DA-Flow: Degradation-Aware Optical Flow Estimation with Diffusion Models

Jaewon Min[1], Jaeeun Lee[1], Yeji Choi[1], Paul Hyunbin Cho[1], Jin Hyeon Kim[1], Tae-Young Lee[2], Jongsik Ahn[2], Hwayeong Lee[2], Seonghyun Park[2], and Seungryong Kim[1][†]

[1]KAIST AI   [2]Hanwha Systems
https://cvlab-kaist.github.io/DA-Flow

Abstract. Optical flow models trained on high-quality data often degrade severely when confronted with real-world corruptions such as blur, noise, and compression artifacts. To overcome this limitation, we formulate **Degradation-Aware Optical Flow**, a new task targeting accurate dense correspondence estimation from real-world corrupted videos. Our key insight is that the intermediate representations of image restoration diffusion models are inherently corruption-aware but lack temporal awareness. To address this limitation, we lift the model to attend across adjacent frames via full spatio-temporal attention, and empirically demonstrate that the resulting features exhibit zero-shot correspondence capabilities. Based on this finding, we present **DA-Flow**, a hybrid architecture that fuses these diffusion features with convolutional features within an iterative refinement framework. DA-Flow substantially outperforms existing optical flow methods under severe degradation across multiple benchmarks.

1   Introduction

Optical flow estimation, the task of estimating per-pixel motion fields between consecutive video frames, is a fundamental dense correspondence problem in computer vision. With the advent of deep neural networks [25, 31, 37], recent methods have achieved remarkable accuracy.

Real-world videos are rarely clean; motion blur, sensor noise, compression artifacts, and low resolution frequently co-exist, severely degrading visual quality. Despite the prevalence of such degradations, how optical flow models behave under such degradations remains largely unexplored. Recently, RobustSpring [28] provided the first comprehensive study on the robustness of dense matching models, benchmarking their generalization from clean synthetic training data to a wide spectrum of real-world degradations. Despite this systematic analysis, a fundamental question remains open: **is it truly impossible to accurately estimate optical flow from corrupted inputs?**

Motivated by this question, we shift the focus from robustness to accuracy by introducing a new task, **Degradation-Aware Optical Flow**, that directly
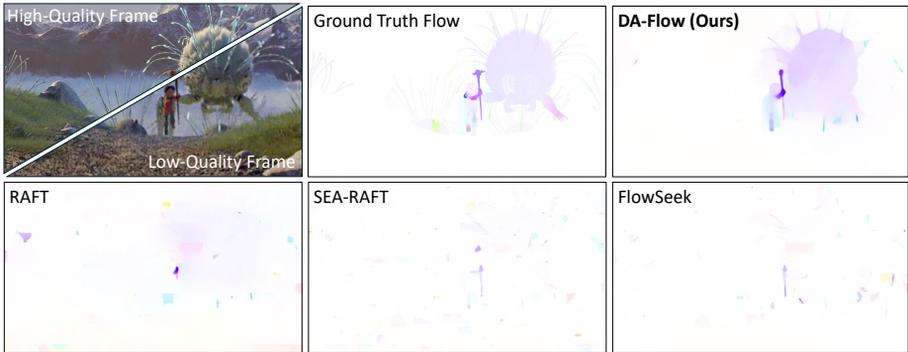

[†]: Corresponding author

Fig. 1: Qualitative comparison of DA-Flow with baselines [25, 31, 37] on Spring benchmark [20]. Under severe degradations, existing optical flow methods fail to recover reliable correspondences, whereas our DA-Flow accurately estimates the underlying motion.

estimates dense correspondences from severely degraded inputs. This task is fundamentally ill-posed: degradations destroy fine textures and attenuate motion boundaries, leaving insufficient visual evidence for reliable matching. In such regimes, correspondence estimation is not merely a matter of distribution shift but becomes inherently ambiguous. Simply augmenting clean training data with synthetic corruptions does not adequately address this challenge; what is needed are representations that are both rich enough to preserve spatial structure for dense matching and sensitive to degradation patterns to recover information lost during corruption, as illustrated in Fig. 1.

Recent works [12–14, 22, 23, 30, 32, 41] have shown that intermediate features of diffusion models encode rich structural and semantic information, achieving strong performance on correspondence tasks [22, 23, 30, 41] as well as diverse downstream vision tasks such as depth estimation [12, 13] and segmentation [14, 32]. These findings suggest that diffusion representations capture geometric and structural cues far beyond what is needed for generation alone. Building on this insight, we observe that diffusion models trained for image restoration [1, 5, 8, 17] offer an even more suitable foundation. Image restoration is likewise a highly underdetermined inverse problem, and models trained for this task must learn to recover clean structures from degraded inputs. As a result, their intermediate features naturally encode degradation patterns while preserving underlying scene geometry. This motivates our core design choice: leveraging restoration diffusion features to obtain representations that are degradation-aware, structurally rich for dense matching, and equipped with generative priors that can reason beyond corrupted observations. However, these features lack temporal awareness, limiting their effectiveness in producing accurate features for optical flow estimation.

Since our task involves video, a natural consideration is whether video restoration diffusion models [6, 39, 43], which jointly model degradation and temporal dynamics, could serve as the backbone. However, such models often encode a

stack of degraded frames into a temporally compressed latent representation through 3D convolutions or temporal attention. This produces a shared latent tensor where the temporal axis is entangled early in the encoding pipeline. While this design suits perceptual video restoration, where temporal smoothness and global consistency are desirable, it is structurally misaligned with dense correspondence estimation. Optical flow requires comparing spatial features extracted independently from each frame to establish pixel-level correspondences. When degraded frames are jointly encoded into a shared spatio-temporal latent space, their per-frame spatial structure is no longer preserved as separable entities, making the representation ill-suited for explicit pairwise feature matching.

To reconcile degradation-aware representation with the structural requirements of dense matching, we take a different approach: instead of adopting a monolithic video diffusion backbone, we start from a pretrained image restoration diffusion model [8] that preserves full spatial resolution at the frame level. We then lift it to handle multiple frames by injecting cross-frame attention across all layers. This design maintains independent spatial latents for each frame, which is crucial for dense matching, while enabling controlled temporal interaction for motion reasoning. By inheriting strong degradation-aware priors from image restoration pretraining and avoiding temporal latent collapse, our architecture yields representations intrinsically suited for dense correspondence estimation under severe corruption, while remaining substantially more efficient than video diffusion architectures.

Building on this representation, we introduce **DA-Flow**, a **D**egradation-**A**ware Optical **Flow** network built on top of RAFT [31]. As illustrated in Fig. 2, DA-Flow combines upsampled diffusion features from the lifted model with conventional CNN-based encoder features into a hybrid representation, enabling the correlation and iterative update stages to benefit from both degradation-aware structural priors and fine-grained spatial detail. Since ground-truth optical flow for real-world degraded videos is unavailable, we train DA-Flow using pseudo ground-truth flow generated by applying a pretrained flow model [37] to a high-quality video, while feeding the corresponding degraded frames as input. We evaluate on degraded versions of established optical flow benchmarks [3, 20, 35] constructed via realistic degradation pipelines [4, 36], and demonstrate that DA-Flow achieves accurate flow estimation even under severe corruption where existing methods fail.

Our main contributions are as follows:

- We formulate **Degradation-Aware Optical Flow**, a new task that estimates accurate dense correspondences from severely corrupted videos.
- We lift a pretrained image restoration diffusion model by introducing inter-frame attention and verify that its features encode geometric correspondence even under severe corruption.
- We introduce **DA-Flow**, a degradation-aware optical flow network that substantially outperforms existing methods on degraded inputs.

2   Related Work

Optical flow estimation. Optical flow estimation aims to model dense pixel-level motion between consecutive frames and serves as a fundamental component in various video-related tasks, including video generation and scene reconstruction. Modern deep learning approaches have significantly advanced flow estimation, among which RAFT [31] establishes a strong baseline by combining dense all-pairs correlation with recurrent iterative refinement. Building on this framework, SEA-RAFT [37] improves efficiency and robustness through a simplified update mechanism and a mixed Laplace loss. Recently, FlowSeek [25] further enhances flow estimation by incorporating stronger priors and more efficient architectures, achieving impressive performance on high-quality inputs.

Geometric correspondence. Establishing reliable geometric correspondence is fundamental to many vision tasks. Classical pipelines rely on handcrafted local descriptors [2,19], while learned CNN and transformer models have substantially improved matching robustness [7,21,29,33]. However, accurately modeling *dense* correspondences for fine-grained geometric alignment remains challenging, especially under large appearance variations. Recent studies show that diffusion models provide spatially informative representations for correspondence. In particular, DIFT demonstrates that correspondence can emerge from image diffusion features without explicit supervision or task-specific fine-tuning [30]. Complementary to diffusion features, DINOv2 offers strong semantic representations, and a simple fusion of diffusion and DINOv2 features yields more robust dense correspondences [24,41]. For videos, DiffTrack further reveals that query-key similarities in selected layers of video diffusion transformers encode temporal correspondences across frames [23].

Restoration diffusion model. Diffusion models have emerged as a powerful paradigm for image restoration, owing to their rich generative priors, stable optimization, and strong generalization through iterative denoising [1,8,17,27,38]. By conditioning on degraded observations, these models recover perceptually sharp and realistic details that GAN-based methods [16] often fail to capture. However, naively extending image restoration diffusion models to video by processing frames independently leads to temporal flickering and visual inconsistency, as they lack a sufficient cross-frame modeling mechanism [40,42]. More recently, video diffusion methods [6,39,43] have been introduced to leverage strong generative priors for temporal modeling. While effective in restoring spatio-temporal content, these approaches incur substantial computational overhead and expose a trade-off between spatial fidelity and temporal coherence [11].

