gain recovered for MVP. Its reduction from rates-only ridge is 0.0825 dB, with a paired 95% interval of $[0.049, 0.123]$ dB. Permuting predictions raises regret to 0.291 dB, exceeding the prior by about 0.067 dB, with a 95% interval of $[0.022, 0.121]$ dB. Both candidate sets support learned, source-dependent valuation; the full-grid results establish this benefit directly over all combinations of metadata configuration and QP.

Rate dependence on the subset. Table 10 partitions the twelve-action analysis into four rate bands. At 2–10 KB, MVP reduces regret relative to rates-only ridge from 0.337 to 0.186 dB (45%). Maximum regret across the subset's frontier falls from 0.53 to 0.28 dB. Above 30 KB, all three models approach the oracle as dense edges become broadly affordable. The twelve-action run takes 838 s on CPUs. These rate-band measurements complement the aggregate full-grid evaluation in Table 3.

H Baselines and evaluation protocol

Two-stage baselines. Following the composed-baseline convention of Zooming SlowMo, the VFI stage runs first on the LR anchor sequence (producing the seven interior LR frames of each interval at 320×180), and the VSR stage runs on the resulting nine-frame LR sequence, including the anchors, to produce the nine HR frames. The official checkpoints are Super SloMo, DAIN, AMT-S (GoPro-trained), EDVR (REDS, large model), BasicVSR (REDS4) and the official DOVE Stage-1 model; bicubic upsampling is the trivial VSR stage. The DOVE Stage-1 baseline uses the officially released weights through a verified port of its inference path.

One-stage baselines. Zooming SlowMo, TMNet, VideoINR, MoTIF and BF-STVSR (with and without $\mathcal{L}_{\mathrm{RAFT}}$, the flow-supervision loss that uses RAFT (Teed & Deng, 2020) during training) with their official checkpoints, run through the BF-STVSR evaluation harness on our interval lists. Under

the center protocol (sampling indices $[0, 0, 4, 8, 8]$), Zooming SlowMo produces positions 0, 4 and 8; its frame 0 is its own $\times 4$ super-resolution of the anchor. It therefore supports Center, the mean of frames 0 and 4, but not Average over frames 0–7.

Metrics. PSNR on the BT.601 luma channel without border crop; SSIM (Wang et al., 2004) with the MATLAB-style Gaussian window on the luma channel after a 5-pixel border crop; both from the BF-STVSR metric implementation. Average is the mean over frames 0–7 of an interval and Center the mean of frames 0 and 4; a shared frame 8 is scored as frame 0 of the next interval, while the final boundary frame is not scored. Incomplete trailing intervals are excluded. Average results pool the distinct scored frames within a dataset (SportsSloMo and Pexels together for HC-Meta). The 1,522 complete GoPro intervals and 4,410 Adobe240 intervals therefore score 12,176 and 35,280 frames, respectively.

Training data of the compared methods. The baselines are used as released: VideoINR, MoTIF and BF-STVSR are trained on Adobe240; Zooming SlowMo and TMNet on Vimeo-90K (Xue et al., 2019); EDVR and BasicVSR on REDS (Nah et al., 2019); AMT-S on GoPro; Super SloMo on Adobe240; DAIN on Vimeo-90K; DOVE on its own real-world VSR corpus.

I Complete experimental results

The complete baseline comparison and metadata ablation are in main-text Tables 1 and 2. Table 11 below tabulates the rate–distortion readings, and Table 6 lists all uniform actions.

Table 11: Values behind Figures 4 and 6a (PSNR Average in dB; rate in KB per interval). MetaST Oracle is the exact optimum within each fixed menu and stated sequence rate model, interpolated at the baseline's rate; H.264 direct values are interpolated, H.266 direct values are measured points (rate in parentheses).

HC-Meta, H.264					HC-Meta, H.266					GoPro, H.264			
			MetaST					MetaST					MetaST
KB	BF	AMT	Oracle	direct	KB	BF	AMT	Oracle	direct	KB	BF	AMT	Oracle
0.80	26.63	26.72	–	–	0.21	24.32	24.32	24.7	–	2.24	27.67	26.97	24.23
1.18	27.18	27.21	28.07	–	0.35	25.47	25.50	25.9	–	3.07	28.15	27.24	25.46
1.76	27.69	27.63	28.84	–	0.65	26.53	26.51	27.0	26.6 (0.75)	4.08	28.56	27.44	26.49
2.60	28.13	27.93	29.56	–	1.23	27.47	27.30	28.3	28.1 (1.24)	5.29	28.88	27.58	27.48
3.77	28.51	28.15	30.23	26.5	2.04	28.07	27.72	29.2	29.6 (2.0)	7.68	29.27	27.71	28.79
5.30	28.79	28.29	30.79	28.6	3.73	28.64	28.03	30.3	32.1 (3.9)	9.69	29.47	27.77	29.56
8.08	29.05	28.39	31.39	31.1						13.28	29.66	27.82	30.57
10.89	29.18	28.43	31.72	32.4						17.68	29.77	27.84	31.07
17.66	29.30	28.45	32.20	34.9						24.94	29.84	27.85	31.31
										33.97	29.89	27.87	31.41