3 Preliminaries

3.1 Optical Flow Estimation

Modern optical flow methods generally follow a three-stage pipeline. Given two consecutive frames, a **feature encoder** $\mathcal{E}$ first encodes each frame into a dense feature representation. A **correlation operator** $\mathcal{C}$ then constructs a cost volume from pairwise similarities between the two feature maps. Finally, an **iterative update operator** $\mathcal{U}$ refines an initial flow estimate by repeatedly querying the cost volume through a recurrent unit, conditioned on context features that provide per-pixel information about the reference frame. Denoting the overall model as $\mathcal{M}$, this pipeline can be written compactly as:

$$\mathcal{M} = \mathcal{U} \circ \mathcal{C} \circ \mathcal{E}, \tag{1}$$

where $\circ$ denotes function composition. While this pipeline achieves strong accuracy on clean inputs, its performance degrades substantially on low-quality (LQ) videos, where noise, compression, and blur corrupt the extracted features and distort the resulting correlation signal.

3.2 DiT-based Image Restoration

Given a paired low-quality and high-quality frame $(\mathbf{I}_{\mathrm{LQ}}^k, \mathbf{I}_{\mathrm{HQ}}^k)$, both are first encoded into the latent space via a pretrained variational autoencoder (VAE) [15]:

$$\mathbf{z}_{\mathrm{LQ}}^k = \mathrm{Enc}(\mathbf{I}_{\mathrm{LQ}}^k), \qquad \mathbf{z}_{\mathrm{HQ}}^k = \mathrm{Enc}(\mathbf{I}_{\mathrm{HQ}}^k). \tag{2}$$

The diffusion process operates exclusively on the clean latent $\mathbf{z}_{\mathrm{HQ}}^k$, while the degraded latent $\mathbf{z}_{\mathrm{LQ}}^k$ serves solely as a conditioning signal. Models such as DiT4SR [8] also accept a text prompt as an additional condition; we omit it from our notation for brevity. During training, a noisy latent $\mathbf{z}_t^k$ is constructed by linearly interpolating between Gaussian noise and the clean target according to a continuous noise level $t \in [0, 1]$:

$$\mathbf{z}_t^k = (1 - t)\,\boldsymbol{\epsilon} + t\,\mathbf{z}_{\mathrm{HQ}}^k, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{3}$$

The DiT-based denoising network $\mathcal{D}$ is then trained to predict the velocity field along this interpolation path, conditioned on the degraded latent:

$$\mathbf{v}_t^k = \mathcal{D}(\mathbf{z}_t^k, t \mid \mathbf{z}_{\mathrm{LQ}}^k). \tag{4}$$

Under the rectified flow formulation [18], the ground-truth velocity is obtained by differentiating $\mathbf{z}_t^k$ with respect to t:

$$\frac{d\mathbf{z}_t^k}{dt} = \frac{d}{dt}\big((1 - t)\,\boldsymbol{\epsilon} + t\,\mathbf{z}_{\mathrm{HQ}}^k\big) = \mathbf{z}_{\mathrm{HQ}}^k - \boldsymbol{\epsilon}. \tag{5}$$

The model is thus trained by minimizing the flow-matching objective:

$$\mathcal{L}_{\mathrm{diff}} = \mathbb{E}_{k,\,t,\,\boldsymbol{\epsilon}}\Big[\big\|\mathbf{v}_t^k - (\mathbf{z}_{\mathrm{HQ}}^k - \boldsymbol{\epsilon})\big\|_2^2\Big]. \tag{6}$$
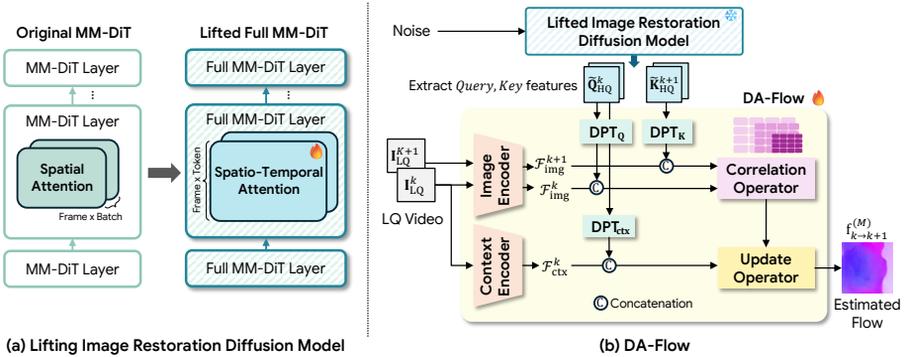
Fig. 2: Overall architecture of DA-Flow.

At inference, the model starts from pure noise and iteratively denoises the latent using the learned velocity field; the final restored image is then obtained by decoding the result with the VAE decoder.

4 Method

In this section, we present our approach to degradation-aware optical flow estimation. We begin by describing how a pretrained DiT-based image restoration model is lifted to the video domain through full spatio-temporal attention in Sec. 4.2. We then analyze the geometric correspondence encoded in the diffusion features across different layers, identifying which layers yield the most correspondence-ready representations in Sec. 4.3. Based on these findings, we introduce **DA-Flow**, our degradation-aware optical flow model that leverages the selected diffusion features to estimate reliable motion from corrupted inputs in Sec. 4.4.

4.1 Problem Formulation

We introduce a new task of **Degradation-Aware Optical Flow**, which aims to estimate accurate flow from corrupted videos. Let $\mathbf{V}_{\mathrm{LQ}}$ and $\mathbf{V}_{\mathrm{HQ}}$ denote a low-quality (LQ) video and its corresponding high-quality (HQ) video, respectively, each represented as a sequence of N RGB frames $\{\mathbf{I}^i\}_{i=1}^N$ with $\mathbf{I}^i \in \mathbb{R}^{3 \times H \times W}$. Our goal is to learn a degradation-aware optical flow model $\mathcal{M}$ that reliably estimates motion from LQ inputs. For a pair of consecutive frames indexed by k and $k+1$, where $k \in \{1, \ldots, N-1\}$, the model estimates:

$$\widehat{\mathbf{f}}_{k \to k+1} = \mathcal{M}\big(\mathbf{I}_{\mathrm{LQ}}^k, \ \mathbf{I}_{\mathrm{LQ}}^{k+1}\big) \approx \mathbf{f}_{k \to k+1}^*, \tag{7}$$

where $\mathbf{f}_{k \to k+1}^*$ denotes the ground-truth flow between frames k and $k+1$. Among the three stages in Eq. 1, the feature encoder $\mathcal{E}$ is most directly affected by input degradation, as corrupted pixels lead to unreliable features that propagate

errors into all downstream stages. We therefore focus on building a degradation-aware feature encoder that produces robust, correspondence-ready representations from LQ inputs, while keeping $\mathcal{C}$ and $\mathcal{U}$ unchanged from existing architectures.

4.2   Lifting Image Restoration Model

The DiT-based image restoration model operates independently on each frame, providing strong per-frame restoration capability but lacking any mechanism for temporal modeling. To preserve this strong generative prior while enabling temporal reasoning, we extend the model with full spatio-temporal attention over tokens across multiple frames.

Multi-modal attention in MM-DiT. Our backbone is based on the Multi-Modal Diffusion Transformer (MM-DiT) [9]. A straightforward way to apply this image-level model to video is to fold the temporal dimension into the batch axis. Each of the F frames in a batch of B video clips is then processed independently, yielding BF separate sequences of T patchified tokens with channel dimension C. Under this scheme, MM-DiT processes three modality-specific token sequences per frame through a Multi-Modal Attention (MM-Attention) mechanism: (i) $\mathbf{F}_{\text{HQ}} \in \mathbb{R}^{(BF) \times T \times C}$, latent tokens representing the current denoising state; (ii) $\mathbf{F}_{\text{LQ}} \in \mathbb{R}^{(BF) \times T \times C}$, conditioning tokens from the degraded input; and (iii) $\mathbf{F}_{\text{Text}} \in \mathbb{R}^{(BF) \times T \times C}$, text tokens encoding semantic priors. Within each block, modality-specific projections produce queries, keys, and values:

$$(\mathbf{Q}_m, \mathbf{K}_m, \mathbf{V}_m) = (\mathbf{F}_m \mathbf{W}_Q^m, \ \mathbf{F}_m \mathbf{W}_K^m, \ \mathbf{F}_m \mathbf{W}_V^m), \quad m \in \{\text{HQ}, \text{LQ}, \text{Text}\}, \quad (8)$$

which are concatenated along the token dimension to form $\mathbf{Q}, \mathbf{K}, \mathbf{V} \in \mathbb{R}^{(BF) \times 3T \times C}$ for joint attention. However, since the temporal dimension remains folded into the batch axis, MM-Attention is applied independently per frame, preventing the model from capturing inter-frame correspondences.

Full spatio-temporal attention. To enable inter-frame reasoning for our task, we reshape each modality stream from $\mathbf{F}_m \in \mathbb{R}^{(BF) \times T \times C}$ to $\tilde{\mathbf{F}}_m \in \mathbb{R}^{B \times (FT) \times C}$, concatenating all spatial tokens across frames into a single sequence per video. Modality-specific projections and concatenation then yield spatio-temporal queries, keys, and values

$$\tilde{\mathbf{Q}}, \tilde{\mathbf{K}}, \tilde{\mathbf{V}} \in \mathbb{R}^{B \times (3FT) \times C}, \quad (9)$$

and full spatio-temporal MM-Attention is computed as

$$\text{MM-Attn} = \text{softmax}\left(\frac{\tilde{\mathbf{Q}}\tilde{\mathbf{K}}^{\top}}{\sqrt{C}}\right) \tilde{\mathbf{V}}, \quad (10)$$

where each token now attends to all spatial tokens across all frames and modalities, enabling explicit inter-frame reasoning. With this full spatio-temporal MM-Attention applied to all layers, we finetune the lifted diffusion model $\mathcal{D}_\phi$ on the

(a) Top-10 Layer-wise Average EPE
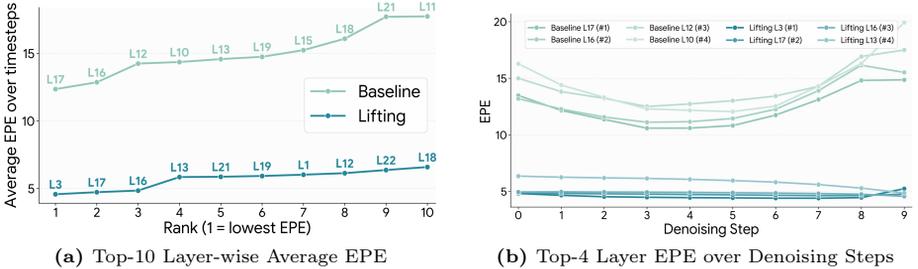
(b) Top-4 Layer EPE over Denoising Steps

Fig. 3: Comparison of zero-shot geometric correspondence between Baseline and Lifting features. (a) Top-10 layers ranked by timestep-averaged EPE (lower is better). Lifting consistently achieves lower EPE across all ranks. (b) EPE over denoising steps for the top-4 layers of each method. Baseline features show high sensitivity to the denoising step, while Lifting features remain stable across the denoising steps.

YouHQ training dataset [42]. After training, we use this lifted model as the feature encoder $\mathcal{E}$ in DA-Flow, leveraging its temporally-aware diffusion features for flow estimation. Further details are provided in Sec. 5.1 and Appendix A.

4.3 Diffusion Feature Analysis

A remaining question is which intermediate representations to extract from the lifted model for flow estimation. Recent work such as DiffTrack [23] shows that query and key features from full spatio-temporal attention layers in video diffusion models exhibit strong geometric correspondence. Motivated by this finding, we extract attention features from the full spatio-temporal MM-Attention layers introduced during lifting. Specifically, given a consecutive frame pair $(k, k+1)$, we take the query feature from frame k and the key feature from frame $k+1$ in the HQ diffusion branch:

$$\tilde{\mathbf{Q}}_{\text{HQ}}^{k}, \ \tilde{\mathbf{K}}_{\text{HQ}}^{k+1} \in \mathbb{R}^{B \times T \times C}. \tag{11}$$

Note that, unlike prior works [23,30] that inject input images into the generation branch at a specific noise level t, our features are extracted during the iterative denoising process, and we accordingly analyze them across denoising timesteps rather than at a single predetermined noise level. A further comparison with an alternative feature type is provided in Appendix B.

Evaluation protocol. To assess the zero-shot geometric correspondence of these diffusion features, we evaluate them through direct flow estimation without any task-specific training. Each feature has $T = h \times w$ tokens corresponding to the spatial dimensions of the latent space. For a single frame pair $(k, k+1)$, the extracted features $\tilde{\mathbf{Q}}_{\text{HQ}}^{k}, \tilde{\mathbf{K}}_{\text{HQ}}^{k+1} \in \mathbb{R}^{T \times C}$ are reshaped to $\mathbb{R}^{h \times w \times C}$, from which we construct a cost volume $\mathbf{C} \in \mathbb{R}^{h \times w \times h \times w}$ by computing pairwise dot-product similarity:

$$\mathbf{C}(i, j) = \tilde{\mathbf{Q}}_{\text{HQ}}^{k}(i) \cdot \tilde{\mathbf{K}}_{\text{HQ}}^{k+1}(j). \tag{12}$$

A flow field $\widehat{\mathbf{f}}_{k \to k+1} \in \mathbb{R}^{h \times w \times 2}$ is then obtained via $\widehat{\mathbf{f}}_{k \to k+1} = \text{softargmax}(\mathbf{C})$ and upsampled to the original image resolution $H \times W$. To evaluate this zero-shot prediction, we obtain a pseudo ground-truth flow $\mathbf{f}^*_{k \to k+1}$ by applying a pretrained optical flow model to the corresponding HQ frame pair $(\mathbf{I}^k_{\text{HQ}}, \mathbf{I}^{k+1}_{\text{HQ}})$, which serves as the reference for measuring correspondence accuracy. We report End-Point Error (EPE) on LQ–HQ video pairs from the YouHQ40 [42] validation set.

Results. We compare two configurations: the *Baseline*, which applies full spatio-temporal attention but is not finetuned, and *Lifting*, which is finetuned on YouHQ as described in Sec. 4.2. As shown in Fig. 3a, the lifted model achieves consistently lower EPE than the baseline across all layer ranks, confirming that finetuning with full spatio-temporal attention enables the model to learn inter-frame correspondences absent in the untrained baseline. The lifted features also remain stable across the entire denoising trajectory in Fig. 3b, in contrast to the baseline which exhibits high sensitivity to the extraction timestep. These results demonstrate that the lifted features possess superior geometric correspondence quality, and provide the basis for selecting which layers to extract features for DA-Flow, as detailed in Sec. 4.4. More detailed analyses are provided in Appendix B.

4.4   DA-Flow

Building upon the lifting architecture in Sec. 4.2 and the empirical analysis in Sec. 4.3, we introduce DA-Flow, a degradation-aware optical flow model built on top of RAFT [31]. As illustrated in Fig. 2, DA-Flow retains the original correlation operator $\mathcal{C}$ and iterative update operator $\mathcal{U}$, while incorporating the lifted diffusion model $\mathcal{D}_\phi$ alongside a conventional feature encoder $\mathcal{E}$. The overall pipeline can be written as:

$$\mathcal{M}_\theta = \mathcal{U} \circ \mathcal{C} \circ (\texttt{Up}(\mathcal{D}_\phi),\ \mathcal{E}), \tag{13}$$

where $\texttt{Up}$ denotes a learnable upsampling stage that maps the coarse diffusion features to a resolution compatible with $\mathcal{E}$. In the following, we describe each component in detail.

Feature upsampling. The diffusion features produced by $\mathcal{D}_\phi$ lie on a coarse spatial grid at $1/16$ of the input resolution. Directly passing them to the correlation operator $\mathcal{C}$ limits the quality of the resulting 4D cost volume, since accurate flow estimation requires fine-grained spatial details for precise boundary localization. We now describe the upsampling stage $\texttt{Up}$ in Eq. 13 that addresses this resolution gap.

Specifically, we aggregate diffusion features from the top-L layers that exhibit the strongest geometric correspondence quality, as identified in Sec. 4.3. The aggregated features are then passed through DPT-based upsampling heads [26] to

Table 1: Quantitative comparison on Sintel, Spring, and TartanAir. All methods are evaluated using End-Point Error (EPE) and outlier rates at 1px, 3px, and 5px thresholds. Best and second best results are highlighted.

Model	Sintel				Spring				TartanAir			
	EPE↓	1px↓	3px↓	5px↓	EPE↓	1px↓	3px↓	5px↓	EPE↓	1px↓	3px↓	5px↓
RAFT [31]	10.693	62.91	37.24	28.63	3.944	39.82	18.65	11.98	9.487	75.17	42.96	30.04
SEA-RAFT [37]	10.185	59.56	34.46	26.15	2.703	41.51	19.31	12.11	8.316	77.85	45.76	32.15
FlowSeek [25]	10.241	64.08	40.71	31.83	2.861	41.53	19.16	12.18	7.694	76.96	45.20	32.00
DA-Flow	6.912	55.80	28.10	20.91	2.207	30.95	13.87	8.91	8.866	72.35	37.61	25.40

recover higher-resolution feature maps. Since the query and key features from the diffusion attention already encode distinct representations, we employ separate DPT heads to preserve this distinction: a query head and a key head produce correspondence features for cost volume construction, while a context head generates spatial conditioning features for the iterative update operator $\mathcal{U}$. Formally, given frame index k, the upsampled features are obtained as:

$$\mathcal{F}_Q^{k,\uparrow} = \mathrm{DPT}_Q\Big(\{\tilde{\mathbf{Q}}_{\mathrm{HQ}}^{k,l}\}_{l=1}^L\Big),$$
$$\mathcal{F}_K^{k+1,\uparrow} = \mathrm{DPT}_K\Big(\{\tilde{\mathbf{K}}_{\mathrm{HQ}}^{k+1,l}\}_{l=1}^L\Big), \qquad (14)$$
$$\mathcal{F}_{\mathrm{ctx}}^{k,\uparrow} = \mathrm{DPT}_{\mathrm{ctx}}\Big(\{\tilde{\mathbf{Q}}_{\mathrm{HQ}}^{k,l}\}_{l=1}^L\Big),$$

where l indexes the selected diffusion layers. All upsampled features share a common spatial resolution of $H/8 \times W/8$, with query and key features having channel dimension c and context features having channel dimension c'.