J The H.266 setting

H.266/vvenc uses the anchor and direct-compression settings in Appendix E. Baselines and MetaST receive the corresponding decoded anchors; sparse metadata is interpolated from those anchors and the transmitted source maps. Figure 4b and Table 11 report the resulting comparison. The MetaST oracle spans 0.13–58.5 KB per interval (23.3–33.1 dB). The strong direct-compression curve narrows the rate range where the reconstruction interface is preferable, while metadata allocation remains above the pixel-only STVSR curves in the plotted range.

K General video: menu and rate–distortion

On GoPro, MetaST's frontier begins at 2.2 KB per interval with the cheapest action (one 25%-density map on QP-39 anchors, 24.1 dB) and is below BF-STVSR until about 10 KB because the anchor ladder of its menu has a large gap between QP 39 (493 B per interval) and QP 25 (4,081 B) and no pose or skeleton family is available to fill the low-rate range cheaply. Above 10 KB sparse-edge actions dominate and the lead grows to 1.5 dB. On Adobe240 (Figure 6b) the menu also contains

anchors-only actions at QP 20–29 and a one-map action at QP 39, so the frontier starts at 1.8 KB per interval; between 2 and 7 KB it is 0.4–0.7 dB below BF-STVSR, 0.2 dB below at 9 KB, ahead from about 11 KB and 0.3 dB ahead at the top of the range. The GoPro menu contains no anchors-only action; adding one, as on Adobe240 and HC-Meta, extends the frontier to lower rates.

L CONTOURS AS TRANSMITTED METADATA

Human contours are a representation family in HC-Meta and Figure 1, evaluated separately from the primary 25-action grid and the twelve-action codec subset. In a fifteen-action menu with contour actions at QP 10, 18 and 25 (interior contours coded with the edge codec, 1,965 B per interval; contours at the anchor positions extracted from the decoded anchors at no cost), the contour actions reach the frontier on 386 of 658 points but with tiny shares (at most 4.6% of intervals) and move it by 0.005 dB (BD-PSNR), with contour actions 2.2–2.7 dB below edge actions at comparable rates in this implementation.

The contour experiment reuses the same polyline coding pipeline and settings as the edge maps. A more powerful codec designed specifically for object contours could better exploit boundary geometry and temporal coherence, making contour compression an important direction for improving this representation. The fixed receiver system assigns edges and contours to the same receiver configuration; contour-specific adaptation could complement improved coding.

M COMPUTE

All inference ran on a machine with eight A100-40GB GPUs. The HC-Meta H.264 rate–distortion experiment cost about 26 GPU-hours for the twelve MetaST actions (a further 76 GPU-hours for the 13 complementary actions of the full grid), 7 GPU-hours for BF-STVSR on nine rungs and 14 GPU-hours for AMT-S + DOVE Stage-1; the fixed-budget MetaST rows on GoPro and Adobe240 cost 54 GPU-hours (22,793 intervals); sparse-edge tree construction is CPU/GPU-bound at about 100 s per 16 intervals and QP. The exact allocation of the HC-Meta menu takes 842 s on 16 CPU threads with a 42 GB peak (967 s for the 25-action grid); the full-grid predictor study takes 1,049 s on CPUs (838 s for the twelve-action study). The mixed-QP validation performs 9,000 interval reconstructions across ten budgets in 2 h 24 min on eight GPUs, completing 198 shard jobs. Receiver fine-tuning ran on four A100-80GB at 16.4 s per iteration.