Hybrid feature encoding. While the upsampled diffusion features $\mathcal{F}_Q^{k,\uparrow}$, $\mathcal{F}_K^{k+1,\uparrow}$, and $\mathcal{F}_{\mathrm{ctx}}^{k,\uparrow}$ provide strong degradation-aware representations, they lack fine-grained spatial localization due to their globally aggregated nature. To compensate, we incorporate the conventional feature encoder $\mathcal{E}$ from Eq. 13, which preserves local spatial details through its convolutional architecture. Following RAFT, $\mathcal{E}$ consists of an image encoder $\mathcal{E}_{\mathrm{img}}$ applied to all input frames and a context encoder $\mathcal{E}_{\mathrm{ctx}}$ applied only to the reference frame:

$$\mathcal{F}_{\mathrm{img}}^k,\ \mathcal{F}_{\mathrm{img}}^{k+1} = \mathcal{E}_{\mathrm{img}}(\mathbf{I}_{\mathrm{LQ}}^k),\ \mathcal{E}_{\mathrm{img}}(\mathbf{I}_{\mathrm{LQ}}^{k+1}),$$
$$\mathcal{F}_{\mathrm{ctx}}^k = \mathcal{E}_{\mathrm{ctx}}(\mathbf{I}_{\mathrm{LQ}}^k), \qquad (15)$$

where all encoder features have spatial resolution $H/8 \times W/8$. The diffusion and CNN features are then concatenated along the channel dimension to form the hybrid representations used for cost volume construction and iterative updates. The hybrid feature maps are then formed by concatenating the diffusion and

Fig. 4: Qualitative results on Sintel [3].

CNN features along the channel dimension:

$$\mathcal{F}^k = \text{Concat}(\mathcal{F}^k_{\text{img}}, \ \mathcal{F}^{k,\uparrow}_{\text{Q}}),$$
$$\mathcal{F}^{k+1} = \text{Concat}(\mathcal{F}^{k+1}_{\text{img}}, \ \mathcal{F}^{k+1,\uparrow}_{\text{K}}), \quad (16)$$
$$\mathcal{F}^k_{\text{h-ctx}} = \text{Concat}(\mathcal{F}^k_{\text{ctx}}, \ \mathcal{F}^{k,\uparrow}_{\text{ctx}}),$$

where $\mathcal{F}^k$ and $\mathcal{F}^{k+1}$ are used to construct the correlation volume via $\mathcal{C}$, and $\mathcal{F}^k_{\text{h-ctx}}$ provides spatial conditioning for the iterative update operator $\mathcal{U}$. The correlation volume and context features are then processed through $\mathcal{U}$, which produces a sequence of refined flow estimates:

$$\{\mathbf{f}^{(i)}_{k\to k+1}\}^M_{i=0}, \quad (17)$$

where $\mathbf{f}^{(i)} \in \mathbb{R}^{H\times W\times 2}$ and M denotes the number of recurrent refinement steps. The final flow field $\mathbf{f}^{(M)}_{k\to k+1}$ is taken from the last iteration.

Loss function. Since obtaining ground-truth optical flow for real-world degraded videos is infeasible, we generate pseudo ground-truth labels from the same YouHQ dataset [42] used to train the lifted diffusion model in Sec. 4.2. Following the protocol in Sec. 4.3, we apply a pretrained optical flow model to the HQ frame pairs to obtain pseudo ground-truth flow $\mathbf{f}^*_{k\to k+1}$, while the corresponding LQ frames serve as input to DA-Flow. We optimize $\mathcal{M}_\theta$ using the standard multi-scale flow loss:

$$\mathcal{L}_{\text{flow}} = \sum_{i=1}^{M} \gamma^{M-i} \left\| \mathbf{f}^{(i)}_{k\to k+1} - \mathbf{f}^*_{k\to k+1} \right\|_1, \quad (18)$$

where γ is a weight decay factor and M denotes the number of refinement iterations.

5 Experiment

5.1 Experimental Setup

Implementation details. We train our model in two stages. In the first stage, the lifted diffusion model $\mathcal{D}_\phi$ is trained with the diffusion loss $\mathcal{L}_{\text{diff}}$ on clips of
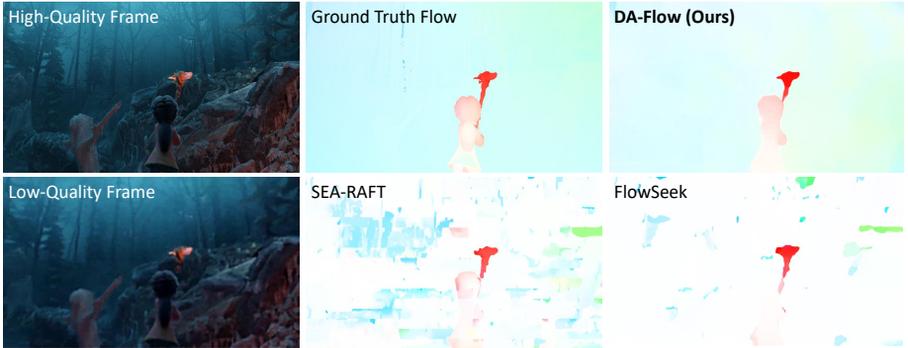
Fig. 5: Qualitative results on Spring [20].

$F = 3$ consecutive frames. In the second stage, $\mathcal{D}_\phi$ is frozen, and the flow estimation pipeline $\mathcal{M}_\theta$ is trained with the flow loss $\mathcal{L}_{\text{flow}}$ using $M = 12$ refinement iterations. For correspondence estimation, we extract diffusion features from four full spatio-temporal attention layers $\{3, 13, 16, 17\}$ ($L = 4$), selected based on the analysis in Sec. 4.3. Both stages are trained for 20K steps each with a batch size of 32 on 4 NVIDIA H100 GPUs and a learning rate of 5×10^{-5}.

Training dataset. We train DA-Flow on YouHQ [42], a high-resolution video dataset comprising 38,576 diverse videos sourced from YouTube. With an average resolution of 1080×1920 and 32 frames per clip, the dataset spans various scenarios including street views, human portraits, animals, and other categories. To generate corresponding LQ pairs, we follow the degradation pipeline of RealBasicVSR [4], which applies the Real-ESRGAN [36] degradation model at the frame level, followed by video compression over the entire sequence. Since ground-truth optical flow is not available, we generate pseudo ground-truth flow from the HQ frame pairs using SEA-RAFT [37]. The same degradation and pseudo ground-truth generation setup is applied to the YouHQ40 [42] validation split used for the feature analysis in Sec. 4.3. During training, input frames are randomly cropped to 512×512.

Evaluation datasets and metrics. We evaluate DA-Flow on three optical flow benchmarks. Spring [20] is a large-scale synthetic dataset providing dense ground-truth flow with highly detailed scenes and complex motion patterns. The Sintel [3] train set offers two rendering passes (clean and final) with varying levels of motion blur and atmospheric effects. Following FlowSeek [25], we additionally construct an evaluation set from the TartanAir [35] validation split using the same setup. For all benchmarks, LQ inputs are generated using the same degradation pipeline employed during training to ensure consistency. We report End-Point Error (EPE) and the percentage of outlier pixels with errors exceeding 1px, 3px, and 5px thresholds as evaluation metrics. The number of iterative
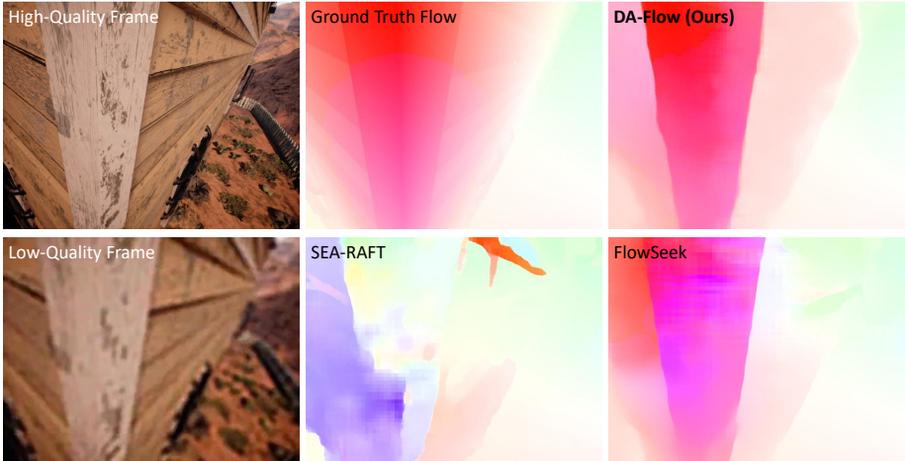
Fig. 6: Qualitative results on TartanAir [35].

updates for all baseline models is set to 12, ensuring a fair comparison. DA-Flow performs 10 denoising steps during inference.

5.2   Quantitative Results

Tab. 1 compares DA-Flow against existing optical flow methods on three benchmarks under synthetic degradations. On Sintel and Spring, DA-Flow achieves the best performance across all metrics, reducing EPE by a clear margin over the strongest baseline. This highlights the effectiveness of leveraging degradation-aware diffusion features for flow estimation under corrupted inputs. On TartanAir, DA-Flow achieves the best outlier rates at all thresholds (1px, 3px, 5px), while showing a higher EPE than FlowSeek. This discrepancy can be attributed to a small number of pixels with large displacement errors; these outlier pixels disproportionately inflate the average endpoint error. The consistently lower outlier rates suggest that DA-Flow produces more accurate estimates over the majority of pixels. Additional results can be found in Appendix C.