N EXTENDED RELATED WORK

Frame interpolation and video super-resolution. Video frame interpolation estimates motion between anchors and synthesizes intermediate frames, with kernel-based (Niklaus et al., 2017; Lee et al., 2020), flow-based (Jiang et al., 2018; Bao et al., 2019; Xu et al., 2019; Sim et al., 2021; Huang et al., 2022; Kong et al., 2022; Reda et al., 2022; Li et al., 2023) and attention-based (Choi et al., 2020; Zhang et al., 2023a) estimators; diffusion models have recently been used to synthesize plausible in-betweens over long gaps (Danier et al., 2024; Jain et al., 2024; Wang et al., 2025b), and human-centric benchmarks and keyframe interpolators with 2D and 3D pose guidance target the articulated motion that makes sports video hard (Chen & Jiang, 2024; Guo et al., 2025). Video super-resolution aggregates temporal evidence with dynamic filters, back-projection, deformable alignment, recurrent propagation and transformers (Jo et al., 2018; Haris et al., 2019; Wang et al., 2019; Tian et al., 2020; Chan et al., 2021; 2022a; Liang et al., 2022; Shi et al., 2022; Liang et al., 2024), and real-world variants train on synthetic degradation pipelines (Wang et al., 2021; Zhang et al., 2021; Chan et al., 2022b). Composing an interpolator with a super-resolver is the classic two-stage STVSR baseline (Xiang et al., 2020), which we evaluate with Super SloMo, DAIN and AMT-S followed by bicubic upsampling, EDVR, BasicVSR or DOVE.

Space-time video super-resolution. One-stage networks share spatio-temporal features across both tasks (Xiang et al., 2020; Haris et al., 2020; Xu et al., 2021; Geng et al., 2022; Huang et al., 2024), and continuous formulations decode arbitrary scales and times from implicit neural representations (Chen et al., 2022; 2023); BF-STVSR combines B-spline temporal mapping with Fourier spatial features and is the strongest published method on the GoPro and Adobe240 protocol we adopt (Kim et al., 2025). Methods in this line optimize the map from received pixels to output video under a fixed

interface; MetaST changes what is presented to the receiver and uses these methods, run on the same anchors, as its pixel-only comparison.

Generative and one-step diffusion restoration. Latent diffusion (Rombach et al., 2022) extended to video (Ho et al., 2022; Blattmann et al., 2023; Yang et al., 2024b) provides priors that synthesize detail only weakly present in the input. For super-resolution, StableSR and ResShift adapt image diffusion priors (Wang et al., 2024a; Yue et al., 2023), and distillation and consistency training compress sampling to a single step (Song et al., 2023; Yin et al., 2024; Wang et al., 2024b; Wu et al., 2024). Video restorers add temporal consistency through motion guidance, text-to-video priors or long-range attention (Zhou et al., 2024; Yang et al., 2024a; Xie et al., 2025; Wang et al., 2025a); DOVE adapts CogVideoX-2B to one-step real-world VSR (Chen et al., 2025) and provides the architecture we adapt for STVSR. Such receivers remain ambiguous where the anchors carry no evidence, which is the gap that transmitted metadata fills.

Guided restoration and structure-conditioned generation. Structural side information has long improved super-resolution when it is available at the receiver: semantic feature transforms (Wang et al., 2018), gradient guidance (Ma et al., 2020) and reference images (Zhang et al., 2019; Yang et al., 2020). Diffusion models accept edges, depth, segmentation or pose through control branches (Zhang et al., 2023b; Mou et al., 2024), and pose-driven human animation conditions video generation on skeletons or SMPL renderings (Hu, 2024; Xu et al., 2024; Zhu et al., 2024). These guidance interfaces generally do not optimize a shared transmission budget; MetaST treats every structural cue as a transmitted, byte-counted stream whose value must be predicted before it is sent, and it fuses the cues through the receiver's own VAE and attention rather than a modality-specific branch.

Rate allocation, side information and generative compression. Rate–distortion optimization allocates bits among coding decisions by Lagrangian or dynamic-programming search (Shoham & Gersho, 1988; Ortega & Ramchandran, 1998; Sullivan & Wiegand, 1998), scalable coding prices enhancement layers on top of a base stream (Schwarz et al., 2007), and second-generation and vertex-based schemes code contours instead of pixels (Kunt et al., 1985; Schuster et al., 1998; Yang et al., 2026). Learned codecs optimize rate and distortion end to end (Ballé et al., 2017; 2018; Lu et al., 2019; Li et al., 2021), and generative codecs trade distortion for realism at very low rates, down to text and sketch descriptions decoded by diffusion models (Agustsson et al., 2019; Mentzer et al., 2020; Blau & Michaeli, 2018; 2019; Lei et al., 2023; Careil et al., 2024). Video coding for machines and joint source–channel coding likewise adapt what is transmitted to the consumer of the signal (Duan et al., 2020; Bourtsoulatze et al., 2019). MetaST combines standard anchors and heterogeneous metadata, allocating bytes by their learned reconstruction value to a fixed receiver system.

Value of information and adaptive acquisition. Value-of-information methods price observations by their expected benefit (Weiss & Taskar, 2013); adaptive acquisition learns policies with set encoders, partial VAEs or conditional mutual information (Shim et al., 2018; Ma et al., 2019; Covert et al., 2023; Gadgil et al., 2024). MetaST applies this principle to source representations, learning their value from a fixed receiver system's response and using listwise supervision (Cao et al., 2007) to guide budgeted transmission.