5.3   Qualitative Results

We present qualitative comparisons against SEA-RAFT and FlowSeek on Sintel, Spring, and TartanAir in Fig. 4, Fig. 5, and Fig. 6. Across all benchmarks, baseline methods produce noisy and inconsistent flow fields under degraded inputs, with artifacts concentrated around motion boundaries and fine-grained structures, which are an expected failure mode for methods designed for clean inputs. DA-Flow, by contrast, consistently recovers sharp, coherent flow fields that closely match the ground truth, successfully localizing motion boundaries on Sintel, maintaining structural coherence over complex scenes on Spring, and producing cleaner estimates under large displacements on TartanAir. These results

Table 2: Ablation on feature source across denoising steps. Baseline* uses the same DA-Flow architecture but extracts features from the full-attention model before finetuning. **Bold** indicates the better result at each step.

Dataset	Metric	Method	Step									
			0	1	2	3	4	5	6	7	8	9
Sintel [3]	EPE ↓	Baseline*	7.4145	7.4542	7.5076	7.4703	7.4124	7.4897	7.5243	7.5752	7.5709	7.6883
		DA-Flow	**7.0210**	**6.7809**	**6.7196**	**6.7160**	**6.7433**	**6.7605**	**6.8029**	**6.8736**	**7.0641**	**7.6397**
	1px ↓	Baseline*	59.47	59.26	59.50	59.26	59.60	60.16	59.85	60.80	61.02	61.79
		DA-Flow	**57.04**	**56.81**	**56.39**	**55.97**	**55.52**	**55.12**	**54.79**	**54.72**	**54.64**	**57.03**
Spring [20]	EPE ↓	Baseline*	2.3269	2.3457	2.2535	2.3073	2.2158	**2.2030**	2.2036	2.2148	2.2066	2.2343
		DA-Flow	**2.2902**	**2.2119**	**2.2069**	**2.2026**	**2.2011**	2.2043	**2.2008**	**2.1928**	**2.1917**	**2.1720**
	1px ↓	Baseline*	32.14	32.74	32.45	32.72	32.23	32.24	32.15	32.63	32.24	31.86
		DA-Flow	**30.49**	**31.01**	**31.06**	**30.99**	**31.05**	**31.23**	**31.23**	**31.06**	**30.88**	**30.54**

demonstrate that the degradation-aware priors encoded in the lifted diffusion features allow DA-Flow to reason about scene geometry even when the underlying visual evidence is severely corrupted. More qualitative results are provided in Appendix D.

5.4   Ablation Study

Comparison with Baseline. To isolate the contribution of the lifted diffusion features, we construct a baseline variant (Baseline*) that uses the same DA-Flow architecture but replaces the lifted features with those from the untrained full-attention model. Following the layer-wise analysis in Sec. 4.3, we select the top-4 layers for each model: {10, 12, 16, 17} for Baseline* and {3, 13, 16, 17} for DA-Flow. Tab. 2 reports the results across all denoising steps on Sintel and Spring. On Sintel, DA-Flow consistently outperforms Baseline* at every step in both EPE and 1px outlier rate. In Spring, the two methods perform comparably in EPE, while DA-Flow maintains a consistent advantage in 1px outlier rate across all steps. In fact, DA-Flow achieves lower EPE at nearly every step, except a single step where the two methods are virtually tied. The advantage is more pronounced in 1px outlier rate, where DA-Flow consistently outperforms Baseline* across all steps. Additional ablation studies are provided in Appendix C.

6   Conclusion

In this work, we propose DA-Flow for degradation-aware optical flow, which estimates dense correspondences directly from corrupted inputs. Our approach starts from a pretrained image restoration diffusion model and extends it to multi-frame processing via spatio-temporal attention. This enables DA-Flow to leverage intermediate representations that encode both degradation-aware priors and geometric correspondence cues, while preserving spatial structure crucial for dense matching. Extensive experiments on degraded optical flow benchmarks demonstrate that DA-Flow consistently improves flow estimation accuracy over existing methods.

References

1. Ai, Y., Zhou, X., Huang, H., Han, X., Chen, Z., You, Q., Yang, H.: Dreamclear: High-capacity real-world image restoration with privacy-safe dataset curation. Advances in Neural Information Processing Systems **37**, 55443–55469 (2024)

2. Bay, H., Tuytelaars, T., Van Gool, L.: Surf: Speeded up robust features. In: European conference on computer vision. pp. 404–417. Springer (2006)

3. Butler, D.J., Wulff, J., Stanley, G.B., Black, M.J.: A naturalistic open source movie for optical flow evaluation. In: European conference on computer vision. pp. 611–625. Springer (2012)

4. Chan, K.C., Zhou, S., Xu, X., Loy, C.C.: Investigating tradeoffs in real-world video super-resolution. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5962–5971 (2022)

5. Chen, H., Zhou, Y., Dong, H., Wang, X., Qiao, Y., Dong, C.: Stablesr: Exploiting diffusion priors for real-world image super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2023)

6. Chen, Z., Zou, Z., Zhang, K., Su, X., Yuan, X., Guo, Y., Zhang, Y.: Dove: Efficient one-step diffusion model for real-world video super-resolution (2025), `https://arxiv.org/abs/2505.16239`

7. DeTone, D., Malisiewicz, T., Rabinovich, A.: Superpoint: Self-supervised interest point detection and description. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 224–236 (2018)

8. Duan, Z.P., Zhang, J., Jin, X., Zhang, Z., Xiong, Z., Zou, D., Ren, J.S., Guo, C., Li, C.: Dit4sr: Taming diffusion transformer for real-world image super-resolution. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 18948–18958 (2025)

9. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., et al.: Scaling rectified flow transformers for high-resolution image synthesis. In: Forty-first international conference on machine learning (2024)

10. Gan, C., Tu, Y., Chen, X., Chen, T., Li, Y., Harandi, M., Lin, W.: Unleashing diffusion transformers for visual correspondence by modulating massive activations. arXiv preprint arXiv:2505.18584 (2025)

11. Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., Fleet, D.J.: Video diffusion models. Advances in neural information processing systems **35**, 8633–8646 (2022)

12. Ke, B., Obukhov, A., Huang, S., Metzger, N., Daudt, R.C., Schindler, K.: Repurposing diffusion-based image generators for monocular depth estimation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9492–9502 (2024)

13. Ke, B., Qu, K., Wang, T., Metzger, N., Huang, S., Li, B., Obukhov, A., Schindler, K.: Marigold: Affordable adaptation of diffusion-based image generators for image analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence (2025)

14. Kim, C., Shin, H., Hong, E., Yoon, H., Arnab, A., Seo, P.H., Hong, S., Kim, S.: Seg4diff: Unveiling open-vocabulary segmentation in text-to-image diffusion transformers. arXiv preprint arXiv:2509.18096 (2025)

15. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)

16. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4681–4690 (2017)

17. Lin, X., He, J., Chen, Z., Lyu, Z., Dai, B., Yu, F., Qiao, Y., Ouyang, W., Dong, C.: Diffbir: Toward blind image restoration with generative diffusion prior. In: European conference on computer vision. pp. 430–448. Springer (2024)

18. Liu, X., Gong, C., Liu, Q.: Flow straight and fast: Learning to generate and transfer data with rectified flow. arXiv preprint arXiv:2209.03003 (2022)

19. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International journal of computer vision $60(2)$, 91–110 (2004)

20. Mehl, L., Schmalfuss, J., Jahedi, A., Nalivayko, Y., Bruhn, A.: Spring: A high-resolution high-detail dataset and benchmark for scene flow, optical flow and stereo. In: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2023)

21. Mishchuk, A., Mishkin, D., Radenovic, F., Matas, J.: Working hard to know your neighbor's margins: Local descriptor learning loss. In: Advances in Neural Information Processing Systems (NeurIPS) (2017)

22. Nam, J., Lee, G., Kim, S., Kim, H., Cho, H., Kim, S., Kim, S.: Diffusion model for dense matching. arXiv preprint arXiv:2305.19094 (2023)

23. Nam, J., Son, S., Chung, D., Kim, J., Jin, S., Hur, J., Kim, S.: Emergent temporal correspondences from video diffusion transformers. arXiv preprint arXiv:2506.17220 (2025)

24. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 (2023)

25. Poggi, M., Tosi, F.: Flowseek: optical flow made easier with depth foundation models and motion bases. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5667–5679 (2025)

26. Ranftl, R., Bochkovskiy, A., Koltun, V.: Vision transformers for dense prediction. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 12179–12188 (2021)

27. Saharia, C., Ho, J., Chan, W., Salimans, T., Fleet, D.J., Norouzi, M.: Image super-resolution via iterative refinement. IEEE transactions on pattern analysis and machine intelligence $45(4)$, 4713–4726 (2022)

28. Schmalfuss, J., Oei, V., Mehl, L., Bartsch, M., Agnihotri, S., Keuper, M., Bruhn, A.: Robustspring: Benchmarking robustness to image corruptions for optical flow, scene flow and stereo. arXiv preprint arXiv:2505.09368 (2025)

29. Sun, J., Shen, Z., Wang, Y., Bao, H., Zhou, X.: Loftr: Detector-free local feature matching with transformers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8922–8931 (2021)

30. Tang, L., Jia, M., Wang, Q., Phoo, C.P., Hariharan, B.: Emergent correspondence from image diffusion. Advances in Neural Information Processing Systems 36, 1363–1389 (2023)

31. Teed, Z., Deng, J.: Raft: Recurrent all-pairs field transforms for optical flow. In: European conference on computer vision. pp. 402–419. Springer (2020)

32. Tian, J., Aggarwal, L., Colaco, A., Kira, Z., Gonzalez-Franco, M.: Diffuse attend and segment: Unsupervised zero-shot segmentation using stable diffusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3554–3563 (2024)

33. Tian, Y., Fan, B., Wu, F.: L2-net: Deep learning of discriminative patch descriptor in euclidean space. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 661–669 (2017)
34. Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D.: Vggt: Visual geometry grounded transformer. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 5294–5306 (2025)
35. Wang, W., Zhu, D., Wang, X., Hu, Y., Qiu, Y., Wang, C., Hu, Y., Kapoor, A., Scherer, S.: Tartanair: A dataset to push the limits of visual slam. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 4909–4916. IEEE (2020)
36. Wang, X., Xie, L., Dong, C., Shan, Y.: Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 1905–1914 (2021)
37. Wang, Y., Lipson, L., Deng, J.: Sea-raft: Simple, efficient, accurate raft for optical flow. In: European Conference on Computer Vision. pp. 36–54. Springer (2024)
38. Wu, R., Yang, T., Sun, L., Zhang, Z., Li, S., Zhang, L.: Seesr: Towards semantics-aware real-world image super-resolution. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 25456–25467 (2024)
39. Xie, R., Liu, Y., Zhou, P., Zhao, C., Zhou, J., Zhang, K., Zhang, Z., Yang, J., Yang, Z., Tai, Y.: Star: Spatial-temporal augmentation with text-to-video models for real-world video super-resolution. arXiv preprint arXiv:2501.02976 (2025)
40. Yang, X., He, C., Ma, J., Zhang, L.: Motion-guided latent diffusion for temporally consistent real-world video super-resolution. In: European conference on computer vision. pp. 224–242. Springer (2024)
41. Zhang, Y., Zhao, W., Zhong, Y., Wu, Z., Avrithis, Y., Vedaldi, A.: A tale of two features: Stable diffusion complements dino for zero-shot semantic correspondence. In: Advances in Neural Information Processing Systems (NeurIPS) (2023)
42. Zhou, S., Yang, P., Wang, J., Luo, Y., Loy, C.C.: Upscale-a-video: Temporal-consistent diffusion model for real-world video super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2535–2545 (2024)
43. Zhuang, J., Guo, S., Cai, X., Li, X., Liu, Y., Yuan, C., Xue, T.: Flashvsr: Towards real-time diffusion-based streaming video super-resolution. arXiv preprint arXiv:2510.12747 (2025)

Appendix

In this appendix, we provide additional details and experiments that supplement the main paper. Sec. A describes the implementation details of DA-Flow. Sec. B extends the feature analysis presented in Sec. 4.3 with further analysis and experiments. Sec. C presents additional experiments and ablation studies. Sec. D provides more qualitative results. Finally, Sec. E discusses limitations and future work.

A   Implementation Details

This section describes the implementation details of our experiments. Unless specified, all settings follow the default configurations.

Lifting image restoration diffusion model. We adopt DiT4SR [8] as our image restoration diffusion model, using the official code with the publicly available `dit4sr_q` weights, as in the original paper. To lift the model, we modify the attention operation at each layer by reshaping the frames, originally processed as a batch, into the spatial token dimension, enabling full attention to be computed across all frames within a single forward pass. We use the YouHQ [42] dataset for training, and adopt the degradation pipeline from the STAR [39] codebase to synthesize low-quality frames as model inputs.

Optical flow network. Our optical flow estimation network $\mathcal{M}_\phi$ is based on RAFT [31], initialized from the `raft-things.pth` weights pretrained on FlyingThings3D. For the feature upsampler, we adopt the DPT architecture following the VGGT [34] codebase. When fusing multi-scale features in DPT, we construct a feature pyramid with resolution scales of $(1, 1, 2, 2)$, where each value indicates the scaling factor relative to the input resolution for the corresponding pyramid level.

Pseudo ground-truth for optical flow. To generate pseudo ground-truth optical flow for feature analysis in Sec. 4.3, we apply SEA-RAFT [37] on the high-quality videos using the official implementation with the `spring-M` configuration (4 recurrent iterations) and the `Tartan-C-T-TSKH-spring540x960-M.pth` weights. For training DA-Flow, we use the same configuration with 20 recurrent iterations to obtain higher-quality pseudo ground-truth flow.

Training and inference details. Unless specified in Sec. 5.1, we follow the training settings of DiT4SR [8]. When training the lifting model, we generate text prompts using the captioner provided by the original model and use them as conditions, enabling the model to effectively learn cross-frame correspondences. For diffusion feature analysis, as well as training and inference of DA-Flow, we

use null prompts for all inputs, as the low-quality input frames make it difficult to generate reliable captions. During inference, we use 10 denoising steps and follow the same procedure as training: the input video is processed in chunks of 3 frames through the model, and the resulting features are used to estimate optical flow with 20 refinement iterations.

B   Additional Feature Analysis

B.1   Full Feature Analysis Results

Fig. 3a presents the results for the top-10 features of both the baseline and lifting models. Extending this analysis, Fig. 7 shows the timestep-averaged EPE across all layers for the baseline and lifting models. Across nearly all feature indices, the lifted features exhibit substantially lower EPE. This demonstrates that lifting enhances correspondence capability by enabling the model to learn cross-frame information.
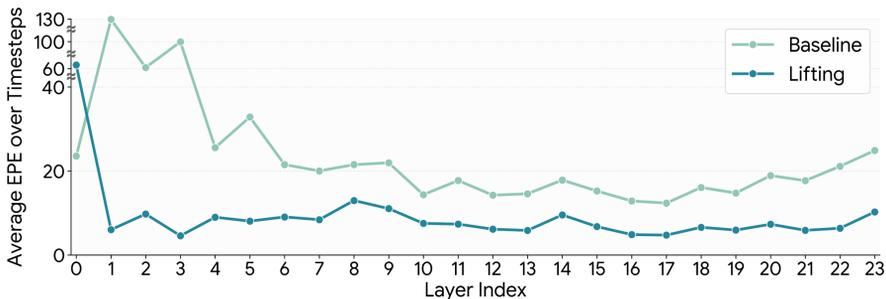

Fig. 7: Comparison of layer-wise average EPE over timesteps.

B.2   Comparison of Alternative Feature Type

DA-Flow utilizes query and key features from the full attention layers of the diffusion model. However, features can also be extracted from other locations within each layer. Among the various candidates, we select post-AdaNorm features for comparison, as DITF [10] has already demonstrated their effectiveness for semantic correspondence, making them a reasonable alternative to examine.

We extract post-AdaNorm features and conduct the same feature analysis described in Sec. 4.3. Fig. 8a ranks the top-10 layers by timestep-averaged EPE for both feature types. At the baseline level, query and key features consistently achieve lower EPE than post-AdaNorm features. After lifting, the gap narrows and the two become comparable except for the top-3 layers, where query and key features retain a clear advantage. To further investigate, we plot the EPE over denoising steps for the top-4 layers of each method in Fig. 8b. Query and key
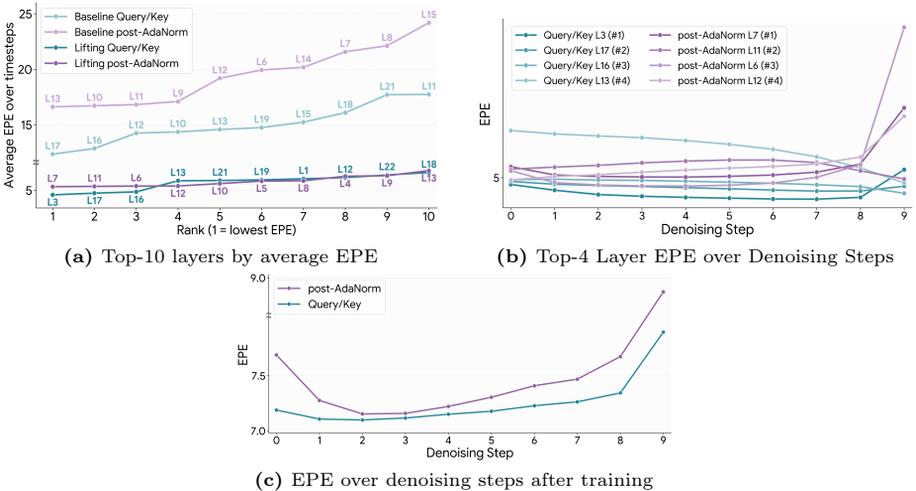
(a) Top-10 layers by average EPE

(b) Top-4 Layer EPE over Denoising Steps

(c) EPE over denoising steps after training

Fig. 8: Comparison of zero-shot geometric correspondence between post-AdaNorm and Query/Key features. (a) Top-10 layers ranked by timestep-averaged EPE (lower is better). (b) EPE over denoising steps for the top-4 layers of each method. (c) EPE on Sintel [3] over denoising steps after training the flow network. DA-Flow uses Query/Key features.

features tend to show lower EPE overall, while post-AdaNorm features exhibit a noticeable spike at the final denoising step. Since the two feature types yield relatively similar trends, we further verify their difference by training the flow network using post-AdaNorm features in place of query and key features, keeping all other settings identical to DA-Flow. Fig. 8c reports the resulting EPE on Sintel [3] over denoising steps, confirming that query and key features lead to better flow estimation after training as well. We conjecture that this advantage stems from the attention mechanism, which inherently encodes pairwise spatial relationships in the query and key projections, making them better suited for geometric correspondence.

B.3 Video Restoration Diffusion Model

Video restoration diffusion models typically compress multiple frames into a single latent, making them inherently unsuitable for extracting frame-by-frame features. Nevertheless, we can still extract query and key features from such models and, analogous to how their VAE decodes a single latent into multiple frames, interpolate a single feature map to obtain per-frame features for the same analysis described in Sec. 4.3. Fig. 9 presents the results on FlashVSR [43]. Due to the interpolation of compressed features, geometric correspondence is significantly inferior to both the baseline and lifting, indicating that features from image diffusion models are more suitable for degradation-aware optical flow than those from video diffusion models.
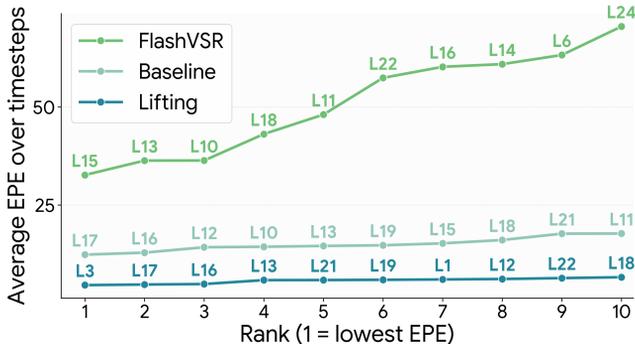
Fig. 9: Zero-shot geometric correspondence of Query/Key features from FlashVSR [43].

C Additional Experiments

C.1 Metrics over Denoising Steps

Tab. 1 reports the quantitative results of DA-Flow averaged over all denoising timesteps. The full per-timestep results are presented in Tab. 3. On Sintel and Spring, DA-Flow consistently outperforms existing methods across all timesteps. On TartanAir, while the timestep-averaged EPE is slightly higher than those of prior methods, DA-Flow achieves superior performance at both the initial (step 0) and final (step 9) denoising steps.

Table 3: Timestep-wise quantitative results of DA-Flow on Sintel [3], Spring [20], and TartanAir [35]. Best and second best results are highlighted.

Dataset	Metric	0	1	2	3	4	5	6	7	8	9	Average	RAFT [31]	SEA-RAFT [37]	FlowSeek [25]
						Step-wise Results of DA-Flow								Method	
Sintel [3]	EPE↓	7.021	6.781	6.720	6.716	6.743	6.761	6.803	6.874	7.064	7.640	6.912	10.69	10.18	10.24
	1px↓	57.04	56.81	56.39	55.97	55.52	55.12	54.79	54.72	54.64	57.03	55.80	62.91	59.56	64.08
	3px↓	28.58	28.06	27.67	27.52	27.56	27.71	27.88	28.14	28.40	29.52	28.10	37.24	34.46	40.71
	5px↓	21.33	20.80	20.49	20.39	20.43	20.57	20.80	21.01	21.29	21.98	20.91	28.63	26.15	31.83
Spring [20]	EPE↓	2.290	2.212	2.207	2.203	2.201	2.204	2.201	2.193	2.192	2.172	2.207	3.944	2.703	2.861
	1px↓	30.49	31.01	31.06	30.99	31.05	31.23	31.23	31.06	30.88	30.54	30.95	39.82	41.51	41.53
	3px↓	14.16	13.95	13.94	13.94	13.95	13.96	13.91	13.82	13.75	13.33	13.87	18.65	19.31	19.16
	5px↓	9.230	9.000	8.990	8.97	8.96	8.940	8.910	8.940	8.940	8.940	8.910	11.98	12.11	12.18
TartanAir [35]	EPE↓	6.674	8.675	8.926	8.955	9.076	9.384	9.696	10.06	9.951	7.257	8.866	9.487	8.316	7.694
	1px↓	72.96	73.07	72.66	72.29	72.01	71.82	71.69	71.69	71.65	73.63	72.35	75.17	77.85	76.96
	3px↓	37.34	38.00	37.71	37.46	37.38	37.42	37.47	37.53	37.47	38.28	37.61	42.96	45.76	45.20
	5px↓	24.57	25.53	25.34	25.19	25.23	25.42	25.62	25.79	25.71	25.59	25.40	30.04	32.15	32.00

C.2 Finetuning RAFT

A straightforward baseline for our degradation-aware optical flow task is to finetune an existing flow network on the same training data. Specifically, we finetune RAFT [31] using the identical setup as DA-Flow: low-quality and high-quality frame pairs generated by applying degradation kernels to YouHQ [42],

Table 4: Average comparison on Sintel between RAFT* and DA-Flow. Lower is better for all metrics.

	Metric			
Method	EPE ↓	1px ↓	3px ↓	5px ↓
RAFT*	7.033	56.99	28.39	**20.70**
DA-Flow	**6.912**	**55.80**	**28.10**	20.91

with pseudo ground-truth flow obtained from SEA-RAFT [37]. Tab. 4 compares the finetuned RAFT [31] (denoted RAFT*) with DA-Flow on Sintel. DA-Flow outperforms RAFT* in EPE, 1px, and 3px metrics, demonstrating that our approach offers clear advantages over simply finetuning a conventional flow network on the same data.

C.3   Additional Ablation Studies

Table 5: Ablation study on the architectural components of DA-Flow. Row (d) corresponds to DA-Flow. **Bold** indicates the best result. Lower is better for all metrics.

	Configuration		Metric			
	Upsample	CNN Encoder	EPE ↓	1px ↓	3px ↓	5px ↓
(a)	Bilinear	✗	7.2236	59.49	29.55	21.59
(b)	DPT	✗	8.0745	64.67	30.07	21.95
(c)	Bilinear	✓	7.1230	57.84	28.96	21.14
(d)	DPT	✓	**6.9122**	**55.80**	**28.10**	**20.91**

In Sec. 4, we adopt DPT-based feature aggregation for upsampling and incorporate the CNN encoder from RAFT [31] to better capture fine-grained local details beyond what diffusion features alone provide. To validate these design choices, we conduct an ablation study on the architectural components of DA-Flow, varying the upsampling strategy and the use of the CNN encoder. Tab. 5 reports EPE and pixel-threshold error rates (%) on Sintel, averaged over all denoising timesteps, where row (d) corresponds to DA-Flow.

CNN encoder. To assess the benefit of incorporating convolutional features from the RAFT [31] encoder, we compare configurations with and without it under the same upsampling strategy. Adding the CNN encoder consistently improves performance in both the bilinear setting ((a) vs. (c)) and the DPT setting ((b) vs. (d)), with the latter showing particularly large gains. This confirms that the RAFT encoder provides complementary fine-grained spatial information that

the diffusion features alone lack, and that the combination of DPT upsampling and CNN encoder features is essential for optimal flow estimation.

Feature upsampling. We examine the effect of the upsampling strategy by comparing configurations with and without DPT-based feature aggregation. Without the CNN encoder, replacing bilinear interpolation with DPT ((a) vs. (b)) does not lead to consistent improvements across all metrics. However, when the CNN encoder is incorporated ((c) vs. (d)), DPT upsampling yields clear gains across all metrics including EPE. This suggests that DPT-based feature aggregation becomes effective when operating on sufficiently detailed features, and that the fine-grained information provided by the CNN encoder enables the DPT to fully leverage multi-scale aggregation for improved flow estimation.

C.4   Application

Several video restoration methods [40,42] leverage off-the-shelf optical flow models such as RAFT [31] to align neighboring frames for improved temporal consistency. However, since these models directly take degraded frames as input, the estimated flow is often inaccurate, limiting the effectiveness of temporal alignment. Among them, MGLD [40] employs a guidance mechanism that warps the current restored frame toward the next frame using estimated optical flow and minimizes the L2 distance between them as a guidance loss, enforcing temporal consistency during the diffusion sampling process. We adopt this guidance technique and combine it with our lifted image restoration diffusion model and DA-Flow. Unlike MGLD [40], which operates in latent space, we perform guidance in image space by decoding the latent at each step. We evaluate on the YouHQ40 dataset used in Sec. 4.3, and additionally include frame-by-frame restoration results from our baseline DiT4SR [8] for comparison. As shown in Tab. 6, our approach achieves strong performance in both reference-based metrics (PSNR, SSIM, LPIPS) and the non-reference metric DOVER, while attaining the best warping error, validating the effectiveness of the accurate flow estimated by DA-Flow. We provide qualitative comparisons of temporal consistency in Fig. 10. Compared to other methods, DA-Flow improves temporal alignment, reducing flickering and maintaining structural stability across consecutive frames.

Table 6: Video restoration results on YouHQ40. Note that $E^*_{\text{warp}} = E_{\text{warp}} \times 10^{-3}$. **Bold** indicates the best result.

Method	PSNR↑	SSIM↑	LPIPS↓	DOVER↑	E^*_{warp} ↓
MGLD	22.50	0.626	0.2661	47.41	4.532
DiT4SR	19.55	0.511	0.335	**74.97**	29.50
DA-Flow + MGLD	**23.47**	**0.646**	**0.215**	57.96	**3.483**

D   Additional Qualitative Results

We provide additional qualitative results on all benchmark datasets in Fig. 11, Fig. 12, and Fig. 13. These examples further demonstrate the effectiveness of DA-Flow across diverse scenarios.
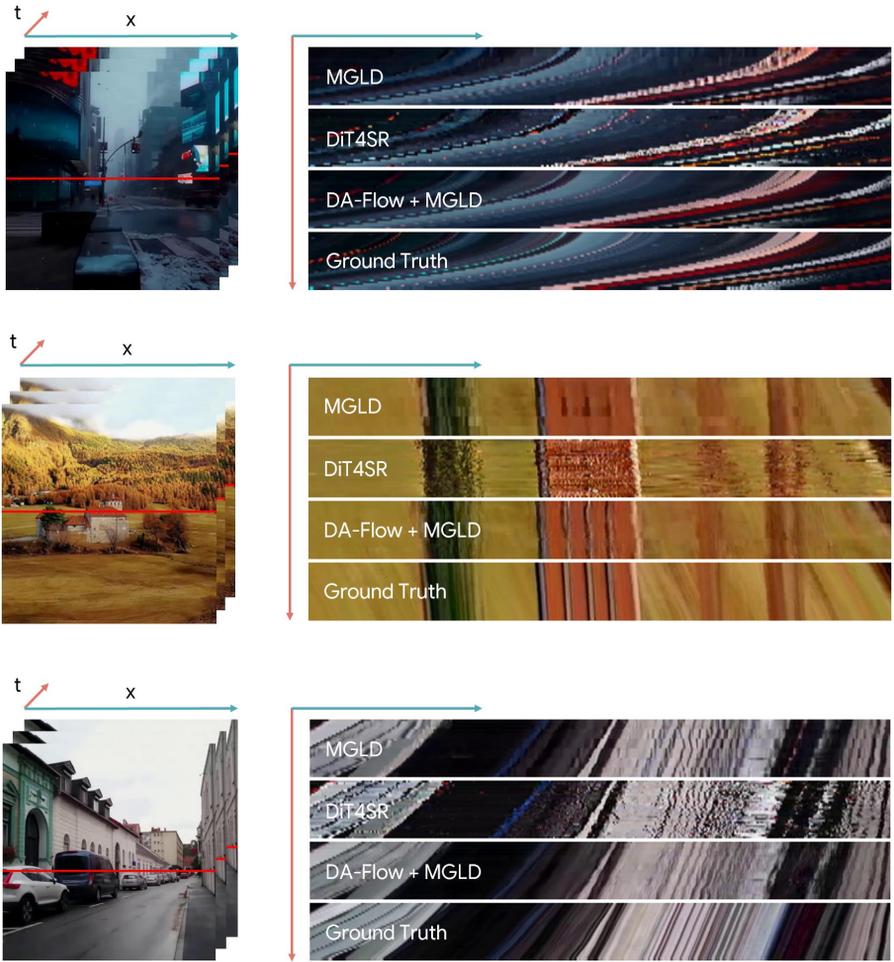
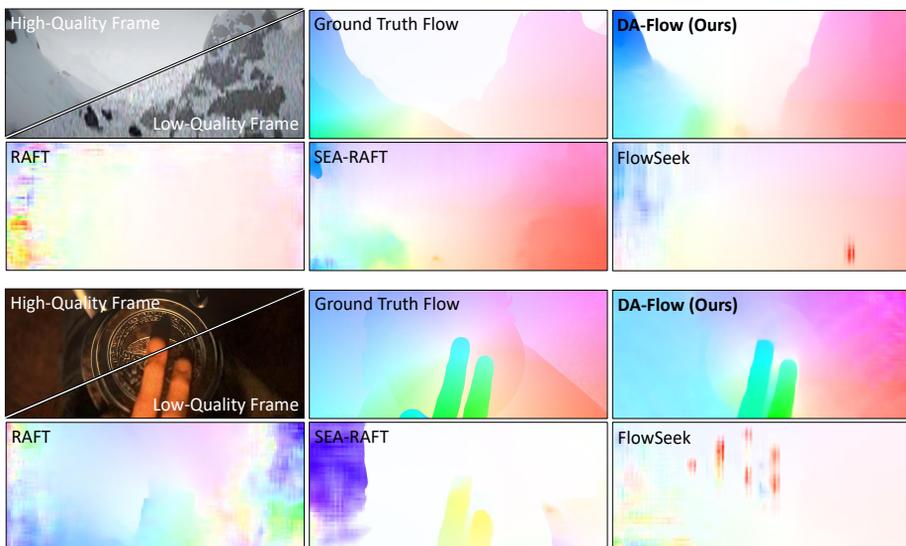Fig. 10: Comparison of temporal consistency in video restoration.

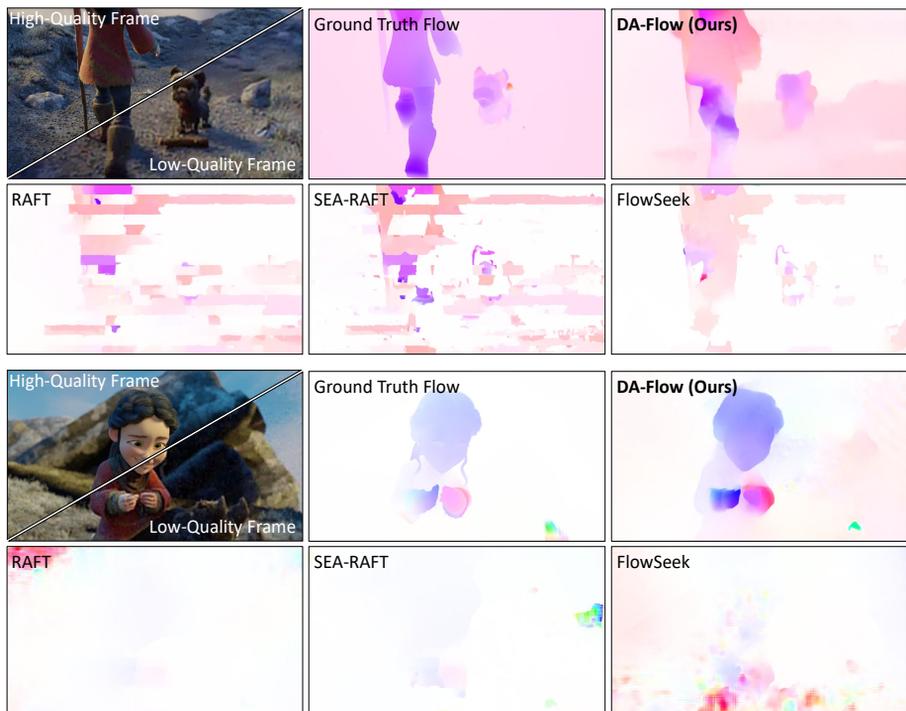Fig. 11: Additional qualitative results on Sintel [3].

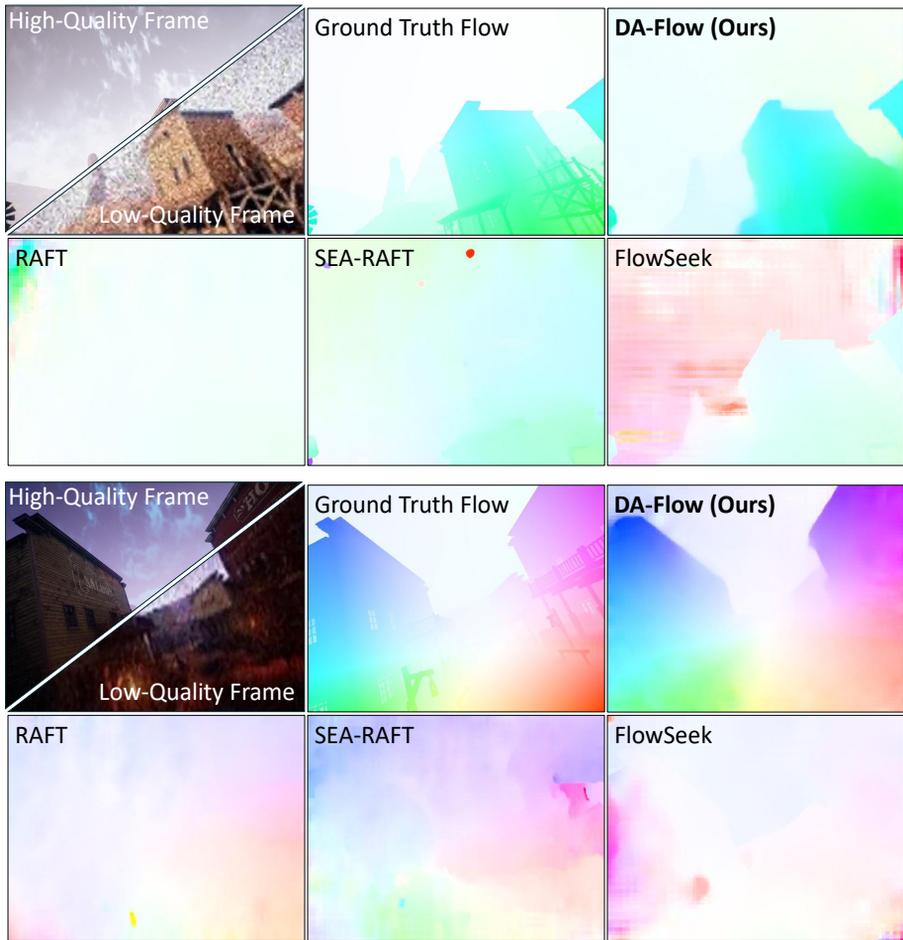Fig. 12: Additional qualitative results on Spring [20].

Fig. 13: Additional qualitative results on TartanAir [35].

E   Limitations and Future Work

In this paper, we introduce degradation-aware optical flow, a new task that aims to accurately estimate flow from degraded video frames. Our approach leverages features from an image restoration diffusion model via lifting, which inherently requires multiple denoising steps at inference time, resulting in slower runtime compared to conventional flow estimation networks. Exploring one-step distillation techniques to reduce the inference cost while preserving estimation quality remains a promising direction for future work.